Available online at www.sciencedirect.com
SCIENCE DIRECT®
Comput. Methods Appl. Mech. Engrg. 194 (2005) 4528–4543
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

The discontinuous Galerkin method with Lax–Wendroff type time discretizations

Jianxian Qiu [a,1], Michael Dumbser [b], Chi-Wang Shu [c,*,2]

[a] *Department of Mechanical Engineering, National University of Singapore, Singapore 119260, Singapore*
[b] *Institut fuer Aero- und Gasdynamik, Pfaffenwaldring 21, 70550 Stuttgart, Germany*
[c] *Division of Applied Mathematics, Brown University, Box F, Providence, RI 02912, USA*

Received 13 January 2004; received in revised form 25 October 2004; accepted 29 November 2004

Abstract

In this paper we develop a Lax–Wendroff time discretization procedure for the discontinuous Galerkin method (LWDG) to solve hyperbolic conservation laws. This is an alternative method for time discretization to the popular total variation diminishing (TVD) Runge–Kutta time discretizations. The LWDG is a one step, explicit, high order finite element method. The limiter is performed once every time step. As a result, LWDG is more compact than Runge–Kutta discontinuous Galerkin (RKDG) and the Lax–Wendroff time discretization procedure is more cost effective than the Runge–Kutta time discretizations for certain problems including two-dimensional Euler systems of compressible gas dynamics when nonlinear limiters are applied.

Keywords: Discontinuous Galerkin method; Lax–Wendroff type time discretization; Runge–Kutta method; Limiter; WENO scheme; High order accuracy

* Corresponding author.
E-mail addresses: mpeqjx@nus.edu.sg (J. Qiu), michael.dumbser@iag.uni-stuttgart.de (M. Dumbser), shu@dam.brown.edu (C.-W. Shu).

[1] The research of the author is supported by NUS Research Project R-265-000-118-112.
[2] The research of this author is supported by NNSFC grant 10028103 while he is in residence at the Department of Mathematics, University of Science and Technology of China, Hefei, Anhui 230026, PR China. Additional support is provided by ARO grant DAAD19-00-1-0405 and NSF grant DMS-0207451.

0045-7825/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2004.11.007

1. Introduction

In this paper, we study an alternative method for time discretization, namely the Lax–Wendroff type time discretization [18], to the popular TVD Runge–Kutta time discretization in [30,8], for the discontinuous Galerkin methods in solving nonlinear hyperbolic conservation law systems

$$\begin{cases} u_t + \nabla \cdot f(u) = 0, \\ u(x,0) = u_0(x). \end{cases} \tag{1.1}$$

The first discontinuous Galerkin (DG) method was introduced in 1973 by Reed and Hill [24], in the framework of neutron transport (steady state linear hyperbolic equations). A major development of the DG method was carried out by Cockburn et al. in a series of papers [9,8,7,6,10], in which they established a framework to easily solve *nonlinear* time dependent hyperbolic conservation laws (1.1) using explicit, nonlinearly stable high order Runge–Kutta time discretizations [30] and DG discretization in space with exact or approximate Riemann solvers as interface fluxes and total variation bounded (TVB) limiter [27] to achieve nonoscillatory properties for strong shocks. These schemes are termed RKDG methods.

In RKDG method, DG is a spatial discretization procedure, namely, it is a procedure to approximate the spatial derivative terms in (1.1). The time derivative term there is discretized by explicit, nonlinearly stable high order Runge–Kutta time discretizations [30]. An alternative approach could be using a Lax–Wendroff type time discretization procedure, which is also called the Taylor type referring to a Taylor expansion in time or the Cauchy–Kowalewski type referring to the similar Cauchy–Kowalewski procedure in partial differential equations (PDEs) [33]. This approach is based on the idea of the classical Lax–Wendroff scheme [18], and it relies on converting all the time derivatives in a temporal Taylor expansion into spatial derivatives by repeatedly using the PDE and its differentiated versions. The spatial derivatives are then discretized by, e.g. the DG approximations.

RKDG methods have the advantage of simplicity, both in concept and in coding. They also enjoy good stability properties when the TVD type Runge–Kutta or multi-step methods are used [30,28]. Thus RKDG methods are very popular in applications. To obtain provable nonlinear stability for scalar cases, a TVD time discretization [30,28] is preferred. There is however an order barrier for TVD Runge–Kutta methods with positive coefficients: they cannot be higher than fourth order accurate [25]. Even if we ignore the TVD requirement for the time discretization, fifth and higher order Runge–Kutta methods need more stages than the order of accuracy, hence the efficiency would be reduced for very high order time discretizations.

The second approach, the Lax–Wendroff type time discretization, which is also referred to as the Taylor–Galerkin method for finite element methods, usually produces the same high order accuracy with a smaller effective stencil than that of the first approach, and it uses more extensively the original PDE. However, the formulation and coding of this procedure could be quite complicated, especially for multi-dimensional systems.

We first review briefly the Taylor–Galerkin methods in the literature. A high order Taylor–Galerkin scheme to solve initial-boundary value problems for first order, linear hyperbolic systems was developed by Safjan and Oden [26]. A predictor–corrector type and unconditionally stable higher order Taylor–Galerkin method scheme was presented by Youn and Park [38]. In Tabarrok and Su [32], a semi-implicit Taylor–Galerkin finite element method was developed for solutions of incompressible flows with heat transfer. In Colin and Rudgyard [12], the authors developed a third order low dissipation Taylor–Galerkin finite-element scheme within an unstructured/hybrid parallel solver for unsteady large eddy simulation. In Lukácová-Medvid'ová and Warnecke [19], a Lax–Wendroff type second order evolution Galerkin method for multi-dimensional hyperbolic systems was discussed. The references quoted above were all based on continuous finite element methods. In [2,3], Choe and Holsapple developed a Taylor–Galerkin method based on discontinuous finite elements. Their method is a one-step, explicit finite element scheme, second order

accurate in both time and space, and is developed for the computation of weak solutions of nonlinear hyperbolic conservation laws in one dimension. Mixed finite element methodology is used to treat the second derivative terms in the Lax–Wendroff procedure, which limits the method to second order accuracy to avoid a global solver. More recently, Dumbser [13] developed the ADER (Arbitrary high order schemes using DERivatives, see [34]) discontinuous Galerkin method for solving the linear aeroacoustics systems. ADER methods also use the Lax–Wendroff procedure to convert time derivatives to spatial derivatives, so the method in [13] is essentially the same as our method in this paper for the linear case. Dumbser and Munz [14] are also extending the ADER discontinuous Galerkin method to the nonlinear case using generalized Riemann solvers [36]. The Lax–Wendroff type time discretization was also used in high order finite volume schemes [15,34,35] and finite difference schemes [20].

In this paper we explore the Lax–Wendroff type time discretization procedure for DG spatial discretizations up to third order (P^2 case). Unlike in [2,3], we do not use a mixed finite element formulation to treat the higher order derivatives, hence in principle our methodology can be easily generalized to arbitrary order of accuracy. The resulting schemes, however, are more complex to formulate and to code than RKDG methods using the TVD Runge–Kutta time discretizations, hence unless there is an advantage in CPU timing for the same accuracy, they will not be competitive. Fortunately, we demonstrate numerically that the Lax–Wendroff time discretization procedure adopted in this paper is more cost effective and produces sharper discontinuity transition than RKDG methods for certain problems including two-dimensional Euler systems of compressible gas dynamics, when nonlinear limiters are used. Of course, this procedure becomes progressively more complicated with more complicated PDEs and/or higher order time accuracy, hence we do not expect it to be always cost effective relative to the standard RKDG methods. The exact break-even point depends on many factors, such as the specific PDE (how complicated the time derivatives can be represented by spatial derivatives via the PDE), the relative cost of the limiter to the core DG procedure, the specific implementation including efficient cache usage, etc.

In this paper we do not address the important issue of time discretization for PDEs with diffusion terms and/or with stiff source terms, which calls for hybrid explicit/implicit time discretization. There are good Runge–Kutta methods to easily achieve this, see, e.g. [40]. The Lax–Wendroff procedure in this paper can also be adapted for such purpose, through careful Taylor expansions.

The organization of this paper is as follows. In Section 2, we describe in detail the construction and implementation of the high order DG method with a Lax–Wendroff type time discretization, for one and two-dimensional scalar and system equation (1.1). In Section 3 we provide extensive numerical examples to demonstrate the behavior of the schemes and to perform a comparison with RKDG methods. Concluding remarks are given in Section 4.

2. Construction and implementation of the scheme

In this section we describe in detail the construction and implementation of the discontinuous Galerkin method with a Lax–Wendroff type time discretization, for one- and two-dimensional scalar and system conservation laws.

2.1. One-dimensional case

Consider the one-dimensional scalar conservation laws:

$$\begin{cases} u_t + f(u)_x = 0, \\ u(x,0) = u_0(x). \end{cases} \tag{2.1}$$

We denote the cells by $I_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, the cell centers by $x_i = \frac{1}{2}\left(x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}}\right)$ and the cell sizes by $\Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$. Let Δt be the time step, $t^{n+1} = t^n + \Delta t$. By a temporal Taylor expansion we obtain

$$u(x, t + \Delta t) = u(x, t) + \Delta t u_t + \frac{\Delta t^2}{2} u_{tt} + \frac{\Delta t^3}{6} u_{ttt} + \cdots. \tag{2.2}$$

If we would like to obtain $(k + 1)$th order accuracy in time, we would need to approximate the first $k + 1$ time derivatives: $u_t, \ldots, \frac{\partial^{(k+1)} u}{\partial t^{k+1}}$. We will proceed up to third order in time in this paper, although the procedure can be naturally extended to any higher orders.

The temporal derivative terms in (2.2) can be replaced with the spatial ones using the governing equation (2.1):

$$u_t = -f(u)_x = -f'(u)u_x,$$

$$u_{tt} = -(f'(u)u_t)_x = -f''(u)u_x u_t - f'(u)u_{xt},$$

$$u_{xt} = -f''(u)(u_x)^2 - f'(u)u_{xx},$$

$$u_{ttt} = -(f''(u)(u_t)^2 + f'(u)u_{tt})_x.$$

Then we can rewrite the approximation to (2.2) up to third order as:

$$u(x, t + \Delta t) = u(x, t) - \Delta t F_x. \tag{2.3}$$

with $F = f + \frac{\Delta t}{2} f'(u)u_t + \frac{\Delta t^2}{6}(f''(u)(u_t)^2 + f'(u)u_{tt})$. The standard discontinuous Galerkin method is then used to discretize F_x in (2.3), described in detail below.

The DG solution as well as the test function space is given by $V_h^k = \{p : p|_{I_i} \in P^k(I_i)\}$, where $P^k(I_i)$ is the space of polynomials of degree $\leqslant k$ on the cell I_i. We adopt a local orthogonal basis over I_i, $\{v_l^{(i)}(x), l = 0, 1, \ldots, k\}$, namely the scaled Legendre polynomials

$$v_0^{(i)}(x) = 1, \quad v_1^{(i)}(x) = \frac{x - x_i}{\Delta x_i}, \quad v_2^{(i)}(x) = \left(\frac{x - x_i}{\Delta x_i}\right)^2 - \frac{1}{12}, \ldots.$$

Other basis functions can be used as well, without changing the numerical method, since the finite element DG method depends only on the choice of space V_h^k, not on the choice of its basis functions.

The numerical solution $u^h(x, t)$ in the space V_h^k can be written as:

$$u^h(x, t) = \sum_{l=0}^{k} u_i^{(l)}(t) v_l^{(i)}(x), \quad \text{for } x \in I_i, \tag{2.4}$$

and the degrees of freedom $u_i^{(l)}(t)$ are the moments defined by

$$u_i^{(l)}(t) = \frac{1}{a_l} \int_{I_i} u^h(x, t) v_l^{(i)}(x)\, \mathrm{d}x, \quad l = 0, 1, \ldots, k,$$

where $a_l = \int_{I_i} (v_l^{(i)}(x))^2\, \mathrm{d}x$ are the normalization constants since the basis is not orthonormal. In order to determine the approximate solution, we evolve the degrees of freedom $u_i^{(l)}$:

$$u_i^{(l)}(t^{n+1}) = u_i^{(l)}(t^n) + \frac{1}{a_l}\left(-\int_{I_i} F \frac{\mathrm{d}}{\mathrm{d}x} v_l^{(i)}(x)\, \mathrm{d}x + \hat{F}_{i+1/2} v_l^{(i)}(x_{i+1/2}) - \hat{F}_{i-1/2} v_l^{(i)}(x_{i-1/2})\right) = 0, \quad l = 0, 1, \ldots, k, \tag{2.5}$$

where $\hat{F}_{i+1/2}$ is a numerical flux which depends on the values of the numerical solution u^h and its spatial derivatives at the cell interface $x_{i+1/2}$, both from the left and from the right. This numerical flux is related

to the so-called generalized Riemann solvers [36]. In this paper, we use the following simple Lax–Friedrichs flux

$$\hat{F}_{i+1/2} = \frac{1}{2}\left(F^-_{i+1/2} + F^+_{i+1/2} - \alpha(u^+_{i+1/2} - u^-_{i+1/2})\right),$$

where $u^{\pm}_{i+1/2}$ and $F^{\pm}_{i+1/2}$ are the left and right limits of the discontinuous solution u^h and the flux F at the cell interface $x_{i+1/2}$, and $\alpha = \max_u |f'(u)|$. For the system case, the maximum is taken for the eigenvalues of the Jacobian $f'(u)$. The integral term in (2.5) can be computed either exactly or by a suitable numerical quadrature accurate to at least $O(\Delta x^{k+l+2})$. In this paper we use two and three point Gaussian quadratures for $k = 1$ and $k = 2$ respectively.

An important component of DG methods for solving conservation laws (1.1) with strong shocks in the solutions is a nonlinear limiter, which is applied to control spurious oscillations. Although many limiters exist in the literature, e.g. [9,8,7,6,10,1,4], they tend to degenerate accuracy when mistakenly used in smooth regions of the solution. In [21–23], we initialized a study of using weighted essentially nonoscillatory (WENO) and Hermite WENO (HWENO) methodology as limiters for RKDG methods. The idea is to first identify "troubled cells", namely those cells where limiting might be needed, then to abandon all moments in those cells except the cell averages and reconstruct those moments from the information of neighboring cells using a WENO or HWENO methodology. This technique works quite well in our one and two-dimensional test problems [21–23]. In this paper we adopt the WENO limiter developed in [21] for $k = 1$ and HWENO limiter in [22,23] for $k = 2$ to maintain the compactness of the DG method. The limiting procedure is performed in every inner Runge–Kutta stage for the RKDG method. Thus, when we use the third order TVD Runge–Kutta time discretization [30], the limiting procedure is performed three times for one time step; but we only need to perform the limiting procedure once per time step for the one step LWDG methods. This actually accounts for a major saving of computational cost of the LWDG methods over the RKDG methods.

For systems of conservation laws (2.1), $u(x,t) = (u^1(x,t),\ldots,u^m(x,t))^{\mathrm{T}}$ is a vector and $f(u) = (f^1(u^1,\ldots,u^m),\ldots,f^m(u^1,\ldots,u^m))^{\mathrm{T}}$ is a vector function of u. As before, the time derivatives in (2.2) are replaced by the spatial derivatives using the PDE. The DG discretization is then performed on each component. In order to achieve better qualities at the price of more complicated and costly computations, we use a local characteristic field decomposition in the limiting procedure. For the details of such local characteristic field decompositions, we refer to [29]. The limiter and the WENO and HWENO reconstructions within the limiters are all performed under local characteristic projections. We note that the second and higher order time derivatives, when converted to spatial derivatives as before, involve expressions like $f'(u)$ which is a matrix (the Jacobian), $f''(u)$ which is a 3D "matrix" (tensor), etc., which could become very complicated. The symbolic manipulator Maple® [16] is used to avoid mistakes. The code is also quite long and messy compared with codes using Runge–Kutta time discretizations. However, we will see in the next section that one can save CPU time by this approach for certain problems.

2.2. Two-dimensional cases

Consider the two-dimensional conservation laws:

$$\begin{cases} u_t + f(u)_x + g(u)_y = 0, \\ u(x,y,0) = u_0(x,y). \end{cases} \tag{2.6}$$

By a temporal Taylor expansion we obtain

$$u(x,y,t+\Delta t) = u(x,y,t) + \Delta t u_t + \frac{\Delta t^2}{2} u_{tt} + \frac{\Delta t^3}{6} u_{ttt} + \cdots.$$

For example, for third order accuracy in time we would need to reconstruct three time derivatives: u_t, u_{tt}, u_{ttt}.

We again use the PDE (2.6) to replace time derivatives by spatial derivatives.

$$u_t = -f(u)_x - g(u)_y = -f'(u)u_x - g'(u)u_y,$$

$$u_{tt} = -(f'(u)u_t)_x - (g'(u)u_t)_y = -(f''(u)u_x u_t + f'(u)u_{xt} + g''(u)u_y u_t + g'(u)u_{yt}),$$

$$u_{xt} = -(f''(u)(u_x)^2 + f'(u)u_{xx} + g''(u)u_x u_y + g'(u)u_{xy}),$$

$$u_{yt} = -(f''(u)u_y u_x + f'(u)u_{xy} + g''(u)(u_y)^2 + g'(u)u_{yy}),$$

$$u_{ttt} = -(f''(u)(u_t)^2 + f'(u)u_{tt})_x - (g''(u)(u_t)^2 + g'(u)u_{tt})_y.$$

Then we rewrite the approximation to (2.6) up to third order as:

$$u(x, t + \Delta t) = u(x, t) - \Delta t(F_x + G_y), \tag{2.7}$$

with

$$F = f + \frac{\Delta t}{2} f'(u)u_t + \frac{\Delta t^2}{6}(f''(u)(u_t)^2 + f'(u)u_{tt}),$$

$$G = g + \frac{\Delta t}{2} g'(u)u_t + \frac{\Delta t^2}{6}(g''(u)(u_t)^2 + g'(u)u_{tt}).$$

The standard discontinuous Galerkin method is then used to discretize F_x and G_y in (2.7).

For systems of conservation laws (2.6), the time derivatives are replaced by the spatial derivatives using the PDE. The DG discretization is then performed on each component. Again the limiter and the WENO, HWENO reconstructions are all performed under local characteristic projections. To avoid mistakes, we again use the symbolic manipulator Maple® [16] to obtain the complicated time derivative terms for the system case.

3. Numerical results

In this section we present the results of our numerical experiments for the LWDG schemes, LWDG2 for P^1 and LWDG3 for P^2, developed in the previous section, and compare them with RKDG schemes, RKDG2 for P^1 with second order Runge–Kutta method and RKDG3 for P^2 with third order Runge–Kutta method in [5–11].

We have used both uniform and nonuniform meshes in the numerical experiments, obtaining similar results. We will only show results with uniform meshes to save space.

We first remark on the important issue of CPU time of LWDG and RKDG methods. In general the LWDG methods have smaller CPU costs for the same mesh and same order of accuracy in our implementation for one and two-dimensional test cases, when the nonlinear limiters are applied, even though the CFL number of LWDG methods is smaller than that of RKDG methods for linear stability. The linear stability limit for RKDG2 is 0.333 and for LWDG2 it is 0.223. For the third order case, this linear stability limit is 0.209 for RKDG3 and it is 0.127 for LWDG3. Similar to that for the RKDG method [11], the CFL limits for the LWDG method are obtained by standard von Neumann analysis with a numerical eigenvalue solver. In our numerical experiments, the CFL numbers are taken as 0.2 and 0.12 for LWDG2 and LWDG3, and as 0.3 and 0.18 for RKDG2 and RKDG3, respectively. In Table 1, we provide a CPU time

Table 1
CPU time (s) for the LWDG and RKDG methods to compute the double Mach reflection problem in Example 3.8 for the two meshes of 120×30 and 240×60 cells

Schemes	LWDG				RKDG			
	$k = 1$		$k = 2$		$k = 1$		$k = 2$	
M	0.01	100	0.01	100	0.01	100	0.01	100
120×30	95.59	77.44	740.96	569.54	131.18	86.20	1071.05	708.64
240×60	789.10	676.51	5925.01	4860.05	920.28	737.04	8361.42	6061.02

comparison between LWDG and RKDG methods for the two-dimensional Euler equations, double Mach reflection test case in Example 3.8. The constant M in the table is the TVB constant in the limiter, see [27,21–23] for details. The computation is performed on a Compaq Digital personal workstation, 600au alpha-599 MHz with 256 MB RAM. We can see that in general the RKDG methods cost about 20–30% more in CPU time than the LWDG methods for this problem, even though the CFL number for the RKDG method used in the computation is 1.5 times of those for LWDG methods because of the limitation on linear stability. We remark that for linear problems, both LWDG and RKDG can be implemented by efficient local matrix–vector multiplications with pre-computed local matrices and vectors containing the coefficients of the solution when represented by local basis functions in the stencil of the scheme (see, e.g. [39]), hence the per time stage cost of LWDG and RKDG is the same for such linear cases, thus LWDG is more efficient than RKDG for such linear cases because LWDG is a one step method, see, e.g. [13].

3.1. Accuracy tests

We first test the accuracy of the schemes on linear scalar problems, nonlinear scalar problems and nonlinear systems. We only show the results of two-dimensional nonlinear scalar and one and two-dimensional nonlinear system problems to save space. We define the standard L_1 and L_∞ error norms by sampling the errors at 40 equally spaced points inside each cell in each direction, emulating the L_1 and L_∞ norms of the error function which is defined everywhere.

Example 3.1. We solve the following nonlinear system of Euler equations

$$u_t + f(u)_x = 0, \tag{3.1}$$

with

$$u = (\rho, \rho v, E)^{\mathrm{T}}, \quad f(u) = (\rho v, \rho v^2 + p, v(E + p))^{\mathrm{T}}.$$

Here ρ is the density, v is the velocity, E is the total energy, p is the pressure, which is related to the total energy by $E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho v^2$ with $\gamma = 1.4$. For the expression of the Jacobian $f'(u)$ for this example we refer to [15]. A similar (but longer) expression exists for the second derivative $f''(u)$. The initial condition is set to be $\rho(x, 0) = 1 + 0.2\sin(\pi x)$, $v(x, 0) = 1$, $p(x, 0) = 1$, with a 2-periodic boundary condition. The exact solution is $\rho(x, t) = 1 + 0.2\sin(\pi(x - t))$, $v(x, t) = 1$, $p(x, t) = 1$. We compute the solution up to $t = 2$. The errors and numerical orders of accuracy of the density ρ for the LWDG scheme in comparison with the results of the RKDG scheme are shown in Table 2. We can see that both schemes achieve their designed orders of accuracy with comparable errors for the same mesh.

Example 3.2. We solve the following nonlinear scalar Burgers equation in two dimensions:

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0, \tag{3.2}$$

Table 2
Euler equations

	N	LWDG				RKDG			
		L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
P^1	10	2.72E−03		6.46E−03		2.32E−03		7.49E−03	
	20	6.25E−04	2.12	2.04E−03	1.67	4.90E−04	2.24	2.07E−03	1.85
	40	1.52E−04	2.04	5.58E−04	1.87	1.16E−04	2.08	5.46E−04	1.93
	80	3.75E−05	2.02	1.45E−04	1.94	2.85E−05	2.03	1.40E−04	1.97
	160	9.32E−06	2.01	3.69E−05	1.97	7.08E−06	2.01	3.53E−05	1.98
	320	2.33E−06	2.00	9.31E−06	1.99	1.76E−06	2.00	8.89E−06	1.99
P^2	10	6.11E−04		1.50E−03		2.78E−03		4.51E−03	
	20	5.48E−05	3.48	2.23E−04	2.75	1.05E−04	4.73	4.46E−04	3.34
	40	4.67E−06	3.55	2.06E−05	3.43	2.31E−05	2.18	4.51E−05	3.31
	80	4.69E−07	3.32	1.82E−06	3.51	3.27E−06	2.82	5.36E−06	3.07
	160	5.32E−08	3.14	2.28E−07	2.99	4.21E−07	2.96	6.97E−07	2.94
	320	6.45E−09	3.04	2.86E−08	3.00	5.30E−08	2.99	9.87E−08	2.82

$\rho(x,0) = 1 + 0.2\sin(\pi x)$, $v(x,0) = 1$, $p(x,0) = 1$. LWDG comparing with RKDG. Local Lax–Friedrichs flux, using N equally spaced cells. $t = 2$. L_1 and L_∞ errors of the density ρ.

with the initial condition $u(x,y,0) = 0.5 + \sin(\pi(x+y)/2)$ and a 4-periodic boundary condition in both directions. When $t = 0.5/\pi$ the solution is still smooth. The errors and numerical orders of accuracy for the LWDG scheme in comparison with the results of the RKDG scheme are shown in Table 3. We can see again that both schemes achieve their designed orders of accuracy with comparable errors for the same mesh.

Example 3.3. We solve the following nonlinear system of Euler equations

$$\xi_t + f(\xi)_x + g(\xi)_y = 0, \tag{3.3}$$

with

$$\xi = (\rho, \rho u, \rho v, E)^{\mathrm{T}}, \quad f(\xi) = (\rho u, \rho u^2 + p, \rho uv, u(E+p))^{\mathrm{T}}, \quad g(\xi) = (\rho v, \rho uv, \rho v^2 + p, v(E+p))^{\mathrm{T}}.$$

Here ρ is the density, (u,v) is the velocity, E is the total energy, p is the pressure, which is related to the total energy by $E = \frac{p}{\gamma-1} + \frac{1}{2}\rho(u^2+v^2)$ with $\gamma = 1.4$. For the expression of the Jacobians $f'(u)$ and $g''(u)$, we refer to [15]. Similar (but longer) expressions exist for the second derivatives $f''(u)$ and $g''(u)$. The initial condition is

Table 3
Burgers equation $u_t + (u^2/2)_x + (u^2/2)_y = 0$

	$N \times N$	LWDG				RKDG			
		L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
P^1	10×10	7.19E−02		5.37E−01		6.42E−02		6.66E−01	
	20×20	1.52E−02	2.25	1.99E−01	1.44	1.54E−02	2.06	2.47E−01	1.43
	40×40	3.66E−03	2.05	6.73E−02	1.56	3.04E−03	2.34	4.33E−02	2.51
	80×80	5.69E−04	2.69	9.13E−03	2.88	5.90E−04	2.37	9.16E−03	2.24
	160×160	1.32E−04	2.10	2.33E−03	1.97	1.42E−04	2.05	2.41E−03	1.93
P^2	10×10	2.99E−02		4.96E−01		2.98E−02		5.03E−01	
	20×20	1.66E−03	4.17	4.01E−02	3.63	1.81E−03	4.04	4.09E−02	3.62
	40×40	1.73E−04	3.26	5.82E−03	2.78	1.73E−04	3.38	6.04E−03	2.76
	80×80	2.09E−05	3.05	9.51E−04	2.61	2.07E−05	3.06	1.00E−03	2.59
	160×160	2.52E−06	3.05	1.30E−04	2.87	2.49E−06	3.06	1.38E−04	2.87

Initial condition $u(x,y,0) = 0.5 + \sin(\pi(x+y)/2)$ and periodic boundary conditions. LWDG comparing with RKDG. Local Lax–Friedrichs flux, $t = 0.5/\pi$. L_1 and L_∞ errors. Uniform meshes with $N \times N$ cells.

set to be $\rho(x,y,0) = 1 + 0.2\sin(\pi(x+y))$, $u(x,y,0) = 0.7$, $v(x,y,0) = 0.3$, $p(x,y,0) = 1$, with a 2-periodic boundary condition. The exact solution is $\rho(x,y,t) = 1 + 0.2\sin(\pi(x+y-(u+v)t))$, $u = 0.7$, $v = 0.3$, $p = 1$. We compute the solution up to $t = 2$. The errors and numerical orders of accuracy of the density ρ for the LWDG scheme in comparison with the results of the RKDG scheme are shown in Table 4. We can see that both schemes achieve their designed orders of accuracy with comparable errors for the same mesh.

3.2. Test cases with shocks

We now test the performance of the LWDG method for problems containing shocks. We have also computed many more problems such as the two-dimensional forward facing step problem, but will not present all the results to save space. We only plot cell averages of the solution in the graphs.

Table 4
Euler equations

	$N \times N$	LWDG				RKDG			
		L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
P^1	10×10	2.59E−02		7.55E−02		3.48E−02		7.34E−02	
	20×20	8.76E−03	1.56	3.58E−02	1.08	6.89E−03	2.34	2.74E−02	1.42
	40×40	1.96E−03	2.16	1.06E−02	1.76	1.21E−03	2.51	7.36E−03	1.89
	80×80	1.65E−04	3.57	1.41E−03	2.91	2.33E−04	2.37	2.02E−03	1.87
	160×160	2.34E−05	2.82	2.87E−04	2.29	5.19E−05	2.17	6.45E−04	1.65
P^2	10×10	2.12E−03		7.42E−03		5.44E−03		1.39E−02	
	20×20	2.45E−04	3.12	9.98E−04	2.89	3.14E−04	4.11	1.22E−03	3.51
	40×40	2.44E−05	3.33	1.30E−04	2.94	2.66E−05	3.56	1.29E−04	3.24
	80×80	2.43E−06	3.33	1.72E−05	2.92	2.35E−06	3.50	1.71E−05	2.92
	160×160	3.16E−07	2.94	2.09E−06	3.04	2.19E−07	3.43	2.17E−06	2.97

Initial condition $\rho(x,y,0) = 1 + 0.2\sin(\pi(x+y))$, $u(x,y,0) = 0.7$, $v(x,y,0) = 0.3$, $p(x,y,0) = 1$ and periodic boundary conditions. LWDG comparing with RKDG. Local Lax–Friedrichs flux, $t = 2.0$. L_1 and L_∞ errors for the density ρ. Uniform meshes with $N \times N$ cells.

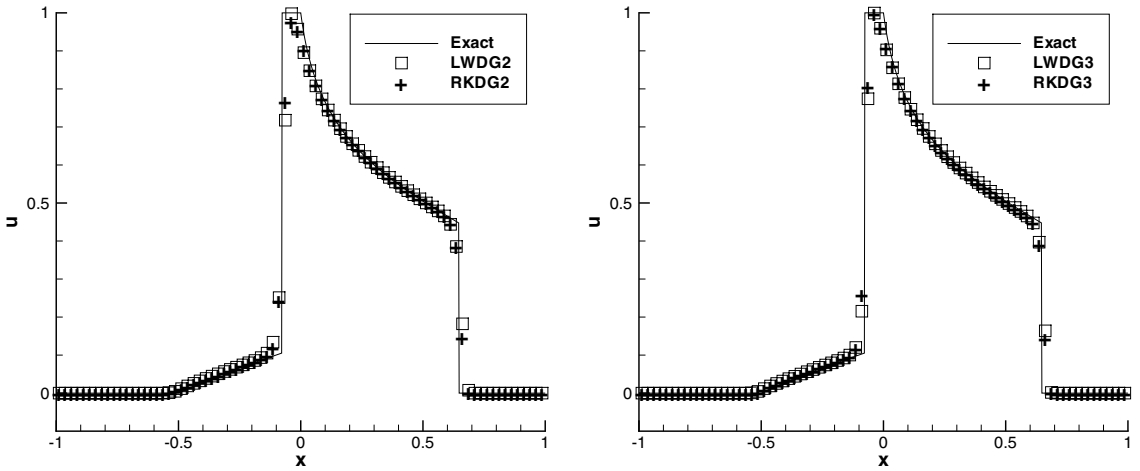

Fig. 1. The Buckley–Leverett problem: $t = 0.4$, $N = 80$ cells. Here and below only the cell averages are plotted. Left: $k = 1$; Right: $k = 2$.

Example 3.4. We solve the nonlinear nonconvex scalar Buckley–Leverett problem

$$u_t + \left(\frac{4u^2}{4u^2 + (1-u)^2} \right)_x = 0, \tag{3.4}$$

with the initial data $u = 1$ when $-\frac{1}{2} \leqslant x \leqslant 0$ and $u = 0$ elsewhere. The solution is computed up to $t = 0.4$. The exact solution is a shock–rarefaction–contact discontinuity mixture. We remark that some high order schemes may fail to converge to the correct entropy solution for this problem. In Fig. 1, the solutions of LWDG comparing with RKDG using $N = 80$ cells are shown. We can see schemes of all orders give good nonoscillatory resolutions to the correct entropy solution for this problem.

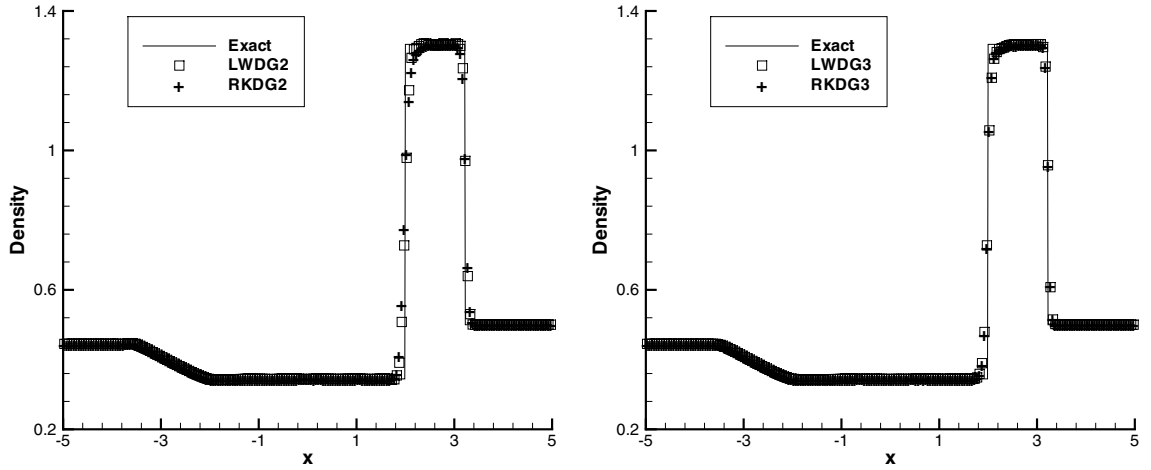

Fig. 2. Lax problem: $t = 1.3$; LWDG together with RKDG; 200 cells; TVB constant $M = 1$; density. Left: $k = 1$; Right: $k = 2$.

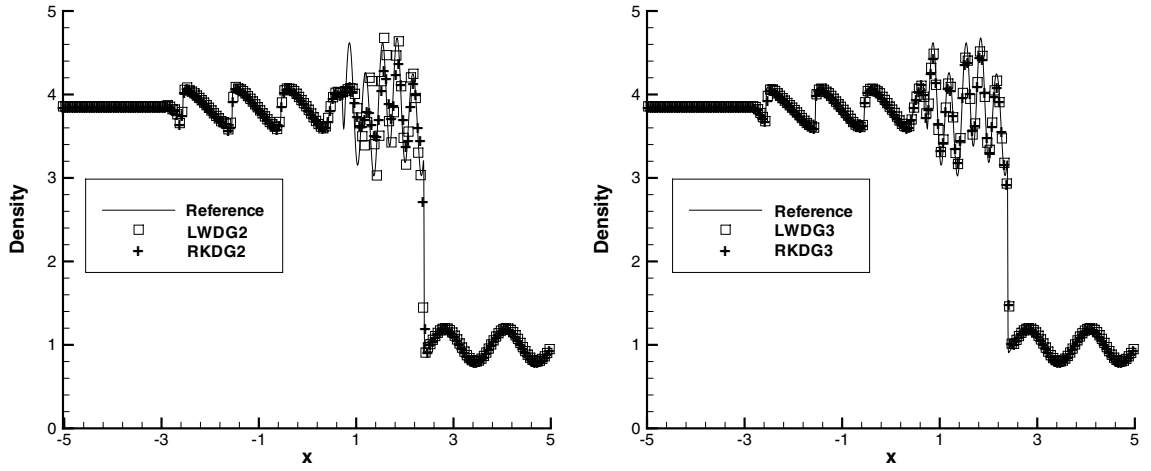

Fig. 3. The shock density wave interaction problem: $t = 1.8$; LWDG together with RKDG; 200 cells; TVB constant $M = 300$; density. Left: $k = 1$; Right: $k = 2$.

Example 3.5. We solve the one-dimensional nonlinear system of Euler equation (3.1). We use the following Riemann initial condition for the Lax problem:

$$(\rho, v, p) = (0.445, 0.698, 3.528) \quad \text{for } x \leqslant 0; \qquad (\rho, v, p) = (0.5, 0, 0.571) \quad \text{for } x > 0.$$

The computed density ρ is plotted at $t = 1.3$ against the exact solution. In Fig. 2, the solutions of LWDG together with RKDG using $N = 200$ cells with TVB constant $M = 1$ in the minmod limiter are shown. We can see that the resolution of the contact discontinuity by LWDG is slightly better than (for the $k = 1$ case) or comparable with (for the $k = 2$ case) that by RKDG.

Example 3.6. A higher order scheme would show its advantage when the solution contains both shocks and complex smooth region structures. A typical example for this is the problem of shock interaction with entropy waves [31]. We solve the Euler equation (3.1) with a moving Mach = 3 shock interacting with sine waves in the density, i.e. initially

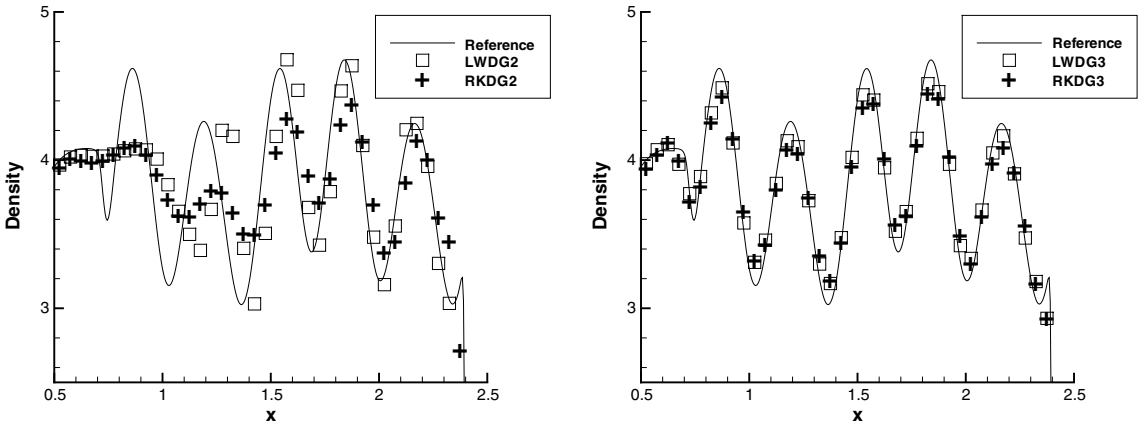

Fig. 4. Details of the oscillatory portion of the solution between $x = 0.5$ and $x = 2.5$ for Fig. 3. Left: $k = 1$; Right: $k = 2$.

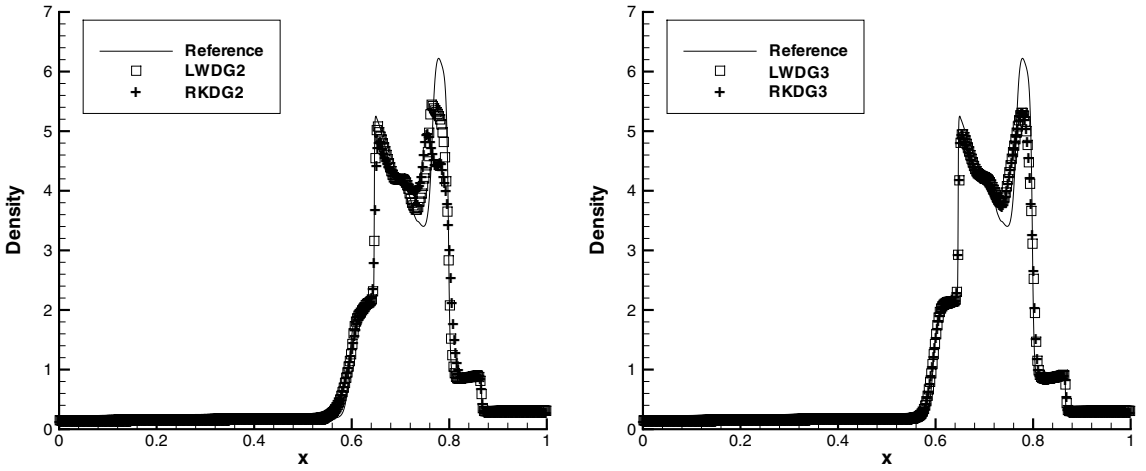

Fig. 5. The blast wave problem: $t = 0.038$; LWDG together with RKDG; 400 cells; TVB constant $M = 300$; density. Left: $k = 1$; Right: $k = 2$.

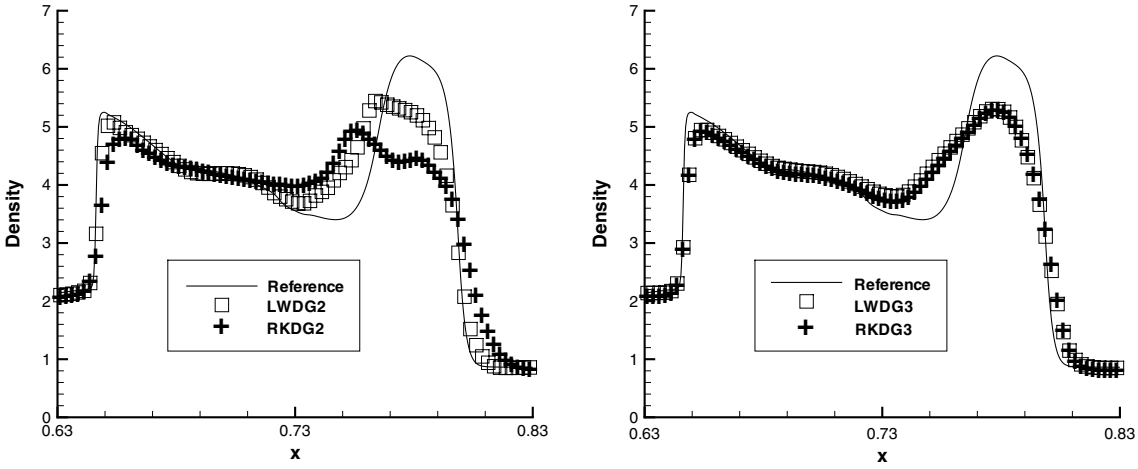

Fig. 6. Detail of solution between $x = 0.5$ and $x = 2.5$ for Fig. 5. Left: $k = 1$; Right: $k = 2$.

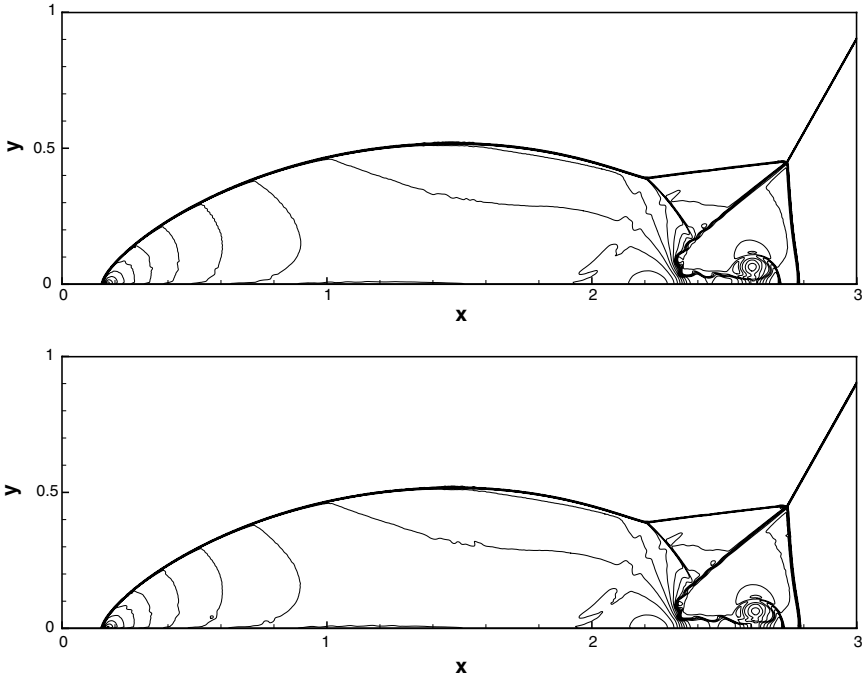

Fig. 7. Double Mach reflection problem: second order ($k = 1$) LWDG (top) and RKDG (bottom) with WENO limiters; 1920×480 cells; TVB constant $M = 100$; 30 equally spaced density contours from 1.5 to 22.7.

$$(\rho, v, p) = (3.857143, 2.629369, 10.333333) \quad \text{for } x < -4;$$

$$(\rho, v, p) = (1 + \varepsilon \sin(5x), 0, 1) \quad \text{for } x \geqslant -4.$$

Here we take $\varepsilon = 0.2$. The computed density ρ is plotted at $t = 1.8$ against the reference "exact" solution, which is a converged solution computed by the fifth order finite difference WENO scheme [17] with 2000 grid points. In Fig. 3, we plot the densities by LWDG and RKDG with WENO limiters using $N = 200$ cells for the TVB constant $M = 300$, and in Fig. 4, we show an enlarged view of the oscillatory portion of the solution between $x = 0.5$ and $x = 2.5$. We can see that LWDG has slightly better resolution than RKDG for the same mesh and same limiter. It seems to be generally true that LWDG often has slightly better resolution than RKDG for the same mesh and same limiter, probably due to the fact that the former is a one-step method using the limiter only once per time step.

Example 3.7. We consider the interaction of blast waves of Euler equation (3.1) with the initial condition:

$$(\rho, v, p) = (1, 0, 1000) \quad \text{for } 0 \leqslant x < 0.1;$$

$$(\rho, v, p) = (1, 0, 0.01) \quad \text{for } 0.1 \leqslant x < 0.9;$$

$$(\rho, v, p) = (1, 0, 100) \quad \text{for } 0.9 \leqslant x.$$

A reflecting boundary condition is applied to both ends, which in the discontinuous Galerkin framework means that a mirror symmetric polynomial is prescribed in the ghost cell outside the boundary point. See [37,15]. The computed density ρ is plotted at $t = 0.038$ against the reference "exact" solution, which is a converged solution computed by the fifth order finite difference WENO scheme [17] with 2000 grid points. In Fig. 5, we plot the densities by LWDG and RKDG with WENO limiters using $N = 400$ cells for the TVB

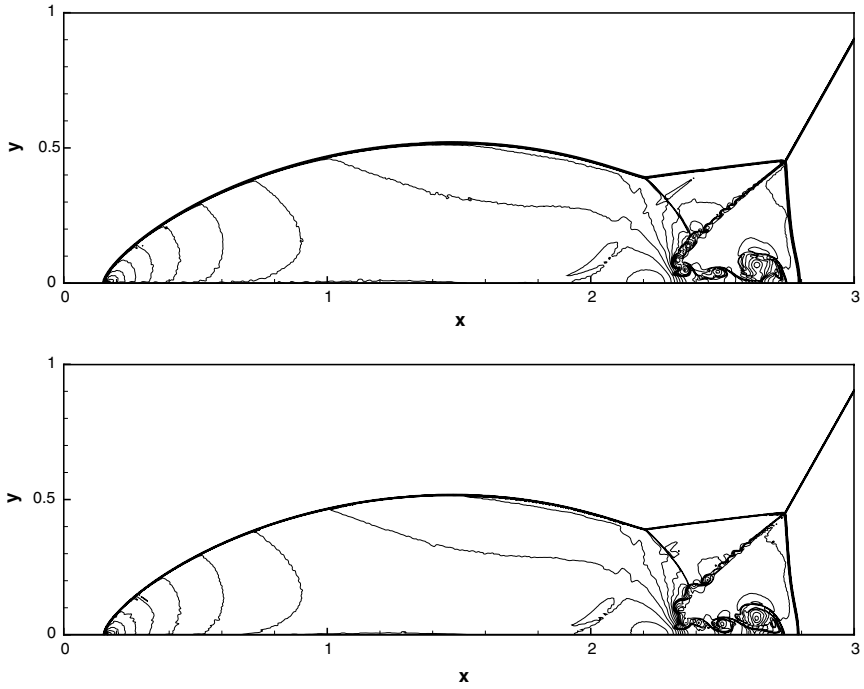

Fig. 8. Double Mach reflection problem: Third order ($k = 2$) LWDG (top) and RKDG (bottom) with WENO limiters; 1920×480 cells; TVB constant $M = 100$; 30 equally spaced density contours from 1.5 to 22.7.

constant $M = 300$, we also show an enlarged view of the solution between $x = 0.63$ and $x = 0.83$ in Fig. 6. We can again see that LWDG has a slightly better resolution than (for the $k = 1$ case) or is comparable with (for the $k = 2$ case) RKDG for the same mesh and the same limiter.

Example 3.8 (Double Mach reflection). This problem is originally from [37]. The computational domain for this problem is $[0,4] \times [0,1]$. The reflecting wall lies at the bottom, starting from $x = \frac{1}{6}$. Initially a right-moving Mach 10 shock is positioned at $x = \frac{1}{6}$, $y = 0$ and makes a 60° angle with the x-axis. For the bottom boundary, the exact post-shock condition is imposed for the part from $x = 0$ to $x = \frac{1}{6}$ and a reflective boundary condition is used for the rest. At the top boundary, the flow values are set to describe the exact motion of a Mach 10 shock. We compute the solution up to $t = 0.2$. Two different uniform meshes, with 960×240 and 1920×480 cells, and two different orders of accuracy for the LWDG and RKDG using WENO limiters with the TVB constant $M = 100$, from $k = 1$ to $k = 2$ (second to third order), are used in the numerical experiments. To save space, we show only the simulation results on the most refined mesh with 1920×480 cells in Figs. 7 and 8, and the "zoomed-in" figures around the double Mach stem to show more details in Fig. 9. All the figures are showing 30 equally spaced density contours from 1.5 to 22.7. For this problem the resolutions of LWDG and RKDG are comparable for the same order of accuracy and the same mesh.

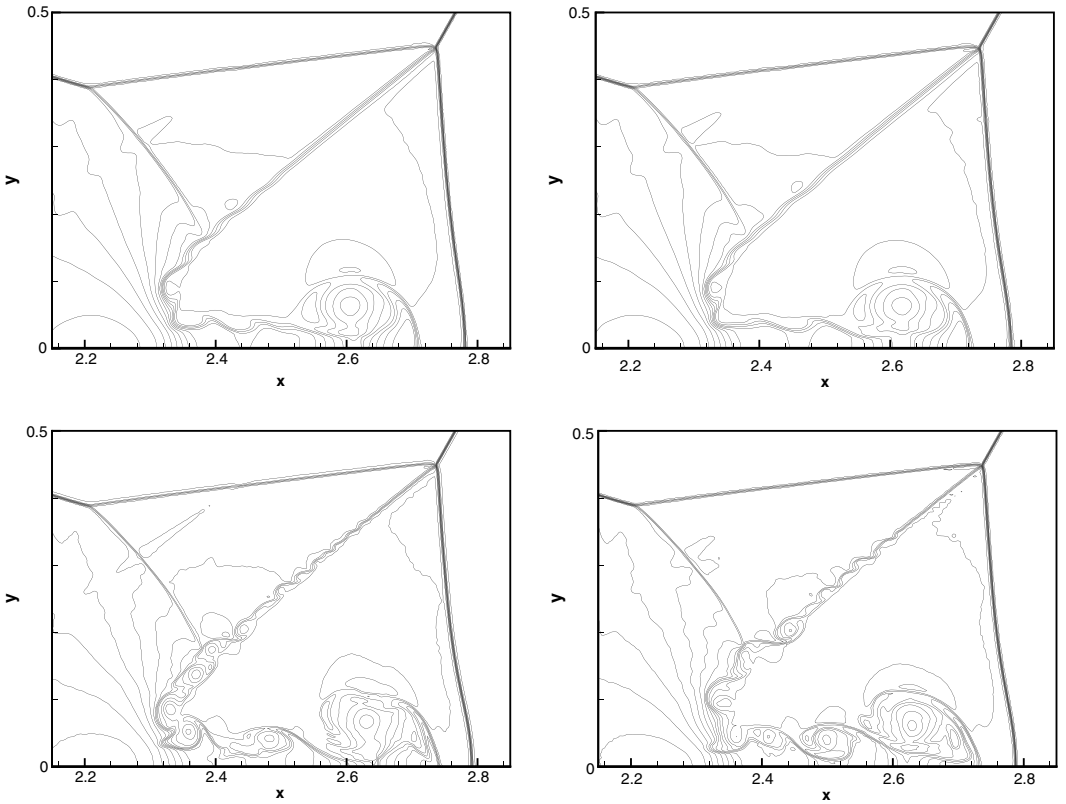

Fig. 9. Double Mach reflection problem: 1920×480 cells; TVB constant $M = 100$; 30 equally spaced density contours from 1.5 to 22.7. Zoomed-in region to show more details. Top: second order ($k = 1$); bottom: third order ($k = 2$). Left: LWDG; Right: RKDG.

4. Concluding remarks

We have developed a Lax–Wendroff time discretization procedure for the discontinuous Galerkin method to solve hyperbolic conservation laws. This is an alternative method for time discretization to the popular TVD Runge–Kutta time discretization in RKDG methods. The LWDG is a one step explicit high order finite element method. The nonlinear limiter for controlling spurious oscillations is performed once per time step. The method relies on converting all the time derivatives in a temporal Taylor expansion into spatial derivatives by repeatedly using the PDE and its differentiated versions. The spatial derivatives are then discretized by the DG approximations. As a result, LWDG is more compact than RKDG and the Lax–Wendroff time discretization procedure is more cost effective than the Runge–Kutta time discretizations, when the nonlinear limiters are used, for certain problems including two-dimensional Euler systems of compressible gas dynamics.

Acknowledgments

The first author would like to thank Professor B.C. Khoo and Professor G.W. Wei for their support and help.

References

[1] R. Biswas, K.D. Devine, J. Flaherty, Parallel, adaptive finite element methods for conservation laws, Appl. Numer. Math. 14 (1994) 255–283.
[2] K.Y. Choe, K.A. Holsapple, The Taylor–Galerkin discontinuous finite element method—an explicit scheme for nonlinear hyperbolic conservation laws, Finite Element Anal. Des. 10 (1991) 243–265.
[3] K.Y. Choe, K.A. Holsapple, The discontinuous finite element method with the Taylor–Galerkin approach for nonlinear hyperbolic conservation laws, Comput. Methods Appl. Mech. Engrg. 95 (1992) 141–167.
[4] A. Burbeau, P. Sagaut, C.H. Bruneau, A problem-independent limiter for high-order Runge–Kutta discontinuous Galerkin methods, J. Comput. Phys. 169 (2001) 111–150.
[5] B. Cockburn, Discontinuous Galerkin methods for convection-dominated problems, in: T.J. Barth, H. Deconinck (Eds.), High-Order Methods for Computational Physics, Lecture Notes in Computational Science and Engineering, vol. 9, Springer, 1999, pp. 69–224.
[6] B. Cockburn, S. Hou, C.-W. Shu, The Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case, Math. Comput. 54 (1990) 545–581.
[7] B. Cockburn, S.-Y. Lin, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems, J. Comput. Phys. 84 (1989) 90–113.
[8] B. Cockburn, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, Math. Comput. 52 (1989) 411–435.
[9] B. Cockburn, C.-W. Shu, The Runge–Kutta local projection P1-discontinuous Galerkin finite element method for scalar conservation laws, Math. Model. Numer. Anal. (M^2AN) 25 (1991) 337–361.
[10] B. Cockburn, C.-W. Shu, The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, J. Comput. Phys. 141 (1998) 199–224.
[11] B. Cockburn, C.-W. Shu, Runge–Kutta Discontinuous Galerkin method for convection-dominated problems, J. Scient. Comput. 16 (2001) 173–261.
[12] O. Colin, M. Rudgyard, Development of high-order Taylor–Galerkin schemes for LES, J. Comput. Phys. 162 (2000) 338–371.
[13] M. Dumbser, ADER discontinuous Galerkin schemes for aeroacoustics, in: Proceedings of the Euromech Colloquium no. 449, Chamonix, France, 2003.
[14] M. Dumbser, C.-D. Munz, Arbitrary high order discontinuous Galerkin schemes, in: S. Cordier, T. Goudon, M. Gutnic, E. Sonnendrucker (Eds.), Numerical Methods for Hyperbolic and Kinetic Problems, IRMA Series in Mathematics and Theoretical Physics, de Gruyter, in press.
[15] A. Harten, B. Engquist, S. Osher, S. Chakravathy, Uniformly high order accurate essentially non-oscillatory schemes III, J. Comput. Phys. 71 (1987) 231–303.

[16] A. Heck, Introduction to Maple, second ed., Springer-Verlag, 1996.
[17] G. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.
[18] P.D. Lax, B. Wendroff, Systems of conservation laws, Commun. Pure Appl. Math. 13 (1960) 217–237.
[19] M. Lukácová-Medvid'ová, G. Warnecke, Lax–Wendroff type second order evolution Galerkin methods for multidimensional hyperbolic systems, East–West J. Numer. Math. 8 (2000) 127–152.
[20] J. Qiu, C.-W. Shu, Finite difference WENO schemes with Lax–Wendroff-type time discretizations, SIAM J. Scient. Comput. 24 (2003) 2185–2198.
[21] J. Qiu, C.-W. Shu, Runge–Kutta discontinuous Galerkin method using WENO limiters, SIAM J. Scient. Comput., in press.
[22] J. Qiu, C.-W. Shu, Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method: one-dimensional case, J. Comput. Phys. 193 (2003) 115–135.
[23] J. Qiu, C.-W. Shu, Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method II: two-dimensional case, Comput. Fluids, in press.
[24] W.H. Reed, T.R. Hill, Triangular mesh methods for neutron transport equation, Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
[25] S.J. Ruuth, R.J. Spiteri, Two barriers on strong-stability-preserving time discretization methods, J. Scient. Comput. 17 (2002) 211–220.
[26] A. Safjan, J.T. Oden, High-order Taylor–Galerkin methods for linear hyperbolic systems, J. Comput. Phys. 120 (1995) 206–230.
[27] C.-W. Shu, TVB uniformly high-order schemes for conservation laws, Math. Comput. 49 (1987) 105–121.
[28] C.-W. Shu, Total-variation-diminishing time discretizations, SIAM J. Scient. Statist. Comput. 9 (1988) 1073–1084.
[29] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in: B. Cockburn, C. Johnson, C.-W. Shu, E. Tadmor (Eds.), Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, in: A. Quarteroni (Ed.), Lecture Notes in Mathematics, vol. 1697, Springer, 1998, pp. 325–432.
[30] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, J. Comput. Phys. 77 (1988) 439–471.
[31] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock capturing schemes II, J. Comput. Phys. 83 (1989) 32–78.
[32] B. Tabarrok, J. Su, Semi-implicit Taylor–Galerkin finite element methods for incompressible viscous flows, Comput. Methods Appl. Mech. Engrg. 117 (1994) 391–410.
[33] M.E. Taylor, Partial differential equation I: Basic theoryApplied Mathematical Sciences, vol. 115, Springer, 1996.
[34] V.A. Titarev, E.F. Toro, ADER: arbitrary high order Godunov approach, J. Scient. Comput. 17 (2002) 609–618.
[35] E.F. Toro, Primitive, conservative and adaptive schemes for hyperbolic conservation laws, in: E.F. Toro, J.F. Clarke (Eds.), Numerical Methods for Wave Propagation, Kluwer Academic Publishers, 1998, pp. 323–385.
[36] E.F. Toro, V.A. Titarev, Solution of the generalized Riemann problem for advection-reaction equations, Proc. Royal Soc. London 458 (2002) 271–281.
[37] P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, J. Comput. Phys. 54 (1984) 115–173.
[38] S. Youn, S. Park, A new direct higher-order Taylor–Galerkin finite element method, Comput. Struct. 56 (1995) 651–656.
[39] M. Zhang, C.-W. Shu, An analysis of three different formulations of the discontinuous Galerkin method for diffusion equations, Math. Model. Meth. Appl. Sci. (M^3AS) 13 (2003) 395–413.
[40] X. Zhong, Additive semi-implicit Runge–Kutta methods for computing high-speed nonequilibrium reactive flows, J. Comput. Phys. 128 (1996) 19–31.