Computers & Fluids 50 (2011) 81–93

Contents lists available at ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

An implicit matrix-free Discontinuous Galerkin solver for viscous and turbulent aerodynamic simulations

A. Crivellini [a,*], F. Bassi [b]

[a] Dipartimento di Energetica, Universitá Politecnica delle Marche, via Brecce Bianche, 60131 Ancona, Italy
[b] Dipartimento di Ingegneria Industriale, Universitá di Bergamo, via Marconi 4, 24044 Dalmine (BG), Italy

ARTICLE INFO

Article history:
Received 13 November 2010
Received in revised form 2 May 2011
Accepted 27 June 2011
Available online 6 July 2011

Keywords:
Discontinuous Galerkin
RANS equations
Computational fluid dynamics
Implicit method
Matrix-free
Newton–Krylov

ABSTRACT

This paper presents some recent advancements of the computational efficiency of a Discontinuous Galerkin (DG) solver for the Navier–Stokes (NS) and Reynolds Averaged Navier Stokes (RANS) equations. The implementation and the performance of a Newton–Krylov matrix-free (MF) method is presented and compared with the matrix based (MB) counterpart. Moreover two solution strategies, developed in order to increase the solver efficiency, are discussed and experimented. Numerical results of some test cases proposed within the EU ADIGMA (Adaptive Higher-Order Variational Methods for Aerodynamic Applications in Industry) project demonstrate the capabilities of the method.

1. Introduction

The growing interest that the DG method has been receiving in the recent years is due to various attractive features of the method. In fact, DG methods are finite element methods which account for the physics of wave propagation by means of Riemann solvers as in upwind finite volume methods but, unlike the latter, they can achieve higher-order accuracy on general unstructured grids using high degree polynomials as it is customary in the classical (continuous) finite element method.

Despite these advantages, DG methods for real life applications still require to be assessed and improved in many respects, such as applicability to complex flow models, shock-capturing properties and computational efficiency. In the present paper the latter topic will be discussed.

It is well known that for turbulent simulations an explicit time integration of the semidiscrete DG equations is not a suitable choice since: (i) the computational grid is highly stretched around the wall boundaries; (ii) the RANS equations are stiff; (iii) the CFL (Courant–Friedrichs–Lewy) stability limit rapidly decreases as the polynomial order of the solution grows. Thus in this case the use of an implicit in time scheme is almost mandatory. Nevertheless the solution of the large block sparse linear system arising from an implicit time DG discretization of the NS and RANS equations becomes prohibitively expensive as the grid density and/or the order of polynomial approximation increases. For this reason effective numerical strategies must be developed and recently this topic has been considered by many researches. A natural choice refers to the Newton–Krylov method: GMRES algorithm preconditioned by a standard incomplete lower–upper factorization (ILU) are adopted both by Bassi et al. [1,2] and Landmann et al. [3] while Person and Peraire [4], Diosady and Darmofal [5] and Shahbazi et al. [6] use a linear p-multigrid (where lower-order approximations serve as "coarse" levels while the same spatial grid elements are used on all levels) algorithm, with different types of smoothers for preconditioning the GMRES iterative solver. This approach seems to be particularly suitable for RANS simulations and, as a matter of fact, all the authors mentioned above have reported high-order turbulent results.

A criticism for this approaches is related to the huge memory requirements needed for the storage of both the Jacobian and the preconditioning matrix, except the approach of Diosady and Darmofal which use an in-place Block-ILU(0) factorization algorithm and thus store only one matrix. For this reason nonlinear h/p-multigrid, also due to the long-term experience made with finite volume methods, is considered a promising alternative.

* Corresponding author.
E-mail addresses: a.crivellini@univpm.it (A. Crivellini), francesco.bassi@unibg.it (F. Bassi).

0045-7930/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2011.06.020

Nevertheless even if nonlinear p-multigrid, which has been recently intensely developed by many authors [7–11], with explicit smoothers does not require any matrix memorization to enhance efficiency, semi-implicit and/or coarse level fully implicit solvers are widely adopted, thus reducing the memory gain of the method.

Moreover Shahbazi et al. [6], in a recent and extensive comparative study about the linear and non-linear p-multigrid methods for DG, claim that the multigrid preconditioned Newton-GMRES yields the most efficient and scalable algorithm. Finally, at the moment, only few turbulent computations have been published in literature, see for instance Bassi et al. [12,13], probably because these kind of algorithms still suffer from a lack of robustness if compared with Newton–Krylov approaches.

With the aim of saving at least one matrix storage, here a preconditioned (by an ILU(0) factorization of the analytically computed Jacobian matrix) Newton-GMRES matrix-free solver has been implemented in a high order DG code. A similar approach has been previously adopted by Rasetarinera and Hussani [14] and K. Hillewaert et al. [15] but only in the context of the DG solution of the compressible Euler inviscid equations. Here both viscous (NS) and turbulent (RANS) computations are performed and an in depth performance comparison with the standard matrix based counterpart is reported.

An analogous analysis, for standard finite element code, can be found in Kennedy et al. [16] and in Behara and Miattal [17] but in those cases the linear system solver is preconditioned by a simple block diagonal matrix which is not a feasible approach for a large, high order, DG simulation due to the stalling of the linear solver related to the high condition number of the iterative matrix arising from the discretization. Thus in this case is not possible to achieve the extremely competitive CPU performance, reported in the above works, which overtakes that of the MB GMRES, when a small Krylov space is employed.

In fact, since in our case the Jacobian matrix is still evaluated, the MF algorithm operation count is expected to be always larger. However, since this matrix is used only for preconditioning purpose, MF enjoys more flexibility, compared to standard algorithm, which is exploited by two approaches here proposed in order to increase the algorithm computational efficiency.

2. Governing equations

The complete set of RANS and k–ω equations can be written as:

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

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

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

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

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

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

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

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

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

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

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

$$\tilde{\omega}_r = \max(\tilde{\omega}, \tilde{\omega}_{r0}). \tag{11}$$

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 [18].

Notice that the RANS and k–ω equations above are not in standard form since, in order to deal with the stiffness of the k–ω equations, we resorted to a non-standard implementation of the two equations differential model. The main features of this implementation consist in (i) using as variable $\tilde{\omega} = \log \omega$ rather than ω and (ii) fulfilling the realizability constraints employing in Eqs. (3), (4) and (10) the variable $\tilde{\omega}_r = \max(\tilde{\omega}, \tilde{\omega}_{r0})$, where $\tilde{\omega}_{r0}$ is the lower bound predicting positive normal turbulent stresses and satisfying the Schwarz inequality for the shear turbulent stresses. A detailed description of the model can be found in [2].

3. DG space discretization

The governing Eqs. (1)–(5) can be written in the following compact form:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{F}_c(\mathbf{u}) + \nabla \cdot \mathbf{F}_v(\mathbf{u}, \nabla \mathbf{u}) + \mathbf{s}(\mathbf{u}, \nabla \mathbf{u}) = \mathbf{0}, \tag{12}$$

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.

Multiplying Eq. (12) by an arbitrary test function ϕ, integrating over the domain Ω and integrating by parts the divergence terms, the weak form of Eq. (12) reads:

$$\int_\Omega \phi \frac{\partial \mathbf{u}}{\partial t}\,d\mathbf{x} - \int_\Omega \nabla\phi \cdot \mathbf{F}(\mathbf{u}, \nabla \mathbf{u})d\mathbf{x} + \int_{\partial\Omega} \phi \mathbf{F}(\mathbf{u}, \nabla \mathbf{u}) \cdot \mathbf{n}\,d\sigma + \int_\Omega \phi \mathbf{s}(\mathbf{u}, \nabla \mathbf{u})d\mathbf{x} = \mathbf{0}, \tag{13}$$

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

In order to construct a DG discretization of Eq. (13), we consider an approximation Ω_h of Ω and a triangulation $\mathscr{T}_h = \{K\}$ of Ω_h, *i.e.*, a set of *ne* non-overlapping elements K not necessarily simplices. We denote with $\mathscr{E}_h^0$ the set of internal element faces, with $\mathscr{E}_h^\partial$ the set of boundary element faces and with $\mathscr{E}_h = \mathscr{E}_h^0 \cup \mathscr{E}_h^\partial$ their union. We moreover set

$$\Gamma_h^0 = \bigcup_{e \in \mathscr{E}_h^0} e, \quad \Gamma_h^\partial = \bigcup_{e \in \mathscr{E}_h^\partial} e, \quad \Gamma_h = \Gamma_h^0 \cup \Gamma_h^\partial. \tag{14}$$

The solution is approximated on $\mathscr{T}_h$ as a piecewise polynomial function possibly discontinuous on element interfaces, *i.e.*, we assume the following space setting for each component $u_{h_i} = u_{h_1}, \ldots, u_{h_M}$ of the numerical solution $\mathbf{u}_h$:

$$u_{h_i} \in \Phi_h \stackrel{\text{def}}{=} \left\{\phi_h \in L^2(\Omega) : \phi_h|_K \in \mathbb{P}_k(K)\ \forall K \in \mathscr{T}_h\right\} \tag{15}$$