



# On the role of Riemann solvers in Discontinuous Galerkin methods for magnetohydrodynamics

V. Wheatley<sup>a,\*</sup>, H. Kumar<sup>b</sup>, P. Huguenot<sup>b</sup>

<sup>a</sup> Centre for Hypersonics, The University of Queensland, Australia

<sup>b</sup> Seminar for Applied Mathematics, ETH Zurich, Switzerland

## ARTICLE INFO

### Article history:

Received 3 March 2009

Received in revised form 2 September 2009

Accepted 4 October 2009

Available online 13 October 2009

### Keywords:

Riemann solvers

Discontinuous Galerkin methods

Limiters

## ABSTRACT

It has been claimed that the particular numerical flux used in Runge–Kutta Discontinuous Galerkin (RKDG) methods does not have a significant effect on the results of high-order simulations. We investigate this claim for the case of compressible ideal magnetohydrodynamics (MHD). We also address the role of limiting in RKDG methods.

For smooth nonlinear solutions, we find that the use of a more accurate Riemann solver in third-order simulations results in lower errors and more rapid convergence. However, in the corresponding fourth-order simulations we find that varying the Riemann solver has a negligible effect on the solutions.

In the vicinity of discontinuities, we find that high-order RKDG methods behave in a similar manner to the second-order method due to the use of a piecewise linear limiter. Thus, for solutions dominated by discontinuities, the choice of Riemann solver in a high-order method has similar significance to that in a second-order method. Our analysis of second-order methods indicates that the choice of Riemann solver is highly significant, with the more accurate Riemann solvers having the lowest computational effort required to obtain a given accuracy. This allows the error in fourth-order simulations of a discontinuous solution to be mitigated through the use of a more accurate Riemann solver.

We demonstrate the minmod limiter is unsuitable for use in a high-order RKDG method. It tends to restrict the polynomial order of the trial space, and hence the order of accuracy of the method, even when this is not needed to maintain the TVD property of the scheme.

© 2009 Elsevier Inc. All rights reserved.

## 1. Introduction

To simulate compressible flows that contain shocks along with small-scale features such as turbulence, we require numerical methods that are shock capturing, but also exhibit high-order accuracy and low numerical dissipation away from shocks [1]. Runge–Kutta Discontinuous Galerkin (RKDG) methods are shock capturing and high-order accurate away from discontinuities, thus they are a candidate method for carrying out such simulations.

Discontinuous Galerkin (DG) methods were first introduced by Hill and Reed [2] for the neutron transport equations (linear hyperbolic equations). LeSaint and Raviart [3] proved a rate of convergence of  $O(\Delta x)^k$  for general triangulations and of  $O(\Delta x)^{k+1}$  for Cartesian meshes, where  $\Delta x$  is the element size and  $k$  is the polynomial order of the approximate solution. In case of general triangulations, this result was then improved by Jhonson and Pitkaranta [4] to  $O(\Delta x)^{k+1/2}$ , which was confirmed to be optimal by Peterson [5].

\* Corresponding author. Tel.: +61 7 33652969.

E-mail address: [v.wheatley@uq.edu.au](mailto:v.wheatley@uq.edu.au) (V. Wheatley).

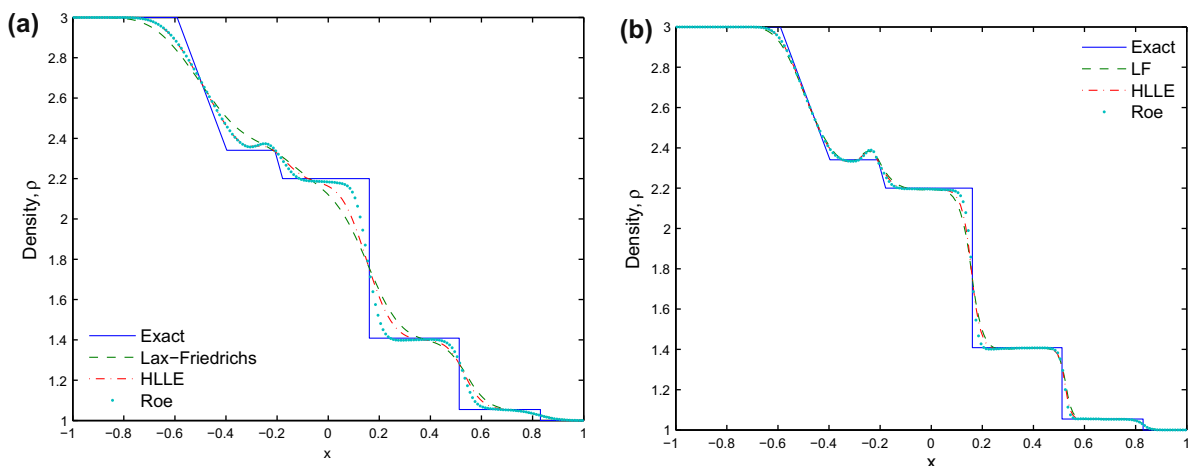
These methods were then generalized for systems of hyperbolic conservation laws by Cockburn and co-workers [6–10]. In space the solution is approximated using piecewise polynomials on each element. Exact or approximate Riemann solvers from finite volume methods are used to compute the numerical fluxes between elements. Limiters are used to achieve non-oscillatory approximate solutions, if they contain shocks [11]. For these reasons, DG methods can be seen as generalization of finite volume methods to higher order. For time integration, the total variation diminishing (TVD) explicit Runge–Kutta (RK) methods proposed by Shu and Osher [12] are used.

RKDG methods have many important advantages. Like finite element methods, RKDG methods are well suited for simulating flows in complicated geometries. These methods can easily handle adaptivity strategies, because of the assumed discontinuity of the solution at element interfaces. This allows refining or unrefining of the triangulation to be done without taking into account the continuity restrictions typical of conforming finite element methods. Similarly, the degree of the polynomial approximation within an element can be changed without affecting the solution on other elements. Another important advantage is that these methods are highly parallelizable because to update the solution on a given element, only information from elements with which it shares a face is needed.

At present, it is believed that the particular numerical flux or Riemann solver used does not have a significant effect on the results of high-order RKDG simulations [13]. Such a conclusion is supported by numerical evidence such as that shown in Fig. 1. This figure shows the results of simulations of the MHD shock tube problem described in Section 4.1 with three different flux calculators (see Section 4 for descriptions), along with the exact solution to the problem. The results of first- and second-order simulations are shown in Fig. 1(a) and (b), respectively. Comparing these results, it appears that the absolute error in the numerical solution is far less sensitive to the choice of numerical flux when the second-order scheme is used. This seems to indicate that as the order of a simulation increases, the choice of numerical flux becomes less significant. This view has led to the simple and highly dissipative Lax–Friedrichs (LF) flux being used within many RKDG methods [13]. Our goal is to rigorously examine the effect of more accurate numerical flux calculators in high-order RKDG methods, with particular emphasis on high-order simulations featuring discontinuities. The influence of accurate flux calculators in high-order RKDG methods is intimately tied to the performance of the limiters in the method. For this reason, we also examine the performance of limiters in high-order simulations.

For the Euler equations, the effect of Riemann solvers has been previously evaluated by Qiu et al. [14]. One- and two-dimensional numerical simulations were carried out to compare various Riemann solvers based on performance measures such as numerical error, resolution of discontinuities and CPU times. The LF flux was shown to require the least CPU-time among all the fluxes that were compared, but it also produced the largest numerical errors. Whereas second-order fluxes such as Lax–Wendroff (LW) and Warming–Beam (WB) were found to be unstable. The Harten, Lax and van Leer (HLL) [15], HLLC [16] and MUSTA [17] fluxes were proposed as good choices for RKDG simulations. However, the data generated was not correlated to demonstrate which scheme is the most computationally efficient, or if the benefits of using more accurate Riemann solvers are dependent on the order of accuracy. In addition, error norms were only computed for a smooth linear problem, while we anticipate that the use of accurate Riemann solvers will be most significant in discontinuous non-linear problems.

This report is organized as follows: in the next section we present the governing equations for the simulations. In Section 3 a brief description of the RKDG method is presented. In Section 4, a number of flux calculators for ideal MHD are tested, leading to the selection of appropriate flux calculators for use in the RKDG method in different circumstances. The limiters used within the RKDG method are described and tested in Section 5. The results of numerical test cases are presented and analyzed in Section 6. Finally, the conclusions that have been drawn from this work are presented in Section 7.



**Fig. 1.** Density profiles at  $t = 0.4$  from (a) first-order and (b) second-order accurate simulations of the MHD shock tube problem described in Section 4.1. The exact solution to the problem is shown along with numerical results using the LF, HLLC and Roe fluxes described in Section 4.

## 2. Governing equations

The governing equations for the simulations presented here are the ideal MHD equations. These govern the evolution of a quasi-neutral conducting fluid and the magnetic field within it, neglecting the magnetization of individual particles, the hall current, ion slip and the time rate of change of the electric field in Maxwell's equations. The complete details of their derivation may be found, for example, in Sutton and Sherman [18]. In the absence of viscosity, thermal conductivity, electrical resistivity and interspecies diffusion, the ideal MHD equations are,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (1)$$

$$\rho \frac{D\mathbf{u}}{Dt} = -\nabla p + \frac{1}{\mu_0} (\nabla \times \mathbf{B}) \times \mathbf{B}, \quad (2)$$

$$\rho \frac{D}{Dt} \left( e + \frac{1}{2} \mathbf{u} \cdot \mathbf{u} \right) = -\frac{1}{\mu_0} (\nabla \times \mathbf{B}) \cdot (\mathbf{u} \times \mathbf{B}) - \nabla \cdot (p\mathbf{u}), \quad (3)$$

$$\nabla \cdot \mathbf{B} = 0, \quad (4)$$

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{u} \times \mathbf{B}). \quad (5)$$

Here,  $\rho$  is the density,  $p$  is the pressure,  $\mathbf{u}$  is the velocity,  $\mathbf{B}$  is the magnetic field,  $e$  is the internal energy per unit mass and  $\mu_0$  is the magnetic permeability of vacuum. It is convenient to normalize  $\mathbf{B}$  by  $\sqrt{\mu_0}$ , thus eliminating  $\mu_0$  from the system of equations.

Eqs. (1)–(5) can be written in conservation form as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_j(\mathbf{U})}{\partial x_j} = 0, \quad (6)$$

where the vector of conserved variables  $\mathbf{U} \equiv \mathbf{U}(x_i, t)$  is,

$$\mathbf{U} = \{\rho, \rho u_i, B_i, \rho e_T\}^T,$$

and the flux vectors  $\mathbf{F}_j(\mathbf{U})$  are,

$$\mathbf{F}_j(\mathbf{U}) = \left\{ \rho u_j, \rho u_i u_j + \left( p + \frac{1}{2} B_k B_k \right) \delta_{ij} - B_i B_j, u_j B_i - u_i B_j, \left( \rho e_T + p + \frac{1}{2} B_k B_k \right) u_j - B_j (B_k u_k) \right\}^T.$$

Here,  $\rho e_T$  is the total energy per unit volume of the plasma. The plasma is assumed to be ideal with constant specific heats, allowing the following equation of state to be used to close the set of equations:

$$\rho e_T = \frac{p}{\gamma - 1} + \frac{1}{2} \rho u_k u_k + \frac{1}{2} B_k B_k.$$

## 3. Runge–Kutta Discontinuous Galerkin method

For hyperbolic problems, solutions may be piecewise continuous, meaning that they are smooth in regions separated by discontinuities. This behavior is mimicked by DG methods as they allow  $L^2$  jumps at the boundaries of subdomains even for operators of higher than first order. This allows any complete set of trial functions to be used to represent the solution on each subdomain. This is not the case in the standard Galerkin formulation where for second-order operators,  $C^0$  continuity is required across subdomains. RKDG methods for systems of hyperbolic conservation laws were developed by Cockburn et al. [6–10]. For completeness, we present the following brief account of their derivation.

### 3.1. Variational form

Consider a domain  $\Omega \in \mathbb{R}^n$ . Let us define a triangulation  $M$  of  $\Omega$  as a finite collection  $\{E_i\}_{i=1}^m$ ,  $m \in \mathbb{N}$ , of non-degenerate polygons such that,

$$\bar{\Omega} = \bigcup \{\bar{E}_i, i = 1, \dots, m\}, \quad (7)$$

$$E_i \cap E_j = \emptyset \iff i \neq j, \quad (8)$$

and for all  $i, j \in \{1, \dots, m\}$ ,  $i \neq j$ , the intersection  $\bar{E}_i \cap \bar{E}_j$  is either  $\emptyset$  or a vertex, edge or face of both  $E_i$  and  $E_j$ . For two-dimensional quadrilateral elements, we introduce a reference element  $\Sigma \equiv [-1, 1] \times [-1, 1]$  and the mapping,

$$\Phi : E \rightarrow \Sigma \quad (9)$$

that maps quadrilateral elements to the reference element. The mapping  $\Phi$  is a bilinear mapping.

It is sufficient for us to consider the following generic scalar advection equation:

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{F} = 0, \tag{10}$$

where  $u$  is a conserved variable and  $\mathbf{F}$  is the inviscid flux vector. The variational form used in the RKDG method is derived by multiplying by the test function  $v$  and integrating over each element separately. After using integration by parts on the divergence term, we obtain,

$$\int_E \frac{\partial u}{\partial t} v dx + \int_{\partial E} \mathbf{vn} \cdot \mathbf{F}(u) ds - \int_E \nabla v \cdot \mathbf{F}(u) dx = 0. \tag{11}$$

The flux vector  $\mathbf{F}(u)$  in the second term must be evaluated on the boundary of the element where  $u$  may be discontinuous and thus has two possible values;  $u_i$  on the interior of the element under consideration and  $u_e$  on the exterior. To account for this, we replace  $\mathbf{F}(u)$  with the numerical flux function  $\hat{\mathbf{F}}(u_i, u_e)$ , which can be computed taking upwind considerations into account. The final variational form is derived by using integration by parts once more on the third term to eliminate the gradient of the test function,

$$\frac{\partial}{\partial t} \int_E uv dx + \int_{\partial E} v(\hat{\mathbf{F}}(u_i, u_e) - \mathbf{F}(u_i)) \cdot \mathbf{n} ds + \int_E \nabla \cdot \mathbf{F}(u) v dx = 0. \tag{12}$$

### 3.2. Basis functions

In the RKDG method, the unknowns and data within each element are expanded in terms of a suitable set of basis functions  $\phi_{pq}(\mathbf{x})$ :

$$f(x, y) = \sum_p \sum_q a_{pq} \phi_{pq}(\xi_1, \xi_2).$$

Here,  $(\xi_1, \xi_2)$  is a local coordinate system associated with that element. The RKDG method we use is based on the Nektar code by Sherwin and Karniadakis [19]. The original code has been extended to include Runge–Kutta time stepping, slope limiters and accurate Riemann solvers, amongst other features. We will refer to the modified version of the code as Nektar-m. The set of polynomial basis functions used in Nektar was proposed by Dubiner [20] in two dimensions and extended to three dimensions in [19]. The two-dimensional basis functions have the form,

$$\phi_{pq}(\xi_1, \xi_2) = \psi_p^a(\xi_1) \psi_q^b(\xi_2),$$

for quadrilaterals and,

$$\phi_{pq}(\xi_1, \xi_2) = \psi_p^a(\eta_1) \psi_q^b(\eta_2),$$

for triangles. The Cartesian coordinates  $\xi_1, \xi_2$  are defined on the reference element as shown in Fig. 2, while  $\eta_1, \eta_2$  are non-Cartesian and are given by,

$$\eta_1 = 2 \frac{1 + \xi_1}{1 - \xi_2} - 1, \quad \eta_2 = \xi_2.$$

The functions  $\psi_p^a(z)$  and  $\psi_q^b(z)$  are,

$$\psi_p^a(z) = P_p^{0,0}(z), \quad \psi_q^b(z) = \left(\frac{1-z}{2}\right)^p P_q^{2p+1,0}(z),$$

where  $P_n^{\alpha,\beta}$  is the  $n$ th-order Jacobi polynomial with weights  $\alpha$  and  $\beta$ . These basis functions are orthogonal in the Legendre inner product over each element, resulting in a diagonal mass matrix. The functions are polynomial in both the Cartesian

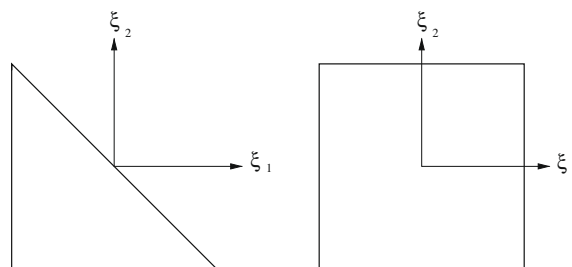


Fig. 2. Reference element local coordinate system.

and non-Cartesian coordinates. It was proved by Warburton and Karniadakis [21] that the coefficients of the basis functions in a solution decay exponentially with polynomial order, thus the numerical solution converges exponentially as the maximum polynomial order of the approximation is increased.

### 3.3. Runge–Kutta time discretization

To advance solutions in time, the RKDG method uses a Runge–Kutta (RK) time marching scheme. In this manuscript we present the results of second-, third- and fourth-order accurate RKDG schemes, thus we limit ourselves to presenting the RK schemes used in these cases. For second- and third-order simulations, we use the TVD RK schemes of [22]. For fourth-order simulations we use the classic scheme.

Consider the semi-discrete ODE,

$$\frac{du_h}{dt} = L_h(u_h).$$

Let  $u_h^n$  be the discrete solution at time  $t^n$ , and let  $\Delta t^n = t^{n+1} - t^n$ . In order to advance a numerical solution from time  $t^n$  to  $t^{n+1}$ , the RK algorithm is as follows:

1. Set  $u_h^{(0)} = u_h^n$ .
2. For  $i = 1, \dots, k + 1$ , compute,

$$u_h^{(i)} = \sum_{l=0}^{i-1} \alpha_{il} u_h^{(l)} + \beta_{il} \Delta t^n L_h(u_h^{(l)}).$$

3. Set  $u_h^{n+1} = u_h^{(k+1)}$ .

The values of the coefficients used in the TVD schemes are shown in Table 1.

For the linear advection equation, it was proved by [9] that the RKDG method is  $L^\infty$ -stable for piecewise linear ( $k = 1$ ) approximate solutions if a second-order RK scheme is used with a time-step that satisfies,

$$c \frac{\Delta t}{\Delta x} \leq \frac{1}{3},$$

where  $c$  is the constant advection speed. The numerical experiments of [23] show that when approximate solutions of polynomial degree  $k$  are used, an order  $k + 1$  RK scheme must be used, which simply corresponds to matching the temporal and spatial accuracy of the RKDG scheme. In this case the  $L^\infty$ -stability condition is

$$c \frac{\Delta t}{\Delta x} \leq \frac{1}{2k + 1}.$$

For the nonlinear case, the same stability conditions are used but with  $c$  replaced by the maximum wave speed of the system.

### 3.4. Algorithm

Here we describe the RKDG algorithm as it is implemented in Nektar-m. The majority of the algorithm remains unchanged from that presented in [21]. It utilizes two sets of quadrature points on element boundaries: The set of interface quadrature points  $Q^I$ , where a Riemann solver is used to compute the numerical fluxes, and the set of edge quadrature points  $Q^E$ , where the fluxes must be evaluated to advance the solution in time.  $Q^I$  corresponds to the Gauss quadrature points along each edge, which do not include the vertices where there are no clearly defined left and right states that could be input to the Riemann solver. The set  $Q^E$  depends on the type of elements used and typically includes the vertices. The basic steps in the RKDG algorithm implemented in Nektar-m are as follows:

1. Read in the initial condition  $\mathbf{U}(\mathbf{x}, 0)$  and evaluate it at all quadrature points, set  $n = 0$ .
2. Limit the solution to ensure that the method is total variation bounded (TVB) (see Section 5).

**Table 1**  
Parameters for TVD Runge–Kutta time marching schemes.

Order		$\alpha_{il}$		$\beta_{il}$	
2	1		1		
	1/2	1/2	0	1/2	
3	1		1		
	3/4	1/4	0	1/4	
	1/3	0	2/3	0	2/3

3. Compute the numerical fluxes  $\widehat{\mathbf{F}}_n$  at  $Q^I$  using a Riemann solver (see Section 4). At the domain boundaries, use the prescribed boundary conditions as the exterior state that is input to the Riemann solver. Interpolate the resulting fluxes onto  $Q^E$ . Scale the fluxes with the ratio of the edge Jacobian to the volume Jacobian. This increases the efficiency of the scheme by allowing the fluxes to be added to the volume terms evaluated at the quadrature points (see [21] for details).
4. Compute the exact fluxes  $\mathbf{F}$  at the element quadrature points. Utilizing the known gradients of the basis functions,  $\nabla \cdot \mathbf{F}$  is readily computed via a mapping.
5. Subtract  $\mathbf{F}$  from  $\widehat{\mathbf{F}}_n$  at the appropriate quadrature points and add this to the result of step 4.
6. Determine the modal coefficients of the result of step 5 by evaluating its inner product with each orthogonal basis function. Evaluate the resulting polynomials at the element quadrature points and store the result in  $\mathbf{Uf}$ .
7. Carry out the first sub-step of the RK time integration scheme:

$$\mathbf{U}^{(1)} = \mathbf{U}(\mathbf{x}, t^n) + \beta_{10} \Delta t \mathbf{Uf}(\mathbf{x}, t^n).$$

8. Repeat steps 2–6 for the remaining sub-steps in the RK scheme.
9. Increment  $n$  by one and  $t^n$  by  $\Delta t$ . Continue steps 2–9 until the desired simulation time is reached.

#### 4. MHD Riemann solvers

Riemann solvers are essential components in finite volume (FV) or RKDG methods. The function of a Riemann solver is to compute the flux across an interface that initially separates two uniform states:

$$\mathbf{U}(x, 0) = \begin{cases} \mathbf{U}_l, & x \leq c \\ \mathbf{U}_r, & x > c \end{cases}$$

where  $c \in \mathbb{R}$ . Such Riemann problems arise at the element boundaries of a computational grid at every time-step. In this context Riemann solvers may also be referred to as flux calculators. In recent years a wide variety of Riemann solvers have been developed for ideal MHD (see e.g. [24–29]). Due to the number of solvers available, selection of the optimal solver for a given simulation is difficult. This issue was addressed by Wesenberg [24], who assessed the computational time required to reach a fixed error bound for finite volume simulations of a suite of test cases using six different Riemann solvers. This was done for both first- and second-order accurate finite volume schemes. Wesenberg concluded that his MHD-HLEM was the cheapest for achieving a fixed error bound. However, since the time of Wesenberg's paper, several attractive new Riemann solvers of the HLLC family have been developed for MHD [25–29]. Our goal here is to assess the performance of a selection of these new schemes so that we may select the best for a more extensive testing in high-order RKDG methods. In particular we will compare the performance of the HLLC solvers of Li [27] (HLLC-L) and Gurski [28] (HLLC-G) with the HLLD solver of Miyoshi and Kusano [29], the Lax–Friedrichs (LF) flux, an MHD Roe solver [30], the HLLC solver [15,31] and the MHD-HLEM solver. We will use a different test case to Wesenberg that more completely exercises the capabilities of the Riemann solvers. In addition to being accurate and efficient, we must also consider the robustness of these Riemann solvers. This issue has been addressed for a number of solvers by Gurski [32]. Physical solutions of the MHD equations possess a positivity preserving property such that positive densities and pressures are always retained. However, non-physical negative densities and pressures can be generated in numerical solutions. Thus it is important that the MHD Riemann solver selected leads to a positivity preserving scheme. The HLLC and HLLC family of Riemann solvers have the additional benefit that they can be simply extended to the case where the plasma is not an ideal gas, which is not the case with the Roe and HLLC-M solvers.

In this section we will first present the results of the test cases, then conclude with the selection of the optimal Riemann solvers to use in certain situations.

##### 4.1. Numerical tests

The primary test case we have selected to evaluate the Riemann solvers listed in the previous subsection is a shock tube problem from Torrilhon [33]. This shock tube problem was selected as it includes all types of admissible MHD waves (according to Falle and Komissarov [34]) within one solution: fast- and slow-mode rarefactions, rotational discontinuities, fast shocks and slow shocks. Thus it will thoroughly exercise the wave resolving capabilities of the Riemann solvers. Torrilhon's shock tube problem is computed on the domain  $-1 < x < 1$ ,  $0 \leq t \leq 0.4$  with the following initial conditions:

$$(\rho, p, u, v, w, B_x, B_y, B_z) = \begin{cases} (3, 3, 0, 0, 0, 1.5, 1, 0), & x < 0, \\ (1, 1, 0, 0, 0, 1.5, \cos(1.5), \sin(1.5)), & x > 0. \end{cases}$$

The boundary conditions for the problem are extrapolated outflow at  $x = \pm 1$ , and periodicity in  $y$ . This test case is computed for an ideal plasma with  $\gamma = 5/3$ .

To facilitate comparison to published results (e.g. [24]), the Riemann solvers are tested within a standard finite volume method. First-order simulations use piecewise constant reconstruction in space and forward Euler time-stepping with a CFL number of 0.45. Second-order simulations use a MINMOD limited piecewise linear reconstruction in space and predictor–corrector time-stepping. For this test case we assess the performance of the schemes via the  $L^2$ -norm of the error in the cell

averages, which in this case gives a measure of the scheme's overall ability to resolve discontinuities, and the CPU-time required to simulate the problem up until the final time,  $T = 0.4$ .

Fig. 3 shows an example of the results of this test case. It shows the final density profiles produced by first-order finite volume simulations on a grid of 200 cells, along with the exact solution. Examining the density allows us to observe the performance of the schemes in the vicinity of the contact discontinuity, where the velocity and magnetic field do not vary. It does not, however, show the rotational discontinuities as these do not alter the density. In terms of resolution of discontinuities on a fixed grid, the Riemann solvers can be divided into three groups; the LF flux which is highly dissipative, the Roe, HLLC, HLLD, and HLLC-M solvers which capture all the discontinuities reasonably well even on this coarse grid, and the HLLC solver, which lies in between. In the first frame of Fig. 3, there appears to be little difference between the most accurate group of solvers. The second frame shows the solutions around the slow shock in greater detail. This allows us to see that the HLLD solution lies closest to the Roe solution, while the HLLC-M solution is even sharper (the shock is smeared over fewer cells) than Roe solution around the slow shock, due to the action of the anti-diffusion terms in the HLLC-M solver.

To assess the performance of the schemes more rigorously, we plot the  $L^2$ -norms of the error in element average density versus the computational effort (normalized CPU-time) in Fig. 4. Curves that lie lower on this plot require less computation effort to reach a fixed error bound, and are therefore said to perform better. From Fig. 4 we see that the LF flux has the worst performance, followed by the HLLC solver. The performance of the HLLC-L, HLLC-G, HLLD, HLLC-M and Roe solvers is very similar, with HLLC-L and HLLC-M being the best for this performance measure at different resolutions, followed by HLLD, Roe, then HLLC-G. These rankings appear to be reasonably independent of resolution. Note, however, that the error in density does not assess the performance of the schemes around the rotational discontinuities, where we expect the HLLD and Roe solvers to be superior. To address this problem, in Fig. 5 we present the  $L^2$ -norms of the error in cell average  $B_z$  versus the CPU-time for the first-order simulations. From this figure, we see that for this performance measure the Roe, HLLC-M and HLLD solvers are superior to the HLLC solvers, with the HLLC-M solver requiring the minimum computational effort to achieve a given error bound.

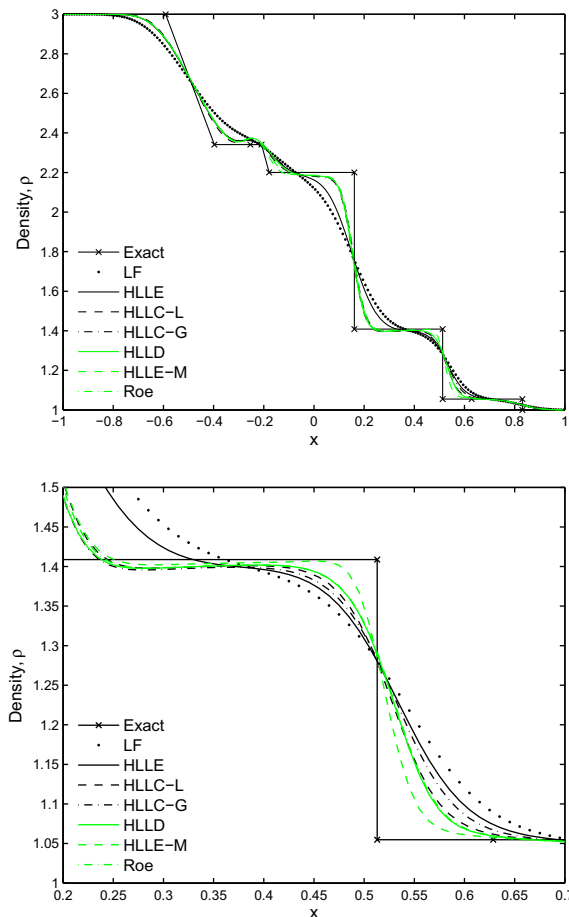


Fig. 3. Density profiles at  $t = 0.4$  from first-order solutions of Torrilhon's MHD shock tube problem on a grid of 200 cells using various Riemann solvers.

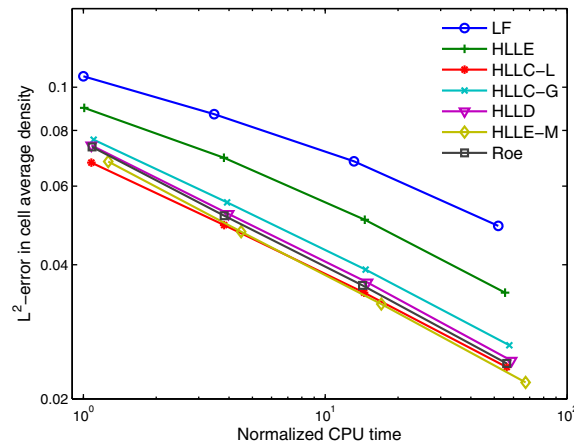


Fig. 4.  $L^2$ -error in element average densities versus normalized CPU-time for first-order simulations of Torrilhon's MHD shock tube problem on grids of 50, 100, 200, and 400 elements using various Riemann solvers.

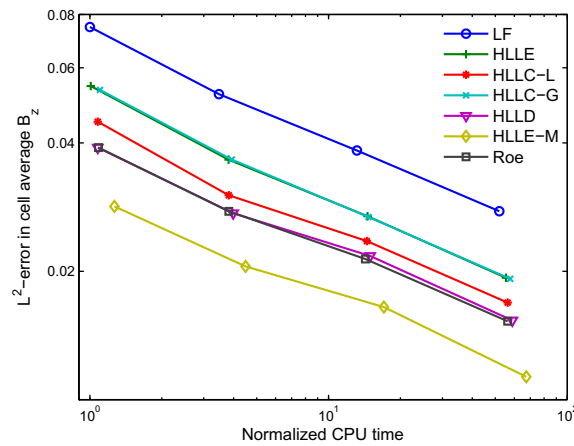


Fig. 5.  $L^2$ -error in element average  $B_z$  versus normalized CPU-time for first-order simulations of Torrilhon's MHD shock tube problem on grids of 50, 100, 200, and 400 elements using various Riemann solvers.

In Fig. 6, we plot error norms versus the computational effort for second-order accurate simulations. The ordering of the schemes in terms of computational efficiency is mostly unchanged from the first-order results. The main exception is that the performance of the Roe, HLLD, HLLC-M, and HLLC-L solvers is practically identical.

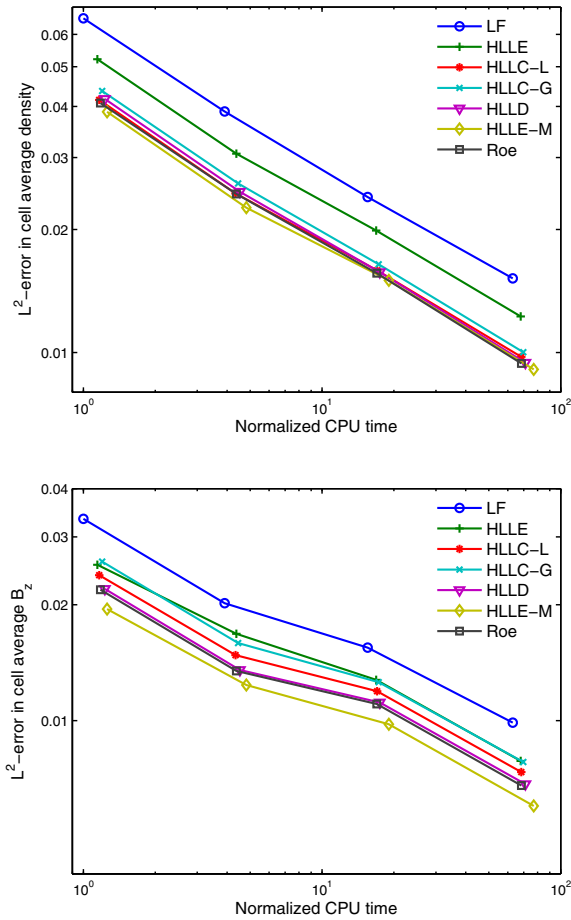
So far we have assessed the accuracy and efficiency of our set of Riemann solvers. The robustness of the Riemann solvers is another important consideration. There are no known issues with the robustness of HLLC and HLLC-type Riemann solvers, provided appropriate estimates for the signal speeds are used. There are however, multiple issues with the robustness of the Roe solver. In addition to the fact that the solver is not guaranteed to be positivity preserving [31], in multiple dimensions it has also been shown to produce what is known as the carbuncle phenomenon [35] when used to simulate shocks that are nearly stationary on the grid.

From the results presented so far, the HLLC-M solver appears to be the optimal choice of solver as it gives the lowest error for a given computational cost. The source of the low error is the anti-diffusion terms in the solver. These cause the numerical representation of the discontinuities in the solution to be more sharply resolved than when the other solvers are used. This increased sharpness comes at a cost, however, as it results in the scheme being "over-compressive" which can lead to oscillations in the vicinity of discontinuities. Such oscillations are seen in the results of second-order accurate finite volume simulations, an example of which is shown in Fig. 7.

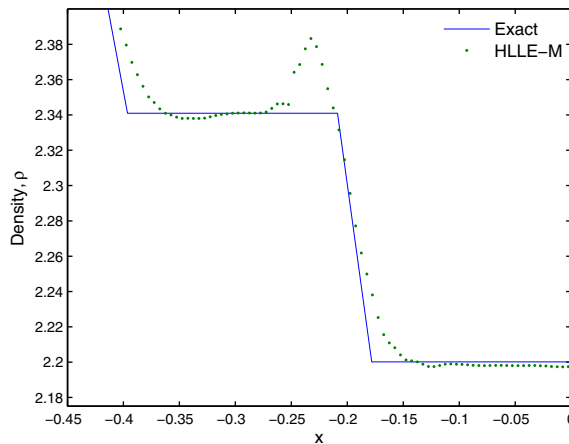
#### 4.2. Selection of solvers

From the results presented in this section, the optimal choices of Riemann solver appears to be the MHD HLLC solver of Miyoshi and Kusano [29] or the HLLC-L solver of Li [27]. These solvers exhibit high computational efficiency and have





**Fig. 6.**  $L^2$ -error in element average  $\rho$  (top) and  $B_z$  (bottom) versus normalized CPU-time for second-order simulations of Torrilhon’s MHD shock tube problem on grids of 50, 100, 200, and 400 cells using various Riemann solvers.



**Fig. 7.** Partial density profile at  $t = 0.4$  from second-order solutions of Torrilhon’s MHD shock tube problem on a grid of 400 cells using the HLLC-M Riemann solver.

similarly low numerical dissipation to the Roe solver, but do not suffer from the same robustness issues. They are not as computationally efficient as the MHD HLLC-M solver, in terms of the error in the magnetic field, but we observe that they do not generate numerical oscillations around discontinuities that we see when the HLLC-M solver is used. Additionally, both solvers have the property that they reduce to the Euler HLLC solver when the magnetic field vanishes, unlike the HLLC-G solver.

The HLLD solver requires slightly less computational effort than HLLC-L to produce the same error in the magnetic field, but the latter has the advantage that it is easier to implement. For this reason, we use the HLLC-L solver in our high-order RKDG simulations. In Section 6 we examine the impact of using this accurate Riemann solver in such simulations, compared to the LF flux.

### 5. Limiters

In order for the RKDG method to be stable in the nonlinear case, we require an entropy inequality and the uniform boundedness of the total variation of the discrete solution  $u_h$  [23]. In general, a limiter function is required for the second condition to hold. We present the limiter functions used in the RKDG method in the context of a one-dimensional nonlinear scalar equation. In order for the total variation of the means to be non-increasing for a forward Euler time-step, the discrete solution must satisfy (see [23] for details):

$$\text{sign}(\bar{u}_{j+1} - \bar{u}_j) = \text{sign}(p_{j+1/2}(u_h|E_{j+1}) - p_{j+1/2}(u_h|E_j)), \tag{13}$$

$$\text{sign}(\bar{u}_j - \bar{u}_{j-1}) = \text{sign}(u_{j+1/2}^- - u_{j-1/2}^-), \tag{14}$$

$$\text{sign}(\bar{u}_{j+1} - \bar{u}_j) = \text{sign}(u_{j+1/2}^+ - u_{j-1/2}^+), \tag{15}$$

where  $\bar{u}_j$  is the average solution on element  $E_j$ ,  $u_{j+1/2}^-$  denotes the limit of the solution at edge  $j + 1/2$  taken from the left,  $u_{j+1/2}^+$  denotes the limit of the solution at edge  $j + 1/2$  taken from the right and,

$$p_{j+1/2}(u_h|E_{j+1}) = \bar{u}_j - \frac{\Delta t}{\Delta_{j+1}} f^+(u_{j+1/2}^-) + \frac{\Delta t}{\Delta_j} f^-(u_{j-1/2}^+).$$

Here,  $\Delta t$  is the forward Euler time-step,  $\Delta_j$  is the size of  $E_j$  and we assume that the flux can be written in the form,

$$F(a, b) = f^+(a) + f^-(b),$$

which can be done for the Enquist–Osher flux, for example. There is no guarantee that the numerical solution will satisfy Eqs. (13)–(15), thus it is necessary to enforce them by means of what [23] refers to as a generalized slope limiter. These alter the numerical solution on an element in such a way that Eq. (13)–(15) are satisfied, while preserving the element average of the solution and, as far as possible, the accuracy of the method.

Following the presentation of Cockburn [23], we will first review TVDM generalized slope limiters for the one-dimensional case, then show how these were modified to obtain schemes which are total variation bounded in the means (TVBM). We will then present the extension of these limiters to the two-dimensional case.

#### 5.1. One-dimensional TVDM generalized slope limiters

A simple set for sufficient conditions on  $u_h$  that imply Eq. (13)–(15) are satisfied can be stated in terms of the minmod function  $m$ ,

$$m(a_1, \dots, a_n) = \begin{cases} s \min_{1 \leq n \leq N} |a_n|, & \text{if } s = \text{sign}(a_1) = \dots = \text{sign}(a_n), \\ 0, & \text{otherwise.} \end{cases}$$

It can be easily shown (see e.g. [23]) that Eq. (13)–(15) are satisfied if, for all  $j = 1, \dots, N$ , we have,

$$u_{j+1/2}^- - \bar{u}_j = m(u_{j+1/2}^- - \bar{u}_j, \bar{u}_j - \bar{u}_{j-1}, \bar{u}_{j+1} - \bar{u}_j), \tag{16}$$

$$\bar{u}_j - u_{j-1/2}^+ = m(\bar{u}_j - u_{j-1/2}^+, \bar{u}_j - \bar{u}_{j-1}, \bar{u}_{j+1} - \bar{u}_j). \tag{17}$$

In the following, let  $v_h$  denote the numerical solution prior to limiting, and let  $\Pi_h$  denote the limiter. Thus the limited numerical solution is given by  $u_h = \Pi_h(v_h)$ . For the case of piecewise linear approximate solutions,

$$v_h|E_j = \bar{v}_j + (x - x_j)v'_j, \quad j = 1, \dots, N,$$

we can apply slightly modified versions of the limiter functions from finite volume schemes. The generalized slope limiter based on the minmod limiter,  $\Pi_h^m$ , is,

$$u_h|E_j = \bar{v}_j + (x - x_j)m\left(v'_j, \frac{\bar{v}_{j+1} - \bar{v}_j}{\Delta_j}, \frac{\bar{v}_j - \bar{v}_{j-1}}{\Delta_j}\right). \tag{18}$$

This satisfies Eqs. (16) and (17). The modification to the standard minmod limiter is the additional input of  $v'_j$  into the minmod function. The minmod generalized slope limiter is equivalent to,

$$u_{j+1/2}^- = \bar{v}_j + m\left(v_{j+1/2}^- - \bar{v}_j, \frac{\bar{v}_j - \bar{v}_{j-1}}{2}, \frac{\bar{v}_{j+1} - \bar{v}_j}{2}\right), \tag{19}$$

$$u_{j-1/2}^+ = \bar{v}_j - m\left(\bar{v}_j - v_{j-1/2}^+, \frac{\bar{v}_j - \bar{v}_{j-1}}{2}, \frac{\bar{v}_{j+1} - \bar{v}_j}{2}\right). \tag{20}$$

Cockburn proposed the following less-restrictive (LR) slope limiter,  $\Pi_h^{lr}$ , that also satisfies Eqs. (16) and (17):

$$u_h|E_j = \bar{v}_j + (x - x_j)m\left(v_j^-, \frac{\bar{v}_{j+1} - \bar{v}_j}{\Delta_j/2}, \frac{\bar{v}_j - \bar{v}_{j-1}}{\Delta_j/2}\right). \tag{21}$$

Compared to the minmod limiter, the LR limiter permits larger values of the internal gradient on an element before limiting occurs. As limiting decreases the accuracy of the solution on an element, this implies that use of the LR limiter will produce more accurate results. The LR slope limiter is equivalent to,

$$u_{j+1/2}^- = \bar{v}_j + m\left(v_{j+1/2}^- - \bar{v}_j, \bar{v}_j - \bar{v}_{j-1}, \bar{v}_{j+1} - \bar{v}_j\right), \tag{22}$$

$$u_{j-1/2}^+ = \bar{v}_j - m\left(\bar{v}_j - v_{j-1/2}^+, \bar{v}_j - \bar{v}_{j-1}, \bar{v}_{j+1} - \bar{v}_j\right). \tag{23}$$

Both the minmod and the LR limiter are defined for piecewise linear solutions. They cannot be directly applied to solutions of higher polynomial order as  $v_h|E_j$  is in general non-constant. Instead, the limiting procedure is broken into two stages: detection and truncation. To detect if an element requires limiting the following procedure is used [23]:

1. Compute  $u_{j-1/2}^+$  and  $u_{j+1/2}^-$  using either Eqs. (19) and (20) (minmod limiter) or Eqs. (22) and (23) (LR limiter).
2. If  $u_{j-1/2}^+ = v_{j-1/2}^+$  and  $u_{j+1/2}^- = v_{j+1/2}^-$ , no limiting is required so set  $u_h|E_j = v_h|E_j$ , otherwise carry out the truncation stage.

In the truncation stage, we set  $u_h|E_j$  equal to a limited lower order polynomial that satisfies Eqs. (16) and (17). The simplest procedure is limiting to a piecewise constant by setting  $u_h|E_j = \bar{v}_j$ . We refer to this as the first-order limiter. The second-order limiter results from truncating  $u_h|E_j$  to a piecewise linear function that satisfies Eqs. (16) and (17). Following [23], this is done by applying Eq. (18) or Eq. (21) to the  $P^1$  part of the solution,

$$v_h^1 = \sum_{l=0}^1 \hat{v}_l \phi_l(x).$$

Thus  $u_h|E_j = \Pi_h^m(v_h^1)$  or  $u_h|E_j = \Pi_h^{lr}(v_h^1)$ . It is important to note that this is more accurate than, for example, direct use Eq. (21) with  $m\left(\frac{v_{j+1/2}^- - \bar{v}_j}{\Delta_j/2}, \frac{\bar{v}_j - v_{j-1/2}^+}{\Delta_j/2}\right)$  as the first argument to the minmod function, which uses quantities already computed in the detection phase. The reason for this is that if limiting is needed due to the high-order modes only, then limiting  $v_h^1$  will allow the original  $P^1$  part of the solution to be preserved, while using the differences computed in the detection step will not.

### 5.2. One-dimensional TVBM generalized slope limiters

At local extrema, the solution is non-monotone thus any limiter based on the minmod function will limit the solution to a piecewise constant no matter how smooth the solution is. This results in a loss in accuracy at local extrema. Following [36], this can be avoided by replacing the minmod function in the limiters with the TVB corrected minmod function  $\bar{m}$  which is defined as,

$$\bar{m}(a_1, \dots, a_n) = \begin{cases} a_1, & \text{if } |a_1| \leq M\Delta x^2, \\ m(a_1, \dots, a_n), & \text{otherwise,} \end{cases}$$

where  $M$  is a constant that is an upper bound for the absolute value of the curvature of the solution at local extrema. In this work, we take  $M = 50$ .

Provided that the value of  $M$  is appropriate, the scheme will not suffer from loss of accuracy at local extrema. The penalty for this is that the scheme is no longer TVD. It can, however, be shown to be total variation bounded in the means (TVBM) [36]. Moreover, provided the CFL condition is satisfied and coefficients  $\alpha_{il}$  from the Runge–Kutta scheme are nonnegative and satisfy,

$$\sum_{l=0}^{i-1} \alpha_{il} = 1,$$

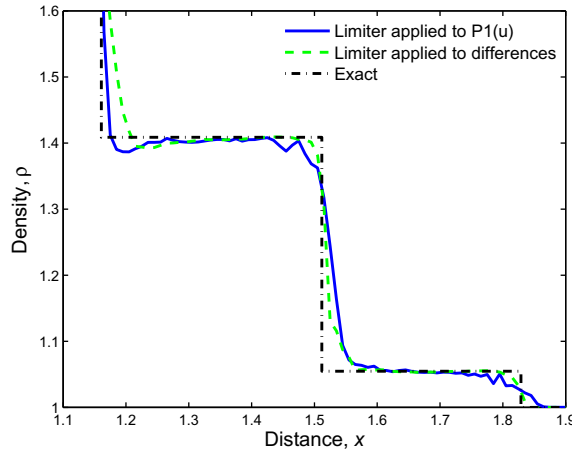
then it can be shown to solutions produced by the RKDG scheme converge in the nonlinear case [23].

### 5.3. Two-dimensional TVBM generalized slope limiters

To extend the generalized slope limiters presented in the previous subsections to the multi-dimensional case, [10] relies on the assumption that spurious oscillations are present in  $u_h$  only if they are present in its  $P^1$  part,  $u_h^1$ . Thus if,

$$u_h^1 = \Pi_h u_h^1, \tag{24}$$

indicating that limiting is not required for  $u_h^1$ , then it is assumed that that no limiting is required for the full solution  $u_h$ , hence,



**Fig. 8.** Density profiles from Torrihion's MHD shock tube problem at  $t = 0.4$  from RKDG schemes using the HLLC-L flux solver and the LR limiter. Results are compared for the following cases: the necessity of limiting is detected using the  $P_1$  projection of the solution; the necessity of limiting is detected using the difference between the element mean and edge mean values.

$$\Pi_h u_h = u_h$$

Conversely, if limiting is required for  $u_h^1$ , the solution is truncated to the limited  $P^1$  part,

$$\Pi_h u_h = \Pi_h u_h^1.$$

This approach has the drawback that it does not reduce to the one-dimensional scheme if the multi-dimensional scheme is applied to simulate a one-dimensional problem. In fact Eqs. (16) and (17) are not guaranteed to be satisfied if only the  $P^1$  part of the solution is used to detect where limiting is required. To demonstrate the implication of this (that the scheme is not TVD), in Fig. 8 we compare simulations of Torrihion's MHD shock tube problem using the proposed multi-dimensional limiter and a scheme that does reduce to the one-dimensional LR limiter described above. We clearly see the appearance of spurious oscillation when only the  $P^1$  part of the solution is used to detect where limiting is required.

The multi-dimensional limiter that we have utilized for rectangular discretizations uses Eqs. (22) and (23) in each direction to detect whether limiting is required, with the exception that the point values at edges are replaced by edge average values. This has the advantage that it reduces to the one-dimensional scheme if it is used to simulate a one-dimensional problem. In detail, we compute,

$$\bar{u}_{i+1/2}^- = \bar{v}_{ij} + \bar{m}(\bar{v}_{ij+1/2}^- - \bar{v}_{ij}, \bar{v}_{ij} - \bar{v}_{i,j-1}, \bar{v}_{i,j+1} - \bar{v}_{ij}), \tag{25}$$

$$\bar{u}_{i-1/2}^+ = \bar{v}_{ij} - \bar{m}(\bar{v}_{ij} - \bar{v}_{i,j-1/2}^+, \bar{v}_{ij} - \bar{v}_{i,j-1}, \bar{v}_{i,j+1} - \bar{v}_{ij}). \tag{26}$$

$$\bar{u}_{i+1/2,j}^- = \bar{v}_{ij} + \bar{m}(\bar{v}_{i+1/2,j}^- - \bar{v}_{ij}, \bar{v}_{ij} - \bar{v}_{i-1,j}, \bar{v}_{i+1,j} - \bar{v}_{ij}), \tag{27}$$

$$\bar{u}_{i-1/2,j}^+ = \bar{v}_{ij} - \bar{m}(\bar{v}_{ij} - \bar{v}_{i-1/2,j}^+, \bar{v}_{ij} - \bar{v}_{i-1,j}, \bar{v}_{i+1,j} - \bar{v}_{ij}), \tag{28}$$

if  $\bar{u}_{i+1/2}^- = \bar{v}_{i,j+1/2}^-$ ,  $\bar{u}_{i-1/2}^+ = \bar{v}_{i,j-1/2}^+$ ,  $\bar{u}_{i+1/2,j}^- = \bar{v}_{i+1/2,j}^-$  and  $\bar{u}_{i-1/2,j}^+ = \bar{v}_{i-1/2,j}^+$ , we assume that no limiting is required so,

$$u_h|E_{ij} = v_h|E_{ij}.$$

Otherwise  $u_h|E_{ij}$  is set to the limited  $P^1$  part of  $v_h|E_{ij}$ . This is computed by setting,

$$\bar{u}_{ij} = \bar{v}_{ij}, \tag{29}$$

$$u_x = \bar{m}(v_x^1, \bar{v}_{ij} - \bar{v}_{i,j-1}, \bar{v}_{i,j+1} - \bar{v}_{ij}), \tag{30}$$

$$u_y = \bar{m}(v_y^1, \bar{v}_{ij} - \bar{v}_{i-1,j}, \bar{v}_{i+1,j} - \bar{v}_{ij}), \tag{31}$$

for  $E_{ij}$ .

## 6. Numerical results

### 6.1. Simple wave problem

Landau and Lifshitz [37] present an exact, non-linear smooth solution to the one-dimensional Euler equations (no magnetic field). The solution only exists if the initial conditions satisfy the following relations:

$$\rho(x, 0) = \rho_0 \left( 1 + \frac{(\gamma - 1)u_x}{2a_0} \right)^{\frac{2}{\gamma-1}},$$

$$p(x, 0) = p_0 \left( 1 + \frac{(\gamma - 1)u_x}{2a_0} \right)^{\frac{2\gamma}{\gamma-1}},$$

$$u_y = u_z = B_x = B_y = B_z = 0.$$

The particular case we utilize has the initial velocity,

$$u_x(x, 0) = u_0 \sin \pi x,$$

on the domain  $-1 < x < 1$  with the boundary condition  $u(1, t) = u(-1, t)$ . The exact solution is initially smooth, but the sinusoidal waves steepen nonlinearly until a shock develops at  $t_{shock}$ , and the exact solution breaks down. The fact that the solution is smooth prior to  $t_{shock}$  implies that the numerical methods we are interested in testing should reproduce the solution to their theoretical order of accuracy, even though this is a nonlinear problem. This is not the case with any solution that involves discontinuities, such as a shock tube problem, because around discontinuities these methods revert to first-order accuracy. The existence of an exact solution to the Euler simple wave problem allows us to study the convergence of RKDG schemes for a nonlinear problem. We will examine the results at  $t = 0.9$ , somewhat prior to the formation of a shock at  $t \approx 0.95$ .

In Fig. 9, the solution from limited second-order simulations are compared to the exact solution. The HLLC-L Riemann solver was used for both simulations. During the simulation, the less-restrictive (LR) limiter did not detect that any limiting

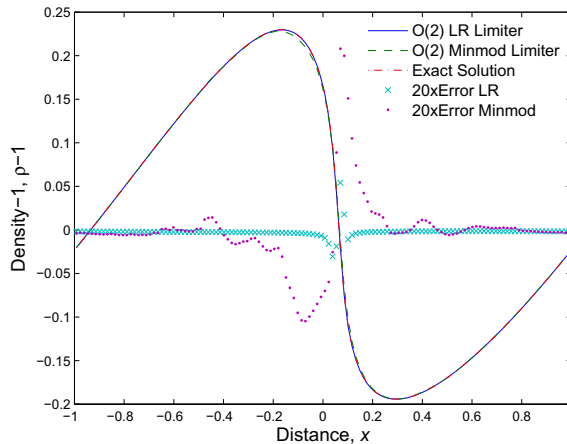


Fig. 9. Solutions from second-order simulations of the Euler simple wave problem. The HLLC-L flux solver was used in all simulations shown. Twenty times the absolute error in each numerical solution is also plotted.

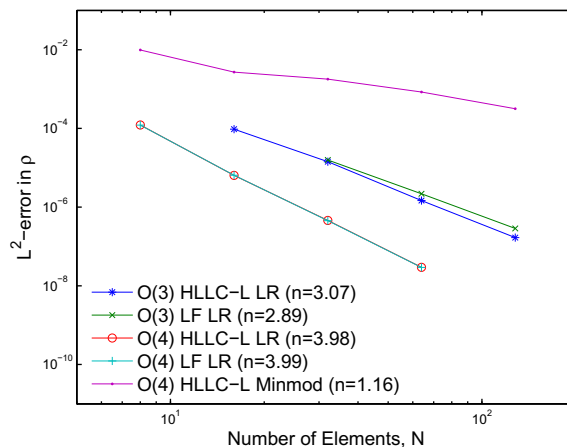
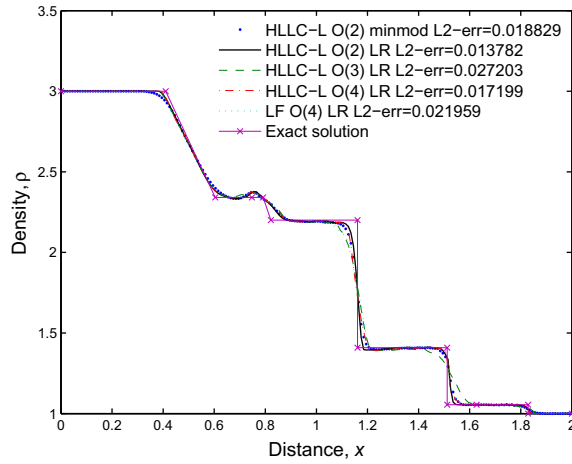
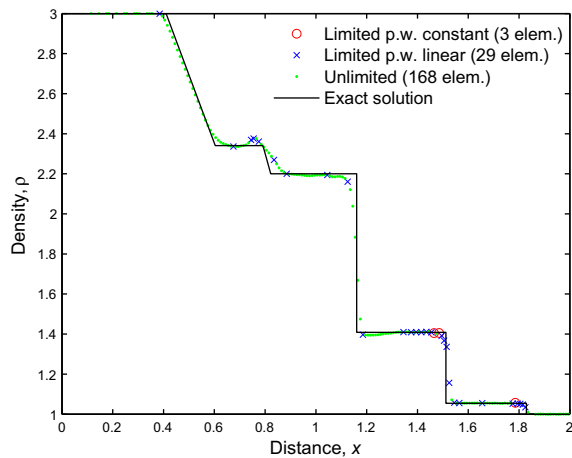
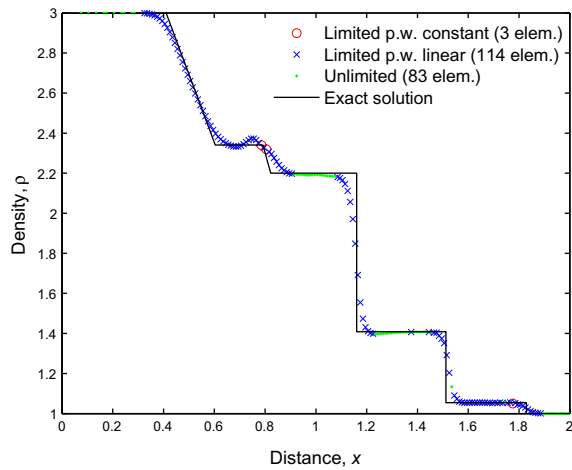


Fig. 10.  $L^2$ -errors in element average density versus  $N$  for the Euler simple wave problem at  $t = 0.5$ . The least-squares fitted order of convergence,  $n$ , for each scheme is indicated in the legend.



**Fig. 11.** Numerical solutions to Torrilhon's Riemann problem at  $t = 0.4$  using second- and third- and fourth-order RKDG schemes. The HLLC-L flux solver was used in all simulations other than one. A fourth-order result using the LF flux is shown for comparison. The  $L^2$ -error in the element average densities for each simulation is also indicated in the legend.

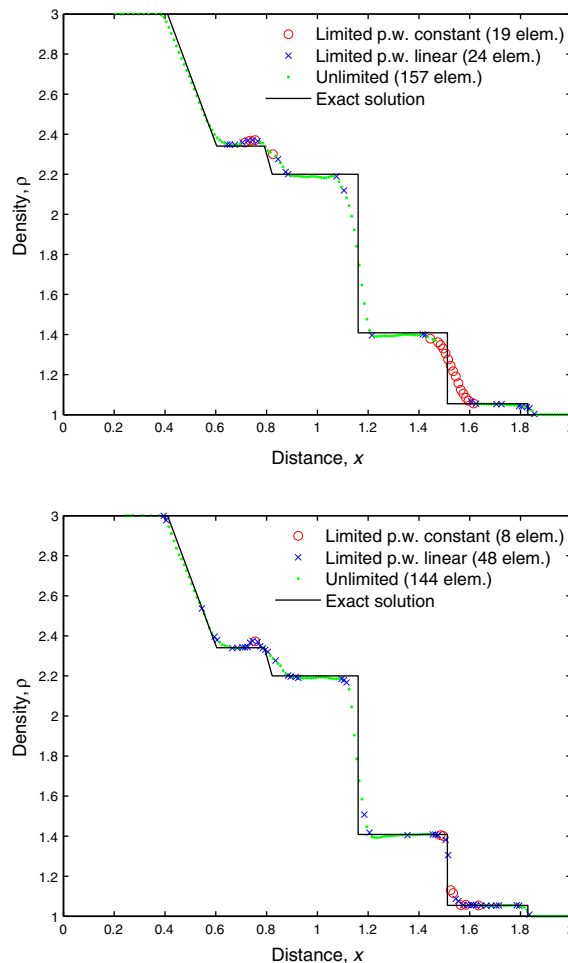


**Fig. 12.** Distribution of limited elements in solutions to Torrilhon's MHD Riemann problem at  $t = 0.4$  from second-order RKDG schemes. The HLLC-L flux solver was used along with the minmod (top) and LR (bottom) limiters.

was required, thus it is unnecessary to show a separate curve for an unlimited simulation. On the other hand the minmod limiter detected that limiting was required on the majority of elements. This indicates a major short-coming of the minmod limiter for high-order simulations: The minmod limiter selects the input gradient with the minimum modulus. If this is not the gradient of the polynomial representation of the field internal to the element, then limiting is carried out and the coefficients of the higher order modes are set to zero. In the RKDG method, the input gradients are the internal gradient, and slope estimates based on the difference between the element average fields and the average fields on the adjacent elements. In general, if the solution has a finite curvature, the internal gradient will not have the minimum modulus, thus the solution will be limited and will revert to (at most) second-order accuracy. This will occur even if limiting is unnecessary, as is the case with the solution we are currently examining. From this we must conclude that the minmod limiter is unsuited for use in a high-order RKDG method. Examining the numerical solutions shown in Fig. 9, we see that the LR limited solution approximates the exact solution very well, while the minmod limited solution slightly under-predicts the peak value.

In Fig. 10, we show how the  $L^2$ -error in element average density converges with  $N$  for various RKDG schemes. The least-squares fitted order of convergence for each scheme is indicated in the figure legend. Note that in these simulations, the LR limiter does not detect that any limiting is required, so there is no need to present separate LR limited and unlimited results. The solutions from the LR limited third- and fourth-order RKDG schemes converge at the expected rate. On the other hand, the results from the minmod limited fourth-order RKDG scheme show an order of convergence of between one and two. This is due to the solution being limited to piecewise linear or piecewise constant on the majority of elements, once again highlighting the short-comings of this limiter.

To investigate the importance of using accurate Riemann solvers in high-order schemes, in Fig. 10 we compare the convergence of solutions from LR limited third- and fourth-order RKDG schemes using LF and HLLC-L Riemann solvers. For the third-order scheme, the solutions produced using the HLLC-L solver have lower errors and converge more rapidly, as might have been anticipated from our low-order results. However, for the fourth-order scheme the errors in the LF and HLLC-L re-



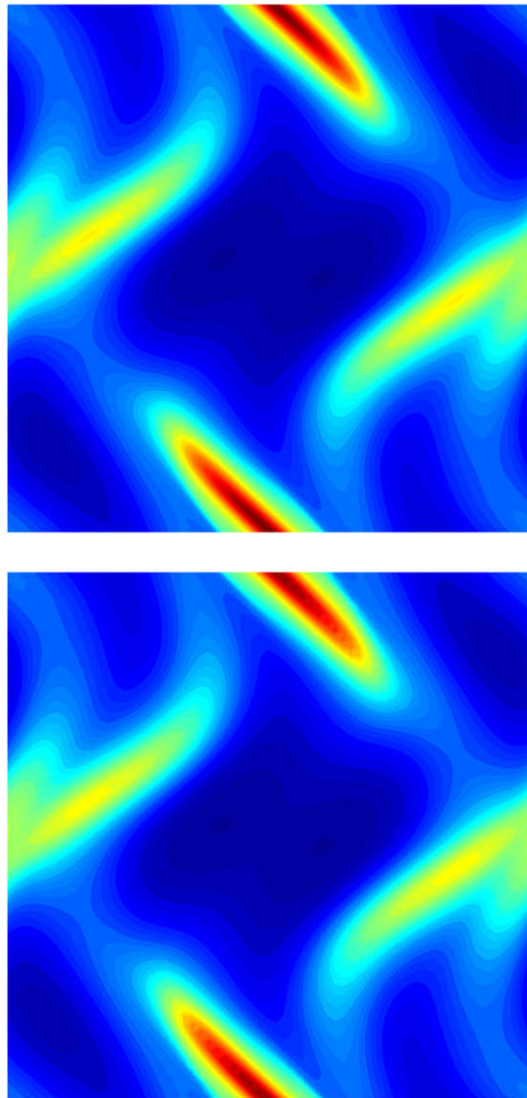
**Fig. 13.** Distribution of limited elements in solutions to Torrilhon's MHD Riemann problem at  $t = 0.4$  from third-order (top) and fourth-order (bottom) RKDG schemes. The HLLC-L flux solver was used along with the LR limiter.

sults are indistinguishable on the scale of the plot for all discretizations used. This shows that high-order accurate solutions to the simple wave problem are insensitive to the choice of Riemann solver over a broad range of resolutions, supporting the view of Cockburn and Shu [13], at least for smooth solutions.

## 6.2. Behavior of limited solutions for problems dominated by discontinuities

To investigate the effect of low-order limiting and accurate Riemann solvers in problems dominated by discontinuities, we compare numerical solutions to Torrilhon's MHD Riemann problem from second-, third- and fourth-order RKDG schemes. These are shown in Fig. 11. Let us first examine the impact of the Riemann solver on solutions from the LR limited fourth-order scheme. When the LF flux is used the  $L^2$ -error in the element average density is 0.0220. Using the more accurate HLLC-L solver, this error is reduced to 0.0172, which is significant. This is in contrast to our earlier results for smooth solutions indicates that when discontinuities are present, the choice of Riemann solver is still significant for high-order methods. The reasons for this will become clear by the end of this discussion.

We now turn our attention to the results of differing order produced using the HLLC-L solver. From Fig. 11, we observe that the solution from the third-order scheme is the least accurate overall, with an  $L^2$ -error in the element average density of 0.0272. The behavior of the second- and fourth-order LR limited schemes is similar, other than in the vicinity of the contact



**Fig. 14.** Density distribution at  $t = 0.8$  from a second-order finite volume scheme on a 256 by 256 cell regular mesh (top) and a fourth-order RKDG scheme on a 64 by 64 element regular mesh (bottom). The HLLC-L flux solver was used along with the LR limiter in the RKDG scheme. The same contour levels are used in both plots.



discontinuity where the second-order scheme is more accurate. This results in the error in the solution from the fourth-order scheme (0.0172) being greater than that from the second-order scheme (0.0138). Both of these two solutions are more accurate than that from the minmod limited second-order scheme, which has a error of 0.0188.

The reason why we do not observe an increase in accuracy as the order of the scheme is increased, and why the LR limiting procedure results in more accurate solutions, can be explained by examining where the limiter functions alter the polynomial order of the solution. For  $t = 0.4$ , this is shown in Fig. 12 for LR and minmod limited second-order RKDG schemes, and in Fig. 13 for LR limited third- and fourth-order RKDG schemes. From Fig. 12, we see that in the second-order solution, the minmod limiter carries out limiting on almost all elements where the solution is non-constant, resulting in 117 of 200 elements being limited. The LR limiter limits far fewer elements, 32 of 200 in this example, mostly those on the edge of discontinuities. This results in the minmod limited solution being more diffuse than the LR limited solution, as can be seen in Fig. 11, and therefore less accurate, as the  $L^2$ -errors revealed. From Fig. 13, we see that 43 and 56 elements are limited to lower order polynomials in the solutions from the LR limited third- and fourth-order schemes, respectively. Also, the unlimited elements are in general located where the solution is close to linear, so that a higher-order representation of the data has little impact on the overall accuracy of the solution. The cumulative result is that the error caused by additional limiting outweighs any gain in accuracy due to the increased order of the scheme, resulting in the third- and fourth-order schemes producing less accurate solutions than the LR limited second-order scheme. For this discontinuity dominated problem, these results indicate that if elements are limited to a maximum polynomial order of one, and the overall scheme is at least second-order accurate, then the  $L^2$ -error in the solution is influenced more by the extent to which the solution must be limited than by the order of accuracy of the scheme.

The remaining issue to be addressed from the results shown in Figs. 12 and 13 is why the third-order scheme produces the solution with the largest  $L^2$ -error in element average densities. The reason appears to be that the limiter function interacts poorly with the piecewise quadratic representation of the data, in that it limits the data to be piecewise constant on a far greater number of elements than for the other schemes, which locally reduces the order of the scheme to one. It is important to note that the distributions of limited elements shown in Fig. 13 are instantaneous. These distributions change significantly with time. In particular, the third-order scheme was observed to require limiting on a far greater number of elements after the first and second sub-steps of the RK scheme than after the third and final sub-step, which is what we see in Fig. 13. For example after the second sub-step at  $t = 0.4$ , the solution is limited to piecewise constants on 27 elements and piecewise linear functions on 29 elements. The fact that the scheme is reduced to first-order on this number of elements results in the solution it produces being less accurate than even the second-order minmod limited solution.

In conclusion, for high-order RKDG simulations of Torrilhon's MHD Riemann problem, the polynomial representation of the data is limited to be at most piecewise linear around the discontinuities. The remaining elements where the representation is piecewise quadratic (for third-order) or cubic (for fourth-order) lie in regions where the exact solution is linear and would thus be represented equally well by a lower-order function. This implies that, at best, a high-order RKDG scheme will behave in the same manner as a second-order scheme for the problem studied here. The only remedy for this would be to develop a limiter function that produces higher-order limited polynomials. Our results imply that for any problem featuring discontinuities, the error in a solution produced by a high-order RKDG method is likely to be dominated by the low-order errors in the vicinity of the discontinuities. Such errors have been shown to be significantly reduced by the use of accurate Riemann solvers in Section 4.1. This explains the improved performance of the fourth-order scheme when the HLLC-L solver was used instead of the LF flux.

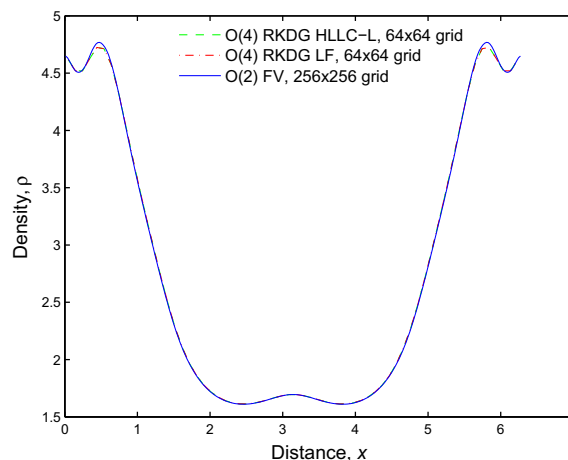


Fig. 15. Density profiles at  $t = 0.8$ ,  $y = \pi$  from a second-order finite volume scheme on a 256 by 256 cell regular mesh and a fourth-order RKDG scheme on a 64 by 64 element regular mesh. The HLLC-L flux solver was used along with the LR limiter in the RKDG scheme.

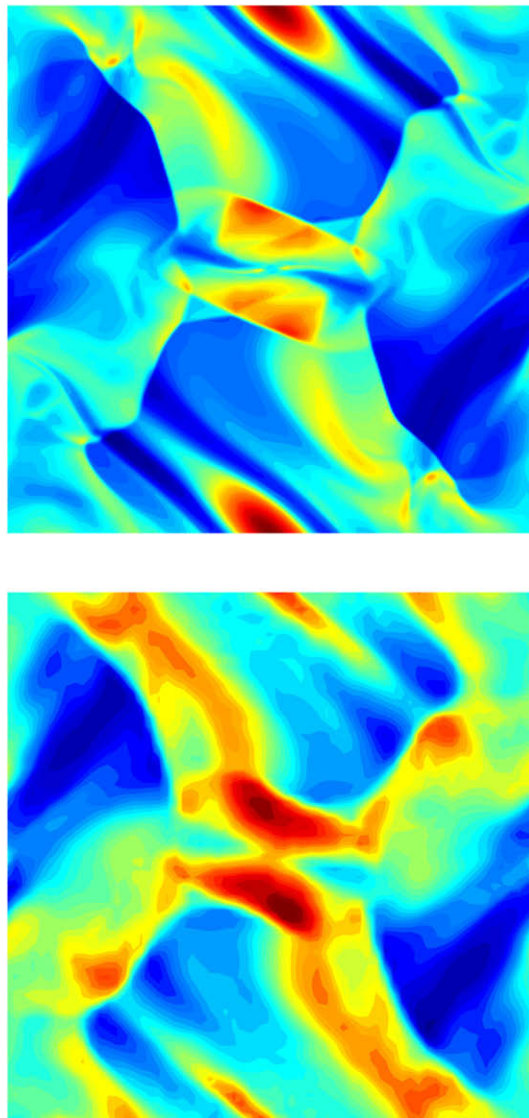
### 6.3. Orzag–Tang vortex

To verify that our findings hold in multiple dimensions, we examine simulations of the Orzag–Tang vortex [38], a commonly used multi-dimensional test case for ideal MHD solvers (see e.g. [39,29]). The problem is very cleanly defined with periodic boundary conditions in both directions. The domain for this test case is  $0 < x, y < 2\pi$ ,  $0 \leq t \leq \pi$ , and the initial conditions are,

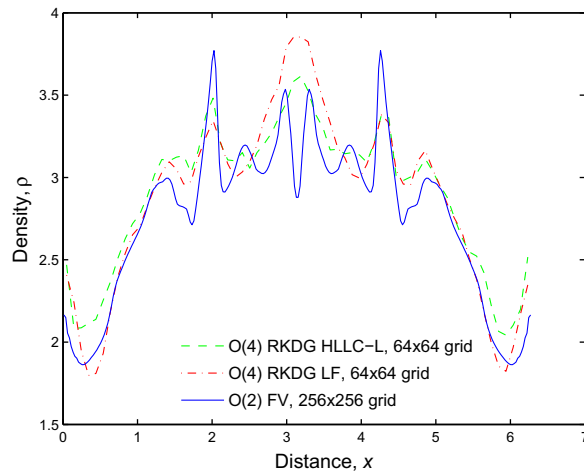
$$(\rho, p, u, v, w, B_x, B_y, B_z) = (\gamma^2, \gamma, -\sin y, \sin x, 0, -\sin y, \sin 2x, 0).$$

The problem is solved for an ideal plasma with  $\gamma = 5/3$ .

We present solutions from fourth-order RKDG simulations using the LF and HLLC-L fluxes, along with a high resolution reference solution. The RKDG simulations were run on a relatively coarse regular grid of 64 by 64 elements. The method of Powell et al. [40] is used to control divergence errors in the magnetic field. The high resolution reference solution is required because an exact solution to this test case does not exist. It was produced using a second-order accurate finite volume scheme on a 256 by 256 cell regular grid. The finite volume scheme was developed by Samtaney [41] and uses the 8-wave upwinding formulation of Powell et al. [40] within an unsplit upwinding method [42]. The solenoidal property of the mag-



**Fig. 16.** Density distribution at  $t = \pi$  from a second-order finite volume scheme on a 256 by 256 cell regular mesh (top) and a fourth-order RKDG scheme on a 64 by 64 element regular mesh (bottom). The HLLC-L flux solver was used along with the LR limiter in the RKDG scheme. The same contour levels are used in both plots.



**Fig. 17.** Density profiles at  $t = \pi$ ,  $y = \pi$  from a second-order finite volume scheme on a 256 by 256 cell regular mesh and a fourth-order RKDG scheme on a 64 by 64 element regular mesh. The HLLC-L flux solver was used along with the LR limiter in the RKDG scheme.

netic field is enforced at each time step using a projection method. A constrained transport step is then used to remove divergence modes with a centered finite difference representation. This uses the formulation prescribed by Toth [43].

During the early development of the vortex, the flow is smooth and limiting is not required. Fig. 14 shows density fields from the RKDG and reference solutions at  $t = 0.8$ , when the solution is still smooth. Only one RKDG solution is shown as the difference between them is negligible. From Fig. 14 we see that there is excellent agreement between the RKDG and reference solutions, indicating that the method performs well when limiting is not required. To show the differences between the solutions more clearly, we show the density profiles along the line  $y = \pi$  in Fig. 15. The only visible deviation of the lower resolution RKDG simulations from the reference solution is a slight under-prediction of the peak density. The RKDG results using the LF and HLLC-L fluxes cannot be distinguished on the scale of this plot. This indicates that as in one-dimension, the results or high-order RKDG simulations of smooth solutions are insensitive to the choice of Riemann solver.

Fig. 16 shows density fields at  $t = \pi$  from the RKDG simulation that used the HLLC-L flux and the reference solution. By this time, the flow has evolved into a complex state featuring numerous discontinuities. At this stage there is a need for extensive limiting, with approximately 37% and 33% of the elements being limited in the RKDG simulations using the HLLC-L and LF fluxes, respectively. Comparing the reference solution and the RKDG solution shown in Fig. 16, it can be seen that while the overall shock structure is similar in both simulations, the RKDG result is far more diffuse. In fact many of the fine scale features are not adequately resolved. To show this more clearly, we show the density profiles from all simulations along the line  $y = \pi$  in Fig. 17. From the figure we see that the RKDG solutions fail to capture the sharp peaks in density, along with the sharp trough in the center of the domain. This is due to the extensive limiting causing the scheme to revert to at most second-order accuracy in these regions around the discontinuities. As for the one-dimensional discontinuous flows studied in the previous subsection, the choice of Riemann solvers now significantly affects the RKDG solutions. From Fig. 17 it can be seen that the more accurate HLLC-L solver allows the scheme to better resolve the features in the center of the domain.

## 7. Conclusions

To investigate the importance of accurate Riemann solvers in RKDG simulations, we examined the results of first-, second-, third-, and fourth-order simulations of both smooth and non-smooth problems. In the presence of discontinuities, the influence of Riemann solver selection is directly tied to the performance of limiters. Thus we also examined the performance of common limiters in these simulations. It was noted that a particular extension of one-dimensional limiters to multiple dimensions can result in the TVD condition being violated.

We first examined the computational efficiency of several recently developed MHD Riemann solvers, in terms of the computing time required to achieve a fixed error bound for a given problem. The performance of these solvers was compared to that of the Roe solver, the LF flux and the MHD HLLC-M solver, which had previously been identified as the most efficient by Wesenberg [24]. This was done for first- and second-order schemes. We find that all of the accurate Riemann solvers are more computationally efficient than the LF flux, with the HLLC-M, HLLC-L, HLLD and Roe solvers exhibiting similar efficiency. The HLLD and HLLC-L solvers were identified as good candidates for use in high-order simulations as of the most efficient group of solvers, these have the most desirable robustness properties.

From our third- and fourth-order results for smooth solutions, we conclude that the minmod limiter is unsuited for use in high-order RKDG methods. The reason for this is that in regions of finite curvature, this limiter tends to restrict the polyno-

mial order of the trial space, reducing the scheme to at most second-order accuracy, even when limiting is not strictly required to keep the scheme TVD. The less-restrictive limiter does not exhibit this behavior. For third-order simulations of smooth problems, we find that using the HLLC-L solver results in lower errors and faster convergence than when the LF flux is used. However, for fourth-order simulations of the same problem, we observe negligible differences in the solutions when the Riemann solver is varied. This supports the view of Cockburn and Shu [13] that the results of high-order RKDG simulations are insensitive to the choice of Riemann solver, at least for smooth solutions.

From our simulations of problems dominated by discontinuities, we find that in the vicinity of discontinuities, high-order RKDG methods behave in a similar manner to the second-order method due to the use of a piecewise linear limiter. The error in numerical solutions to such problems is dominated by the error in these regions. Thus, for such solutions, the choice of Riemann solver used in a high-order method has a similar significance as for a second-order method. Our analysis of second-order methods indicates that the choice of Riemann solvers is very significant, with the more accurate Riemann solvers having the best performance. This results in fourth-order simulations of discontinuous solutions using the HLLC-L solver having considerably lower error than those using the LF flux, in contrast to the result for smooth solutions.

## Acknowledgment

The authors would like to acknowledge Professors R. Jeltsch, C. Schwab and R. Hiptmair for their support and constructive discussions on this work. Professor G. E. Karniadakis provided the authors with the original version of Nektar, which is gratefully acknowledged.

## References

- [1] D.J. Hill, D.I. Pullin, Hybrid tuned center-difference-WENO method for large eddy simulations in the presence of strong shocks, *J. Comput. Phys.* 194 (2004) 435–450.
- [2] T.R. Hill, W.H. Reed, Triangular mesh methods for neutron transport equation, Tech. Rep. LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
- [3] P. LeSaint, P.A. Raviart, On a finite element method for solving neutron transport equation, in: C. de Boor (Ed.), *Mathematical Aspects of Finite Element in Partial Differential Equations*, Academic Press, 1974, pp. 89–145.
- [4] C. Johnson, J. Pitkaranta, An analysis of discontinuous Galerkin method for scalar hyperbolic equation, *Math. Comput.* 46 (1986) 1–26.
- [5] T. Peterson, A note on the convergence of discontinuous Galerkin method for a scalar hyperbolic equation, *SIAM J. Numer. Anal.* 28 (1991) 133–140.
- [6] S. Cockburn, B. 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] S.Y. Cockburn, B. 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  $p^1$ -discontinuous Galerkin finite element method for scalar conservation laws, *Math. Model. Numer. Anal. (M<sup>2</sup>AN)* 25 (1991) 337–361.
- [10] B. Cockburn, C.W. Shu, The Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws V: the multidimensional systems case, *J. Comput. Phys.* 141 (1998) 199–224.
- [11] G. Chavent, B. Cockburn, The local projection  $p^0p^1$ -discontinuous Galerkin finite element method for scalar conservation law, *Math. Model. Numer. Anal. (M<sup>2</sup>AN)* 23 (1989) 565–592.
- [12] C.W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock capturing schemes, *J. Comput. Phys.* 77 (1988) 439–471.
- [13] B. Cockburn, C.W. Shu, Runge–Kutta discontinuous Galerkin methods for convection-dominated problems, *J. Sci. Comput.* 16 (2001) 173–261.
- [14] J.X. Qiu, B.C. Khoo, C.W. Shu, A numerical study for the performance of the Runge–Kutta discontinuous Galerkin method based on different numerical fluxes, *J. Comput. Phys.* 212 (2006) 540–565.
- [15] A. Harten, P.D. Lax, B. van Leer, On upstream differencing and Godunov-type methods for hyperbolic conservation laws, *SIAM Rev.* 25 (1983) 35.
- [16] E.F. Toro, M. Spruce, W. Speares, Restoration of the contact surface in the HartenLaxvan Leer Riemann solver, *J. Shock Waves* 4 (1994) 25–34.
- [17] V.A. Titarev, E.F. Toro, Finite-volume WENO schemes for three-dimensional convection laws, *J. Comput. Phys.* 201 (2004) 238–260.
- [18] G.W. Sutton, A. Sherman, *Engineering Magnetohydrodynamics*, McGraw-Hill, 1965.
- [19] S.J. Sherwin, G.E. Karniadakis, A new triangular and tetrahedral basis for high-order finite element methods, *Int. J. Numer. Methods Eng.* 123 (1995) 3775.
- [20] M. Dubiner, Spectral methods on triangles and other domains, *J. Sci. Comput.* 6 (1991) 345.
- [21] T.C. Warburton, G.E. Karniadakis, A discontinuous Galerkin method for the viscous MHD equations, *J. Comput. Phys.* 152 (1999) 608–641.
- [22] C.W. Shu, Total-variation-diminishing time discretizations, *SIAM J. Sci. Stat. Comput.* 9 (1988) 1073–1084.
- [23] B. Cockburn, *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, An Introduction to the Discontinuous Galerkin Method for Convection-dominated Problems*, Lecture Notes in Mathematics, Springer, 1998, pp. 151–268.
- [24] M. Wessenberg, Efficient MHD Riemann solvers for simulations on unstructured triangular grids, *J. Numer. Math.* 10 (2002) 37–71.
- [25] P. Janhunen, A positive conservative method for magnetohydrodynamics based on HLL and Roe methods, *J. Comput. Phys.* 160 (2000) 649–661.
- [26] T. Linde, A practical, general-purpose, two-state HLL Riemann solver for hyperbolic conservation laws, *Int. J. Numer. Methods Fluids* 40 (2002) 391–402.
- [27] S. Li, An HLLC Riemann solver for magneto-hydrodynamics, *J. Comput. Phys.* 203 (2005) 344.
- [28] K.F. Gurski, An HLLC-type approximate Riemann solver for ideal magnetohydrodynamics, *SIAM J. Sci. Comput.* 25 (2004) 2165.
- [29] T. Miyoshi, K. Kusano, A multi-state HLL approximate Riemann solver for ideal magnetohydrodynamics, *J. Comput. Phys.* 208 (2005) 315–344.
- [30] D. Ryu, T.W. Jones, Numerical magnetohydrodynamics in astrophysics: algorithm and tests for one-dimensional flow, *Astrophys. J.* 442 (1995) 228.
- [31] B. Einfeldt, C.D. Munz, P.L. Roe, B. Sjögreen, On Godunov-type methods near low densities, *J. Comput. Phys.* 92 (1991) 273.
- [32] K.F. Gurski, A Comparison of Two Intermediate State HLLC Solvers for Ideal Magnetohydrodynamics, in: N.V. Porelov, E. Audit, G.P. Zank (Eds.), *Numerical Modeling of Space Plasma Flows: Astronom 2007, ASP Conference Series*, vol. 385, 2008, pp. 341–348.
- [33] M. Torrilhon, Non-uniform convergence of finite volume schemes for Riemann problems of ideal magnetohydrodynamics, *J. Comput. Phys.* 192 (2003) 73–94.
- [34] S.A.E.G. Falle, S.S. Komissarov, On the inadmissibility of non-evolutionary shocks, *J. Plasma Phys.* 65 (1) (2001) 29–58.
- [35] J.J. Quirk, A contribution to the great Riemann solver debate, *Int. J. Num. Methods Fluids* 18 (1994) 555.
- [36] C.W. Shu, TVB uniformly high order schemes for conservation laws, *Math. Comput.* 49 (1987) 105–121.
- [37] L.D. Landau, E.M. Lifshitz, *Fluid Mechanics*, second ed., Butterworth-Heinemann, 1987. p. 378 (Chapter 10).

- [38] A. Orzag, C.M. Tang, Small-scale structure of two-dimensional magnetohydrodynamic turbulence, *J. Fluid Mech.* 90 (1979) 129.
- [39] G. Toth, The  $\nabla \cdot \mathbf{B}$  constraint in shock-capturing magnetohydrodynamics codes, *J. Comput. Phys.* 161 (2000) 605–652.
- [40] K.G. Powell, P.L. Roe, T.J. Linde, T.I. Gombosi, D.L. DeZeeuw, A solution-adaptive upwind scheme for ideal magnetohydrodynamics, *J. Comput. Phys.* 154 (1999) 284–309.
- [41] R. Samtaney, Suppression of the Richtmyer–Meshkov instability in the presence of a magnetic field, *Phys. Fluids* 15 (8) (2003) L53–L56.
- [42] P. Colella, Multidimensional upwind methods for hyperbolic conservation laws, *J. Comput. Phys.* 87 (1990) 171–200.
- [43] G. Toth, The  $\nabla \cdot \mathbf{B} = 0$  constraint in shock-capturing magnetohydrodynamics codes, *J. Comput. Phys.* 161 (2000) 2605–2652.