A Conservation Constrained Runge-Kutta Discontinuous Galerkin Method with the Improved CFL Condition for Conservation Laws

Zhiliang Xu [‡], Yingjie Liu [§]

February 16, 2010

Abstract

We present a new formulation of the Runge-Kutta discontinuous Galerkin (RKDG) method [6, 5, 4, 3] for conservation Laws. The new formulation requires the computed RKDG solution in a cell to satisfy additional conservation constraint in adjacent cells and does not increase the complexity or change the compactness of the RKDG method. We use this new formulation to solve conservation laws on one-dimensional grids with piecewise cubic polynomial approximation as well as on two-dimensional unstructured grids with piecewise quadratic polynomial approximation. The hierarchical reconstruction [12, 24] is applied as a limiter to eliminate spurious oscillations. Numerical computations for scalar and systems of nonlinear hyperbolic conservation laws are performed. We find that: 1) this new formulation improves the CFL number over the original RKDG formulation and thus reduces the overall complexity; 2) the new formulation improves the robustness of the DG scheme with the current limiting strategy and improves the resolution of the numerical solutions in multi-dimensions.

1 Introduction

In this paper, we introduce a simple, yet effective technique to improve the Courant-Friedrichs-Lewy (CFL) condition of the Runge-Kutta discontinuous Galerkin (RKDG) method for solving nonlinear conservation laws. The discontinuous Galerkin method (DG) was firstly introduced by Reed and Hill [17] as a technique to solve neutron transport problems. In a series of papers by Cockburn, Shu *et al.* [6, 5, 4, 3], the RKDG method has been developed for solving nonlinear hyperbolic conservation laws and related equations, in which DG is used for spatial discretization with flux values at cell edges computed by either Riemann solvers

[‡](E-mail: zxu2@nd.edu)
Department of Mathematics, University of Notre Dame, Notre Dame, IN 46556. Research was supported in part by NSF grant DMS-0800612.

[§](E-mail: yingjie@math.gatech.edu)
School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332. Research was supported in part by NSF grant DMS-0810913.

or monotone flux functions, the total variation bounded (TVB) limiter [18, 6] is employed to eliminate spurious oscillations and the total variation diminishing (TVD) Runge-Kutta (RK) method [20] is used for the temporal discretization to ensure the stability of the numerical approach while simplifying the implementation. The RKDG method has enjoyed great success in solving the Euler equations for gas dynamics, compressible Navier-Stokes equations, viscous MHD equations and many other equations, and motivated many related new techniques.

In [6], the RKDG method is shown to be linearly stable when the CFL factor is bounded by $\frac{1}{2q+1}$ for the second order and the third order schemes in the one-dimensional (1D) space, where q is the degree of the polynomial approximating the solution. It would be great if there is a simple technique to increase its CFL number without too much overhead while still being compact and maintaining its other nice properties. In this paper, we present a strategy which is to mix the RKDG method with some of the finite volume reconstruction features (e.g. Abgrall [1]), and use them as the extra constraint. We would like to refer to a recent work of van Leer and Nomura [11], in which the diffusive flux for DG is approximated by using a reconstructed polynomial supported on the union of adjacent cells out of a piecewise polynomial. In a paper by Warburton and Hagstrom [23], the RKDG solution is projected to the staggered covolume mesh to obtain distributional derivatives and then is projected back on each Runge-Kutta step which is analytically shown in 1D to significantly increase the CFL number. It is found in [14] that the central DG scheme on overlapping cells with Runge-Kutta time-stepping can afford larger CFL numbers than the RKDG method on non-overlapping cells when the order of accuracy is above the first order.

In the present paper, we enforce a few local cell average constraints on the RKDG method in order to obtain a larger CFL number and better quality of numerical solutions after limiting. The resulting method is termed as the conservation constrained RKDG method or the constrained RKDG method. We are going to test the effectiveness of our technique in the fourth order case for 1D (while the third order case in 1D becomes a finite volume scheme) and in the third order case on two-dimensional (2D) triangular meshes. Further study on higher order cases and theoretical analysis will be reported in the future.

Using finite volume limiting techniques on solutions computed by the RKDG method for conservation laws has been explored by many researchers. In [16, 26], the WENO finite volume reconstruction procedures are used as the limiter on "trouble cells". In [15], Luo *et al.* develop a Hermite WENO-based limiter for the second order RKDG method on unstructured meshes following [16]. Since the RKDG method is a compact method, it would be ideal to use a compact limiting technique. It is a challenging task to use adjacent high order information in the limiting procedure to remove spurious oscillations in the vicinities of discontinuities while preserving high resolution. The first of such limiters is the TVB projection limiter by Cockburn and Shu, which uses the lowest and (limited) first Legendre moments locally where non-smoothness is detected. Other compact limiting techniques which are supposed to remove spurious oscillations using information only from adjacent cells for any orders include the moment limiter [2] and the recently developed hierarchical reconstruction (HR) [12]. Besides the above related techniques, there are also many research works of compact limiters for high order schemes on various problems. HR as a limiting technique can be applied without using local characteristic decomposition. One goal of the paper is to verify if our technique for improving the CFL number of RKDG works well with HR. In [24], HR

on 2D triangular meshes has been studied for the piecewise quadratic DG method; a partial neighboring cell technique has been developed and a component-wise WENO-type linear reconstruction is used on each hierarchical level. This new technique has good resolution and accuracy on unstructured meshes and is easy to implement since the weights on each hierarchical level are trivial to compute and essentially independent of the mesh.

We find that the constrained RKDG method increases the CFL number over the original RKDG method, reduces the magnitude of numerical errors for the multi-dimensional case, and further improves the resolution of the numerical solutions limited by HR.

The paper is organized as follows. Section 2 describes the conservation constrained RKDG formulation and summarizes the limiting procedure. Results of numerical tests are presented in Section 3. Concluding remarks and a plan for the future work are included in Section 4.

2 Algorithm Formulation

In this section, we formulate the conservation constrained Runge-Kutta discontinuous Galerkin finite element method for solving time dependent hyperbolic conservation laws (2.1)

$$\begin{cases} \frac{\partial u_k}{\partial t} + \nabla \cdot \mathbf{F}_k(\mathbf{u}) = 0 \ , \quad k = 1, .., p, \quad \text{in } \Omega \times (0, T) \ , \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \ , \end{cases} \tag{2.1}$$

where $\Omega \subset R^d$, $\mathbf{x} = (x_1, ..., x_d)$, d is the dimension, $\mathbf{u} = (u_1, ..., u_p)^T$ and the flux vectors $\mathbf{F}_k(\mathbf{u}) = (F_{k,1}(\mathbf{u}), ..., F_{k,d}(\mathbf{u}))$.

The method of lines approach is used to evolve the solution. The 3^{rd} and 4^{th} order accurate TVD Runge-Kutta time-stepping methods are used for the test problems presented in the paper. At each time level the semi-discrete constrained DG method is used for spatial discretization. In the vicinities of discontinuities of the solution, the computed piecewise polynomial solution is reconstructed by the hierarchical reconstruction to remove spurious oscillations.

2.1 Conservation constrained discontinuous Galerkin Method

We describe the conservation constrained DG formulation here. First, the physical domain Ω is partitioned into a collection of $\mathcal{N}$ non-overlapping cells $\mathcal{T}_h = \{\mathcal{K}_i : i = 1, ..., \mathcal{N}\}$ so that $\Omega = \bigcup_{i=1}^{\mathcal{N}} \mathcal{K}_i$. In 2D, we use triangular meshes and for simplicity, we assume that there are no hanging nodes. Let the basis function set which spans the finite element space on cell $\mathcal{K}_i$ be

$$\mathcal{B}_i = \{\phi_m(\mathbf{x}) : m = 0, ..., r\} \ . \tag{2.2}$$

In the present study, we choose the basis function set to be a polynomial basis function set of degree q in a cell $\mathcal{K}_i$, which consists of the monomials of multi-dimensional Taylor expansions about the cell centroid. For instance, for a 2D triangular cell $\mathcal{K}_i$, the basis function set (2.2) in the (x, y) coordinate is

$$\begin{aligned} \mathcal{B}_i &= \{\phi_m(x - x_i, y - y_i) : m = 0, ..., r\} \\ &= \{1, x - x_i, y - y_i, (x - x_i)^2, (x - x_i)(y - y_i), (y - y_i)^2, \cdots, (y - y_i)^q\} \ , \end{aligned} \tag{2.3}$$

where $\mathbf{x}_i \equiv (x_i, y_i)$ is the centroid of $\mathcal{K}_i$ and $r = (q+1)(q+2)/2 + 1$. The finite element space on cell $\mathcal{K}_i$ is the span of these basis functions.

In each cell $\mathcal{K}_i$, the approximate solution $u_{h,k}$ of the k^{th} equation of (2.1) is expressed as

$$u_{h,k} = \sum_{m=0}^{r} c_m(t)\phi_m(\mathbf{x}) \ . \tag{2.4}$$

Let's assume that the immediate neighbors (sharing same edges) of $\mathcal{K}_i$ are collected as the set $\{\mathcal{K}_J : J = 1, 2, .., M.\}$ (which also contains cell $\mathcal{K}_i$). The semi-discrete DG formulation of the k^{th} equation of (2.1) is to find an approximate solution u_h of the form (2.4) (neglecting its subscript k for convenience) such that

$$\frac{d}{dt}\int_{\mathcal{K}_i} u_h v_h d\mathbf{x} + \int_{\partial\mathcal{K}_i} \mathbf{F}_k(\mathbf{u}_h)\cdot\mathbf{n}_i v_h d\Gamma - \int_{\mathcal{K}_i} \mathbf{F}_k(\mathbf{u}_h)\cdot\nabla v_h d\mathbf{x} = 0 \ , \tag{2.5}$$

for any $v_h \in \text{span}\{\mathcal{B}_i\}$, where $\mathbf{n}_i$ is the outer unit normal vector of $\mathcal{K}_i$.

Since the approximate solution $\mathbf{u}_h$ is discontinuous across cell edges, the interfacial fluxes are not uniquely determined. The flux function $\mathbf{F}_k(\mathbf{u}_h)\cdot\mathbf{n}_i$ appearing in equation (2.5) can be replaced by the Lax-Friedrich flux function (see e.g. [19]) defined as

$$h_k(\mathbf{x}, t) = h_k(\mathbf{u}_h^{in}, \mathbf{u}_h^{out}) = \frac{1}{2}(\mathbf{F}_k(\mathbf{u}_h^{in})\cdot\mathbf{n}_i + \mathbf{F}_k(\mathbf{u}_h^{out})\cdot\mathbf{n}_i) + \frac{\alpha}{2}(u_h^{in} - u_h^{out}) \quad , \quad k = 1, ..., m \ ,$$

where α is the largest characteristic speed,

$$\mathbf{u}_h^{in}(\mathbf{x}, t) = \lim_{\mathbf{y}\to\mathbf{x}, \mathbf{y}\in\mathcal{K}_i^{\text{int}}} \mathbf{u}_h(\mathbf{y}, t) \ ,$$
$$\mathbf{u}_h^{out}(\mathbf{x}, t) = \lim_{\mathbf{y}\to\mathbf{x}, \mathbf{y}\notin\bar{\mathcal{K}}_i} \mathbf{u}_h(\mathbf{y}, t) \ .$$

Equation (2.5) then becomes

$$\frac{d}{dt}\int_{\mathcal{K}_i} u_h v_h d\mathbf{x} + \int_{\partial\mathcal{K}_i} h_k v_h d\Gamma - \int_{\mathcal{K}_i} \mathbf{F}_k(\mathbf{u}_h)\cdot\nabla v_h d\mathbf{x} = 0 \ . \tag{2.6}$$

The resulting systems of ordinary differential equations can be solved by a TVD Runge-Kutta method [20] which builds on convex combinations of several forward Euler schemes of (2.6). Our additional conservation constraint is performed within each of the component forward Euler scheme. A forward Euler scheme of (2.6) can be written as

$$\int_{\mathcal{K}_i} u_h^{n+1} v_h d\mathbf{x} = \int_{\mathcal{K}_i} u_h^n v_h d\mathbf{x} - \Delta t_n \int_{\partial\mathcal{K}_i} h_k^n v_h d\Gamma + \Delta t_n \int_{\mathcal{K}_i} \mathbf{F}_k^n(\mathbf{u}_h)\cdot\nabla v_h d\mathbf{x} \ , \tag{2.7}$$

where the superscript n denotes the time level t_n, $\Delta t_n = t_{n+1} - t_n$. In particular, letting $v_h \equiv 1$, we obtain the cell average of u_h^{n+1} over cell $\mathcal{K}_i$, denoted by $\overline{u_i^{n+1}}$, just as with a finite volume scheme.

Now suppose the cell averages $\{\overline{u_i^{n+1}}\}$ have been computed on all cells. We do not compute the rest of the moments of u_h^{n+1} on cell $\mathcal{K}_i$ by using equation (2.7). Instead, we let u_h^{n+1} on cell $\mathcal{K}_i$ minimize an energy functional (variational to (2.7)) subject to that it

conserves additional given cell averages not only in cell $\mathcal{K}_i$ but also in some of its neighbors. Rewrite (2.7) in cell $\mathcal{K}_i$ as

$$\int_{\mathcal{K}_i} u_h^{n+1} v_h d\mathbf{x} = \mathcal{L}(v_h) \ , \tag{2.8}$$

where $\mathcal{L}(v_h)$ represents the right-hand-side of (2.7), which is a linear bounded functional defined on the finite element space on $\mathcal{K}_i$. The variational form of (2.8) is to find u_h^{n+1} in the finite element space on $\mathcal{K}_i$ such that it minimizes the energy functional

$$E(v_h) = \frac{1}{2} \int_{\mathcal{K}_i} (v_h)^2 d\mathbf{x} - \mathcal{L}(v_h) \ . \tag{2.9}$$

Finally, our new conservation constrained RKDG formulation on cell $\mathcal{K}_i$ can be described as replacing each component forward Euler scheme by finding u_h^{n+1} in the finite element space on $\mathcal{K}_i$, such that

$$\begin{array}{rl} E(u_h^{n+1}) = & \text{Minimum of } \{E(v_h) : v_h \in \text{span}\{\mathcal{B}_i\}\}, \\ & \text{subject to } \frac{1}{|\mathcal{K}_J|} \int_{\mathcal{K}_J} v_h d\mathbf{x} = \overline{u_J^{n+1}} \ , \quad J = 1, ..., M \ . \end{array} \tag{2.10}$$

This constrained minimization problem can be solved by the method of Lagrange as follows

$$\begin{array}{l} \int_{\mathcal{K}_i} u_h^{n+1} v_h d\mathbf{x} - \mathcal{L}(v_h) = \sum_{J=1}^{M} \frac{\lambda_J}{|\mathcal{K}_J|} \int_{\mathcal{K}_J} v_h d\mathbf{x}, \ \forall \ v_h \in \text{span}\{\mathcal{B}_i\} \\ \frac{1}{|\mathcal{K}_J|} \int_{\mathcal{K}_J} u_h^{n+1} d\mathbf{x} = \overline{u_J^{n+1}} \ , \quad J = 1, ..., M \ , \end{array} \tag{2.11}$$

where $\{\lambda_J\}$ are Lagrangian multipliers. Coefficients $\{c_m\}$ of u_h^{n+1} (see equation (2.4)) is determined by the above linear system. Note that the left-hand-side of the first equation of (2.11) is in the same form as equation (2.8) or (2.7), and $M = 4$ for the 2D triangular meshes since the set $\{\mathcal{K}_J\}$ contains the cell $\mathcal{K}_i$ and its three adjacent neighbors (sharing common edges with $\mathcal{K}_i$). ($M = 3$ in 1D, the set $\{\mathcal{K}_J\}$ contains the cell $\mathcal{K}_i$ and its left and right neighbors.)

To summarize, assume we employ a s-stage TVD Runge-Kutta method to solve equation (2.6), which can be written in the form:

$$\begin{array}{rcl} \int_{\mathcal{K}_i} u_h^{(j)} v_h d\mathbf{x} & = & \sum_{l=0}^{j-1} \alpha_{jl} \left(\int_{\mathcal{K}_i} u^{(l)} v_h d\mathbf{x} + \Delta t_n \beta_{jl} L(u_h^{(l)}, v_h) \right) \ , \quad j = 1, ..., s \\ & \equiv & \sum_{l=0}^{j-1} \alpha_{jl} \int_{\mathcal{K}_i} u_h^{(j,l+1)} v_h d\mathbf{x} \ , \end{array} \tag{2.12}$$

with

$$u_h^{(0)} = u_h^n, \qquad u_h^{(s)} = u_h^{n+1} \ . \tag{2.13}$$

Here α_{jl} and β_{jl} are coefficients of the Runge-Kutta method at the j^{th} stage, and

$$L(u_h, v_h) = -\int_{\partial \mathcal{K}_i} h_k v_h d\Gamma + \int_{\mathcal{K}_i} \mathbf{F}_k(\mathbf{u}_h) \cdot \nabla v_h d\mathbf{x} \ .$$

In particular, $u_h^{(j,l+1)}$ is determined by

$$\int_{\mathcal{K}_i} u_h^{(j,l+1)} v_h d\mathbf{x} = \int_{\mathcal{K}_i} u^{(l)} v_h d\mathbf{x} + \Delta t_n \beta_{jl} L(u_h^{(l)}, v_h) \ , \ \forall \ v_h \in \text{span}\{\mathcal{B}_i\} \ .$$

This is a forward Euler scheme as in (2.7) with the time step size $\Delta t_n \beta_{jl}$, and will be replaced similarly by the modification as in (2.11).

2.2 Limiting by hierarchical reconstruction

To prevent non-physical oscillations in the vicinity of discontinuities, we apply HR [24] at each Runge-Kutta stage to the DG solution. Since shock waves or contact discontinuities are all local phenomena, we apply the HR limiting procedure to a small region covering discontinuities. Specifically, we employ a local limiting procedure by using a detector [3] to identify "bad cells", *i.e.*, cells which may contain oscillatory solutions. HR is then applied to solutions supported on these "bad cells". We give a brief description of the HR limiting procedure here. More details can be found in [24].

HR decomposes the job of limiting a high-order polynomial supported on a cell (which may contain spurious oscillations) into a series of smaller jobs, each of which only involves the non-oscillatory reconstruction of a linear polynomial, which can be easily achieved through classical processes such as the MUSCL reconstruction [8, 9, 10] used in [12], or a WENO-type combination used in [24]. Since the reconstruction of a linear polynomial can only use information from adjacent cells, HR can be formulated in multi dimensions on a compact stencil. Using the basis function set (2.3), the approximate solution $u_h(\mathbf{x}-\mathbf{x}_i)$ on cell $\mathcal{K}_i$ is in the Taylor expansion around cell centroid $\mathbf{x}_i$. $u_h(\mathbf{x}-\mathbf{x}_i)$ may contain spurious oscillations. The hierarchical reconstruction procedure is to recompute the coefficients of polynomial $u_h(\mathbf{x}-\mathbf{x}_i)$ by using polynomials in cells adjacent to $\mathcal{K}_i$ (or partial neighboring cells [24]). These adjacent cells (or partial cells) are collected as the set $\{\mathcal{K}_j\}$ (which also contains cell $\mathcal{K}_i$) and the polynomials (of degree q) supported on them are denoted as $\{u_{h,j}(\mathbf{x}-\mathbf{x}_j)\}$ respectively. HR recomputes a set of new coefficients

$$\frac{1}{\mathbf{m}!}\widetilde{u}_h^{(\mathbf{m})}(\mathbf{0}), \qquad |\mathbf{m}| = q, q-1, \ldots, 0$$

to replace the original coefficients $\frac{1}{\mathbf{m}!}u_h^{(\mathbf{m})}(\mathbf{0})$ of $u_h(\mathbf{x}-\mathbf{x}_i)$ iteratively from the highest to the lowest degree terms without losing the order of accuracy if the piecewise polynomial solution is locally smooth, and eliminates spurious oscillations of $u_h(\mathbf{x}-\mathbf{x}_i)$ otherwise.

To obtain $\widetilde{u}_h^{(\mathbf{m})}(\mathbf{0})$, we first compute *candidates* of $u_h^{(\mathbf{m})}(\mathbf{0})$, and then let the new value for $u_h^{(\mathbf{m})}(\mathbf{0})$ be

$$\widetilde{u}_h^{(\mathbf{m})}(\mathbf{0}) = F\big(\text{ candidates of } u_h^{(\mathbf{m})}(\mathbf{0})\ \big),$$

where F is a convex limiter of its arguments (e.g., the center biased minmod function used in [13], or the WENO-type combination in [24]), $F(a_1, a_2, \cdots, a_l) = \sum_{i=1}^{l}\theta_i a_i$, for some $\theta_i \geq 0$ and $\sum_{i=1}^{l}\theta_i = 1$.

In order to find these candidates of $u_h^{(\mathbf{m})}(\mathbf{0})$, $|\mathbf{m}| = m$, we take a $(m-1)^{th}$ order partial derivative of $u_h(\mathbf{x}-\mathbf{x}_i)$ (and also polynomials in adjacent cells), and express

$$\partial^{m-1}u_h(\mathbf{x}-\mathbf{x}_i) = L_h(\mathbf{x}-\mathbf{x}_i) + R_h(\mathbf{x}-\mathbf{x}_i),$$

where L_h is the linear part (containing the zeroth and first degree terms) and R_h is the remainder. Clearly, every coefficient in the first degree terms of L_h is in the set $\{u_h^{(\mathbf{m})}(\mathbf{0}) : |\mathbf{m}| = m\}$. And for every $\mathbf{m}$ subject to $|\mathbf{m}| = m$, one can always take some $(m-1)^{th}$ order partial derivatives of $u_h(\mathbf{x}-\mathbf{x}_i)$ so that $u_h^{(\mathbf{m})}(\mathbf{0})$ is a coefficient in a first degree term of L_h. Thus, a "candidate" for a coefficient in a first degree term of L_h is also the candidate for the corresponding $u_h^{(\mathbf{m})}(\mathbf{0})$.

In order to find a set of candidates for all coefficients in the first degree terms of $L_h(\mathbf{x}-\mathbf{x}_i)$, we need to know the new approximate cell averages of $L_h(\mathbf{x}-\mathbf{x}_i)$ on $d+1$ distinct mesh cells adjacent to cell $\mathcal{K}_i$, which is a key step. Assume $\mathcal{K}_{j_0}, \mathcal{K}_{j_1}, \cdots, \mathcal{K}_{j_d} \in \{\mathcal{K}_j\}$ are these cells and $\overline{L}_{j_0}, \overline{L}_{j_1}, \cdots, \overline{L}_{j_d}$ are the corresponding new approximate cell averages. For example, in order to obtain $\overline{L}_{j_1}$, we first compute

$$A_{j_1} = \frac{1}{|\mathcal{K}_{j_1}|} \int_{\mathcal{K}_{j_1}} \partial^{m-1} u_{h,j_1}(\mathbf{x}-\mathbf{x}_{j_1}) d\mathbf{x},$$

then

$$D_{j_1} = \frac{1}{|\mathcal{K}_{j_1}|} \int_{\mathcal{K}_{j_1}} \tilde{R}_h(\mathbf{x}-\mathbf{x}_i) d\mathbf{x},$$

where $\tilde{R}_h(\mathbf{x}-\mathbf{x}_i)$ is the $R_h(\mathbf{x}-\mathbf{x}_i)$ with its coefficients replaced by previously computed new values. Finally we can set $\overline{L}_{j_1} = A_{j_1} - D_{j_1}$.

More details for the HR implementation of our one-dimensional test problems in this paper can be found in [25].

3 Numerical Examples

3.1 One-dimensional Tests

3.1.1 1D Burgers' equation with a smooth solution

We first test the capability of the constrained RKDG method to achieve the desired order of accuracy with a large CFL number, using the 1D scalar Burgers' equation

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

with periodic boundary conditions and the initial condition $u(x,0) = \frac{1}{2} + \sin(\pi x)$, $-1 \le x \le 1$.

The uniform mesh is used to solve this test problem. The cell size, denoted by h, is listed in Tables shown in this section. Tables 1 and 2 show the accuracy test results for the 3^{rd} and 4^{th} order accurate constrained RKDG methods. From Table 1, we can see that the 3^{rd} order constrained RKDG method is stable with the CFL number close to 1. Tables 2 and 3 show that the 4^{th} order constrained RKDG method becomes unstable when the CFL number is equal to 0.7; while the 4^{th} order constrained RKDG method is stable when the CFL number is equal to 0.6. For the RKDG method, we found that the 3^{rd} order RKDG method is stable when CFL < 0.23, and the 4^{th} order RKDG method is stable when CFL < 0.17. See Tables 4 and 5 for the accuracy test results of the 3^{rd} and 4^{th} order RKDG methods respectively.

3.1.2 1D Euler equations with discontinuous solutions

We assess the resolution and the non-oscillatory property of numerical solutions computed by the constrained RKDG method and limited by HR. We compute solutions of the 1D Euler equations

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0$$

Table 1: Accuracy test results of the 3^{rd} order constrained RKDG method solving the 1D Burgers' equation. CFL = 0.9.

h	L_1 error	order	L_∞ error	order
1/40	5.11E-5	-	7.84E-4	-
1/80	6.30E-6	3.02	9.89E-5	2.99
1/160	7.83E-7	3.01	1.25E-5	2.98
1/320	9.76E-8	3.00	1.56E-6	3.00
1/640	1.22E-8	3.00	1.95E-7	3.00
1/1280	1.52E-9	3.00	2.44E-8	3.00
1/2560	1.90E-10	3.00	3.05E-9	3.00
1/5120	2.40E-11	2.98	3.64E-10	3.07

Table 2: Accuracy test results of the 4^{th} order constrained RKDG method solving the 1D Burgers' equation. CFL = 0.6.

h	L_1 error	order	L_∞ error	order
1/40	1.10E-6	-	2.75E-5	-
1/80	7.20E-8	3.93	1.86E-6	3.89
1/160	4.69E-9	3.94	1.20E-7	3.95
1/320	2.97E-10	3.98	7.49E-9	4.00
1/640	1.91E-11	3.96	4.70E-10	3.99

Table 3: Accuracy test results of the 4^{th} order constrained RKDG method solving the 1D Burgers' equation. CFL = 0.7.

h	L_1 error	order	L_∞ error	order
1/40	1.16E-6	-	2.82E-5	-
1/80	7.43E-8	3.96	1.86E-6	3.94
1/160	1.45E-7	-	7.51E-6	-
1/320	6.34E-3	-	4.99E-1	-

with $\mathbf{u} = (\rho, \rho v, E)^T$, $\mathbf{f}(\mathbf{u}) = (\rho v, \rho v^2 + p, v(E+p))^T$, $p = (\gamma - 1)(E - \frac{1}{2}\rho v^2)$ and $\gamma = 1.4$.

Example 3.1.2.1. 1D Shu-Osher problem [21]. It is the Euler equations with an initial data

$$\begin{aligned}(\rho, v, p) &= (3.857143,\ 2.629369,\ 10.333333), \quad \text{for } x < -4,\\ (\rho, v, p) &= (1 + 0.2\sin(5x),\ 0,\ 1), \quad \text{for } x \geq -4.\end{aligned}$$

We compute the numerical solutions using 300 equal size cells and the 4^{th} order constrained RKDG method and the 4^{th} order RKDG method respectively. The density profiles of the solutions are plotted at the time $T = 1.8$ in Fig. 1. We can clearly see that the 4^{th} order constrained RKDG solution and the 4^{th} order RKDG solution have almost identical resolution for this test problem.

Example 3.1.2.2. 1D Woodward-Colella blast wave problem [22]. It is the Euler equations with an initial data

$$\begin{aligned}(\rho, \rho v, E) &= (1,\ 0,\ 2500), \quad \text{for } 0 < x < 0.1,\\ (\rho, \rho v, E) &= (1,\ 0,\ 0.025), \quad \text{for } 0.1 < x < 0.9,\\ (\rho, \rho v, E) &= (1,\ 0,\ 250), \quad \text{for } 0.9 < x < 1.\end{aligned}$$

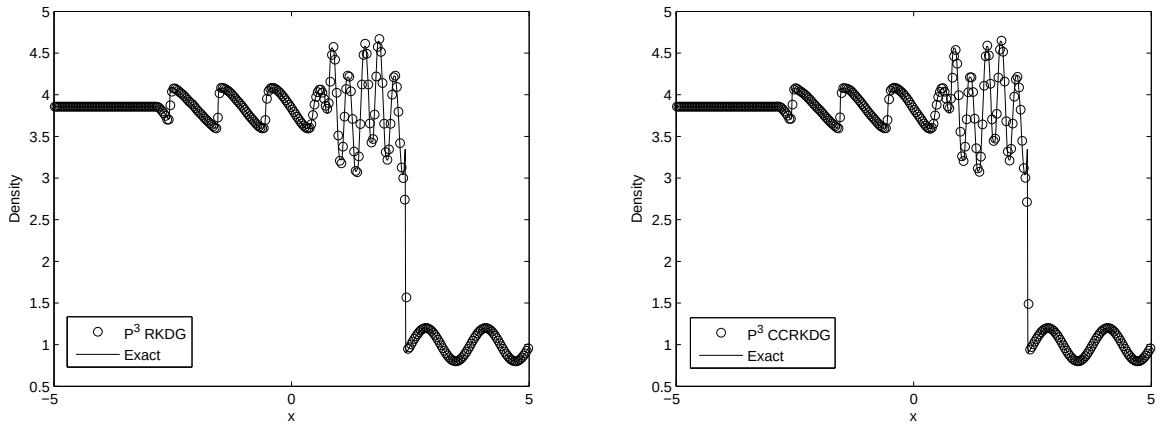
We compute the numerical solutions using 400 equal size cells and the 4^{th} order constrained RKDG method and the 4^{th} order RKDG method respectively. The density profiles of the solutions are plotted at the time $T = 0.038$ in Fig. 2. We can clearly see that the 4^{th}

Table 4: Accuracy test results of the 3^{rd} order RKDG method solving the 1D Burgers' equation. CFL = 0.23.

h	L_1 error	order	L_∞ error	order
1/40	3.62E-6	-	4.93E-5	-
1/80	4.49E-7	3.01	6.68E-6	2.88
1/160	5.56E-8	3.01	8.75E-7	2.93
1/320	6.90E-9	3.01	1.12E-7	2.97
1/640	3.24E-3	-	2.80E-1	-

Table 5: Accuracy test results of the 4^{th} order RKDG method solving the 1D Burgers' equation. CFL = 0.17.

h	L_1 error	order	L_∞ error	order
1/40	5.50E-8	-	8.75E-7	-
1/80	3.44E-9	4.00	5.51E-8	3.99
1/160	2.18E-10	3.98	3.49E-9	3.98
1/320	1.38E-11	3.98	2.20E-10	3.99
1/640	3.21E-10	-	5.97E-8	-



(a) (b)

Figure 1: Solutions of the 1D Shu-Osher problem computed on 300 cells. (a) The 4^{th} order RKDG solution compared with the "exact" solution; (b) The 4^{th} order constrained RKDG solution compared with the "exact" solution.

order constrained RKDG solution and the 4^{th} order RKDG solution have almost identical resolution for the 1D Woodward-Colella blast wave problem.

Example 3.1.2.3. 1D Lax problem [7]. It is the Euler equations with the Lax's initial data: the density ρ, momentum ρv and total energy E are 0.445, 0.311 and 8.928 in $(-1, 0)$; and are 0.5, 0 and 1.4275 in $(0, 1)$. We compute the numerical solutions using 200 equal size cells and the 4^{th} order constrained RKDG method and the 4^{th} order RKDG method respectively. The density profiles of the solutions are plotted at the time $T = 0.26$ in Fig. 3. We can clearly see that the 4^{th} order constrained RKDG solution and the 4^{th} order RKDG solution have almost identical resolution for the 1D Lax test problem.

From these 1D compressible gas flow test problems, we conclude that the constraint RKDG method combined with HR limiter, gives good quality results for problems containing strong shock waves in the solution.

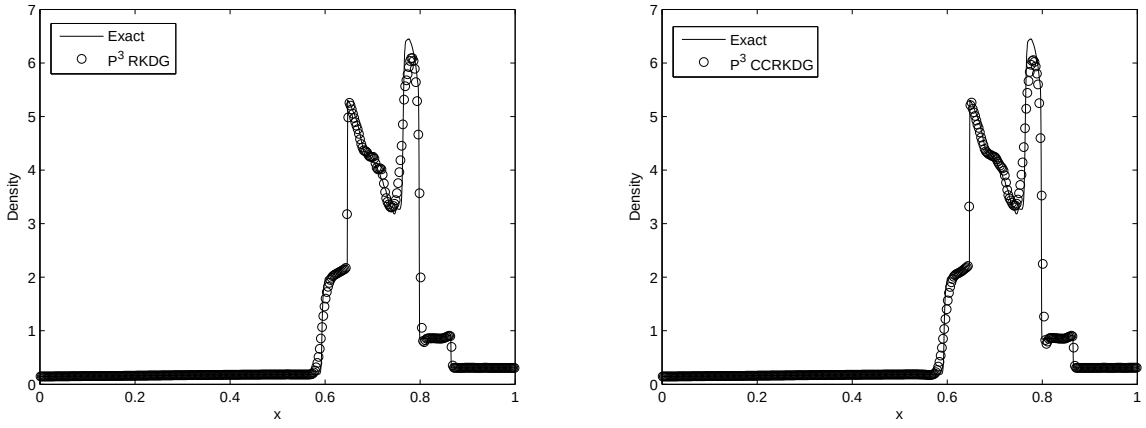
3.2 Two-dimensional Tests

We start with the 2D Burgers' equation

$$u_t + \left(\tfrac{1}{2}u^2\right)_x + \left(\tfrac{1}{2}u^2\right)_y = 0, \quad \text{in } (0,T)\times\Omega, \tag{3.1}$$

with a smooth solution to assess the limit of the CFL number for the 3^{rd} order constrained RKDG method. We then employ the Euler equations for gas dynamics to assess the resolution of the constrained RKDG method. The 2D Euler equations can be expressed in a conservative form as

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x + \mathbf{g}(\mathbf{u})_y = 0\ , \tag{3.2}$$



(a) (b)

Figure 2: Solutions of the 1D blast wave problem computed on 400 cells. (a) The 4^{th} order RKDG solution compared with the "exact" solution; (b) The 4^{th} order constrained RKDG solution compared with the "exact" solution.

where $\mathbf{u} = (\rho, \rho u, \rho v, E)$, $\mathbf{f}(\mathbf{u}) = (\rho u, \rho u^2 + p, \rho uv, u(E + p))$, and $\mathbf{g}(\mathbf{u}) = (\rho v, \rho uv, \rho v^2 + p, v(E + p))$. Here ρ is the density, (u, v) is the velocity, E is the total energy, p is the pressure, and $E = \frac{p}{\gamma-1} + \frac{1}{2}\rho(u^2 + v^2)$. γ is equal to 1.4 for all test cases.

To assess the CFL condition for the constrained RKDG method on 2D triangular meshes, we use the following definition of the CFL number, which is the maximum of

$$\frac{\triangle t(|\mathbf{u}| + c)}{\mathcal{D}},$$

where $\mathcal{D}$ is the diameter of the inscribed circle of a triangle, c is the speed of sound and $|\mathbf{u}|$, defined by $(|\mathbf{u}| = \sqrt{u^2 + v^2})$, is the speed of flow both evaluated by the local cell average value.

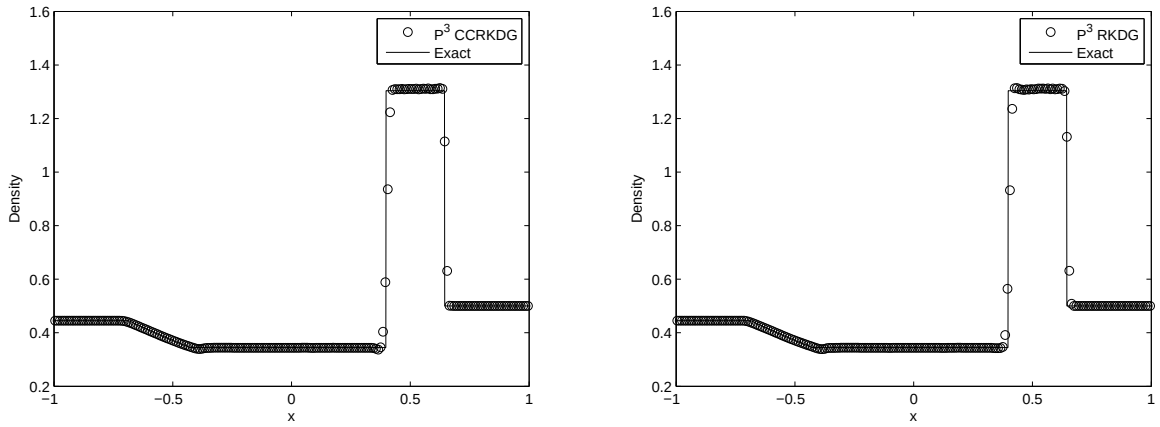
3.2.1 2D Burgers' equation with a smooth solution

To assess the CFL condition for the 3^{rd} order constrained RKDG method, we first utilize the 2D Burgers' equation (3.1) with the following initial condition and the periodic boundary condition

$$u(t = 0, x, y) = \tfrac{1}{4} + \tfrac{1}{2}\sin(2\pi(x + y)), \quad (x, y) \in \Omega,$$

where the domain Ω is the square $[0, 1] \times [0, 1]$. At $T = 0.1$, the exact solution is smooth. The convergence test is conducted on triangular meshes. See Fig. 4 for a typical mesh. The typical triangle edge length, denoted by h, is listed in Tables shown in this section. The errors presented are for u.

We found that the 2D 3^{rd} order constrained RKDG method is stable when CFL ≤ 0.36; while the 2D 3^{rd} order RKDG method is stable when CFL ≤ 0.23. Tables 6 and 7 show the accuracy test results of the constrained RKDG method when CFL $= 0.34$ and 0.36



(a) (b)

Figure 3: Solutions of the 1D Lax shock tube problem computed on 200 cells. (a) The 4^{th} order constrained RKDG solution compared with the "exact" solution; (b) The 4^{th} order RKDG solution compared with the "exact" solution.

respectively. Tables 8 and 9 show the accuracy test results of the RKDG method when CFL = 0.22 and 0.23 respectively.

The constrained RKDG method improves the CFL number over the RKDG method by about 50% for the 3^{rd} order case on triangular meshes for solving the 2D Burgers' equation. Moreover, we note that the magnitude of L_1 and L_∞ errors of the numerical solution of this test problem computed by the constrained RKDG method is less than that computed by the RKDG method. See Tables 6 and 8 for this comparison.

Table 6: Accuracy test results of the 3^{rd} order constrained RKDG method solving the 2D Burgers' equation. CFL = 0.34.

h	L_1 error	order	L_∞ error	order
1/8	7.93E-3	-	5.34E-2	-
1/16	1.37E-3	2.53	1.60E-2	1.74
1/32	1.97E-4	2.80	3.51E-3	2.19
1/64	2.76E-5	2.84	4.65E-4	2.92
1/128	3.81E-6	2.86	7.41E-5	2.65
1/256	5.25E-7	2.86	1.19E-5	2.64
1/512	7.15E-8	2.88	2.07E-6	2.52

3.2.2 2D Euler equations with a smooth solution

A two-dimensional gas dynamics problem [19] for the Euler equations is used to assess the CFL condition for the 3^{rd} order constrained RKDG method on triangular meshes again. The

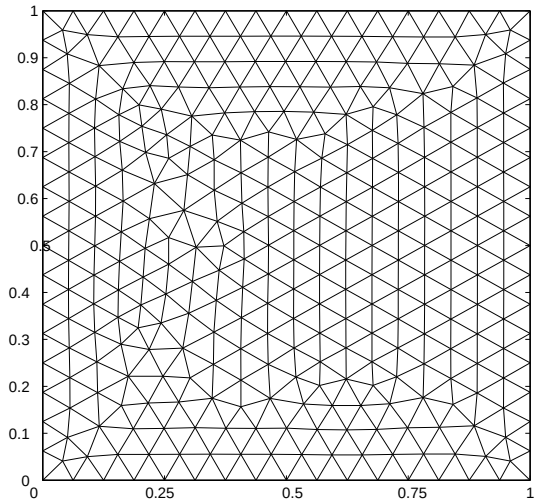


Figure 4: Representative mesh for accuracy test.

Table 7: Accuracy test results of the 3^{rd} order constrained RKDG method solving the 2D Burgers' equation. CFL = 0.36.

h	L_1 error	order	L_∞ error	order
1/8	8.18E-3	-	5.32E-2	-
1/16	1.37E-3	2.58	1.60E-2	1.73
1/32	1.97E-4	2.80	3.50E-3	2.19
1/64	2.93E-5	2.75	4.64E-4	2.92
1/128	4.83E-6	2.60	1.04E-4	2.16
1/256	1.77E-4	-	4.94E-1	-

Table 8: Accuracy test results of the 3^{rd} order RKDG method solving the 2D Burgers' equation. CFL = 0.22.

h	L_1 error	order	L_∞ error	order
1/8	1.23E-2	-	7.12E-2	-
1/16	2.31E-3	2.41	1.16E-2	2.62
1/32	3.61E-4	2.68	2.99E-3	1.96
1/64	5.49E-5	2.72	7.00E-4	2.07
1/128	8.06E-6	2.77	1.86E-4	1.91
1/256	1.17E-6	2.78	4.32E-5	2.11
1/512	1.68E-7	2.80	9.66E-6	2.16

exact solution is given by $\rho = 1 + 0.5\sin(x + y - (u + v)t)$, $u = 1.0$, $v = -0.7$ and $p = 1$. The convergence test is conducted on triangular meshes in the spatial domain $[0, 1] \times [0, 1]$ from the time $T = 0$ to $T = 0.1$. The typical triangle edge length, denoted by h, is listed in Tables shown in this section. The errors presented are for the density.

We found that the 2D 3^{rd} order constrained RKDG method is stable when CFL < 0.36; while the 3^{rd} order RKDG method is stable when CFL < 0.25. Tables 10 and 11 show the accuracy test results of the constrained RKDG method when CFL $= 0.35$ and 0.36 respectively. Tables 12 and 13 show the accuracy test results of the RKDG method when CFL $= 0.24$ and 0.25 respectively.

The constrained RKDG method improves the CFL number over the RKDG method by about 50% for the 3^{rd} order case on triangular meshes for this test problem. We also note that the magnitude of L_1 error of the numerical solution of this gas dynamics test problem computed by the constrained RKDG method becomes less than that computed by the RKDG method under the mesh refinement. See also Tables 10 and 12 for comparing L_1 errors.

Table 9: Accuracy test results of the 3^{rd} order RKDG method solving the 2D Burgers' equation. CFL = 0.23.

h	L_1 error	order	L_∞ error	order
1/8	1.23E-2	-	7.11E-2	-
1/16	2.30E-3	2.42	1.16E-2	2.62
1/32	3.61E-4	2.67	2.99E-3	1.96
1/64	5.50E-5	2.71	7.00E-4	2.10
1/128	8.38E-6	2.71	1.86E-4	1.91
1/256	Solution blows up	-	Solution blows up	-

Table 10: Accuracy test results of the 3^{rd} order constrained RKDG method solving the 2D Euler equations with smooth sine evolution. CFL = 0.35.

h	L_1 error	order	L_∞ error	order
1/4	8.32E-5	-	2.58E-4	-
1/8	1.36E-5	2.61	3.03E-5	3.09
1/16	1.77E-6	2.94	5.17E-6	2.55
1/32	2.23E-7	2.99	8.09E-7	2.68
1/64	2.72E-8	3.04	8.91E-8	3.18
1/128	1.36E-9	4.32	1.32E-8	2.75
1/256	1.33E-10	3.35	1.43E-9	3.21
1/512	1.66E-11	3.00	2.64E-10	2.44

Table 11: Accuracy test results of the 3^{rd} order constrained RKDG method solving the 2D Euler equations with smooth sine evolution. CFL = 0.36.

h	L_1 error	order	L_∞ error	order
1/4	8.32E-5	-	2.58E-4	-
1/8	1.36E-5	2.61	3.03E-5	3.09
1/16	1.77E-6	2.94	5.17E-6	2.55
1/32	2.23E-7	2.99	8.09E-7	2.68
1/64	2.72E-8	3.04	8.91E-8	3.18
1/128	3.39E-9	3.00	1.32E-8	2.75
1/256	1.33E-10	4.67	2.27E-9	2.54
1/512	Solution blows up	-	Solution blows up	-

Table 12: Accuracy test results of the 3^{rd} order RKDG method solving the 2D Euler equations with smooth sine evolution. CFL = 0.24.

h	L_1 error	order	L_∞ error	order
1/4	2.15E-5	-	5.93E-5	-
1/8	4.62E-6	2.22	1.18E-5	2.33
1/16	7.49E-7	2.62	2.54E-6	2.22
1/32	1.06E-7	2.82	4.29E-7	2.57
1/64	3.23E-8	1.71	5.79E-8	2.89
1/128	1.70E-9	4.25	1.13E-8	2.36
1/256	2.56E-10	2.73	1.46E-9	2.95
1/512	3.41E-11	2.91	1.98E-10	2.88

Table 13: Accuracy test results of the 3^{rd} order RKDG solution solving the 2D Euler equations with smooth sine evolution. CFL = 0.25.

h	L_1 error	order	L_∞ error	order
1/4	2.15E-5	-	5.86E-5	-
1/8	4.62E-6	2.22	1.18E-5	2.31
1/16	7.49E-7	2.62	2.54E-6	2.22
1/32	1.06E-7	2.82	4.29E-7	2.57
1/64	3.23E-8	1.71	5.79E-8	2.89
1/128	5.68E-9	2.51	1.13E-8	2.36
1/256	Solution blows up	-	Solution blows up	-

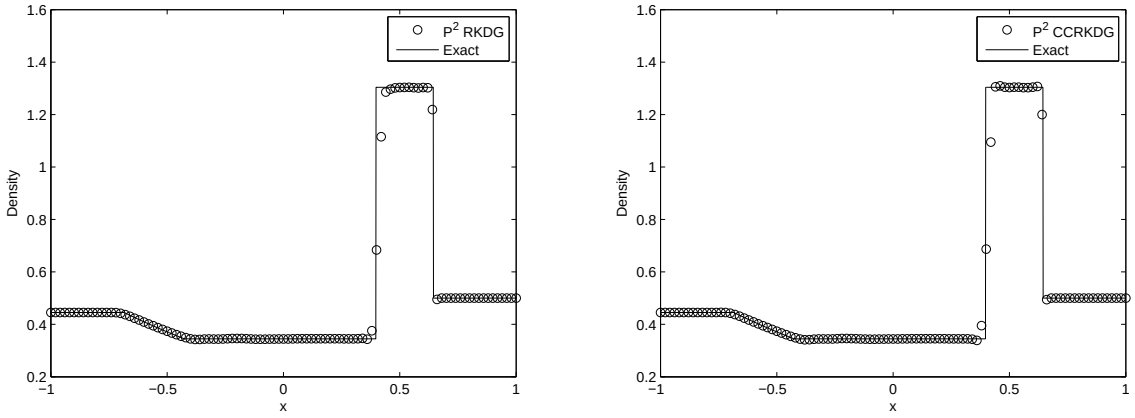
3.2.3 2D Euler equations with discontinuous solutions

We test 2D problems with discontinuities in solutions to assess the non-oscillatory property of numerical solutions computed by the 2D constrained RKDG method together with HR limiter, again using the Euler equations for gas dynamics.

Example 3.2.3.1. 2D Lax problem. This test problem is set by modifying the Lax shock tube problem taken from [7]. We solve the Euler equations in a rectangular domain of $[-1, 1] \times [0, 0.2]$, with a triangulation of approximately 101 vertices in the x-direction and 11 vertices in the y-direction. The initial data is

$$(\rho, u, p) = \begin{cases} (0.445, 0.698, 3.528), & \text{if } x \leq 0 \\ (0.5, 0, 0.571), & \text{if } x > 0 \ . \end{cases} \tag{3.3}$$

Initially, the y-component of the velocity is zero. The density profile at $t = 0.26$ is shown here. Figures 5(a) and 5(b) are obtained by interpolating the numerical solutions along the line $y = 0.1$ on 101 equally spaced points. We can see that both of the 3^{rd} order RKDG and constrained RKDG methods computed numerical solutions with almost identical and high resolution and with almost no noise after a component-wise HR limiting for this test problem.



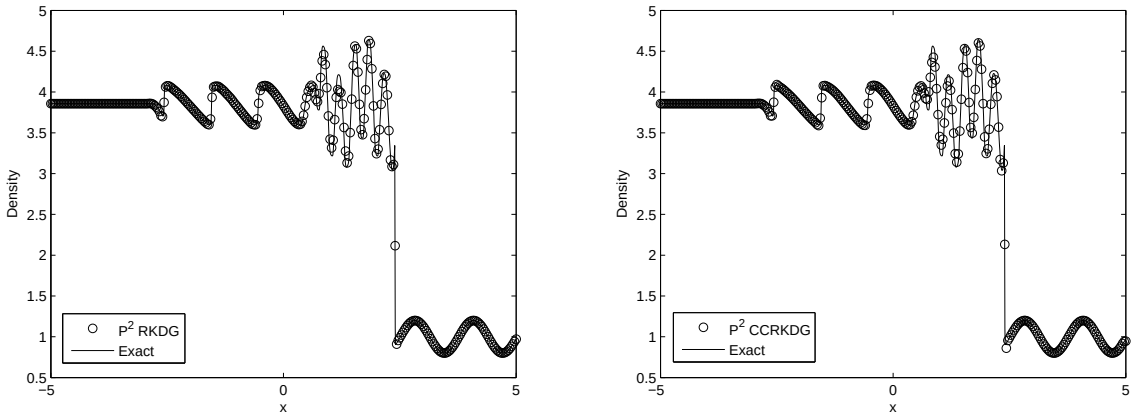
(a) (b)

Figure 5: Solutions of the 2D Lax problem. Third-order results. Density ρ. (a) The 3^{rd} order RKDG solution; (b) The 3^{rd} order constrained RKDG solution.

Example 3.2.3.2. 2D Shu-Osher problem. This test problem is set by modifying the Shu-Osher problem [21]. We solve the Euler equations in a rectangular domain of $[-5, 5] \times [0, 0.1]$ with a triangulation of about 301 vertices in the x-direction and 4 vertices in the y-direction. The initial data is

$$(\rho, u, p) = \begin{cases} (3.857143, 2.629369, 10.333333) & \text{if } x \leq -4 \\ (1 + 0.2\sin(5x), 0, 1) & \text{if } x \geq -4 \ . \end{cases} \tag{3.4}$$

Initially, the y-velocity is zero. At $t = 1.8$, the density profiles along $y = 0.05$ are shown in Figures 6(a) and 6(b). Both of the 3^{rd} order RKDG and constrained RKDG methods computed solutions with almost identical and high resolution and with almost no noise after a component-wise HR limiting for the 2D Shu-Osher test problem.



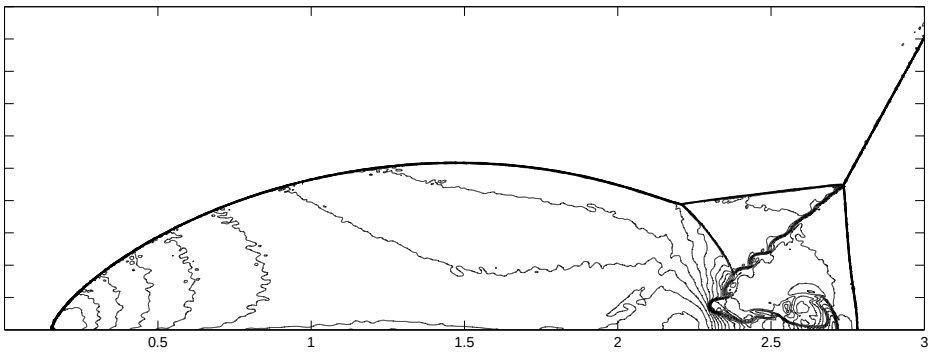
(a) (b)

Figure 6: Solutions of the 2D Shu-Osher problem. Third-order results. Density ρ. (a) The 3^{rd} order RKDG solution; (b) The 3^{rd} order constrained RKDG solution.

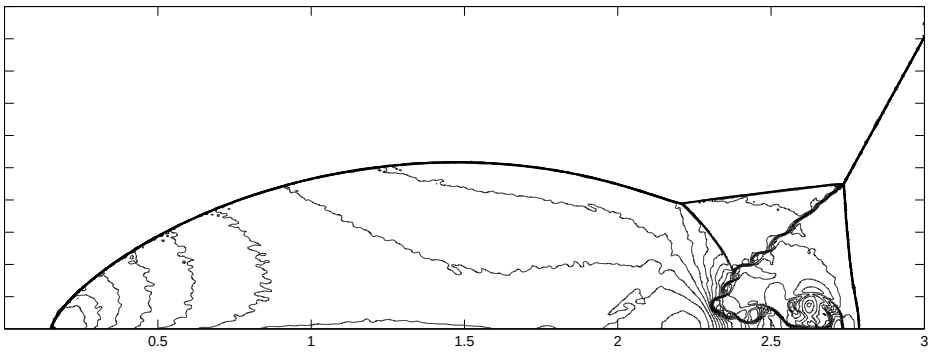
Example 3.2.3.3. Double Mach reflection. The Double Mach reflection problem is taken from [22]. We solve the Euler equations in a rectangular computational domain of $[0, 4] \times [0, 1]$. A reflecting wall lies at the bottom of the domain starting from $x = \frac{1}{6}$. Initially a right-moving Mach 10 shock is located at $x = \frac{1}{6}, y = 0$, making a 60^0 angle with the x axis and extends to the top of the computational domain at $y = 1$. The reflective boundary condition is used at the wall.

We test our method on unstructured meshes with the triangle edge length roughly equal to $\frac{1}{400}$. The density contour of the flow in the $[0, 3] \times [0, 1]$ region at the time $t = 0.2$ is shown with 30 equally spaced contour lines. Fig. 7 is the contour plot of the numerical solutions computed by the 3^{rd} order RKDG and constrained RKDG methods respectively. Fig. 8 shows the "blown-up" portion around the double Mach region. We can see that while both of the RKDG and constrained RKDG methods successfully reproduce the vortex sheet roll-up; the solution computed by the constrained RKDG method is better than the one computed by the RKDG method, namely the constrained RKDG method picks up more roll-up and computes smoother contour lines.

Example 3.2.3.4. Flow past a forward facing step. This flow problem is again taken from [22]. The setup of the problem is the following: a right-going Mach 3 uniform flow enters a wind tunnel of 1 unit wide and 3 units long. The step is 0.2 units high and is located 0.6 units from the left side of the tunnel. The problem is initialized by a uniform, right-going Mach 3 flow, which has density 1.4, pressure 1.0, and velocity 3.0. The initial state of the gas is also used at the left side boundary. At the right side boundary, the out-flow boundary



(a)

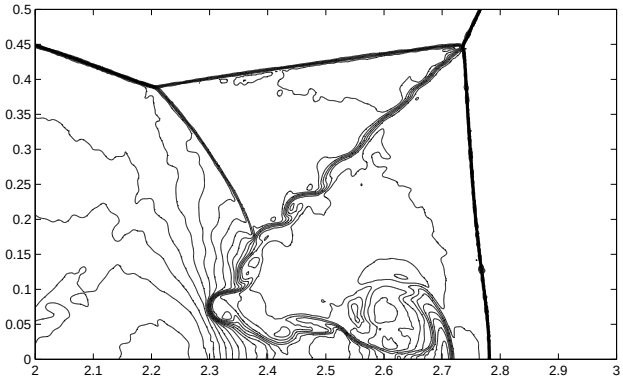


(b)

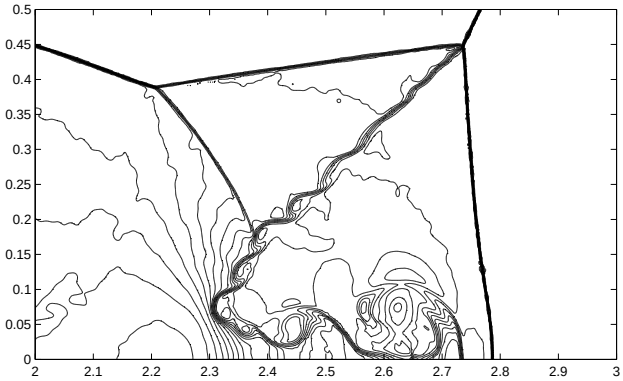
Figure 7: Double Mach reflection problem. Third-order results. Density ρ. (a) The 3^{rd} order RKDG solution; (b) The 3^{rd} order constrained RKDG solution.

condition is applied there. Reflective boundary condition is applied along the walls of the tunnel.

The corner of the step is a singularity. Unlike in [22] and in other studies, we do not modify our scheme near the corner, which is known to lead to an erroneous entropy layer at the downstream bottom wall, as well as a spurious Mach stem at the bottom wall. Instead, we use the approach taken in [3], which is to locally refine the mesh near the corner, to decrease these artifacts. The edge length of the triangle away from the corner is roughly equal to $\frac{1}{160}$. Near the corner, the edge length of the triangle is roughly equal to $\frac{1}{320}$. Fig. 9 is the contour plot of the numerical solutions computed by the 3^{rd} order RKDG and constrained RKDG methods respectively. Comparing results in Fig. 9, we can see that the resolution of the solution computed by the constrained RKDG method is better, especially for the contour lines around the triple point. Smoother contour lines are obtained in the constrained RKDG case.



(a)

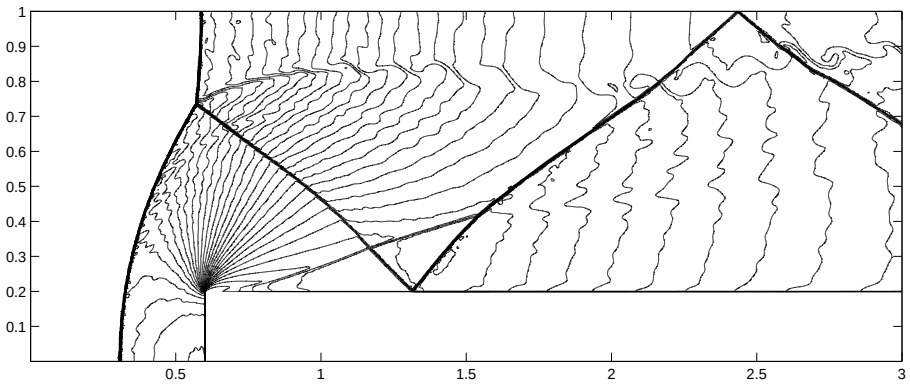


(b)

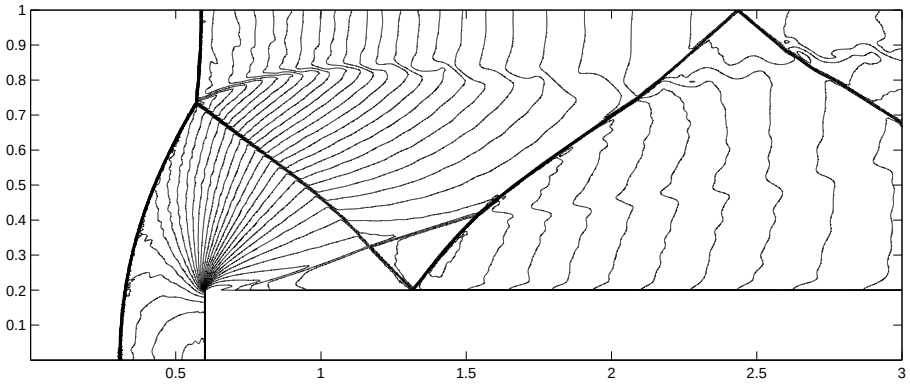
Figure 8: Double Mach reflection problem. Blown-up region around the double Mach stems. Third-order results. Density ρ. (a) The 3^{rd} order RKDG solution; (b) The 3^{rd} order constrained RKDG solution.

3.3 Remark on Computational Cost of Constrained RKDG Method

To estimate the computational cost of the constrained RKDG method, we use the 2D Burgers' equation with a smooth solution as a test case. See Section 3.2.1 for the description of this benchmark problem. We employ a mesh with the triangle edge length roughly equal to $\frac{1}{128}$. The code is written in C and is compiled with "g++ -O3". Simulations are performed on a Linux workstation with an Intel i7 2.93 GHz processor. Table 14 shows cpu times spent by the RKDG and constrained RKDG methods for CFL = 0.2 and 0.3 respectively. To conclude, when CFL = 0.2, the cpu time of the constrained RKDG method is about 12% more than that of the RKDG method; while the constrained RKDG method with CFL = 0.3 saves about 26% cpu time over the RKDG method with CFL = 0.2.



(a)



(b)

Figure 9: Forward-facing step problem. Third-order results. Density ρ. (a) The 3^{rd} order RKDG solution; (b) The 3^{rd} order constrained RKDG solution.

4 Concluding Remarks

We have developed a conservation constrained RKDG method for solving conservation Laws. The new formulation requires the computed RKDG solution in a cell to satisfy additional conservation constraint in adjacent cells and does not increase the complexity or change the compactness of the original RKDG method. This conservation constrained RKDG method

Table 14: Cpu time comparison between the RKDG and constrained RKDG methods

	CFL = 0.2	CFL = 0.3
RKDG	cpu time = 344 sec.	-
Constrained RKDG	cpu time = 386 sec.	cpu time = 255 sec.

improves the CFL number over the RKDG method as well as the robustness of the RKDG scheme. The CFL number is improved around 50% for the 3^{rd} order case on 2D triangular meshes and 250% for the 4^{th} order case in 1D. Moreover, for the multi-dimensional smooth solution test problems, the constrained RKDG method also reduces the magnitude of the solution error (as least for L_1 error). For the multi-dimensional test problems with discontinuous solutions, the constrained RKDG method together with HR limiter also improves the resolution of numerical solutions.

In the future, we will explore the higher order (≥ 4) constrained DG formulation in multi-dimensions with TVD Runge-Kutta time-stepping as well as other time-stepping methods.

Acknowledgment

Simulations were performed on the Notre Dame Biocomplexity Cluster supported in part by NSF MRI Grant No. DBI-0420980.

References

[1] R. Abgrall. On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation, *J. Comput. Phys.*, 114:45–58, 1994.

[2] R. Biswas, K. Devine and J. Flaherty. Parallel, adaptive finite element methods for conservation laws. *Appl. Numer. Math.*, 14:255–283, 1994.

[3] B. Cockburn and C.-W. Shu. The TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws V: multidimensional systems. *J. Comput. Phys.*, 141:199–224, 1998.

[4] B. Cockburn, S. Hou and C.-W. Shu. The TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case. *Math. Comp.*, 54:545–581, 1990.

[5] 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. *J. Comput. Phys.*, 52:411–435, 1989.

[6] B. Cockburn and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework. *Math. Comp.*, 52:411–435, 1989.

[7] P. Lax. Weak solutions of nonlinear hyperbolic equations and their numerical computations. *Comm. Pure Appl. Math.*, 7:159, 1954.

[8] B. van Leer. Toward the ultimate conservative difference scheme: II. Monotonicity and conservation combined in a second order scheme. *J. Comput. Phys.*, 14:361–370, 1974.

[9] B. van Leer. Towards the ultimate conservative difference scheme: IV. A new approach to numerical convection. *J. Comput. Phys.*, 23:276–299, 1977.

[10] B. van Leer. Towards the ultimate conservative difference scheme: V. A second order sequel to Godunov's method. *J. Comput. Phys.*, 32:101–136, 1979.

[11] B. van Leer and S. Nomura. Discontinuous Galerkin for diffusion. *AIAA-2005-5108*, 2005.

[12] Y.-J. Liu, C.-W. Shu, E. Tadmor and M.-P. Zhang. Central discontinuous Galerkin methods on overlapping cells with a non-oscillatory hierarchical reconstruction. *SIAM J. Numer. Anal.*, 45:2442–2467, 2007.

[13] Y.-J. Liu, C.-W. Shu, E. Tadmor and M.-P. Zhang. Non-oscillatory hierarchical reconstruction for central and finite volume schemes. *Comm. Comput. Phys.*, 2:933–963, 2007.

[14] Y.-J. Liu, C.-W. Shu, E. Tadmor and M.-P. Zhang. L2 stability analysis of the central discontinuous Galerkin method and a comparison between the central and regular discontinuous Galerkin methods. *ESAIM: Math. Mod. Numer. Anal.*, 42:593–607, 2008.

[15] H. Luo, J.D. Baum and R. Lohner, A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids, *J. Comput. Phys.*, 225:686–713, 2007.

[16] J. Qiu and C.-W. Shu, Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method. II: Two dimensional case. *Comput. Fluids*, 34:642–663, 2005.

[17] W. Reed and T. Hill. Triangular mesh methods for the neutron transport equation. Tech. report la-ur-73-479, Los Alamos Scientific Laboratory, 1973.

[18] C.-W. Shu. TVB uniformly high-order schemes for conservation laws. *Math. Comp.*, 49:105–121, 1987.

[19] 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, Berlin. Springer.* , 1697, 1998.

[20] C.-W. Shu and S. Osher. Efficient Implementation of essentially non-oscillatory shock capturing schemes. *J. Comput. Phys.*, 77:439–471, 1988.

[21] C.-W. Shu and S. Osher. Efficient Implementation of essentially non-oscillatory shock capturing schemes, II. *J. Comput. Phys.*, 83:32–78, 1989.

[22] P. Woodward and P. Colella. Numerical simulation of two-dimensional fluid flows with strong shocks. *J. Comput. Phys.*, 54:115, 1984.

[23] T. Warburton and T. Hagstrom. Taming the CFL number for discontinuous Galerkin methods on structured meshes. *SIAM J. Numer. Anal.*, 46: 3151–3180, 2008.

[24] Z.-L. Xu and Y.-J. Liu and C.-W. Shu. Hierarchical reconstruction for discontinuous Galerkin methods on unstructured grids with a WENO type linear reconstruction and partial neighboring cells, *J. Comput. Phys.*, 228:2194–2212, 2009.

[25] Z.-L. Xu and G. Lin. Spectral/hp element method with hierarchical reconstruction for solving nonlinear hyperbolic conservation laws, *Acta Mathematica Scientia*, 29(B):1737–1748, 2009.

[26] J. Zhu, J.-X. Qiu, C.-W. Shu and M. Dumbser. Runge-Kutta discontinuous Galerkin method using WENO limiters II: unstructured meshes. *J. Comput. Phys.*, 227:4330–4353, 2008.