Runge-Kutta discontinuous Galerkin method with a simple and compact Hermite WENO limiter on unstructured meshes¹

Jun Zhu², Xinghui Zhong³, Chi-Wang Shu⁴ and Jianxian Qiu⁵

Abstract

In this paper we generalize a new type of compact Hermite weighted essentially nonoscillatory (HWENO) limiter for the Runge-Kutta discontinuous Galerkin (RKDG) method, which was recently developed in [38] for structured meshes, to two dimensional unstructured meshes. The main idea of this HWENO limiter is to reconstruct the new polynomial by the usage of the entire polynomials of the DG solution from the target cell and its neighboring cells in a least squares fashion [11] while maintaining the conservative property, then use the classical WENO methodology to form a convex combination of these reconstructed polynomials based on the smoothness indicators and associated nonlinear weights. The main advantage of this new HWENO limiter is the robustness for very strong shocks and simplicity in implementation especially for the unstructured meshes considered in this paper, since only information from the target cell and its immediate neighbors is needed. Numerical results for both scalar and system equations are provided to test and verify the good performance of this new limiter.

Key Words: Runge-Kutta discontinuous Galerkin method, HWENO limiter, unstructured mesh.

AMS(MOS) subject classification: 65M60, 35L65

¹The research of J. Zhu and J. Qiu is partially supported by NSFC grants 11372005, 91230110 and 11571290. The research of C.-W. Shu is partially supported by DOE grant DE-FG02-08ER25863 and NSF grant DMS-1418750.

²College of Science, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu 210016, P.R. China. E-mail: zhujun@nuaa.edu.cn

³Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, UT 84112, USA. E-mail: zhongxh@sci.utah.edu

⁴Division of Applied Mathematics, Brown University, Providence, RI 02912, USA. E-mail: shu@dam.brown.edu

⁵School of Mathematical Sciences and Fujian Provincial Key Laboratory of Mathematical Modeling & High-Performance Scientific Computing, Xiamen University, Xiamen, Fujian 361005, P.R. China. E-mail: jxqiu@xmu.edu.cn

1 Introduction

In this paper we consider solving the two dimensional conservation law, given by

$$\begin{cases} u_t + f(u)_x + g(u)_y = 0, \\ u(x, y, 0) = u_0(x, y), \end{cases}$$
(1.1)

using the Runge-Kutta discontinuous Galerkin (RKDG) method [6, 7, 8, 9] on unstructured triangular meshes. RKDG methods use explicit, nonlinearly stable high order Runge-Kutta methods [33] to discretize the temporal variable and the DG methods to discretize the spatial variables, with exact or approximate Riemann solvers as interface fluxes. For a detailed discussion on DG methods for solving conservation laws, we refer the readers to the review paper [10] and the books and lecture notes [5, 15, 21, 32].

DG methods can compute the numerical solution to (1.1) without further modification provided the solution either is smooth or contains weak discontinuities. However, for problems containing strong shocks or contact discontinuities, there are spurious oscillations in the numerical solution near these discontinuities, which may cause nonlinear instability. One common strategy to control these oscillations is to apply nonlinear limiters to RKDG methods. Many limiters have been studied in the literature for RKDG methods, such as the minmod type total variation bounded (TVB) limiter [6, 7, 8, 9], the moment based limiter [3] and an improved moment limiter [4] and so on. These limiters belong to the slope type limiters and they do control oscillations very well at the price of possibly degrading the accuracy of the numerical solution at smooth extrema. Another type of limiters is the WENO type limiters, which are based on the weighted essentially non-oscillatory (WENO) methodology [14, 16, 17, 23] and can achieve both high-order accuracy and non-oscillatory property near discontinuities. This type of limiters includes the WENO limiter [27, 36] and the HWENO limiter [24, 26, 29], which use the classical WENO finite volume methodology for reconstruction and thus require a wide stencil, especially for higher order methods. Therefore, it is difficult to implement these limiters for multi-dimensional problems, especially on unstructured meshes. Moreover, these limiters may have the issue of negative linear weights. An alternative family of DG limiters which serves at the same time as a new PDE-based limiter, as well as a troubled cells indicator, was introduced by Dumbser et al. [13].

More recently, a particularly simple and compact WENO limiter was developed by Zhong and Shu [35] for RKDG schemes, and then was generalized to the unstructured mesh in [37]. This simple WENO limiter utilizes fully the advantage of DG schemes in that a complete polynomial is available in each cell without the need of reconstruction. The major advantages of this simple WENO limiter include the compactness of its stencil, the simplicity in its implementation, and the freedom in choosing linear weights, which can be set arbitrarily so long as their summation is one and each of them is nonnegative. However, it was observed in [35] that the limiter might not be robust enough for problems containing very strong shocks or low pressure problem, especially for higher order polynomials, for example the blast wave problems [30, 34] and the double rarefaction wave problem [22]. In order to overcome this difficulty, without compromising the advantages of compact stencil and simplicity of linear weights, we present a modification of the limiter in the step of preprocessing the polynomials in the immediate neighboring cells before applying the WENO reconstruction procedure. This preprocessing is necessary to maintain strict conservation, and is designed in [35] to be a simple addition of a constant to make the cell average of the preprocessed neighboring cell polynomial in the target cell matching the original cell average. In this paper, a more involved least square process [11] is used in this step. The objective is to achieve strict conservation while maintaining more information of the original neighboring cell polynomial before applying the WENO procedure. Numerical experiments indicate that this modification does improve the robustness of the limiter.

This paper is organized as follows: in Section 2, we briefly review the RKDG methods for solving (1.1) on triangular meshes and present the details of the new HWENO procedure for two dimensional scalar and system problems on unstructured meshes. Numerical examples are provided in Section 3 to verify the compactness, accuracy and stability of this new approach. Concluding remarks are given in Section 4.

2 New HWENO limiter to RKDG method on unstructured mesh

In this section, we describe the details of using the new HWENO reconstruction procedure as a limiter for the RKDG methods. It is a generalization to unstructured meshes of the procedure in [38] for structured meshes. The general framework of the HWENO limiting procedure consists of the following two steps.

The first step is to identify the troubled cells, namely those cells which may need the HWENO limiting procedure. This step is an important issue for limiters. If too many cells are identified as troubled cells, then the computational cost associated with the second step will increase. If too few cells are identified as troubled cells, then the oscillations may not be avoided. We remark here that the main focus of this paper is the development of a compact, simple HWENO limiter on unstructured meshes. We refer the reads to [28] for a comparison between different trouble cell indicators. The KXRCF shock detection technique in [20] is used in our numerical tests to detect troubled cells. As discussed in [20], let \mathcal{J} denote the normalization of the jump in the numerical solution component (both the density ρ and the total energy E are used in our numerical tests) across the inflow edges (faces) of the target cell, to an "average" convergence rate. $\mathcal{J} \to 0$ as the mesh size $h \to 0$ in smooth solution regions, whereas $\mathcal{J} \to \infty$ near a discontinuity. Thus the KXRCF discontinuity detection scheme is: if $\mathcal{J} > C_k$, the indicated solution component is discontinuous; if $\mathcal{J} \leq C_k$, the indicated solution component is discontinuous; if $\mathcal{J} = C_k$, the indicated solution component is a constant and we set $C_k = 1$ unless otherwise specified in our numerical tests.

The second step is to reconstruct a new polynomial using the HWENO limiting procedure in order to replace the solution polynomial on the troubled cell. The new polynomial should maintain the cell average and high order accuracy of the original DG solution polynomial, but should be less oscillatory.

We will first briefly review the RKDG method for solving two dimensional problems on unstructured meshes in Section 2.1. Then the details of this second step will be discussed for the scalar case in Section 2.2 and for the system case in Section 2.3.

2.1 Review of the RKDG method on unstructured mesh

This section provides a review of the RKDG methods for solving two dimensional conservation laws (1.1) on the triangular meshes.

We first use DG methods to discretize the spatial variables. Given a triangulation of the computational domain consisting of cells Δ_j , the DG method has its solution as well as the test function space given by $V_h^k = \{v(x, y) : v(x, y)|_{\Delta_j} \in \mathbb{P}^k(\Delta_j)\}$, where $\mathbb{P}^k(\Delta_j)$ denotes the set of polynomials of degree at most k defined on Δ_j . The semi-discrete DG method for solving (1.1) is defined as follows: find the unique function $u_h \in V_h^k$, such that

$$\int_{\Delta_0} (u_h)_t v \, dx \, dy = \int_{\Delta_0} \left(f(u_h) v_x + g(u_h) v_y \right) \, dx \, dy - \int_{\partial \Delta_0} \left(f(u^h), \widehat{g(u^h)} \right)^T \cdot n \, v \, ds \qquad (2.1)$$

holds for all the test functions $v \in V_h^k$. Here $n = (n_x, n_y)^T$ is the outward unit normal of the triangle boundary $\partial \triangle_0$, and $(\widehat{f(u^h)}, \widehat{g(u^h)})^T \cdot n$ is a monotone numerical flux for the scalar case and an exact or approximate Riemann solver for the system case. The Lax-Friedrichs flux is used in all our numerical tests.

For time discretization, we can use, for example, the third order strong stability preserving (SSP) Runge-Kutta methods [33]:

$$u^{(1)} = u^{n} + \Delta t L(u^{n}),$$

$$u^{(2)} = \frac{3}{4}u^{n} + \frac{1}{4}u^{(1)} + \frac{1}{4}\Delta t L(u^{(1)}),$$

$$u^{n+1} = \frac{1}{3}u^{n} + \frac{2}{3}u^{(2)} + \frac{2}{3}\Delta t L(u^{(2)}).$$

(2.2)

Other SSP time discretization method can also be used here.

2.2 New HWENO limiting procedure: scalar case

In this subsection, the details of the HWENO limiting procedure are presented for the scalar case. The idea of this new and simple HWENO limiter is that the reconstructed polynomial on the troubled cell is a convex combination of the DG solution polynomial on the target cell and the "modified" DG solution polynomials on its neighboring cells. The modification procedure is in a least squares fashion [11]. The construction of the nonlinear weights in the convex combination coefficients follows the classical WENO procedure.

Now assume Δ_0 is identified as a troubled cell by our trouble cell indicator. The procedure to reconstruct a new polynomial on the troubled cell Δ_0 by using the HWENO reconstruction procedure is summarized as follows:



Figure 2.1: The stencil $S = \{ \triangle_0, \triangle_1, \triangle_2, \triangle_3 \}$

Step 1.1. Denote the reconstruction stencil as $S = \{ \Delta_0, \Delta_1, \Delta_2, \Delta_3 \}$ shown in Figure 2.1, and denote the DG solutions on these four cells as $p_{\ell}(x, y), \ell = 0, 1, 2, 3$, respectively. We need to modify the DG solutions on the neighboring cells first and denote the modified version of $p_{\ell}(x, y), \ell = 1, 2, 3$ as $\tilde{p}_{\ell}(x, y), \ell = 1, 2, 3$. The modification procedure is defined as follows: $\tilde{p}_1(x, y)$ is the solution to the constrained minimization problem:

$$\min_{\forall \phi(x,y) \in \mathbb{P}^k(\Delta_1)} \left\{ \left(\int_{\Delta_1} (\phi(x,y) - p_1(x,y))^2 dx dy \right) + \sum_{\ell \in \mathbb{L}_1} \left(\int_{\Delta_\ell} (\phi(x,y) - p_\ell(x,y)) dx dy \right)^2 \right\},$$

subject to $\bar{\phi} = \bar{p}_0$, where

$$\bar{\bar{\phi}} = \frac{1}{|\Delta_0|} \int_{\Delta_0} \phi(x, y) \, dx dy, \quad \bar{\bar{p}}_0 = \frac{1}{|\Delta_0|} \int_{\Delta_0} p_0(x, y) \, dx dy$$

and

$$\mathbb{L}_1 = \{2, 3\} \cap \{\ell : |\bar{p}_\ell - \bar{p}_0| < \max(|\bar{p}_2 - \bar{p}_0|, |\bar{p}_3 - \bar{p}_0|)\}.$$

Here and below $\overline{\star}$ denotes the cell average of the function \star on the target cell while $\overline{\star}$ denotes the cell average of the function \star on its own associated cell.

The modified polynomial $\tilde{p}_1(x, y)$ has the same cell average as the polynomial on the troubled cell, \bar{p}_0 , and it optimizes the distance to $p_1(x; y)$ and to the cell averages of those "useful" polynomial(s) on the other neighboring cells. The "useful" polynomial is chosen by comparing the distance between the cell averages of the polynomials on the other neighboring cells and the cell average of p_0 on the target cell. If one is not the farthest, then this polynomial is considered "useful".

Similarly, $\tilde{p}_2(x, y)$ is the solution to the constrained minimization problem:

$$\min_{\forall \phi(x,y) \in \mathbb{P}^k(\Delta_2)} \left\{ \left(\int_{\Delta_2} (\phi(x,y) - p_2(x,y))^2 dx dy \right) + \sum_{\ell \in \mathbb{L}_2} \left(\int_{\Delta_\ell} (\phi(x,y) - p_\ell(x,y)) dx dy \right)^2 \right\},$$

subject to $\bar{\phi} = \bar{p}_0$, where

$$\mathbb{L}_{2} = \{1, 3\} \cap \{\ell : |\bar{p}_{\ell} - \bar{p}_{0}| < \max(|\bar{p}_{1} - \bar{p}_{0}|, |\bar{p}_{3} - \bar{p}_{0}|)\}$$

 $\tilde{p}_3(x, y)$ is the solution to the constrained minimization problem:

$$\min_{\forall \phi(x,y) \in \mathbb{P}^k(\Delta_3)} \left\{ \left(\int_{\Delta_3} (\phi(x,y) - p_3(x,y))^2 dx dy \right) + \sum_{\ell \in \mathbb{L}_3} \left(\int_{\Delta_\ell} (\phi(x,y) - p_\ell(x,y)) dx dy \right)^2 \right\},$$

subject to $\bar{\phi} = \bar{p}_0$, where

$$\mathbb{L}_3 = \{1, 2\} \cap \{\ell : |\bar{p}_\ell - \bar{p}_0| < \max(|\bar{p}_1 - \bar{p}_0|, |\bar{p}_2 - \bar{p}_0|)\}.$$

We also define $\tilde{p}_0(x, y) = p_0(x, y)$ to keep notation consistency.

Step 1.2. Choose the linear weights denoted by $\gamma_0, ..., \gamma_3$. Notice that, since $\tilde{p}_i(x, y)$, for i = 0, 1, 2, 3, are all (k + 1)-th order approximations to the exact solution in smooth regions, there is no requirement on the values of these linear weights for accuracy besides $\gamma_0 + \gamma_1 + \gamma_2 + \gamma_3 = 1$. The choice of these linear weights is then solely based on the consideration of a balance between accuracy and ability to achieve essentially nonoscillatory shock transition. In all of our numerical tests, following the practice in [12, 35], we take $\gamma_0 = 0.997$ and $\gamma_1 = \gamma_2 = \gamma_3 = 0.001$. Step 1.3. Compute the smoothness indicators, denoted by β_i , i = 0, ..., 3, which measure how smooth the functions $\tilde{p}_i(x, y)$, for i = 0, ..., 3, are on the target cell Δ_0 . The smaller these smoothness indicators, the smoother the functions are on the target cell. We use the similar recipe for the smoothness indicators as in [1, 17, 31]:

$$\beta_i = \sum_{|\ell|=1}^k |\Delta_0|^{|\ell|-1} \int_{\Delta_0} \left(\frac{1}{|\ell|!} \frac{\partial^{|\ell|}}{\partial x^{\ell_1} \partial y^{\ell_2}} \tilde{p}_i(x, y) \right)^2 dx dy, \tag{2.3}$$

where $\ell = (\ell_1, \ell_2)$ and $|\ell| = \ell_1 + \ell_2$.

Step 1.4. Compute the nonlinear weights based on the smoothness indicators:

$$\omega_i = \frac{\tilde{\omega}_i}{\sum_{\ell=0}^3 \tilde{\omega}_\ell}, \qquad \tilde{\omega}_\ell = \frac{\gamma_\ell}{(\varepsilon + \beta_\ell)^2}.$$
(2.4)

Here ε is a small positive number to avoid the denominator becoming zero. We take $\varepsilon = 10^{-6}$ in our computation.

Step 1.5. The final nonlinear HWENO reconstruction polynomial $p_0^{new}(x, y)$ is defined by a convex combination of the four (modified) polynomials in the stencil:

$$p_0^{new}(x,y) = \omega_0 \tilde{p}_0(x,y) + \omega_1 \tilde{p}_1(x,y) + \omega_2 \tilde{p}_2(x,y) + \omega_3 \tilde{p}_3(x,y).$$
(2.5)

It is easy to verify that $p_0^{new}(x, y)$ has the same cell average and order of accuracy as the original one $p_0(x, y)$ on the condition that $\sum_{i=0}^{3} \omega_i = 1$.

2.3 HWENO limiting procedure: system case

In this subsection, the details of the HWENO limiting procedure are presented for the systems case.

Consider equation (1.1) where u, f(u) and g(u) are vectors with m components. In order to achieve better nonoscillatory property, the HWENO reconstruction limiter is used with a local characteristic decomposition, see [31] for a discussion on the rationale in adopting such a decomposition. In this paper, we only consider the following Euler systems and set m = 4.

$$u_t + f(u)_x + g(u)_y = \frac{\partial}{\partial t} \begin{pmatrix} \rho \\ \rho \mu \\ \rho \nu \\ E \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} \rho \mu \\ \rho \mu^2 + p \\ \rho \mu \nu \\ \mu(E+p) \end{pmatrix} + \frac{\partial}{\partial y} \begin{pmatrix} \rho \nu \\ \rho \mu \nu \\ \rho \nu^2 + p \\ \nu(E+p) \end{pmatrix} = 0, \quad (2.6)$$

with $u(x, y, 0) = u_0(x, y)$, where ρ is the density, μ is the x-direction velocity, ν is the ydirection velocity, E is the total energy, $p = \frac{E}{\gamma-1} - \frac{1}{2}\rho(\mu^2 + \nu^2)$ is the pressure and $\gamma = 1.4$ in our test cases. We denote the Jacobian matrices as $(f'(u), g'(u)) \cdot \mathbf{n_i}$ and $\mathbf{n_i} = (\mathbf{n_{i_x}}, \mathbf{n_{i_y}})^{\mathrm{T}}, i =$ 1, 2, 3, are the outward unit normals to different edges of the target cell. We then give the left and right eigenvectors of such Jacobian matrices as:

$$L_{i} = \begin{pmatrix} \frac{B_{2} + (\mu \mathbf{n}_{ix} + \nu \mathbf{n}_{iy})/c}{2} & -\frac{B_{1}\mu + \mathbf{n}_{ix}/c}{2} & -\frac{B_{1}\nu + \mathbf{n}_{iy}/c}{2} & \frac{B_{1}}{2} \\ \mathbf{n}_{iy}\mu - \mathbf{n}_{ix}\nu & -\mathbf{n}_{iy} & \mathbf{n}_{ix} & 0 \\ 1 - B_{2} & B_{1}\mu & B_{1}\nu & -B_{1} \\ \frac{B_{2} - (\mu \mathbf{n}_{ix} + \nu \mathbf{n}_{iy})/c}{2} & -\frac{B_{1}\mu - \mathbf{n}_{ix}/c}{2} & -\frac{B_{1}\nu - \mathbf{n}_{iy}/c}{2} & \frac{B_{1}}{2} \end{pmatrix}, \quad (2.7)$$

and

$$R_{i} = \begin{pmatrix} 1 & 0 & 1 & 1 \\ \mu - c\mathbf{n}_{ix} & -\mathbf{n}_{iy} & \mu & \mu + c\mathbf{n}_{ix} \\ \nu - c\mathbf{n}_{iy} & \mathbf{n}_{ix} & \nu & \nu + c\mathbf{n}_{iy} \\ H - c(\mu\mathbf{n}_{ix} + \nu\mathbf{n}_{iy}) & -\mathbf{n}_{iy}\mu + \mathbf{n}_{ix}\nu & \frac{\mu^{2} + \nu^{2}}{2} & H + c(\mu n_{ix} + \nu\mathbf{n}_{iy}) \end{pmatrix}, i = 1, 2, 3,$$
(2.8)

where $c = \sqrt{\gamma p/\rho}$, $B_1 = (\gamma - 1)/c^2$, $B_2 = B_1(\mu^2 + \nu^2)/2$ and $H = (E + p)/\rho$.

Assuming Δ_0 is the troubled cell detected by the KXRCF technique [20], we denote the four polynomial vectors on the troubled cell and its three neighboring cells as p_0 , p_1 , p_2 , p_3 . Note that each of them has four components. We then perform the HWENO limiting procedure as follows:

Step 2.1. In each $\mathbf{n_i}$ -direction among three normal directions of $\partial \triangle_0$, we reconstruct new polynomial vectors $(p_0)_i^{new}$, i = 1, 2, 3, by using the characteristic-wise HWENO limiting procedure with the associated Jacobian $f'(u)\mathbf{n_{ix}} + g'(u)\mathbf{n_{iy}}$, i = 1, 2, 3:

– Step 2.1.1. Project the polynomial vectors p_0 , p_1 , p_2 and p_3 into the characteristic fields $\tilde{\tilde{p}}_{i_l} = L_i \cdot p_l$, i = 1, 2, 3, l = 0, 1, 2, 3. $\tilde{\tilde{p}}_{i_l}$ a 4-component vector with each component being a polynomial of degree up to k.

– Step 2.1.2. For each component of $\tilde{\tilde{p}}_{i_l}$, perform Step 1.1 to Step 1.5 of the HWENO limiting procedure that has been specified for the scalar case, to obtain a new 4-component vector on the troubled cell Δ_0 as $\tilde{\tilde{p}}_{i_l}^{new}$.

– Step 2.1.3. Project $\tilde{\tilde{p}}_{i,0}^{new}$ into the physical space $p_{0,i}^{new} = R_i \cdot \tilde{\tilde{p}}_{i,0}^{new}$, i = 1, 2, 3. Step 2.2. The final new 4-component vector on the troubled cell Δ_0 is defined as

$$p_0^{new} = \frac{\sum_{i=1}^3 p_{i,0}^{new} |\Delta_i|}{\sum_{i=1}^3 |\Delta_i|}.$$

3 Numerical results

In this section, we provide numerical results to demonstrate the performance of the HWENO limiters for the RKDG methods on unstructured meshes described in Section 2. For all of our accuracy tests, the refinement is performed by a structured refinement (each triangle is divided into four similar smaller triangles for every level of the refinement). We perform the HWENO limiting procedure on every cell of the computational domain for the accuracy tests, in order to fully testify the influence of the limiter upon accuracy. The CFL number is set to be 0.3 for the second order (k = 1), 0.18 for the third order (k = 2) and 0.1 for the fourth order (k = 3) RKDG methods with and without the HWENO limiters.

Example 3.1. We solve the following scalar Burgers equation in two dimensions:

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0, (x, y) \in [-2, 2] \times [-2, 2], \tag{3.1}$$

with the initial condition $u(x, y, 0) = 0.5 + \sin(\pi(x+y)/2)$ and periodic boundary conditions in both directions. The final computing time is $t = 0.5/\pi$, when the solution is still smooth. For this test case, the sample mesh is shown in Figure 3.1. In order to fully test the effect of the HWENO limiter on accuracy, we perform the HWENO limiting procedure on every cell of the computational domain. The L^1 , L^2 , L^∞ errors and numerical orders of accuracy for the RKDG methods with the HWENO limiters comparing with the original RKDG methods without limiters are shown in Table 3.1. It is observed that the new HWENO limiters maintain the designed order of accuracy.

Example 3.2. We solve the Euler equations (2.6). The computational field is $[0,2] \times [0,2]$. The initial conditions are: $\rho(x, y, 0) = 1 + 0.2 \sin(\pi(x+y)), \ \mu(x, y, 0) = 0.7, \ \nu(x, y, 0) = 0.3,$



Figure 3.1: Burgers equation. Sample mesh.

Table 3.1: $u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0$. $u(x, y, 0) = 0.5 + \sin(\pi(x+y)/2)$. Periodic boundary conditions in both directions. $T = 0.5/\pi$. L^1 , L^∞ and L^2 errors.

			RKD	G with HW	ENO lii	miter	RKDG without limiter						
	cell	L^1 error	order	L^{∞} error	order	$L^2 \operatorname{error}$	order	L^1 error	order	L^{∞} error	order	L^2 error	order
	232	3.25E-2		2.57E-1		4.52E-2		1.43E-2		1.06E-1		2.07E-2	
	928	9.40E-3	1.79	7.71E-2	1.74	1.43E-2	1.66	3.54E-3	2.01	2.92E-2	1.86	5.38E-3	1.94
P^1	3712	2.11E-3	2.15	1.84E-2	2.06	3.58E-3	2.00	8.68E-4	2.02	7.79E-3	1.91	1.36E-3	1.98
	14848	3.72E-4	2.50	4.40E-3	2.07	6.85E-4	2.39	2.15E-4	2.01	2.13E-3	1.87	3.45E-4	1.98
	59392	5.85E-5	2.66	6.85E-4	2.68	9.91E-5	2.79	5.36E-5	2.00	5.58E-4	1.93	8.70E-5	1.99
	232	2.12E-3		6.59E-2		3.83E-3		1.92E-3		2.27E-2		3.16E-3	
	928	3.12E-4	2.77	1.19E-2	2.47	5.97 E-4	2.68	2.98E-4	2.69	4.33E-3	2.39	5.16E-4	2.61
P^2	3712	4.30E-5	2.86	2.03E-3	2.55	8.86E-5	2.75	4.23E-5	2.81	7.58E-4	2.52	7.96E-5	2.70
	14848	5.65E-6	2.93	3.32E-4	2.61	1.22E-5	2.86	5.70E-6	2.89	1.18E-4	2.68	1.15E-5	2.79
	59392	7.39E-7	2.93	4.87E-5	2.77	1.66E-6	2.88	7.67E-7	2.89	1.77E-5	2.73	1.63E-6	2.82
	232	3.24E-4		1.81E-2		7.68E-4		3.07E-4		5.40E-3		7.04E-4	
	928	2.42E-5	3.74	1.81E-3	3.33	6.51E-5	3.56	2.20E-5	3.80	5.46E-4	3.31	5.88E-5	3.58
P^3	3712	1.54E-6	3.97	1.24E-4	3.86	4.33E-6	3.91	1.28E-6	4.10	3.91E-5	3.80	3.64E-6	4.01
	14848	1.03E-7	3.89	7.77E-6	4.00	2.97 E-7	3.86	$8.01\overline{\text{E-8}}$	4.00	$2.59\overline{\text{E}}$ -6	3.91	$2.28\overline{\text{E}}$ -7	4.00
	59392	$7.26\overline{\text{E}}-9$	3.84	4.84E-7	4.00	2.17E-8	-3.77	$5.14\overline{\text{E-9}}$	3.96	1.73E-7	3.90	1.46E-8	3.96

p(x, y, 0) = 1. Periodic boundary conditions are applied in both directions. The exact solution is $\rho(x, y, t) = 1 + 0.2 \sin(\pi(x + y - t))$. The final computing time is t = 2. For this test case the sample mesh is shown in Figure 3.2. Similar to the previous example, we define all cells in the computational field as troubled cells and perform the HWENO limiting procedure on every cell. The L^1 , L^2 , L^∞ errors and numerical orders of accuracy of the density for the RKDG methods with the HWENO limiters comparing with the original RKDG methods without limiters are shown in Table 3.2. Similar to the previous example, the new HWENO limiting procedure can maintain the desired order of accuracy even though the cells in smooth regions are all "intentionally" identified as troubled cells.



Figure 3.2: 2D-Euler equations. Sample mesh.

Example 3.3. We solve the nonlinear Burgers equation (3.1) with the same computational field $[-2, 2] \times [-2, 2]$ and the same initial condition $u(x, y, 0) = 0.5 + \sin(\pi(x+y)/2)$, except that we plot the results at $t = 1.5/\pi$ when a shock has already appeared in the solution. The solutions with the constant $C_k=0.001$ in the troubled cell indicator, and when all cells are defined as troubled cells in the computational field, are shown in Figure 3.3 for comparisons. We can see the schemes could give non-oscillatory shock transitions for this problem in either case.

Example 3.4. We solve the subsonic flow past a circular cylinder [24] with Mach number $M_{\infty} = 0.38$. This test is chosen to verify the ability of the HWENO limiter in maintaining the order of accuracy of the DG methods for problems with curved boundaries. Four successively

Table 3.2: 2D-Euler equations: initial data $\rho(x, y, 0) = 1 + 0.2 \sin(\pi(x+y)), u(x, y, 0) = 0.7, v(x, y, 0) = 0.3$, and p(x, y, 0) = 1. Periodic boundary conditions in both directions. T = 2.0. L^1, L^{∞} and L^2 errors.

		RKDG with HWENO limiter							RKDG without limiter					
	cell	L^1 error	order	L^{∞} error	order	L^2 error	order	L^1 error	order	L^{∞} error	order	L^2 error	order	
	232	5.21E-2		1.14E-1		6.13E-2		3.62E-3		1.60E-2		4.59E-3		
	928	1.74E-2	1.58	4.13E-2	1.47	1.99E-2	1.62	6.51E-4	2.48	3.46E-3	2.22	8.68E-4	2.40	
P^1	3712	4.11E-3	2.08	1.47E-2	1.49	5.58E-3	1.83	1.45E-4	2.17	8.68E-4	1.99	1.98E-4	2.13	
	14848	6.93E-4	2.56	4.59E-3	1.68	1.24E-3	2.17	3.47E-5	2.06	2.21E-4	1.97	4.77E-5	2.05	
	59392	1.37E-4	2.33	1.19E-3	1.94	2.17E-4	2.51	8.53E-6	2.03	5.61E-5	1.98	1.17E-5	2.02	
	232	3.74E-4		2.03E-3		4.80E-4		3.36E-4		2.35E-3		4.44E-4		
	928	7.80E-5	2.26	3.72E-4	2.45	9.27E-5	2.37	4.32E-5	2.96	3.38E-4	2.80	5.74E-5	2.95	
P^2	3712	1.24E-5	2.65	4.93E-5	2.92	1.44E-5	2.69	5.10E-6	3.08	3.63E-5	3.22	6.70E-6	3.09	
	14848	1.69E-6	2.87	7.42E-6	2.73	1.96E-6	2.88	5.96E-7	3.09	4.56E-6	2.99	7.89E-7	3.08	
	59392	2.19E-7	2.94	1.13E-6	2.70	2.54E-7	2.94	7.16E-8	3.06	5.63E-7	3.02	9.58E-8	3.04	
	232	2.41E-5		2.58E-4		3.16E-5		1.07E-5		8.42E-5		1.51E-5		
	928	1.53E-6	3.98	3.09E-5	3.06	2.24E-6	3.82	5.79E-7	4.22	5.89E-6	3.84	8.47E-7	4.16	
P^3	3712	9.57E-8	4.00	2.06E-6	3.91	1.57E-7	3.83	3.41E-8	4.08	3.93E-7	3.91	5.14E-8	4.04	
	14848	6.12E-9	3.96	1.31E-7	3.97	1.06E-8	3.89	2.06E-9	4.05	2.20E-8	4.15	3.15E-9	4.03	
	59392	4.05E-10	3.91	8.48E-9	3.96	7.20E-10	3.88	1.27E-10	4.02	1.51E-9	3.86	1.95E-10	4.01	



Figure 3.3: Burgers equation. $T = 1.5/\pi$. The surface of the solution. RKDG with HWENO limiter. Top: $C_k=0.001$; bottom: all cells are defined as troubled cells. Left: second order (k = 1); middle: third order (k = 2); right: fourth order (k = 3).

refined triangular meshes are used in the computation, which consist of 16×11 (320 cells), 32×21 (1280 cells), 64×41 (5120 cells), and 128×81 (20480 cells) points, respectively. The first number refers to the number of points in the circular direction, and the second designates the number of concentric circles in the mesh. The sample mesh and its zoomedin mesh are shown in Figure 3.4. The radius of the cylinder is 0.5 and the computational domain is set as $\{(x, y) : 0.5 \leq \sqrt{x^2 + y^2} \leq 20\}$. Mach number contours are shown in Figure 3.5 and Figure 3.6. Following [24], we measure the entropy production given by

$$\frac{S_0 - S_\infty}{S_\infty} = \frac{\frac{p_0}{\rho_0^\gamma}}{\frac{p_\infty}{\rho_\infty^\gamma}} - 1, \qquad (3.2)$$

as the error measurement, where S_0 is the local entropy and S_{∞} is the far field entropy. The L^2 errors and numerical orders of accuracy for the entropy for the RKDG methods with the HWENO limiters on every cell are shown in Table 3.3. We can see that the new HWENO limiter can maintain the designed high order accuracy of the RKDG method even in the extreme situation that HWENO limiter is applied on every cell.



Figure 3.4: Subsonic cylinder test case. Sample mesh and zoomed-in mesh.

Example 3.5. Double Mach reflection problem. This model problem is originally from [34]. We solve the Euler equations (2.6) in a computational domain of $[0, 4] \times [0, 1]$. The reflection



Figure 3.5: Subsonic cylinder test case. RKDG with HWENO limiter on every cell. Zoomedin pictures around the cylinder. 30 equally spaced Mach number contours from 0.04 to 0.94. Left: second order (k = 1); middle: third order (k = 2); right: fourth order (k = 3). From top to bottom: the numbers of points on the inner and outer boundaries are the same as 16 and 32.

Table 3.3:Subsonic cylinder test case.	L^2	entropy	errors	and	orders	of	convergence.	RKDG
with HWENO limiter on every cell.								

RKDG with HWENO limiter on every cell										
	P^1		P^2		P^3					
cell	L^2 error	order	L^2 error	order	L^2 error	order				
320	8.73E-3		3.11E-3		9.72E-4					
1280	1.42E-3	2.62	3.88E-4	3.00	6.68E-5	3.86				
5120	2.88E-4	2.30	3.65E-5	3.41	3.87E-6	4.10				
20480	5.97E-5	2.27	4.31E-6	3.08	2.69E-7	3.85				



Figure 3.6: Subsonic cylinder test case. RKDG with HWENO limiter on every cell. Zoomedin pictures around the cylinder. 30 equally spaced Mach number contours from 0.04 to 0.94. Left: second order (k = 1); middle: third order (k = 2); right: fourth order (k = 3). From top to bottom: the numbers of points on the inner and outer boundaries are the same as 64 and 128.

boundary condition is used at the wall, while for the rest of the bottom boundary (the part from x = 0 to $x = \frac{1}{6}$), the exact post-shock condition is imposed. At the top boundary is the exact motion of the Mach 10 shock. The results shown are at t = 0.2. Three different orders of accuracy for the RKDG methods with the HWENO limiters, k=1, k=2 and k=3 (second order, third order and fourth order), are used in this numerical experiment. A sample mesh is shown in Figure 3.7. In Table 3.4 we give the percentage of cells declared to be troubled cells for different RKDG methods with the HWENO limiters. The simulation results are shown in Figure 3.8. The "zoomed-in" pictures around the double Mach stem to show more details are given in Figure 3.9. The troubled cells identified at the last time step are shown in Figure 3.10. Clearly, the resolution improves with an increasing k on the same mesh although the percentage of troubled cells is simultaneously increasing.



Figure 3.7: Double Mach reflection problem. Sample mesh.

Table 3.4: Double Mach reflection problem. The maximum and average percentages of troubled cells subject to the HWENO limiting.

	Percentage of the troubled cells									
	cell length h	1/200		cell length h	1/200		cell length h	1/200		
P^1	maximum percent	3.50	P^2	maximum percent	8.50	P^3	maximum percent	15.3		
	average percent	1.53		average percent	5.12		average percent	9.41		

Example 3.6. A Mach 3 wind tunnel with a step. This model problem is also originally from [34]. The setup of the problem is as follows. The wind tunnel is 1 length unit wide and 3 length units long. The step is 0.2 length units high and is located 0.6 length units from the left-hand end of the tunnel. The problem is initialized by a right-going Mach 3 flow.



Figure 3.8: Double Mach reflection problem. RKDG with HWENO limiter. Top: second order (k = 1); middle: third order (k = 2); bottom: fourth order (k = 3). 30 equally spaced density contours from 1.5 to 21.5. The mesh points on the boundary are uniformly distributed with cell length h = 1/200.



Figure 3.9: Double Mach reflection problem. RKDG with HWENO limiter. Left: second order (k = 1); right: third order (k = 2); bottom: fourth order (k = 3). Zoomed-in pictures around the Mach stem. 30 equally spaced density contours from 1.5 to 21.5. The mesh points on the boundary are uniformly distributed with cell length h = 1/200.



Figure 3.10: Double Mach reflection problem. RKDG with HWENO limiter. Top: second order (k = 1); middle: third order (k = 2); bottom: fourth order (k = 3). Troubled cells. Circles denote triangles which are identified as troubled cells subject to the HWENO limiting at the last time step. The mesh points on the boundary are uniformly distributed with cell length h = 1/200.

Reflective boundary conditions are applied along the wall of the tunnel and inflow/outflow boundary conditions are applied at the entrance/exit. The results are shown at t = 4. We present a sample triangulation of the whole region $[0,3] \times [0,1]$ in Figure 3.11. In Table 3.5 we give the percentage of cells declared to be troubled cells for different RKDG methods with the HWENO limiters. In Figure 3.12, we show 30 equally spaced density contours from 0.32 to 6.15 computed by the second order, third order and fourth order RKDG methods with the HWENO limiters, respectively. The troubled cells identified at the last time step are shown in Figure 3.13. We can clearly observe that the fourth order scheme gives better resolution than the former two schemes, especially for the resolution of the physical instability and roll-up of the contact line.



Figure 3.11: Forward step problem. Sample mesh.

Table 3.5: Forward step problem. The maximum and average percentages of troubled cells subject to the HWENO limiting.

	Percentage of the troubled cells									
	cell length h	1/100		cell length h	1/100		cell length h	1/100		
P^1	maximum percent	5.94	P^2	maximum percent	9.25	P^3	maximum percent	14.5		
	average percent	4.02		average percent	6.21		average percent	9.59		

Example 3.7. We consider inviscid Euler transonic flow past a single NACA0012 airfoil configuration with Mach number $M_{\infty} = 0.8$, angle of attack $\alpha = 1.25^{\circ}$ and with $M_{\infty} = 0.85$, angle of attack $\alpha = 1^{\circ}$. The computational domain is $[-15, 15] \times [-15, 15]$. The mesh used in the computation is shown in Figure 3.14, consisting of 9340 elements with the maximum



Figure 3.12: Forward step problem. RKDG with HWENO limiter. Top: second order (k = 1); middle: third order (k = 2); bottom: fourth order (k = 3). 30 equally spaced density contours from 0.32 to 6.15. The mesh points on the boundary are uniformly distributed with cell length h = 1/100.



Figure 3.13: Forward step problem. RKDG with HWENO limiter. Top: second order (k = 1); middle: third order (k = 2); bottom: fourth order (k = 3). Troubled cells. Circles denote triangles which are identified as troubled cell subject to the HWENO limiting at the last time step. The mesh points on the boundary are uniformly distributed with cell length h = 1/100.

diameter of the circumcircle being 1.4188 and the minimum diameter being 0.0031 near the airfoil. The mesh uses curved cells near the airfoil. The second order, third order and fourth order RKDG methods with the HWENO limiters are used in the numerical experiment. In Table 3.6, we document the percentage of cells declared to be troubled cells for different orders of RKDG methods with the HWENO limiters. Mach number distributions are shown in Figure 3.15. Figure 3.16 shows the entropy (3.2) distributions plotted with a two-point line, a three-point line and a four-point line on each cell-face for solutions obtained by secondorder, third-order and fourth order RKDG methods with HWENO limiters, respectively. Figure 3.17 shows the pressure (C_p) distributions plotted with a two-point line, a threepoint line and a four-point line on each cell-face for solutions obtained by secondorder, third-order and fourth order RKDG methods with HWENO limiters, respectively. Figure 3.17 shows the pressure (C_p) distributions plotted with a two-point line, a threepoint line and a four-point line on each cell-face for solutions obtained by second-order, third-order and fourth-order RKDG methods with HWENO limiters, respectively. We can see that the third order and fourth order schemes have better resolutions than the second order scheme. The troubled cells identified at the last time step are shown in Figure 3.18 and very few cells are identified as troubled cells.



Figure 3.14: NACA0012 airfoil mesh. Zoomed-in mesh.

Example 3.8. The two dimensional Sedov problem [18, 30]. The initial conditions are: $\rho=1, \mu=0, \nu=0, E=10^{-12}$ everywhere except that the energy in the lower left corner cell is the constant $\frac{0.244816}{\Delta x \Delta y}$ and $\gamma = 1.4$. Symmetry boundary conditions are applied at the left and



Figure 3.15: NACA0012 airfoil. RKDG with HWENO limiter. Top: second order (k = 1); middle: third order (k = 2); bottom: fourth order (k = 3). Mach number. Left: $M_{\infty} = 0.8$, angle of attack $\alpha = 1.25^{\circ}$, 30 equally spaced Mach number contours from 0.172 to 1.325; right: $M_{\infty} = 0.85$, angle of attack $\alpha = 1^{\circ}$, 30 equally spaced Mach number contours from 0.158 to 1.357.



Figure 3.16: NACA0012 airfoil. RKDG with HWENO limiter. Top: second order (k = 1), a two-point line on each cell-face; middle: third order (k = 2), a three-point line on each cell-face; bottom: fourth order (k = 3), a four-point line on each cell-face. Entropy. Left: $M_{\infty} = 0.8$, angle of attack $\alpha = 1.25^{\circ}$; right: $M_{\infty} = 0.85$, angle of attack $\alpha = 1^{\circ}$.



Figure 3.17: NACA0012 airfoil. RKDG with HWENO limiter. Top: second order (k = 1), a two-point line on each cell-face; middle: third order (k = 2), a three-point line on each cell-face; bottom: fourth order (k = 3), a four-point line on each cell-face. Pressure distribution. Left: $M_{\infty} = 0.8$, angle of attack $\alpha = 1.25^{\circ}$; right: $M_{\infty} = 0.85$, angle of attack $\alpha = 1^{\circ}$.



Figure 3.18: NACA0012 airfoil. RKDG with HWENO limiter. Top: second order (k = 1); middle: third order (k = 2); bottom: fourth order (k = 3). Troubled cells. Circles denote triangles which are identified as troubled cells subject to the HWENO limiting at the last time step. Left: $M_{\infty} = 0.8$, angle of attack $\alpha = 1.25^{\circ}$; right: $M_{\infty} = 0.85$, angle of attack $\alpha = 1^{\circ}$.

Table 3.6: NACA0012 airfoil problem. The maximum and average percentages of troubled cells subject to the HWENO limiting.

	$M_{\infty} = 0.8$, angle of att	tack $\alpha = 1.25^{\circ}$	$M_{\infty} = 0.85$, angle of attack $\alpha = 1^{\circ}$			
P^1	maximum percentage	15.2	maximum percentage	15.9		
	average percentage	8.79	average percentage	9.22		
P^2	maximum percentage	21.2	maximum percentage	22.5		
	average percentage	12.2	average percentage	14.1		
P^3	maximum percentage	28.6	maximum percentage	29.3		
	average percentage	15.1	average percentage	17.3		

bottom boundaries, thus making it possible to compute only the upper-right quarter of the whole problem. The final computing time is t=1. We present a sample triangulation of the whole region $[0, 1.1] \times [0, 1.1]$ in Figure 3.19. In Table 3.7, we document the percentage of cells declared to be troubled cells for different orders of RKDG methods with the HWENO limiters. The results of the second, third and fourth order RKDG methods with the HWENO limiters are shown in Figure 3.20. This is a rather extreme test case, many limiters may fail to control the appearance of negative pressure, causing instability, including the one in [37]. We can see from Figure 3.20 that our new limiter works well for this test case.



Figure 3.19: 2D Sedov problem. Sample mesh.

Table 3.7: 2D Sedov problem. The maximum and average percentages of troubled cells subject to the HWENO limiting.



Figure 3.20: 2D Sedov problem. RKDG with HWENO limiter. Top: second order (k = 1); middle: third order (k = 2); bottom: fourth order (k = 3). From left to right: 30 equally spaced density contours from 0.95 to 5; density is projected to the radical coordinates; circles denote triangles which are identified as troubled cells subject to the HWENO limiting at the last time step. Solid line: the exact solution; squares: the numerical results. The mesh points on the boundary are uniformly distributed with cell length h = 1.1/80.

4 Concluding remarks

We have generalized the Runge-Kutta discontinuous Galerkin (RKDG) methods with a new type of simple Hermite weighted essentially non-oscillatory (HWENO) limiters for solving hyperbolic conservation laws on two dimensional unstructured meshes. The procedure of the new HWENO limiters for the RKDG methods is specified as follows: the KXRCF technique [20] is used to detect the troubled cells which need further HWENO reconstruction, then the new polynomials are reconstructed using the available DG solution polynomials on the troubled cell and its three adjacent neighbors with suitable modification for sustaining the conservative property. The modification procedure is performed in a least squares fashion [11]. Several numerical benchmark tests of scalar equation and compressible inviscid Euler equations are given to demonstrate the good performance in comparison with those specified in earlier literature which use wider stencils and more sophisticated WENO or HWENO limiters. In future work, we would like to extend such simple HWENO limiting procedure to three dimensional tetrahedral meshes.

References

- D.S. Balsara and C.-W. Shu, Monotonicity preserving weighted essentially nonoscillatory schemes with increasingly high order of accuracy, Journal of Computational Physics, 160 (2000), 405-452.
- F. Bassi and S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, Journal of Computational Physics, 138 (1997), 251-285.
- [3] R. Biswas, K.D. Devine and J. Flaherty, Parallel, adaptive finite element methods for conservation laws, Applied Numerical Mathematics, 14 (1994), 255-283.
- [4] A. Burbeau, P. Sagaut and C.H. Bruneau, A problem-independent limiter for high-order Runge-Kutta discontinuous Galerkin methods, Journal of Computational Physics, 169

(2001), 111-150.

- B. Cockburn, Discontinuous Galerkin methods for convection-dominated problems, in T. Barth and H. Deconinck, editors, High-Order Methods for Computational Physics, Volume 9 of Lecture Notes in Computational Science and Engineering, Springer, Berlin, 1999, 69-224.
- B. Cockburn, S. Hou and C.-W. Shu, The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case, Mathematics of Computation, 54 (1990), 545-581.
- B. Cockburn, S.-Y. Lin and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one dimensional systems, Journal of Computational Physics, 84 (1989), 90-113.
- [8] B. Cockburn and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, Mathematics of Computation, 52 (1989), 411-435.
- B. Cockburn and C.-W. Shu, The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, Journal of Computational Physics, 141 (1998), 199-224.
- [10] B. Cockburn and C.-W. Shu, Runge-Kutta discontinuous Galerkin method for convection-dominated problems, Journal of Scientific Computing, 16 (2001), 173-261.
- [11] M. Dumbser, D.S. Balsara, E.F. Toro and 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.

- [12] M. Dumbser and M. Käser, Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems, Journal of Computational Physics, 221 (2007), 693-723.
- [13] M. Dumbser, O. Zanotti, R. Loubère and S. Diot, A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws, Journal of Computational Physics, 278 (2014), 47-75.
- [14] O. Friedrichs, Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids, Journal of Computational Physics, 144 (1998), 194-212.
- [15] J. Hesthaven and T. Warburton, Nodal Discontinuous Galerkin Methods, Springer, New York, 2008.
- [16] C. Hu and C.-W. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, Journal of Computational Physics, 150 (1999), 97-127.
- [17] G. Jiang and C.-W. Shu, Efficient implementation of weighted ENO schemes, Journal of Computational Physics, 126 (1996), 202-228.
- [18] V.P. Korobeinikov, Problems of Point-Blast Theory, American Institute of Physics, 1991.
- [19] L. Krivodonova and M. Berger, High-order implementation of solid wall boundary conditions in curved geometries, Journal of Computational Physics, 211 (2006), 492-512.
- [20] L. Krivodonova, J. Xin, J.-F. Remacle, N. Chevaugeon and J.E. Flaherty, Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws, Applied Numerical Mathematics, 48 (2004), 323-338.
- [21] B. Li, Discontinuous Finite Elements in Fluid Dynamics and Heat Transfer, Birkhauser, Basel, 2006.

- [22] T. Linde and P.L. Roe, *Robust Euler codes*, in: 13th Computational Fluid Dynamics Conference, AIAA Paper-97-2098.
- [23] X. Liu, S. Osher and T. Chan, Weighted essentially non-oscillatory schemes, Journal of Computational Physics, 115 (1994), 200-212.
- [24] H. Luo, J.D. Baum and R. Lohner, A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids, Journal of Computational Physics, 225 (2007), 686-713.
- [25] H. Luo, J.D. Baum and R. Lohner, On the computation of steady-state compressible flows using a discontinuous Galerkin method, International Journal for Numerical Methods in Engineering, 73 (2008), 597-623.
- [26] J. Qiu and C.-W. Shu, Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method: one dimensional case, Journal of Computational Physics, 193 (2003), 115-135.
- [27] J. Qiu and C.-W. Shu, Runge-Kutta discontinuous Galerkin method using WENO limiters, SIAM Journal on Scientific Computing, 26 (2005), 907-929.
- [28] J. Qiu and C.-W. Shu, A comparison of troubled-cell indicators for Runge-Kutta discontinuous Galerkin methods using weighted essentially nonoscillatory limiters, SIAM Journal on Scientific Computing, 27 (2005), 995-1013.
- [29] J. Qiu and C.-W. Shu, Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method II: two dimensional case, Computers and Fluids, 34 (2005), 642-663.
- [30] L.I. Sedov, Similarity and Dimensional Methods in Mechanics, Academic Press, New York, 1959.

- [31] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, In Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, B. Cockburn, C. Johnson, C.-W. Shu and E. Tadmor (Editor: A. Quarteroni), Lecture Notes in Mathematics, volume 1697, Springer, 1998, 325-432.
- [32] C.-W. Shu, Discontinuous Galerkin methods: general approach and stability, in Numerical Solutions of Partial Differential Equations, S. Bertoluzza, S. Falletta, G. Russo and C.-W. Shu, Advanced Courses in Mathematics CRM Barcelona, Birkhäuser, Basel, 2009, 149-201.
- [33] C.-W. Shu and S. Osher, Efficient implementation of essentially non-oscillatory shockcapturing schemes, Journal of Computational Physics, 77 (1988), 439-471.
- [34] P. Woodward and P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, Journal of Computational Physics, 54 (1984), 115-173.
- [35] X. Zhong and C.-W. Shu, A simple weighted essentially nonoscillatory limiter for Runge-Kutta discontinuous Galerkin methods, Journal of Computational Physics, 232 (2013), 397-415.
- [36] J. Zhu, J. Qiu, C.-W. Shu and M. Dumbser, Runge-Kutta discontinuous Galerkin method using WENO limiters II: Unstructured meshes, Journal of Computational Physics, 227 (2008), 4330-4353.
- [37] J. Zhu, X. Zhong, C.-W. Shu and J. Qiu, Runge-Kutta discontinuous Galerkin method using a new type of WENO limiters on unstructured meshes, Journal of Computational Physics, 248 (2013), 200-220.
- [38] J. Zhu, X. Zhong, C.-W. Shu and J. Qiu: Runge-Kutta discontinuous Galerkin method with a simple and compact Hermite WENO limiter, Communications in Computational Physics, 19 (2016), 944-969.