

A reconstructed discontinuous Galerkin method based on a Hierarchical WENO reconstruction for compressible flows on tetrahedral grids



Hong Luo^{a,*}, Yidong Xia^{a,2}, Seth Spiegel^{a,2}, Robert Nourgaliev^{b,3}, Zonglin Jiang^{c,4}

^a Department of Mechanical and Aerospace Engineering, North Carolina State University, Raleigh, NC 27695, United States

^b Idaho National Laboratory, Idaho Falls, ID 83415, United States

^c Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China

ARTICLE INFO

Article history:

Received 10 August 2012

Received in revised form 22 November 2012

Accepted 27 November 2012

Available online 10 December 2012

Keywords:

Discontinuous Galerkin method
WENO reconstruction
Unstructured grids

ABSTRACT

A reconstructed discontinuous Galerkin (RDG) method based on a hierarchical WENO reconstruction, termed HWENO (P_1P_2) in this paper, designed not only to enhance the accuracy of discontinuous Galerkin methods but also to ensure the nonlinear stability of the RDG method, is presented for solving the compressible Euler equations on tetrahedral grids. In this HWENO (P_1P_2) method, a quadratic polynomial solution (P_2) is first reconstructed using a Hermite WENO reconstruction from the underlying linear polynomial (P_1) discontinuous Galerkin solution to ensure the linear stability of the RDG method and to improve the efficiency of the underlying DG method. By taking advantage of handily available and yet invaluable information, namely the derivatives in the DG formulation, the stencils used in the reconstruction involve only von Neumann neighborhood (adjacent face-neighboring cells) and thus are compact. The first derivatives of the quadratic polynomial solution are then reconstructed using a WENO reconstruction in order to eliminate spurious oscillations in the vicinity of strong discontinuities, thus ensuring the nonlinear stability of the RDG method. The developed HWENO (P_1P_2) method is used to compute a variety of flow problems on tetrahedral meshes to demonstrate its accuracy, robustness, and non-oscillatory property. The numerical experiments indicate that the HWENO (P_1P_2) method is able to capture shock waves within one cell without any spurious oscillations, and achieve the designed third-order of accuracy: one order accuracy higher than the underlying DG method.

© 2012 Elsevier Inc. All rights reserved.

1. Introduction

The discontinuous Galerkin methods [1–28] (DGM) have recently become popular for the solution of systems of conservation laws. Originally introduced for the solution of neutron transport equations [1], nowadays they are widely used in computational fluid dynamics, computational acoustics, and computational magneto-hydrodynamics. The discontinuous Galerkin methods combine two advantageous features commonly associated with finite element and finite volume methods.

* Corresponding author.

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

¹ Associate Professor, Department of Mechanical and Aerospace Engineering.

² Ph.D. student, Department of Mechanical and Aerospace Engineering.

³ Senior Scientist, Thermal Science and Safety Analysis Department.

⁴ Professor, State Key Laboratory of High Temperature Gas Dynamics.

As in classical finite element methods, accuracy is obtained by means of high-order polynomial approximation within an element rather than by wide stencils as in the case of finite volume methods. The physics of wave propagation is, however, accounted for by solving the Riemann problems that arise from the discontinuous representation of the solution at element interfaces. In this respect, the DG methods are similar to finite volume methods. The discontinuous Galerkin methods have many attractive features: (1) They have several useful mathematical properties with respect to conservation, stability, and convergence; (2) The methods can be easily extended to higher-order (>2nd) approximation; (3) The methods are well suited for complex geometries since they can be applied on unstructured grids. In addition, the methods can also handle non-conforming elements, where the grids are allowed to have hanging nodes; (4) The methods are highly parallelizable, as they are compact and each element is independent. Since the elements are discontinuous, and the inter-element communications are minimal, domain decomposition can be efficiently employed. The compactness also allows for structured and simplified coding for the methods; (5) They can easily handle adaptive strategies, since refining or coarsening a grid can be achieved without considering the continuity restriction commonly associated with the conforming elements. The methods allow easy implementation of *hp*-refinement, for example, the order of accuracy, or shape, can vary from element to element; (6) They have the ability to compute low Mach number flow problems without recourse to the time-preconditioning techniques normally required for the finite volume methods. In contrast to the enormous advances in the theoretical and numerical analysis of the DGM, the development of a viable, attractive, competitive, and ultimately superior DG method over the more mature and well-established second order finite volume methods is relatively an untouched area. This is mainly due to the fact that the DGM have a number of weaknesses that have yet to be addressed, before they can be robustly used for flow problems of practical interest in a complex configuration environment. In particular, how to effectively control spurious oscillations in the presence of strong discontinuities, and how to reduce the computing costs for the DGM remain the two most challenging and unresolved issues in the DGM. Indeed, compared to the finite element methods and finite volume methods, the DGM require solutions of systems of equations with more unknowns for the same grids. Consequently, these methods have been recognized as expensive in terms of both computational costs and storage requirements especially in the context of implicit methods, where the memory requirement for the Jacobian matrix grows quadratically with the order of the DG methods, thus leading to a significant increase in computational cost.

In order to reduce high costs associated with the DGM, Dumbser et al. [18–20] have introduced a new family of reconstructed DGM, termed P_nP_m schemes and referred to as RDG (P_nP_m) in this paper, where P_n indicates that a piecewise polynomial of degree *n* is used to represent a DG solution, and P_m represents a reconstructed polynomial solution of degree *m* (*m* ≥ *n*) that is used to compute the fluxes. The RDG (P_nP_m) schemes are designed to enhance the accuracy of the discontinuous Galerkin method by increasing the order of the underlying polynomial solution. The beauty of RDG (P_nP_m) schemes is that they provide a unified formulation for both finite volume and DGM, and contain both classical finite volume and standard DG methods as two special cases of RDG (P_nP_m) schemes, and thus allow for a direct efficiency comparison. When *n* = 0, i.e., a piecewise constant polynomial is used to represent a numerical solution, RDG (P₀P_m) is nothing but classical high order finite volume schemes, where a polynomial solution of degree *m* (*m* ≥ 1) is reconstructed from a piecewise constant solution. When *m* = *n*, the reconstruction reduces to the identity operator, and RDG (P_nP_n) scheme yields a standard DG method.

Obviously, the construction of an accurate and efficient reconstruction operator is crucial to the success of the RDG (P_nP_m) schemes. In Dumbser's work, a higher order polynomial solution is reconstructed using a L₂ projection, requiring it indistinguishable from the underlying DG solutions in the contributing cells in the weak sense. The resultant over-determined system is then solved using a least-squares method that guarantees exact conservation, not only of the cell averages but also of all higher order moments in the reconstructed cell itself, such as slopes and curvatures. However, this conservative least-squares reconstruction approach is computationally expensive, as the L₂ projection, i.e., the operation of integration, is required to obtain the resulting over-determined system. Furthermore, the reconstruction might be problematic for a boundary cell, where the number of the face-neighboring cells might be not enough to provide the necessary information to recover a polynomial solution of a desired order. Fortunately, the projection-based reconstruction is not the only way to obtain a polynomial solution of higher order from the underlying discontinuous Galerkin solutions. In a reconstructed DG method using a Taylor basis [26–28] developed by Luo et al. for the solution of the compressible Euler and Navier–Stokes equations on arbitrary grids, a higher order polynomial solution is reconstructed by use of a strong interpolation, requiring point values and derivatives to be interpolated on the face-neighboring cells. The resulting over-determined linear system of equations is then solved in the least-squares sense. This reconstruction scheme only involves von Neumann neighborhood, and thus is compact, simple, robust, and flexible. Like the projection-based reconstruction, the strong reconstruction scheme guarantees exact conservation, not only of the cell averages but also of their slopes due to a judicious choice of the Taylor basis. More recently, Zhang et al. [29,30] presented a class of hybrid DG/FV methods for the conservation laws, where the second derivatives in a cell are obtained from the first derivatives in the cell itself and its neighboring cells using a Green–Gauss reconstruction widely used in the finite volume methods. This provides a fast, simple, and robust way to obtain higher-order polynomial solutions. Lately, Luo et al. [31,32] have conducted a comparative study for these three reconstructed discontinuous Galerkin methods RDG (P₁P₂) by solving 2D Euler equations on arbitrary grids. It is found that all three reconstructed discontinuous Galerkin methods can deliver the desired third order of accuracy and significantly improve the accuracy of the underlying second-order DG method, although the least-squares reconstruction method provides the best performance in terms of both accuracy and robustness.

Unfortunately, the attempt to extend our RDG method to solve 3D Euler equations on tetrahedral grids was not successful. Like the second order cell-centered finite volume methods RDG (P₀P₁), the resultant RDG (P₁P₂) method is unstable.

Although RDG (P_0P_1) methods are in general stable in 2D and on Cartesian or structured grids in 3D, they suffer from the so-called *linear instability* on unstructured tetrahedral grids, when the reconstruction stencils only involve von Neumann neighborhood, i.e., adjacent face-neighboring cells [33]. The RDG (P_1P_2) method exhibits the same linear instability, which can be overcome by using extended stencils. However, this is achieved at the expense of sacrificing the compactness of the underlying DG methods. Furthermore, these linear reconstruction-based DG methods will suffer from non-physical oscillations in the vicinity of strong discontinuities for the compressible Euler equations. Alternatively, ENO, WENO, and HWENO can be used to reconstruct a higher-order polynomial solution, thereby not only enhancing the order of accuracy of the underlying DG method but also achieving both linear and non-linear stability. This type of hybrid HWENO + DG schemes has been developed on 1D and 2D structured grids by Balsara et al. [34], where the HWENO reconstruction is relatively simple and straightforward.

The objective of the effort discussed in this paper is to develop a reconstructed discontinuous Galerkin method based on a hierarchical WENO reconstruction, HWENO (P_1P_2), using a Taylor basis [13] for the solution of the compressible Euler equations on unstructured tetrahedral grids. The HWENO (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. In this HWENO (P_1P_2) method, a quadratic solution is first reconstructed to enhance the accuracy of the underlying DG method in two steps: (1) all second derivatives on each cell are first reconstructed using the solution variables and their first derivatives from adjacent face-neighboring cells via a strong interpolation; (2) the final second derivatives on each cell are then obtained using a WENO strategy based on the reconstructed second derivatives on the cell itself and its adjacent face-neighboring cells. This reconstruction scheme, by taking advantage of handily available and yet valuable information namely the gradients in the context of the DG methods, only involves von Neumann neighborhood and thus is compact, simple, robust, and flexible. As the underlying DG method is second-order, and the basis functions are at most linear functions, fewer quadrature points are then required for both domain and face integrals, and the number of unknowns (the number of degrees of freedom) remains the same as for the DG (P_1). Consequently, this RDG method is more efficient than its third order DG (P_2) counterpart. The gradients of the quadratic polynomial solutions are then modified using a WENO reconstruction in order to eliminate non-physical oscillations in the vicinity of strong discontinuities, thus ensuring the non-linear stability of the RDG method. The developed HWENO (P_1P_2) method is used to compute a variety of flow problems on tetrahedral grids to demonstrate its accuracy, robustness, and non-oscillatory performance. The presented numerical results indicate that this HWENO (P_1P_2) method is able to capture shock waves within once cell without any spurious oscillations, and achieve the designed third-order of accuracy: one order accuracy higher than the underlying DG (P_1) method, and thus significantly increase its accuracy without significant increase in computing costs and memory requirements. The remainder of this paper is organized as follows. The governing equations are listed in Section 2. The developed reconstructed discontinuous Galerkin method is presented in Section 3. Extensive numerical experiments are reported in Section 4. Concluding remarks are given in Section 5.

2. Governing equations

The Euler equations governing unsteady compressible inviscid flows can be expressed as

$$\frac{\partial \mathbf{U}(x, t)}{\partial t} + \frac{\partial \mathbf{F}_k(\mathbf{U}(x, t))}{\partial x_k} = 0 \tag{2.1}$$

where the summation convention has been used. The conservative variable vector \mathbf{U} , and inviscid flux vector \mathbf{F} are defined by

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u_i \\ \rho e \end{pmatrix} \quad \mathbf{F}_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p \delta_{ij} \\ u_j(\rho e + p) \end{pmatrix} \tag{2.2}$$

Here ρ , p , and e denote the density, pressure, and specific total energy of the fluid, respectively, and u_i is the velocity of the flow in the coordinate direction x_i . The pressure can be computed from the equation of state

$$p = (\gamma - 1)\rho \left(e - \frac{1}{2} u_j u_j \right) \tag{2.3}$$

which is valid for perfect gas, where γ is the ratio of the specific heats.

3. Reconstructed discontinuous Galerkin method

3.1. Discontinuous Galerkin formulation

The governing Eq. (2.1) is discretized using a discontinuous Galerkin finite element formulation. We first introduce some notations. We assume that the domain Ω is subdivided into a collection of non-overlapping tetrahedral elements Ω_e . We introduce the following broken Sobolev space \mathbf{V}_h^p

$$V_h^p = \left\{ v_h \in [L_2(\Omega)]^m : v_h|_{\Omega_e} \in [V_p^m] \forall \Omega_e \in \Omega \right\}, \tag{3.1}$$

which consists of discontinuous vector-values polynomial functions of degree p , and where m is the dimension of the unknown vector and V_p is the space of all polynomials of degree $\leq p$. To formulate the discontinuous Galerkin method, we introduce the following weak formulation, which is obtained by multiplying the above conservation law (2.1) by a test function \mathbf{W}_h , integrating over an element Ω_e , and then performing an integration by parts,

Find $\mathbf{U}_h \in V_h^p$ such as

$$\frac{d}{dt} \int_{\Omega_e} \mathbf{U}_h \mathbf{W}_h d\Omega + \int_{\Gamma_e} \mathbf{F}_k(\mathbf{U}_h) \mathbf{n}_k \mathbf{W}_h d\Gamma - \int_{\Omega_e} \mathbf{F}_k(\mathbf{U}_h) \frac{\partial \mathbf{W}_h}{\partial x_k} d\Omega = 0, \quad \forall \mathbf{W}_h \in V_h^p \tag{3.2}$$

where \mathbf{U}_h and \mathbf{W}_h are represented by piecewise-polynomial functions of degrees p , which are discontinuous between the cell interfaces and \mathbf{n}_k denotes the unit outward normal vector to Γ_e : the boundary of Ω_e . Assume that B_i is the basis of polynomial function of degrees p , this is then equivalent to the following system of N equations,

$$\frac{d}{dt} \int_{\Omega_e} \mathbf{U}_h B_i d\Omega + \int_{\Gamma_e} \mathbf{F}_k(\mathbf{U}_h) \mathbf{n}_k B_i d\Gamma - \int_{\Omega_e} \mathbf{F}_k(\mathbf{U}_h) \frac{\partial B_i}{\partial x_k} d\Omega = 0, \quad 1 \leq i \leq N \tag{3.3}$$

where N is the dimension of the polynomial space. Since the numerical solution \mathbf{U}_h is discontinuous between element interfaces, the interface fluxes are not uniquely defined. The flux function $\mathbf{F}_k(\mathbf{U}_h) \mathbf{n}_k$ appearing in the second terms of Eq. (3.3) is replaced by a numerical Riemann flux function $\mathbf{H}_k(\mathbf{U}_h^L, \mathbf{U}_h^R, \mathbf{n}_k)$ where \mathbf{U}_h^L and \mathbf{U}_h^R are the conservative state vector at the left and right side of the element boundary, and which is computed using HLLC scheme [43] in the present work. This scheme is called discontinuous Galerkin method of degree p , or in short notation DG (P) method. Note that discontinuous Galerkin formulations are very similar to finite volume schemes, especially in their use of numerical fluxes. Indeed, the classical first-order cell-centered finite volume scheme exactly corresponds to the DG (P₀) method, i.e., to the discontinuous Galerkin method using a piecewise-constant polynomial. Consequently, the DG (P_k) methods with $k > 0$ can be regarded as a natural generalization of finite volume methods to higher order methods. By simply increasing the degree p of the polynomials, the DG methods of corresponding higher order are obtained. The domain and boundary integrals in Eq. (3.3) are calculated using Gauss quadrature formulas. The number of quadrature points used is chosen to integrate exactly polynomials of order of $2p$ and $2p + 1$ for volume and surface inner products in the reference element. In the traditional DGM, termed nodal discontinuous Galerkin methods, numerical polynomial solutions \mathbf{U}_h in each element are expressed using either standard Lagrange finite element or hierarchical node-based basis as following

$$\mathbf{U}_h = \sum_{i=1}^N \mathbf{U}_i(t) B_i(x), \tag{3.4}$$

where B_i are the finite element basis functions. As a result, the unknowns to be solved for are the variables at the nodes \mathbf{U}_i .

In the present work, the numerical polynomial solutions are represented using a Taylor series expansion at the center of the cell. For example, if we do a Taylor series expansion at the center of the cell, a quadratic polynomial solution can be expressed as follows:

$$\begin{aligned} \mathbf{U}_h = & \mathbf{U}_c + \frac{\partial \mathbf{U}}{\partial x} \Big|_c (x - x_c) + \frac{\partial \mathbf{U}}{\partial y} \Big|_c (y - y_c) + \frac{\partial \mathbf{U}}{\partial z} \Big|_c (z - z_c) + \frac{\partial^2 \mathbf{U}}{\partial x^2} \Big|_c \frac{(x - x_c)^2}{2} + \frac{\partial^2 \mathbf{U}}{\partial y^2} \Big|_c \frac{(y - y_c)^2}{2} + \frac{\partial^2 \mathbf{U}}{\partial z^2} \Big|_c \frac{(z - z_c)^2}{2} \\ & + \frac{\partial^2 \mathbf{U}}{\partial x \partial y} \Big|_c (x - x_c)(y - y_c) + \frac{\partial^2 \mathbf{U}}{\partial x \partial z} \Big|_c (x - x_c)(z - z_c) + \frac{\partial^2 \mathbf{U}}{\partial y \partial z} \Big|_c (y - y_c)(z - z_c) \end{aligned} \tag{3.5}$$

which can be further expressed as cell-averaged values and their derivatives at the center of the cell:

$$\begin{aligned} \mathbf{U}_h = & \tilde{\mathbf{U}} + \frac{\partial \mathbf{U}}{\partial x} \Big|_c (x - x_c) + \frac{\partial \mathbf{U}}{\partial y} \Big|_c (y - y_c) + \frac{\partial \mathbf{U}}{\partial z} \Big|_c (z - z_c) + \frac{\partial^2 \mathbf{U}}{\partial x^2} \Big|_c \left(\frac{(x - x_c)^2}{2} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{(x - x_c)^2}{2} d\Omega \right) \\ & + \frac{\partial^2 \mathbf{U}}{\partial y^2} \Big|_c \left(\frac{(y - y_c)^2}{2} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{(y - y_c)^2}{2} d\Omega \right) + \frac{\partial^2 \mathbf{U}}{\partial z^2} \Big|_c \left(\frac{(z - z_c)^2}{2} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{(z - z_c)^2}{2} d\Omega \right) \\ & + \frac{\partial^2 \mathbf{U}}{\partial x \partial y} \Big|_c \left((x - x_c)(y - y_c) - \frac{1}{\Omega_e} \int_{\Omega_e} (x - x_c)(y - y_c) d\Omega \right) \\ & + \frac{\partial^2 \mathbf{U}}{\partial x \partial z} \Big|_c \left((x - x_c)(z - z_c) - \frac{1}{\Omega_e} \int_{\Omega_e} (x - x_c)(z - z_c) d\Omega \right) \\ & + \frac{\partial^2 \mathbf{U}}{\partial y \partial z} \Big|_c \left((y - y_c)(z - z_c) - \frac{1}{\Omega_e} \int_{\Omega_e} (y - y_c)(z - z_c) d\Omega \right) \end{aligned} \tag{3.6}$$

where $\tilde{\mathbf{U}}$ is the mean value of \mathbf{U} in this cell. The unknowns to be solved for in this formulation are the cell-averaged variables and their derivatives at the center of the cells. The dimension of the polynomial space is 10 and the basis functions are

$$\begin{aligned}
 B_1 &= 1, \quad B_2 = x - x_c, \quad B_3 = y - y_c, \quad B_4 = z - z_c \\
 B_5 &= \frac{B_2^2}{2} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{B_2^2}{2} d\Omega, \quad B_6 = \frac{B_3^2}{2} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{B_3^2}{2} d\Omega, \quad B_7 = \frac{B_4^2}{2} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{B_4^2}{2} d\Omega, \\
 B_8 &= B_2B_3 - \frac{1}{\Omega_e} \int_{\Omega_e} B_2B_3 d\Omega, \quad B_9 = B_2B_4 - \frac{1}{\Omega_e} \int_{\Omega_e} B_2B_4 d\Omega, \quad B_{10} = B_3B_4 - \frac{1}{\Omega_e} \int_{\Omega_e} B_3B_4 d\Omega
 \end{aligned} \tag{3.7}$$

The discontinuous Galerkin formulation then leads to the following 10 equations

$$\begin{aligned}
 \frac{d}{dt} \int_{\Omega_e} \tilde{\mathbf{U}} d\Omega + \int_{\Gamma_e} H_k(\mathbf{U}_h^L, \mathbf{U}_h^R, \mathbf{n}_k) d\Gamma &= 0 \\
 M_{9 \times 9} \frac{d}{dt} \left(\frac{\partial \mathbf{U}}{\partial x} \Big|_c, \frac{\partial \mathbf{U}}{\partial y} \Big|_c, \frac{\partial \mathbf{U}}{\partial z} \Big|_c, \frac{\partial^2 \mathbf{U}}{\partial x^2} \Big|_c, \frac{\partial^2 \mathbf{U}}{\partial y^2} \Big|_c, \frac{\partial^2 \mathbf{U}}{\partial z^2} \Big|_c, \frac{\partial^2 \mathbf{U}}{\partial x \partial y} \Big|_c, \frac{\partial^2 \mathbf{U}}{\partial x \partial z} \Big|_c, \frac{\partial^2 \mathbf{U}}{\partial y \partial z} \Big|_c \right)^T + R_{9 \times 1} &= 0
 \end{aligned} \tag{3.8}$$

Note that in this formulation, equations for the cell-averaged variables are decoupled from equations for their derivatives due to the judicious choice of the basis functions and the fact that

$$\int_{\Omega_e} B_i B_j d\Omega = 0 \quad 2 \leq i \leq 10 \tag{3.9}$$

In the implementation of this DG method, the basis functions are actually normalized in order to improve the conditioning of the system matrix (3.3) as follows:

$$\begin{aligned}
 B_1 &= 1, \quad B_2 = \frac{x - x_c}{\Delta x}, \quad B_3 = \frac{y - y_c}{\Delta y}, \quad B_4 = \frac{z - z_c}{\Delta z} \\
 B_5 &= \frac{B_2^2}{2} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{B_2^2}{2} d\Omega, \quad B_6 = \frac{B_3^2}{2} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{B_3^2}{2} d\Omega, \quad B_7 = \frac{B_4^2}{2} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{B_4^2}{2} d\Omega, \\
 B_8 &= B_2B_3 - \frac{1}{\Omega_e} \int_{\Omega_e} B_2B_3 d\Omega, \quad B_9 = B_2B_4 - \frac{1}{\Omega_e} \int_{\Omega_e} B_2B_4 d\Omega, \quad B_{10} = B_3B_4 - \frac{1}{\Omega_e} \int_{\Omega_e} B_3B_4 d\Omega,
 \end{aligned} \tag{3.10}$$

where $\Delta x = 0.5(x_{\max} - x_{\min})$, $\Delta y = 0.5(y_{\max} - y_{\min})$, and $\Delta z = 0.5(z_{\max} - z_{\min})$, and x_{\max} , x_{\min} , y_{\max} , y_{\min} , z_{\max} , and z_{\min} are the maximum and minimum coordinates in the cell Ω_e in x -, y -, and z -directions, respectively. A quadratic polynomial solution can then be rewritten as

$$\mathbf{U}_h = \tilde{\mathbf{U}} B_1 + \mathbf{U}_x B_2 + \mathbf{U}_y B_3 + \mathbf{U}_z B_4 + \mathbf{U}_{xx} B_5 + \mathbf{U}_{yy} B_6 + \mathbf{U}_{zz} B_7 + \mathbf{U}_{xy} B_8 + \mathbf{U}_{xz} B_9 + \mathbf{U}_{yz} B_{10} \tag{3.11}$$

where

$$\begin{aligned}
 \mathbf{U}_x &= \frac{\partial \mathbf{U}}{\partial x} \Big|_c \Delta x, \quad \mathbf{U}_y = \frac{\partial \mathbf{U}}{\partial y} \Big|_c \Delta y, \quad \mathbf{U}_z = \frac{\partial \mathbf{U}}{\partial z} \Big|_c \Delta z, \quad \mathbf{U}_{xx} = \frac{\partial^2 \mathbf{U}}{\partial x^2} \Big|_c \Delta x^2, \quad \mathbf{U}_{yy} = \frac{\partial^2 \mathbf{U}}{\partial y^2} \Big|_c \Delta y^2, \\
 \mathbf{U}_{zz} &= \frac{\partial^2 \mathbf{U}}{\partial z^2} \Big|_c \Delta z^2, \quad \mathbf{U}_{xy} = \frac{\partial^2 \mathbf{U}}{\partial x \partial y} \Big|_c \Delta x \Delta y, \quad \mathbf{U}_{xz} = \frac{\partial^2 \mathbf{U}}{\partial x \partial z} \Big|_c \Delta x \Delta z, \quad \mathbf{U}_{yz} = \frac{\partial^2 \mathbf{U}}{\partial y \partial z} \Big|_c \Delta y \Delta z
 \end{aligned} \tag{3.12}$$

This formulation belongs to the so-called modal discontinuous Galerkin method and has a number of attractive, distinct, and useful features. First, cell-averaged variables and their derivatives are handily available in this formulation. This makes the implementation of both in-cell and inter-cell reconstruction schemes straightforward and simple [26–28,30,31]. Secondly, the Taylor basis is hierarchic. This greatly facilitates implementation of p -multigrid methods [16,17] and p -refinement. Thirdly, the same basis functions are used for any shapes of elements: tetrahedron, pyramid, prism, and hexahedron. This makes the implementation of DGM on arbitrary meshes straightforward.

3.2. Reconstructed discontinuous Galerkin methods

In comparison with reconstructed FV methods, the DGM have a significant drawback in that they require more degrees of freedom, additional domain integration, and more Gauss quadrature points for the boundary integration, and therefore more computational costs and storage requirements. On the one hand, the reconstruction methods that FV methods use to achieve higher-order accuracy are relatively inexpensive but less accurate and robust. On the other hand, the DGM that can be viewed as a different way to extend a FV method to higher orders are accurate and robust but costly. It is only natural and tempting to combine the efficiency of the reconstruction methods and the accuracy of the DG methods. This idea was originally introduced by Dumbser et al. in the frame of $P_n P_m$ scheme [18–20], termed RDG ($P_n P_m$) in this paper, where P_n indicates that a piecewise polynomial of degree of n is used to represent a DG solution, and P_m represents a reconstructed polynomial solution of degree of m ($m \geq n$) that is used to compute the fluxes and source terms. The beauty of RDG ($P_n P_m$) schemes is that they provide a unified formulation for both finite volume and DG methods, and contain both classical finite volume and standard DG methods as two special cases of RDG ($P_n P_m$) schemes, and thus allow for a direct efficiency

comparison. When $n = 0$, i.e., a piecewise constant polynomial is used to represent a numerical solution, RDG (P_0P_m) is nothing but classical high order finite volume schemes, where a polynomial solution of degree m ($m \geq 1$) is reconstructed from a piecewise constant solution. When $m = n$, the reconstruction reduces to the identity operator, and RDG (P_nP_m) scheme yields a standard DG method. Clearly, an accurate and efficient reconstruction is the key ingredient in extending the underlying DG method to higher order accuracy. Our discussion in this work is mainly focused on a third order RDG (P_1P_2) method, as the benefits of higher-order ($>3rd$) methods diminish dramatically for aerodynamic applications. Nevertheless, its extension to higher order DG methods is straightforward, as demonstrated by Dumbser et al. in Ref. [18]. The RDG (P_1P_2) method is based on a hierarchical Hermite WENO reconstruction and 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 ensuring the nonlinear stability of the RDG method. Similar to moment limiters, the hierarchical reconstruction methods [35] reconstruct the derivatives in a hierarchical manner. In the case of the RDG (P_1P_2) method, the second derivatives (curvatures) are first reconstructed and the first derivatives (gradients) are then reconstructed, which are describes in the next two sub-sections.

3.2.1. WENO reconstruction at P_2 : WENO (P_1P_2)

The reconstruction of the second derivatives consists of two steps: a quadratic polynomial solution (P_2) is first reconstructed using a least-squares method from the underlying linear polynomial (P_1) discontinuous Galerkin solution, and the final quadratic polynomial solution is then obtained using a WENO reconstruction, which is necessary to ensure the linear stability of the RDG method [36]. The resulting RDG method is referred to as WENO (P_1P_2) in this paper, where the quadratic polynomial solution is obtained from the underlying linear DG solution via a WENO reconstruction.

3.2.1.1. Least-squares reconstruction. In the case of DG (P_1) method, a linear polynomial solution \mathbf{U}_i in any cell i is expressed as

$$\mathbf{U}_i = \tilde{\mathbf{U}}_i + \mathbf{U}_{xi}B_2 + \mathbf{U}_{yi}B_3 + \mathbf{U}_{zi}B_4 \tag{3.13}$$

A quadratic polynomial solution \mathbf{U}_i^R can be reconstructed using the underlying linear polynomial DG solution in the neighboring cells as follows:

$$\mathbf{U}_i^R = \tilde{\mathbf{U}}_i^R + \mathbf{U}_{xi}^R B_2 + \mathbf{U}_{yi}^R B_3 + \mathbf{U}_{zi}^R B_4 + \mathbf{U}_{xxi}^R B_5 + \mathbf{U}_{yyi}^R B_6 + \mathbf{U}_{zzi}^R B_7 + \mathbf{U}_{xyi}^R B_8 + \mathbf{U}_{xzi}^R B_9 + \mathbf{U}_{yzi}^R B_{10} \tag{3.14}$$

In order to maintain the compactness of the DG methods, the reconstruction is required to involve only von Neumann neighborhood, i.e., the adjacent cells that share a face with the cell i under consideration. There are 10 degrees of freedom, and therefore 10 unknowns must be determined. The first four unknowns can be trivially obtained, by requiring the consistency of the RDG with the underlying DG: (1) The reconstruction scheme must be conservative, and (2) The values of the reconstructed first derivatives are equal to the ones of the first derivatives of the underlying DG solution at the centroid i . Due to the judicious choice of Taylor basis in our DG formulation, these four degrees of freedom simply coincide with the ones from the underlying DG solution, i.e.,

$$\tilde{\mathbf{U}}_i^R = \tilde{\mathbf{U}}_i, \mathbf{U}_{xi}^R = \mathbf{U}_{xi}, \mathbf{U}_{yi}^R = \mathbf{U}_{yi}, \mathbf{U}_{zi}^R = \mathbf{U}_{zi} \tag{3.15}$$

As a result, only six second derivatives need to be determined. This can be accomplished by requiring that the point-wise values and first derivatives of the reconstructed solution are equal to these of the underlying DG solution at the cell centers for all the adjacent face neighboring cells. Considering an adjacent neighboring cell j , one obtains

$$\begin{aligned} \mathbf{U}_j &= \tilde{\mathbf{U}}_j + \mathbf{U}_{xi}B_2 + \mathbf{U}_{yi}B_3 + \mathbf{U}_{zi}B_4 + \mathbf{U}_{xxi}B_5 + \mathbf{U}_{yyi}B_6 + \mathbf{U}_{zzi}B_7 + \mathbf{U}_{xyi}B_8 + \mathbf{U}_{xzi}B_9 + \mathbf{U}_{yzi}B_{10} \\ \frac{\partial \mathbf{U}}{\partial x} \Big|_j &= \mathbf{U}_{xi} \frac{1}{\Delta x_i} + \mathbf{U}_{xxi}^R \frac{B_2}{\Delta x_i} + \mathbf{U}_{xyi}^R \frac{B_3}{\Delta x_i} + \mathbf{U}_{xzi}^R \frac{B_4}{\Delta x_i} \\ \frac{\partial \mathbf{U}}{\partial y} \Big|_j &= \mathbf{U}_{yi} \frac{1}{\Delta y_i} + \mathbf{U}_{yyi}^R \frac{B_3}{\Delta y_i} + \mathbf{U}_{xyi}^R \frac{B_2}{\Delta y_i} + \mathbf{U}_{yzi}^R \frac{B_4}{\Delta y_i} \\ \frac{\partial \mathbf{U}}{\partial z} \Big|_j &= \mathbf{U}_{zi} \frac{1}{\Delta z_i} + \mathbf{U}_{zzi}^R \frac{B_4}{\Delta z_i} + \mathbf{U}_{xzi}^R \frac{B_2}{\Delta z_i} + \mathbf{U}_{yzi}^R \frac{B_3}{\Delta z_i} \end{aligned} \tag{3.16}$$

where the basis functions B are evaluated at the center of cell j , i.e., $B = \mathbf{B}(x_j, y_j, z_j)$. This can be written in a matrix form as follows:

$$\begin{pmatrix} B_5^j & B_6^j & B_7^j & B_8^j & B_9^j & B_{10}^j \\ B_2^j & 0 & 0 & B_3^j & B_4^j & 0 \\ 0 & B_3^j & 0 & B_2^j & 0 & B_4^j \\ 0 & 0 & B_4^j & 0 & B_2^j & B_3^j \end{pmatrix} \begin{pmatrix} \mathbf{U}_{xxi}^R \\ \mathbf{U}_{yyi}^R \\ \mathbf{U}_{zzi}^R \\ \mathbf{U}_{xyi}^R \\ \mathbf{U}_{xzi}^R \\ \mathbf{U}_{yzi}^R \end{pmatrix} = \begin{pmatrix} \mathbf{U}_j - (\mathbf{U}_i B_1^j + \mathbf{U}_{xi} B_2^j + \mathbf{U}_{yi} B_3^j + \mathbf{U}_{zi} B_4^j) \\ \frac{\Delta x_i}{\Delta x_j} \mathbf{U}_{xj} - \mathbf{U}_{xi} \\ \frac{\Delta y_i}{\Delta y_j} \mathbf{U}_{yj} - \mathbf{U}_{yi} \\ \frac{\Delta z_i}{\Delta z_j} \mathbf{U}_{zj} - \mathbf{U}_{zi} \end{pmatrix} = \begin{pmatrix} \mathbf{R}_1^j \\ \mathbf{R}_2^j \\ \mathbf{R}_3^j \\ \mathbf{R}_4^j \end{pmatrix} \tag{3.17}$$

where \mathbf{R} is used to represent the right-hand-side for simplicity. Similar equations can be written for all cells connected to the cell i with a common face, which leads to a non-square matrix. The number of face-neighboring cells for a tetrahedral is four. Consequently, the size of the resulting non-square matrix is 16×6 . In the present work, this over-determined linear system of 16 equations for 6 unknowns is solved in the least-squares sense using both normal equation approach and the QR decomposition to obtain the second derivatives of the reconstructed quadratic polynomial solution. One can easily verify that this least-squares reconstruction satisfies the so-called 2-exactness, i.e., it can reconstruct a quadratic polynomial function exactly.

3.2.1.2. WENO reconstruction. This least-squares reconstructed discontinuous Galerkin method: RDG (P_1P_2) has been successfully used to solve the 2D compressible Euler equations for smooth flows on arbitrary grids [26–28,30,31] and is able to achieve the designed third order of accuracy and significantly improve the accuracy of the underlying second-order DG method. However, when extended to solve the 3D compressible Euler equations on tetrahedral grids, this RDG method suffers from the so-called linear instability, that is also observed in the second-order cell-centered finite volume methods, i.e., RDG (P_0P_1) [33]. This linear instability is attributed to the fact that the reconstruction stencils only involve von Neumann neighborhood, i.e., adjacent face-neighboring cells [33]. The linear stability can be achieved using extended stencils, which will unfortunately sacrifice the compactness of the underlying DG methods. Furthermore, such a linear reconstruction-based DG method cannot maintain the non-linear instability, leading to non-physical oscillations in the vicinity of strong discontinuities. Alternatively, ENO/WENO can be used to reconstruct a higher-order polynomial solution, which can not only enhance the order of accuracy of the underlying DG method but also achieve both linear and non-linear stability. Specifically, the WENO scheme introduced by Dumber et al. [40] is adopted in this work, where an entire quadratic polynomial solution on cell i is obtained using a nonlinear WENO reconstruction as a convex combination of the least-squares reconstructed second derivatives at the cell itself and its four face-neighboring cells,

$$\left. \frac{\partial^2 \mathbf{U}}{\partial x_m \partial x_n} \right|_i^{\text{WENO}} = \sum_{k=1}^5 w_k \left. \frac{\partial^2 \mathbf{U}}{\partial x_m \partial x_n} \right|_k \tag{3.18}$$

where the normalized nonlinear weights w_k are computed as

$$w_k = \frac{\tilde{w}_k}{\sum_{i=1}^5 \tilde{w}_k} \tag{3.19}$$

The non-normalized nonlinear weights \tilde{w}_i are functions of the linear weights λ_i and the so-called oscillation indicator o_i

$$\tilde{w}_i = \frac{\lambda_i}{(\varepsilon + o_i)^\gamma} \tag{3.20}$$

where ε is a small positive number used to avoid division by zero, and γ an integer parameter to control how fast the non-linear weights decay for non-smooth stencils. The oscillation indicator for the reconstructed second order polynomials is simply defined as

$$o_k = \left[\left(\left. \frac{\partial^2 \mathbf{U}}{\partial x_m \partial x_n} \right|_k \right)^2 \right]^{1/2} \tag{3.21}$$

The linear weights λ_i can be chosen to balance the accuracy and the non-oscillatory property of the RDG method. Note that the least-squares reconstructed polynomial at the cell itself serves as the central stencil and the least-squares reconstructed polynomials on its four face-neighboring cells act as biased stencils in this WENO reconstruction. This reconstructed quadratic polynomial solution is then used to compute the domain and boundary integrals of the underlying DG (P_1) method in Eq. (3.3). As demonstrated in Ref. [36], the resulting WENO (P_1P_2) method is able to achieve the designed third order of accuracy, maintain the linear stability, and significantly improve the accuracy of the underlying second-order DG method without significant increase in computing costs and storage requirements. Note that this RDG method is not compact anymore, as neighbor's neighbors are used in the solution update. However, the stencil used in the reconstruction is compact, involving only von Neumann neighbors. Consequently, the resultant RDG method can be implemented in a compact manner.

3.2.2. WENO reconstruction at P_1 : HWENO (P_1P_2)

Although the WENO (P_1P_2) method does not introduce any new oscillatory behavior for the reconstructed curvature terms (second derivatives) due to the WENO reconstruction, it cannot remove inherent oscillations in the underlying DG (P_1) solutions. Consequently, the WENO (P_1P_2) method still suffers from the non-linear instability for flows with strong discontinuities. In order to eliminate non-physical oscillations in the vicinity of strong discontinuities and thus maintain the non-linear stability, the first derivatives need to be reconstructed using a WENO reconstruction. The resulting reconstructed discontinuous Galerkin method based on this Hierarchical WENO reconstruction is termed as HWENO (P_1P_2) in this paper, where a hierarchical reconstruction (successively from high order to low order) strategy [35] is adopted.

The WENO reconstruction for the first derivatives is based on the reconstructed quadratic polynomial solutions of the flow variables for each cell in the mesh. The stencils are only chosen in the von Neumann neighborhood. More precisely, for a cell i , the following four stencils (i, j_1, j_2, j_3) , (i, j_1, j_2, j_4) , i.e., (j_1, j_3, j_4) , and (i, j_2, j_3, j_4) , where j_1, j_2, j_3 , and j_4 designate the four adjacent face-neighboring cells of the cell i are chosen to construct a Lagrange polynomial such that

$$\mathbf{U}_j = \tilde{\mathbf{U}}_i + \mathbf{U}_{xi}^R B_2 + \mathbf{U}_{yi}^R B_3 + \mathbf{U}_{zi}^R B_4 + \mathbf{U}_{xxi}^R B_5 + \mathbf{U}_{yyi}^R B_6 + \mathbf{U}_{zzi}^R B_7 + \mathbf{U}_{xyi}^R B_8 + \mathbf{U}_{xzi}^R B_9 + \mathbf{U}_{yzi}^R B_{10} \tag{3.22}$$

where \mathbf{U}_j refers to the pointwise value of the reconstructed polynomial solution at centroid of cell j and the basis functions B are evaluated at the center of cell j , i.e., $B = \mathbf{B}(x_j, y_j, z_j)$. In addition, the following four stencils (i, j_1) , (i, j_2) , (i, j_3) , and (i, j_4) are chosen to construct a Hermite polynomial such that

$$\begin{aligned} \left. \frac{\partial \mathbf{U}}{\partial x} \right|_j &= \mathbf{U}_{xi}^R \frac{1}{\Delta x_i} + \mathbf{U}_{xxi}^R \frac{B_2}{\Delta x_i} + \mathbf{U}_{xyi}^R \frac{B_3}{\Delta x_i} + \mathbf{U}_{xzi}^R \frac{B_4}{\Delta x_i} \\ \left. \frac{\partial \mathbf{U}}{\partial y} \right|_j &= \mathbf{U}_{yi}^R \frac{1}{\Delta y_i} + \mathbf{U}_{yyi}^R \frac{B_3}{\Delta y_i} + \mathbf{U}_{xyi}^R \frac{B_2}{\Delta y_i} + \mathbf{U}_{yzi}^R \frac{B_4}{\Delta y_i} \\ \left. \frac{\partial \mathbf{U}}{\partial z} \right|_j &= \mathbf{U}_{zi}^R \frac{1}{\Delta z_i} + \mathbf{U}_{zzi}^R \frac{B_4}{\Delta z_i} + \mathbf{U}_{xzi}^R \frac{B_2}{\Delta z_i} + \mathbf{U}_{yzi}^R \frac{B_3}{\Delta z_i} \end{aligned} \tag{3.23}$$

These eight reconstructed gradients $(\mathbf{U}_{xi}^R, \mathbf{U}_{yi}^R, \text{ and } \mathbf{U}_{zi}^R)$ serving as the biased stencils and the gradient from the DG solution itself at cell i (\mathbf{U}_{xi} , \mathbf{U}_{yi} , and \mathbf{U}_{zi}) acting as the central stencil are used to modify the first derivatives based on the WENO reconstruction as a convex combination of these nine derivatives,

$$\left. \frac{\partial \mathbf{U}}{\partial x_m} \right|_i^{\text{WENO}} = \sum_{k=1}^9 w_k \left. \frac{\partial \mathbf{U}}{\partial x_m} \right|_k \tag{3.24}$$

where the normalized nonlinear weights w_k are computed as

$$w_k = \frac{\tilde{w}_k}{\sum_{i=1}^9 \tilde{w}_k} \tag{3.25}$$

The non-normalized nonlinear weights \tilde{w}_i are functions of the linear weights λ_i and the so-called oscillation indicator o_i

$$\tilde{w}_i = \frac{\lambda_i}{(\varepsilon + o_i)^\gamma} \tag{3.26}$$

where ε is a small positive number used to avoid division by zero, and γ an integer parameter to control how fast the non-linear weights decay for non-smooth stencils. The oscillation indicator is simply defined as

$$o_k = \left[\left(\left. \frac{\partial \mathbf{U}}{\partial x_m} \right|_k \right)^2 \right]^{1/2} \tag{3.27}$$

The present choice of stencils is symmetric, and compact, as von Neumann neighbors are only involved in the reconstruction. This means that no additional data structure is required for our HWENO ($P_1 P_2$) method. Note that this WENO reconstruction at P_1 is the extension of a HWENO limiter developed for the DG (P_1) by the authors in Ref. [15]. From the perspective of both computational cost and solution accuracy, the above WENO reconstruction on P_1 should only be used in the regions where strong discontinuities exist. This can be accomplished using the so-called discontinuity detectors, which are helpful to distinguish regions where solutions are smooth and discontinuous. The beauty of this WENO reconstruction is that in case that the reconstruction is mistakenly applied in the smooth cells, the uniform high-order accuracy can still be maintained, unlike the slope limiters, which, when applied near smooth extrema, will have a profoundly adverse impact on solution in the smooth region, leading to the loss of the original high-order accuracy. This remarkable feature of the WENO reconstruction in turn alleviates the burden on the discontinuity detectors, as no discontinuity detectors can really either in theory or in practice make a distinction between a stagnation point and a shock wave, as flow gradients near the stagnation point are even larger than the ones near the shock wave in some cases. All numerical experiments presented in the next section are performed by applying the P_1 reconstruction everywhere in an effort to ensure that the computational results are not affected by a shock detector, and to demonstrate the superior properties of the designed HWENO ($P_1 P_2$) method.

4. Numerical examples

The hierarchical WENO reconstruction method has been implemented in a well-tested DG code [13–17] to solve a variety of the compressible flow problems on tetrahedral grids, where a fast implicit p -multigrid method [16,17] is used to obtain steady state solutions considered in this paper. A few examples are presented in this section to demonstrate that the developed HWENO ($P_1 P_2$) method is able to maintain the non-linear stability and achieve the designed third order of accuracy. The first two test cases are chosen to demonstrate that the developed HWENO ($P_1 P_2$) method is able to achieve the designed third-order of convergence for smooth flows. The next two test cases are used to assess the non-oscillatory property of

the HWENO (P₁P₂) method for flows with strong discontinuities. The final test case is presented to illustrate the applicability of the HWENO (P₁P₂) method to solve problems of scientific and industrial interests for complex configurations. All the grids in our numerical experiments are generated using an advancing front method described in Refs. [41,42]. In the grid convergence study of the first two test cases, a sequence of the three successively refined tetrahedral grids is used, where the cell size is halved between consecutive meshes. The number of elements on a successively refined mesh is not exactly eight times the coarse mesh's elements due to the nature of unstructured grid generation. The length scale, characterizing the cell size of an unstructured grid, is defined as $1/\sqrt[3]{nDOFs}$, where $nDOFs$ is the total number of degrees of freedom. Since the analytical solutions are unknown, the following L²-norm of the entropy production is used as the error measurement

$$\|\varepsilon\|_{L^2(\Omega)} = \sqrt{\int_{\Omega} \varepsilon^2 d\Omega}$$

where the entropy production ε defined as

$$\varepsilon = \frac{S - S_{\infty}}{S_{\infty}} = \frac{p}{p_{\infty}} \left(\frac{\rho_{\infty}}{\rho} \right)^{\gamma} - 1$$

Note that the entropy production, where the entropy is defined as $S = p/\rho^{\gamma}$, is a very good criterion to measure accuracy of the numerical solutions, since the flow under consideration is smooth and therefore isentropic.

4.1. Subsonic flow through a channel with a smooth bump

The problem under consideration is a subsonic internal flow inside a 3D channel with a smooth bump on the lower surface. The height, width, and length of the channel are 0.8, 0.8, and 3, respectively. The shape of the lower wall is defined by the function $0.0625 \exp(-25 \times x^2)$ from $x = -1.5$ to $x = 1.5$. The inflow condition is prescribed at a Mach number of 0.5, and an angle of attack of 0°. Fig. 1 shows the three successively refined tetrahedral grids used in this the grid convergence study. The numbers of elements, points, and boundary points for the coarse, medium, and fine grids are (889,254,171), (6986,1555,691), and (55703,10822,2711), respectively. Numerical solutions to this problem are computed using the RDG (P₁P₁), WENO (P₁P₂), and HWENO (P₁P₂) methods on these three grids to obtain a quantitative measurement of the order of accuracy and discretization errors. Fig. 2 illustrates the computed velocity contours in the flow field obtained by the HWENO (P₁P₂) methods on these three grids. The errors and the convergence rates for the three methods are reported in Table 1. It shows the mesh size, the L²-error of the error function, and the order of convergence. Considering that this is a 3D simulation of a 2D problem, and unstructured tetrahedral grids are not symmetric by nature, thus causing error in the z-direction, the second order RDG (P₁P₁) method can be viewed to offer the designed second order of accuracy. Both WENO (P₁P₂) and HWENO (P₁P₂) methods are able to deliver the designed third order of convergence, adding one order of accuracy to the underlying DG (P₁) method. As expected, the WENO reconstruction on P₁ increases slightly the absolute error. However, it

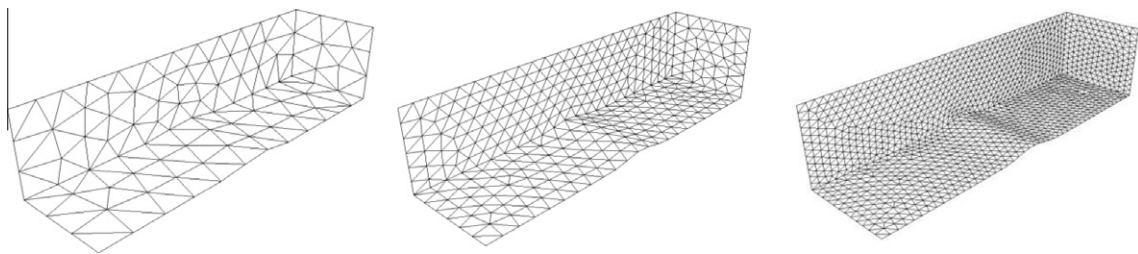


Fig. 1. A sequence of three successively globally refined unstructured meshes used for computing a subsonic flow in a channel with a smooth bump.

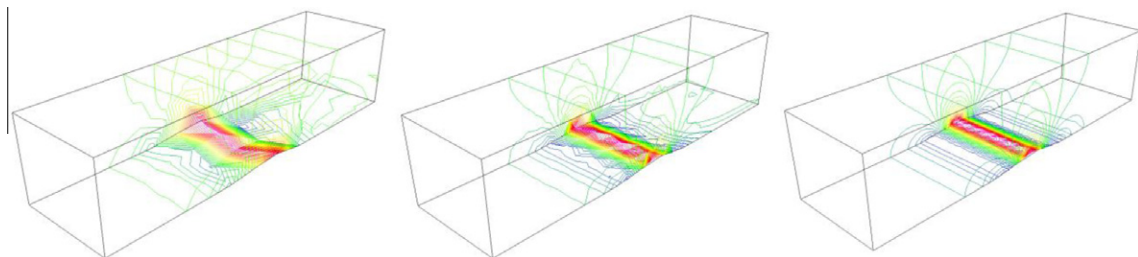


Fig. 2. Computed velocity contours in the flow field obtained by the HWENO (P₁P₂) on a sequence of three successively globally refined unstructured grids for a subsonic flow through a channel with a bump on the lower surface at $M_{\infty} = 0.5$.

Table 1

L^2 -error and order of convergence for the RDG (P_1P_1), WENO (P_1P_2), and HWENO (P_1P_2) methods.

Length scale	RDG (P_1P_1)		WENO (P_1P_2)		HWENO (P_1P_2)	
	L^2 -error	Order	L^2 -error	Order	L^2 -error	Order
6.552E-2	2.438E-3		2.183E-3		2.220E-3	
3.295E-2	7.356E-4	1.744	2.794E-4	2.992	2.851E-4	2.987
1.650E-2	1.807E-4	2.032	4.539E-05	2.626	4.565E-5	2.647

does not destroy the order of accuracy. This example demonstrates that the HWENO (P_1P_2) method can indeed achieve a third-order rate of convergence for this smooth internal flow.

4.2. Subsonic flow past a sphere

A subsonic flow past a sphere at a Mach number of $M_\infty = 0.5$ is considered in this test case. A sequence of the three successively refined tetrahedral grids used in this grid convergence study is shown in Fig. 3. The numbers of elements, points, and boundary points for the coarse, medium, and fine grids are (535,167,124), (2426,598,322), and (16467,3425,1188), respectively. Like the previous test case, the computations are performed on these three grids using the RDG (P_1P_1), WENO (P_1P_2), and HWENO (P_1P_2) methods. Fig. 4 illustrates the computed velocity contours in the flow field obtained by the HWENO (P_1P_2) method on these three grids. The errors and the orders of accuracy for the three methods are reported in Table 2. All three methods achieve higher than the expected rates of convergence, being 2.36, 3.55, and 3.50 respectively. One can observe again that the HWENO (P_1P_2) keeps the designed order of accuracy, although the WENO reconstruction on P_1 increases slightly the absolute errors of the WENO (P_1P_2) method. This example demonstrates that the HWENO (P_1P_2) method is able to deliver a third-order rate of convergence for smooth external flows.

4.3. Transonic flow past an ONERA M6 wing

A transonic flow over the ONERA M6 wing at a Mach number of $M_\infty = 0.84$ and an attack of angle of $\alpha = 3.06^\circ$ is considered in this example. This standard test case is chosen to demonstrate that the developed HWENO (P_1P_2) method can effectively

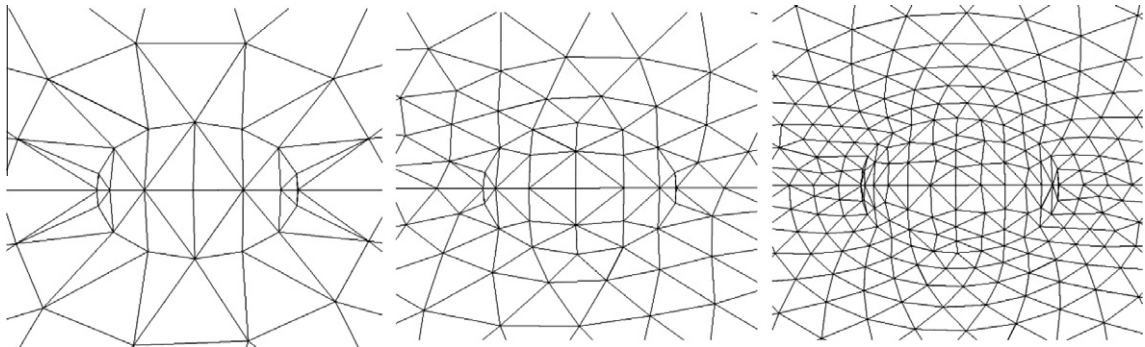


Fig. 3. A series of four successively globally refined tetrahedral meshes for computing a subsonic flow past a sphere at $M_\infty = 0.5$.

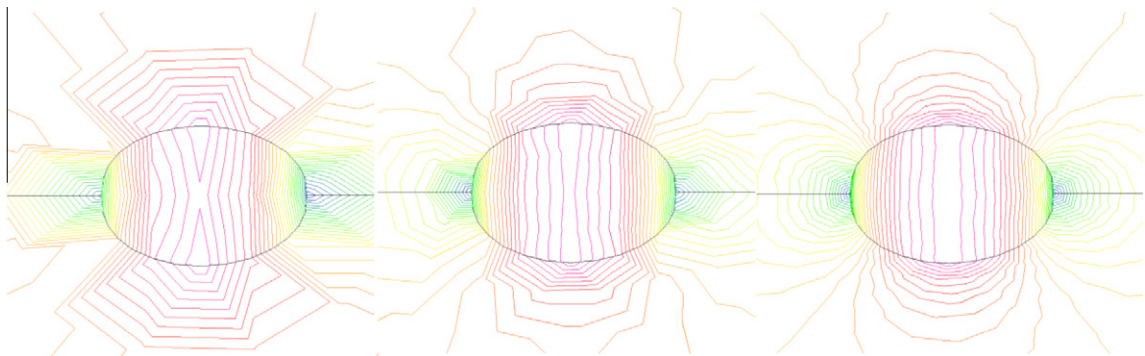


Fig. 4. Computed velocity contours in the flow field obtained by the HWENO (P_1P_2) on a sequence of three successively globally refined unstructured grids for a subsonic flow past a sphere at $M_\infty = 0.5$.

Table 2

L^2 -error and order of convergence for the RDG (P_1P_1), WENO (P_1P_2), and HWENO (P_1P_2) methods.

Length scale	RDG (P_1P_1)		WENO (P_1P_2)		HWENO (P_1P_2)	
	L^2 -error	Order	L^2 -error	Order	L^2 -error	Order
7.760E-2	1.783E-2		1.052E-2		1.117E-2	
4.688E-2	5.010E-3	2.519	1.317E-3	4.124	1.503E-3	3.980
2.476E-2	1.232E-3	2.198	1.978E-4	2.964	2.201E-4	3.009

suppress spurious oscillations in the vicinity of strong discontinuities and provide high accuracy solution in comparison with the second-order finite volume method, WENO (P_0P_1) [37]. The flow solutions are presented using the WENO (P_0P_1) method on a fine mesh and the HWENO (P_1P_2) method on a coarse mesh, respectively. The coarse mesh contains 95,266 elements, 18,806 points, and 5,287 boundary points, and the fine one 593,169 elements, 110,282 points, and 19,887 boundary points. Fig. 5 shows the computed pressure contours on the upper wing surface obtained by these two solutions, respectively. The computed pressure coefficients obtained by these two solutions are compared with experimental data [38] at six span-wise stations in Fig. 6. The computed pressure coefficients for the HWENO (P_1P_2) solution are plotted by a straight line connecting the two nodes of each triangle that intersect with the desired cut plane, thus truly reflecting the discontinuous nature of a DG solution. The results obtained by both RDG methods compare closely with the experimental data, except at the root stations, due to the lack of viscous effects. The leading edge suction peak is extremely well captured by both solutions in spite of the coarseness of the grids used in both solutions. However, one can observe that the third-order HWENO (P_1P_2) solution on the coarse mesh is more accurate than the second-order WENO (P_0P_1) solution on a globally refined grid, which is especially evident by judging the entropy production on the surface of the wing at these six span-wise stations as shown in Fig. 7. Note that the entropy production corresponds directly to the error of the numerical methods, as it should be zero everywhere with exception of shock waves where it should increase. The shocks are virtually captured within one cell without any oscillations in the HWENO (P_1P_2) solution, clearly demonstrating the high accuracy and non-oscillatory property of our HWENO (P_1P_2) method.

4.4. Transonic flow past a wing/pylon/finned-store configuration

A transonic flow past a wing/pylon/finned-store configuration reported in Ref. [39] is computed in this test case using HWENO (P_1P_2) method. The configuration consists of a clipped delta wing with a 45° sweep comprised from a constant NACA 64010 symmetric airfoil section. The wing has a root chord of 15 in. a semi-span of 13 in. and a taper ratio of 0.134. The pylon is located at the mid-span station and has a cross-section characterized by a flat plate closed at the leading and trailing edges by a symmetrical ogive shape. The width of the pylon is 0.294 in. The four fins on the store are defined by a constant NACA 0008 airfoil section with a leading-edge sweep of 45° and a truncated tip. The mesh used in the computation contains 319,134 elements, 61,075 grid points, and 14,375 boundary points. The flow solution is presented at a Mach number of 0.95 and an angle of attack of 0° . Fig. 8 shows the computed pressure contours on the upper and lower wing surface, respectively. The computed pressure coefficient distributions are compared with experimental data at two span-wise stations in Fig. 9. The comparison with experimental data is excellent on both upper and lower surface up to 70% chord. As expected from the Euler solution, the computation predicts a shock location that is downstream of that measured by the experiment

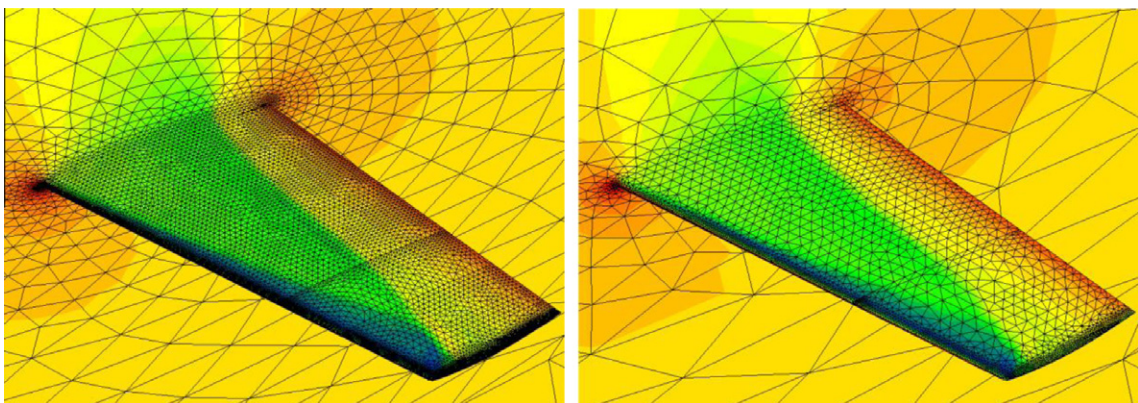


Fig. 5. Computed pressure contours on the unstructured surface mesh obtained by the WENO (P_0P_1) solution on the fine mesh (left, $n_{elem} = 593,169$, $n_{poin} = 110,282$, $n_{boun} = 19,887$), and the HWENO (P_1P_2) solution on the coarse mesh (right, $n_{elem} = 95,266$, $n_{poin} = 18,806$, $n_{boun} = 5287$) for a transonic flow past a M6 wing at $M_\infty = 0.84$, $\alpha = 3.06^\circ$.

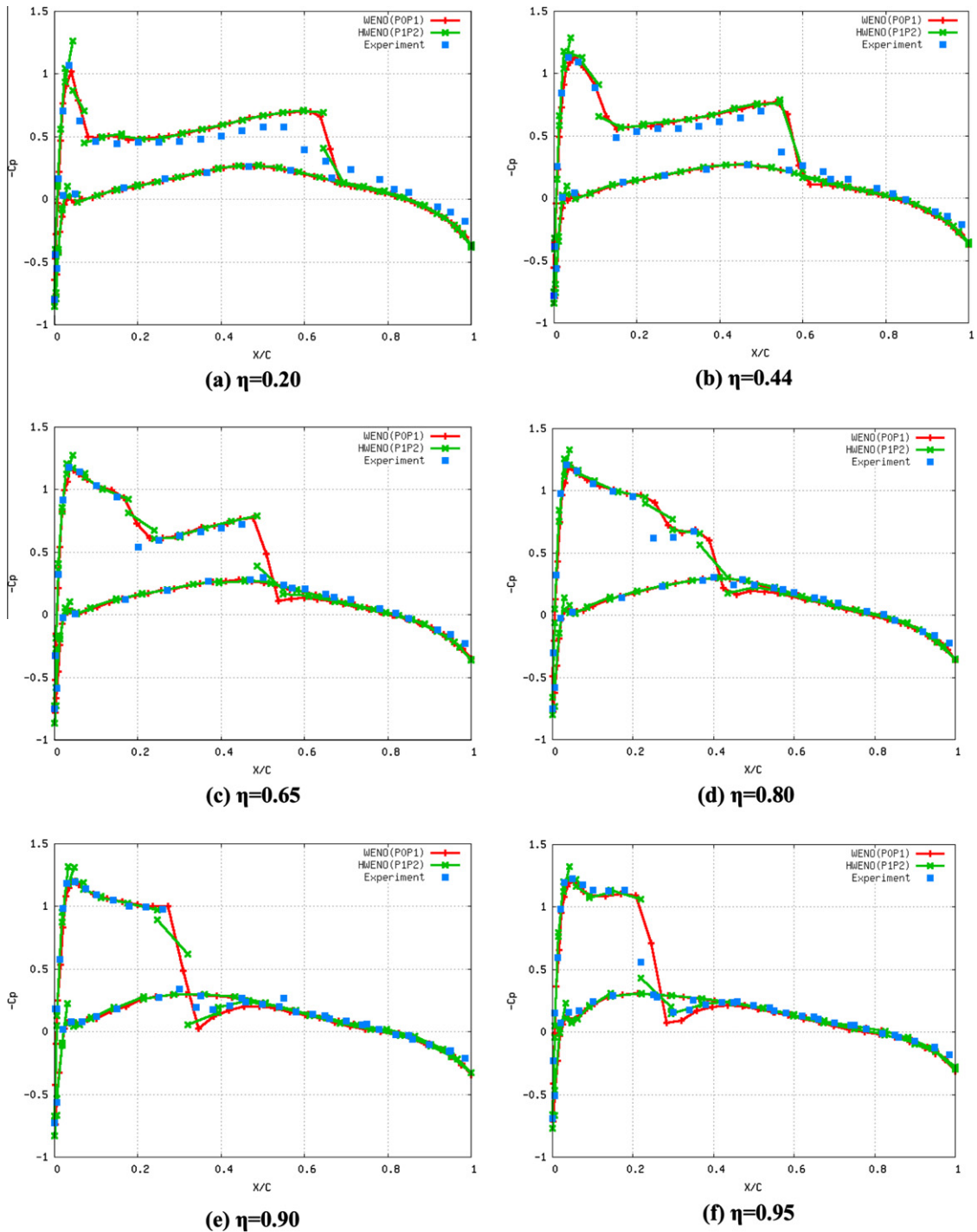


Fig. 6. Comparison of the computed pressure coefficient distributions obtained by WENO (P_0P_1) and HWENO (P_1P_2) solutions with experimental data at six span-wise locations for a transonic flow past the ONERA M6 wing at $M_\infty = 0.84$, and $\alpha = 3.06^\circ$.

due to the lack of viscous effect. Again, our third-order HWENO (P_1P_2) method captures the shock waves very sharply within one cell without any visible under- and over-shoots.

4.5. Transonic flows past a Boeing 747 aircraft

Finally, a transonic flow past a complete Boeing 747 aircraft is presented in this test case. The 747 configuration includes the fuselage, wing, horizontal and vertical tails, under-wing pylons, and flow-through engine nacelle. The mesh used in the

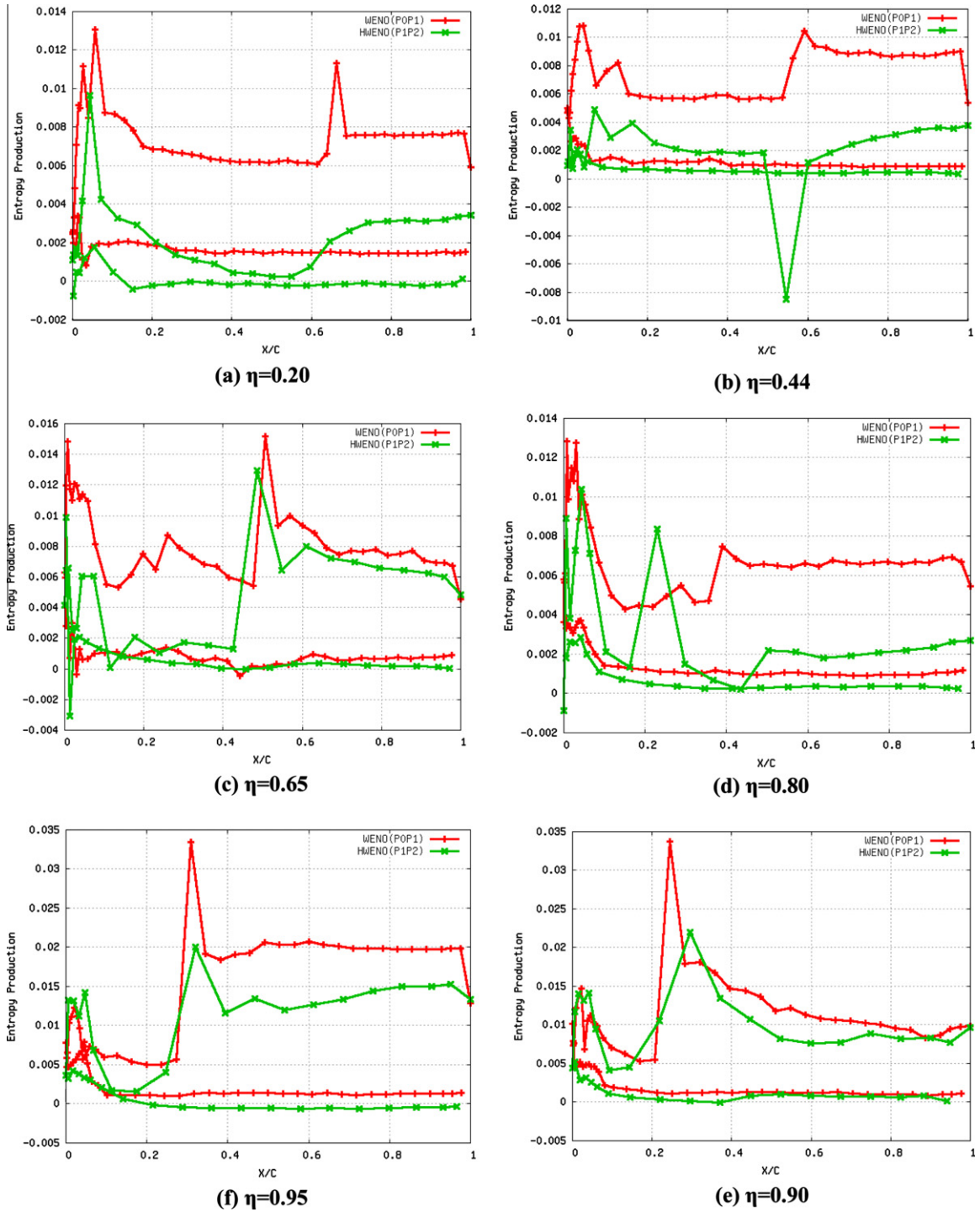


Fig. 7. Comparison of the computed entropy production distributions obtained by the WENO (P_0P_1) and HWENO (P_1P_2) solutions at six span-wise locations for a transonic flow past the ONERA M6 wing at $M_\infty = 0.84$, and $\alpha = 3.06^\circ$.

computation contains 48,851 grid points, 253,577 elements, and 11,802 boundary points for the half-span airplane. The solution is computed at a free stream of Mach number of 0.85 and an angle of attack of 2° . The computed Mach number contours on the surface of the airplane, along with the surface mesh, are shown in Fig. 10. One can see that the shock waves on the upper surface of the wing are captured well within one cell, confirming the accuracy and robustness of the HWENO (P_1P_2) method for computing complicated flows of practical importance.

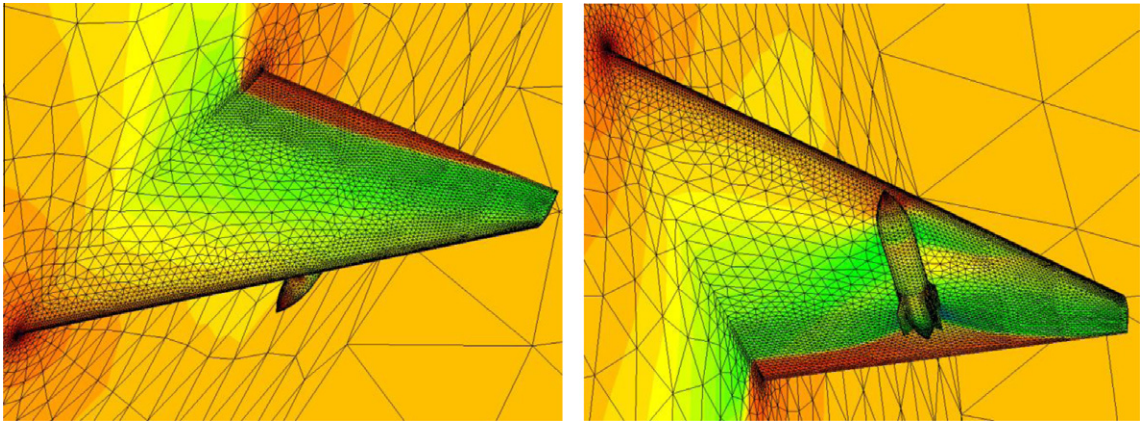


Fig. 8. Computed pressure contours on the unstructured surface mesh obtained by the HWENO (P₁P₂) solution (nelem = 319,134, npoin = 61,075, nboun = 14,373) for a transonic flow past a wing/pylon/finned-store configuration at $M_\infty = 0.95$, and $\alpha = 0^\circ$.

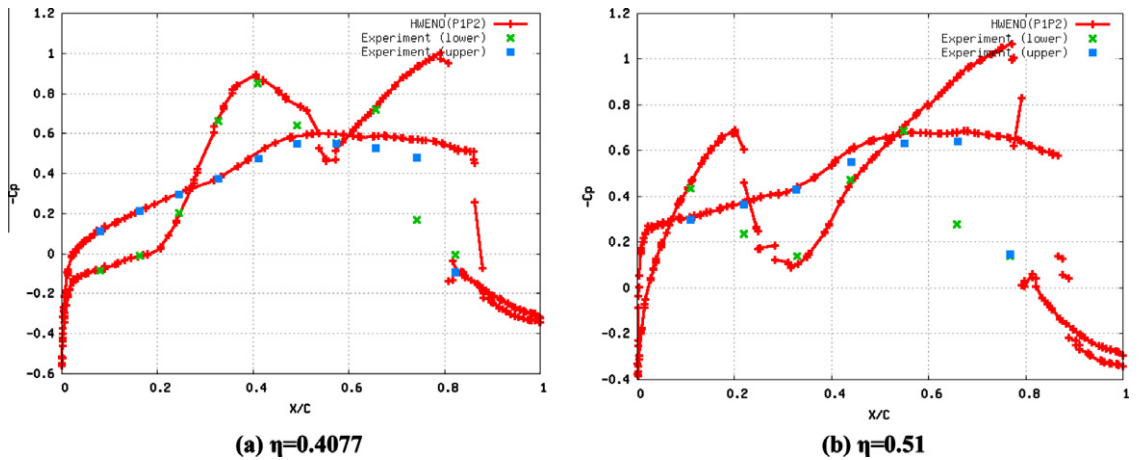


Fig. 9. Comparison of the computed pressure coefficient distributions with experimental data at two span-wise locations for a transonic flow past a wing/pylon/finned-store configuration at $M_\infty = 0.95$, and $\alpha = 0^\circ$.

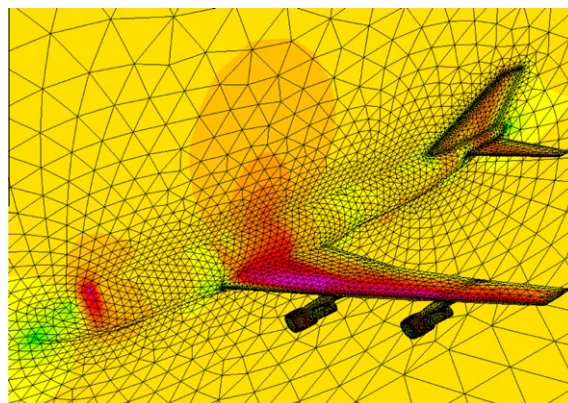


Fig. 10. Computed Mach number contours and unstructured surface mesh for transonic flow past a complete B747 aircraft (nelem = 253,577, npoin = 48,851, nboun = 11,802) at $M_\infty = 0.85$, and $\alpha = 2^\circ$.

5. Conclusions

A reconstructed discontinuous Galerkin method based on a hierarchical Hermite WENO reconstruction, HWENO (P_1P_2), has been presented for solving the compressible Euler equations on tetrahedral grids. The HWENO (P_1P_2) method is designed not only to enhance the accuracy of the discontinuous Galerkin method, but also to avoid non-physical oscillations in the vicinity of discontinuities. A number of numerical experiments for a variety of flow conditions have been conducted to demonstrate the accuracy, robustness, and non-oscillatory performance of the HWENO (P_1P_2) method. The numerical results obtained indicate that the developed HWENO (P_1P_2) method is able to provide sharp resolution of shock waves without over- and under-shoots for flows with strong discontinuities and achieve the designed third-order of accuracy for smooth flows: one order accuracy higher than the underlying DG method, thus significantly increasing the accuracy of the underlying DG method without significant increase in computing costs and memory requirements.

Acknowledgments

This research is partially supported using funding received from the DOE Office of Nuclear Energy's Nuclear Engineering University Program. The first author would like to acknowledge the partial support for this work provided by the fundamental research program of DTRA under Grant No. HDTR1-10-1-0.123. Dr. Suhithi Peiris serves as the technical monitor.

References

- [1] W.H. Reed, T.R. Hill, Triangular Mesh Methods for the Neutron Transport Equation, Los Alamos Scientific Laboratory Report, LA-UR-73-479, 1973.
- [2] B. Cockburn, S. Hou, C.W. Shu, TVD Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case, *Mathematics of Computation*, 55 (1990) 545–581 (P.L. George, Automatic Mesh Generation, John Wiley & Sons, 1991).
- [3] B. Cockburn, C.W. Shu, The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional system, *Journal of Computational Physics* 141 (1998) 199–224.
- [4] B. Cockburn, G. Karniadakis, C.W. Shu, The development of discontinuous Galerkin method, in: B. Cockburn, G.E. Karniadakis, C.W. Shu (Eds.), *Discontinuous Galerkin Methods, Theory, Computation, and Applications*, Lecture Notes in Computational Science and Engineering, Springer-Verlag, New York, 2000, pp. 5–50. vol. 11.
- [5] F. Bassi, S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, *Journal of Computational Physics* 138 (1997) 251–285.
- [6] H.L. Atkins, C.W. Shu, Quadrature free implementation of discontinuous Galerkin method for hyperbolic equations, *AIAA Journal* 36 (5) (1998).
- [7] F. Bassi, S. Rebay, GMRES discontinuous Galerkin solution of the compressible Navier–Stokes equations, in: B. Cockburn, G.E. Karniadakis, C.W. Shu (Eds.), *Discontinuous Galerkin Methods, Theory, Computation, and Applications*, Lecture Notes in Computational Science and Engineering, Springer-Verlag, New York, 2000, pp. 197–208. vol. 11.
- [8] T.C. Warburton, G.E. Karniadakis, A discontinuous Galerkin method for the viscous MHD equations, *Journal of Computational Physics* 152 (1999) 608–641.
- [9] J.S. Hesthaven, T. Warburton, *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*, Texts in Applied Mathematics 56 (2008).
- [10] P. Rasetarinera, M.Y. Hussaini, An efficient implicit discontinuous spectral Galerkin method, *Journal of Computational Physics* 172 (2001) 718–738.
- [11] B.T. Helenbrook, D. Mavriplis, H.L. Atkins, Analysis of p -multigrid for continuous and discontinuous finite element discretizations, *AIAA Paper* 2003-3989, 2003.
- [12] K.J. Fidkowski, T.A. Oliver, J. Lu, D.L. Darmofal, p -Multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier–Stokes equations, *Journal of Computational Physics* 207 (1) (2005) 92–113.
- [13] H. Luo, J.D. Baum, R. Löhner, A discontinuous Galerkin method using Taylor basis for compressible flows on arbitrary grids, *Journal of Computational Physics* 227 (20) (2008) 8875–8893.
- [14] H. Luo, J.D. Baum, R. Löhner, On the computation of steady-state compressible flows using a discontinuous Galerkin method, *International Journal for Numerical Methods in Engineering* 73 (5) (2008) 597–623.
- [15] H. Luo, J.D. Baum, R. Löhner, A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids, *Journal of Computational Physics* 225 (1) (2007) 686–713.
- [16] H. Luo, J.D. Baum, R. Löhner, A p -multigrid discontinuous Galerkin method for the Euler equations on unstructured grids, *Journal of Computational Physics* 211 (2) (2006) 767–783.
- [17] H. Luo, J.D. Baum, R. Löhner, Fast, p -multigrid discontinuous Galerkin method for compressible flows at all speeds, *AIAA Journal* 46 (3) (2008) 635–652.
- [18] M. Dumbser, D.S. Balsara, E.F. Toro, C.D. Munz, A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes, *Journal of Computational Physics* 227 (2008) 8209–8253.
- [19] M. Dumbser, O. Zanotti, Very high order PNP schemes on unstructured meshes for the resistive relativistic MHD equations, *Journal of Computational Physics* 228 (2009) 6991–7006.
- [20] M. Dumbser, Arbitrary high order PNP schemes on unstructured meshes for the compressible Navier–Stokes equations, *Computers and Fluids* 39 (2010) 60–76.
- [21] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, *Journal of Computational Physics* 131 (1997) 267–279.
- [22] F. Bassi, S. Rebay, Discontinuous Galerkin solution of the Reynolds-averaged Navier–Stokes and k - ω turbulence model equations, *Journal of Computational Physics* 34 (2005) 507–540.
- [23] B. Cockburn, C.W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion system, *SIAM, Journal of Numerical Analysis* 16 (2001) 301–318.
- [24] C.E. Baumann, J.T. Oden, A discontinuous hp finite element method for the Euler and Navier–Stokes equations, *International Journal for Numerical Methods in Fluids* 31 (1999).
- [25] H. Luo, L. Luo, K. Xu, A discontinuous Galerkin method based on a BGK scheme for the Navier–Stokes equations on arbitrary grids, *Advances in Applied Mathematics and Mechanics* 1 (3) (2009) 301–318.
- [26] H. Luo, L. Luo, R. Nourgaliev, A reconstructed discontinuous Galerkin method for the Euler equations on arbitrary grids, *Communication in Computational Physics* 12 (5) (2012) 1495–1519.
- [27] H. Luo, L. Luo, R. Nourgaliev, V.A. Mousseau, N. Dinh, A reconstructed discontinuous Galerkin method for the compressible Navier–Stokes equations on arbitrary grids, *Journal of Computational Physics* 229 (2010) 6961–6978.
- [28] H. Luo, L. Luo, A. Ali, R. Nourgaliev, C. Cai, A parallel, reconstructed discontinuous Galerkin method for the compressible flows on arbitrary grids, *Communication in Computational Physics* 9 (2) (2011) 363–389.

- [29] L.P. Zhang, W. Liu, L.X. He, X.G. Deng, H.X. Zhang, A class of hybrid DG/FV methods for conservation laws I: basic formulation and one-dimensional system, *Journal of Computational Physics* 23 (4) (2012) 1081–1103.
- [30] L.P. Zhang, W. Liu, L.X. He, X.G. Deng, H.X. Zhang, A class of hybrid DG/FV methods for conservation laws II: two dimensional cases, *Journal of Computational Physics* 23 (4) (2012) 1104–1120.
- [31] H. Luo, Y. Xia, R. Nourgaliev, C. Cai, A class of reconstructed discontinuous Galerkin methods for the compressible flows on arbitrary grids, AIAA-2011-0199, 2011.
- [32] H. Luo, H. Xiao, R. Nourgaliev, C. Cai, A comparative study of different reconstruction schemes for reconstructed discontinuous Galerkin methods for the compressible flows on arbitrary grids, AIAA-2011-3839, 2011.
- [33] D.F. Haider, J.P. Croisille, B. Courbet, Stability analysis of the cell centered finite-volume MUSCL method on unstructured grids, *Numerische Mathematik* 113 (4) (2009) 555–600.
- [34] D. Balsara, C. Altmann, C.D. Munz, M. Dumbser, A sub-cell based indicator for troubled zones in RKDG schemes and a novel class of hybrid RKDG + HWENO schemes, *Journal of Computational Physics* 226 (2007) 586–620.
- [35] Z. Xu, Y. Liu, H. Du, C.W. Shu, Point-wise hierarchical reconstruction for discontinuous Galerkin and finite volume method for solving conservation laws, *Journal of Computational Physics* 230 (2011) 6843–6865.
- [36] H. Luo, Y. Xia, S. Li, R. Nourgaliev, C. Cai, A Hermite WENO reconstruction-based discontinuous Galerkin method for the Euler equations on tetrahedral grids, *Journal of Computational Physics* 231 (2012) 5489–5503.
- [37] S. Spiegel, H. Luo, A finite volume method based on WENO reconstruction for compressible flows on hybrid grids, submitted for publication at 51th AIAA Sciences Meeting, 2013.
- [38] V. Schmitt, E. Charpin, Pressure distributions on the ONERA M6-wing at transonic mach numbers, experimental data base for computer program assessment, AGARD AR-138, 1979.
- [39] E.R. Ileim, CFD wing/pylon/finned store mutual interference wind tunnel experiment, AEDC-TSR-91-P4, Arnold Engineering Development Center, Arnold AFB, TN, January 1991.
- [40] M. Dumbser, M. Kaser, Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic system, *Journal of Computational Physics* 221 (2007) 693–723.
- [41] R. Löhner, P. Parikh, Three-dimensional grid generation by the advancing front method, *International Journal for Numerical Methods in Fluids* (8) (1988) 1135–1149.
- [42] D. Sharov, H. Luo, J.D. Baum, R. Löhner, Unstructured Navier–Stokes grid generation at corners and ridges, *International Journal for Numerical Methods in Fluids* 43 (2003) 717–728.
- [43] H. Luo, J.D. Baum, R. Löhner, Extension of HLLC scheme for flows at all speeds, *AIAA Journal* 43 (6) (2005) 1160–1166.