Computers and Fluids 160 (2018) 26–41

Contents lists available at ScienceDirect

Computers and Fluids

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

A reconstructed discontinuous Galerkin method for compressible turbulent flows on 3D curved grids

Xiaodong Liu [a], Yidong Xia [b], Hong Luo [a,*]

[a] Department of Mechanical and Aerospace Engineering, North Carolina State University, Raleigh, NC 27695, USA
[b] Energy and Environment Science & Technology Directorate, Idaho National Laboratory, Idaho Falls, ID 83415, USA

ARTICLE INFO

Article history:
Received 8 March 2017
Revised 5 October 2017
Accepted 11 October 2017
Available online 20 October 2017

Keywords:
Discontinuous Galerkin
WENO
RANS
Curved elements

ABSTRACT

A third-order accurate reconstructed discontinuous Galerkin method, namely rDG(P_1P_2), is presented to solve the Reynolds-Averaged Navier–Stokes (RANS) equations, along with the modified one-equation model of Spalart and Allmaras (SA) on 3D curved grids. In this method, a piecewise quadratic polynomial solution (P_2) is obtained using a least-squares method from the underlying piecewise linear DG(P_1) solution. The reconstructed quadratic polynomial solution is then used for computing the inviscid and the viscous fluxes. Furthermore, Hermite Weighted Essentially Non-Oscillatory (WENO) reconstruction is used to guarantee the stability of the developed rDG method. A number of benchmark test cases based on a set of uniformly refined quadratic curved meshes are presented to assess the performance of the resultant rDG(P_1P_2) method for turbulent flow problems. The numerical results demonstrate that the rDG(P_1P_2) method is able to obtain reliable and accurate solutions to 3D compressible turbulent flows at a cost slightly higher than its underlying second-order DG(P_1) method.

1. Introduction

The discontinuous Galerkin (DG) methods, originally introduced for solving the neutron transport by Reed and Hill [1], have become popular for the solution of systems of conservation laws in recent decades. Nowadays, they are widely used in computational fluid dynamics (CFD), computational acoustics, and computational magneto-hydrodynamics (MHD) [2]. A lot of attractive features of DG methods have been listed in [3–7,11,22]. However, the DG methods also have a number of weaknesses that have yet to be addressed, e.g., how to reduce the high computational costs and how to develop more efficient time integration methods.

In order to reduce the high costs associated to the DG methods, Dumbser et al. [8–10] introduced a new family of so-called reconstructed DG, termed P_nP_m schemes and referred to as rDG (P_nP_m) in this paper. P_n indicates that a piecewise polynomial of degree of n is used to represent the underlying DG solution, and P_m represents a polynomial solution of degree of m ($m \geq n$) that is reconstructed from the underlying P_n polynomial and used to compute the fluxes. The P_nP_m schemes can be constructed based on a few different algorithms, e.g., the recovery approach [12], the reconstruction approach [14,25], and the Gauss–Green approach [16,17], all of which were proved to deliver the designed grid convergence of $\mathcal{O}(h^{m+1})$ [18]. Indeed, implicit methods can especially benefit from the use of rDG(P_nP_m) methods as the costs can be substantially reduced in two aspects [19,20]. Firstly, fewer spatial integration points are required for evaluating the residual vector and Jacobian matrix. For instance, the third-order rDG (P_1P_2) only needs 4 points for triangular boundary integral whereas the equivalent DG (P_2) requires 7. Secondly, the Jacobian matrix of rDG (P_nP_m) is based on the underlying DG (P_n), and thus requires much less storage than the equivalent DG (P_m). For example, for RANS-SA system, the memory needed for the diagonal part of the Jacobian matrix of rDG(P_1P_2) is 576 word versus 3600 needed by DG (P_2) for 3D cases.

In our latest work, a rDG(P_1P_2) method based on a Hierarchical WENO reconstruction has been successfully used to solve the compressible flows on unstructured grids, ranging from inviscid [19–22,43] to viscous flows [13,20,28,43,45]. This rDG(P_1P_2) method is designed not only to reduce the high computing costs of the DGM, but also to avoid spurious oscillations in the vicinity of strong discontinuities, thus effectively addressing the two shortcomings of the DGM. However, most practical applications belong to 3D turbulent flows. Therefore, the great success for rDG(P_1P_2) has motivated us to extend this promising method to solution of the practical 3D turbulent flows.

In recent years, development and application of high-order methods for computing high Reynolds number turbulent flows

* Corresponding author.
E-mail address: hong_luo@ncsu.edu (H. Luo).

https://doi.org/10.1016/j.compfluid.2017.10.014

governed by the Reynolds-averaged Navier–Stokes (RANS) equations have been an active research topic all over the world, as demonstrated by the two European projects: ADIGMA and IDIHOM [49,50]. It is well known that the use of high-order methods for computing RANS problems is not commonplace, mainly due to the severe numerical instability by the highly non-smooth behavior between the turbulent and non-turbulent flow regions. Although it is still a challenging problem, there are several successful implementations for the one-equation Spalart and Allmaras (SA) model and the two-equation $k - \omega$ model. In [33] by Bassi and Rebay, they successfully solved the RANS equations with a modified $k - \omega$ model based on a high-order DG framework. In their work, the logarithm of ω rather than ω itself is used as the unknown, which has been found very useful to enhance stabilty of the method. Similarly, Hartmann et al. [34] developed a DG code for 3D turbulent flow computation with adaptive mesh refinement based on the $k - \omega$ model. In particular, some test cases with increasing complexity have been used to validate the solver. Burgess et al. [35] and Wang et al. [29] developed high-order DG methods for solving a fully coupled RANS-SA system respectively. The modified SA model is particularly designed to make the original SA model insensitive to negative values of turbulence working variables as numerical experiments show that turbulence working variable often drops several orders of magnitudes at the edge of the turbulent boundary layer. Ceze and Fidkowski [36] applied a high-order output-based adaptive solution technique to the 2D RANS equations closed with the modified negative SA model which originally proposed by Allmaras et al. [37]. Compared with uniform refinement at second order, high-order yields faster convergence. Nguyen et al. [39] implemented the SA model equation based on a DG framework with an artificial viscosity modification for SA equation. It is aimed to accomodate high-order RANS approximations on too coarse grids. Besides that, Crivellini et al. [38] analyzed and implemented a modified SA model based on high-order DG methods for incompressible flows. For modified SA model, they introduce an SA model implementation that dealt with negative $\tilde{\nu}$ values by modifying the source and diffusion terms in the SA model equation only when the working variable or one of the model closure functions became negative. Zhou et al. [40] successfully solved RANS-SA system in a high-order correction procedure via reconstruction (CPR) framework. In their work, a high-order solver based on CPR has been developed for the Eikonal equation to compute the nearest distance to the wall.

In addition, the use of high-order curved boundary elements is essential for high-order schemes to deliver an overall accurate solution [29]. Poor representation of the actual geometry could result in a significant amount of artificial entropy produced along the geometry surface, thus degrading the solution accuracy. Furthermore, since the elements are of high-aspect-ratio through the thin boundary layer for high Reynolds number flow, the interior elements are also required to be curvilinear to avoid the negative cells.

The objective of this paper is to develop a reconstructed discontinuous Galerkin method for the solution of RANS-SA system on 3D curved grids. First, the quadratic curved elements generated by Gmsh [30] or simple agglomeration are shown satisfatory for the solution of high order reconstructed discontinuous Galerkin method. Second, grid convergence study on uniformly refined meshes with high-order reconstructed discontinuous Galerkin is reported about the 3D turbulent benchmark test cases with increasing complexity from NASA official website [47]. This provides direct and valuable comparison with second-order finite volume method. Since a WENO reconstruction has to be used for unstructured meshes, e.g.prism and tetrahedron, to guarantee the stability of rDG(P_1P_2), it is necessary to demonstrate that WENO(P_1P_2) could still guarantee the accuracy of the designed rDG(P_1P_2) method. Thus, third, the accuracy of WENO(P_1P_2) is also validated by comparison with the output of rDG(P_1P_2) without WENO reconstruction on subsonic turbulent flow cases, i.e. Zero Pressure Gradient Flat Plate.

The outline of the rest of this paper is organized as follows. The governing equations are described in Section 2. The developed reconstructed discontinuous Galerkin method is presented in Section 3. The use of curved elements is introduced in Section 4. Extensive numerical experiments are reported in Section 5. Concluding remarks are given in Section 6.

2. Governing equations

The conservation form of the compressible RANS equations with the modified one-equation Spalart–Allmaras (SA) turbulence model [44] is given as below,

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}_j(\boldsymbol{U})}{\partial x_j} = \frac{\partial \mathbf{G}_j(\boldsymbol{U})}{\partial x_j} + \boldsymbol{S} \tag{1}$$

where the summation convention ($j = 1, 2, 3$) has been used. In Eq. (1), the conservative variables $\boldsymbol{U}$ are defined as

$$\boldsymbol{U} = (\rho, \rho u_j, \rho e, \rho\tilde{\nu})^T \tag{2}$$

where ρ, p and e denote the density, pressure and specific total energy of the fluid, respectively, u_j is the velocity components of the flow in the coordinate direction x_j, and $\tilde{\nu}$ represents the turbulence working variable in the modified SA model.

The invisid and viscous flux vector, i.e. $\boldsymbol{F}$ and $\boldsymbol{G}$, are defined by

$$\boldsymbol{F}_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p\delta_{ij} \\ u_j(\rho e + p) \\ \rho u_j \tilde{\nu} \end{pmatrix}, \qquad \boldsymbol{G}_k = \begin{pmatrix} 0 \\ \tau_{ij} \\ u_i\tau_{ij} + q_j \\ \frac{1}{\sigma}\mu(1+\psi)\frac{\partial\tilde{\nu}}{\partial x_j} \end{pmatrix} \tag{3}$$

and the source term $\boldsymbol{S}$ is defined by

$$\boldsymbol{S}^T = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & s \end{pmatrix} \tag{4}$$

where $s = c_{b1}\tilde{S}\mu\psi + \frac{c_{b2}}{\sigma}\rho\nabla\tilde{\nu}\cdot\nabla\tilde{\nu} - c_{w1}\rho f_w(\frac{\nu\psi}{d})^2 - \frac{1}{\sigma}\nu(1+\psi)\nabla\rho\cdot\nabla\tilde{\nu}$.

The Newtonian fluid with the Stokes hypothesis is valid under the current framework, since only air is considered. The viscous stress tensor τ is defined by

$$\tau_{ij} = (\mu + \mu_t)\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) \tag{5}$$

where δ_{ij} is the Kronecker delta function, μ represents the molecular viscosity coefficient, which can be determined through Sutherland's law, and μ_t denotes the turbulence eddy viscosity, which is given by:

$$\mu_t = \begin{cases} \rho\tilde{\nu}f_{v1} & if \quad \tilde{\nu} \geq 0 \\ 0 & if \quad \tilde{\nu} < 0 \end{cases} \tag{6}$$

The heat flux vector q_j, which is formulated according to Fourier's law, is given by

$$q_j = -c_p\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial T}{\partial x_j} \tag{7}$$

where c_p is the specific heat capacity at constant pressure , Pr is the nondimensional laminar Prandtl number, Pr_t is the turbulent Prandtl number, and the temperature of the fluid T is determined by $T = \frac{p}{\rho R}$.