J. Math. Study
doi: 10.4208/jms.v47n3.14.02

Vol. **47**, No. 3, pp. 250-273
September 2014

Adaptive Runge-Kutta Discontinuous Galerkin Methods with Modified Ghost Fluid Method for Simulating the Compressible Two-Medium Flow

Jun Zhu[1] and Jianxian Qiu[2,*]

[1] *College of Science, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu 210016, P.R. China.*
[2] *School of Mathematical Sciences and Fujian Provincial Key Laboratory of Mathematical Modeling and High-Performance Scientific Computation, Xiamen University, Xiamen, Fujian 361005, P.R. China.*

Received 6 March 2014; Accepted 19 May 2014

Abstract. In this paper, we investigate using the adaptive Runge-Kutta discontinuous Galerkin (RKDG) methods with the modified ghost fluid method (MGFM) in conjunction with the adaptive RKDG methods for solving the level set function to simulate the compressible two-medium flow in one and two dimensions. A shock detection technique (KXRCF method) is adopted as an indicator to identify the troubled cell, which serves for further numerical limiting procedure which uses a modified TVB limiter to reconstruct different degrees of freedom and an adaptive mesh refinement procedure. If the computational mesh should be refined or coarsened, and the detail of the implementation algorithm is presented on how to modulate the hanging nodes and redefine the numerical solutions of the two-medium flow and the level set function on such adaptive mesh. Extensive numerical tests are provided to illustrate the proposed adaptive methods may possess the capability of enhancing the resolutions nearby the discontinuities inside of the single medium flow region and material interfacial vicinities of the two-medium flow region.

AMS subject classifications: 65M60, 65M99, 35L65

Chinese Library Classifications: O241

Key words: RKDG method, MGFM, two-medium flow, adaptive mesh.

1 Introduction

In general, algorithms proposed for solving the two-medium compressible flow consist of two parts: One is to treat the material interface accurately and the other is the method

*Corresponding author. *Email addresses:* zhujun@nuaa.edu.cn (J. Zhu), jxqiu@xmu.edu.cn (J. Qiu)

for simulating the single medium fluid precisely. Kapila *et al.* [13] proposed an important class of five-equation models to solve such two-medium problem. And then, Kreeft and Koren [15] provided a new formulation based on Kapila's model [13] for inviscid, non-heat-conducting and compressible two-fluid flow. Some other references of five-equation models for solving two-medium flow were found in [2,21,22,29]. A relatively crucial difficulty for simulating the compressible two-medium flow is the treatment of the moving material interfaces and their immediate vicinities. In 1999, Fedkiw *et al.* [10] provided a ghost fluid method (GFM) to treat the two-medium flow simulations. Essentially, the GFM makes the interface "invisible" during calculation by defining the ghost cell and the ghost fluid. There are subsequent variants of the original GFM with other applications [1, 14]. But it is precisely the manner of treatment of the single medium across the interface in the GFM that may cause numerical inaccuracy when there is a strong shock wave interaction with the interface. Liu *et al.* presented a modified GFM (MGFM) [18–20] to overcome this drawback of the original GFM.

On the other hand, we would like to mention the developing history of the famous RKDG method which is used for solving the single medium fluid. In 1973, Reed and Hill introduced the first discontinuous Galerkin (DG) method [27] in the framework of neutron transport. A major development of the DG method was carried out by Cockburn and Shu in a series of papers [4–8]. They employed the explicit, total variation diminishing or strong stability preserving high order Runge-Kutta time discretizations [32] and DG discretization in space with exact or approximate Riemann solvers as interface fluxes and total variation bounded (TVB) limiter [30] to achieve non-oscillatory properties for strong shocks. In [23, 24, 37], Qiu *et al.* investigated using DG methods with GFM and MGFM for two-medium flow simulations.

As we known, the solutions of the two-medium flow might have numerous local fluid structures including shock waves, contact discontinuities and rarefaction waves. So we would like to use mesh refining and coarsening procedures to assemble cells in the regions covering such fluid structures in an adaptive manner. The RKDG methods are the finite element methods and easy to deal with an adaptive strategy since the mesh refining and coarsening procedures can be applied without taking into account the continuity restrictions through cell's interface (in single medium) or material interface (in two-medium). Flaherty and his cooperators proposed a series of excellent works [3, 9, 28] on studying parallel and adaptive finite element methods for simulating conservation laws in single medium flow. Also, recently, Zhu and Qiu gave the procedures of the adaptive RKDG methods for solving hyperbolic conservation laws in [35, 36].

In this paper, following the study of [24], we would like to investigate using the adaptive RKDG methods with the MGFM for solving the two-medium flow in one and two dimensions. A shock detection technique called KXRCF method [16] is applied as a troubled cell indicator to identify the cells where the mesh should be refined or coarsened, and the detail of implementation procedures is presented on how to refine or coarse the adaptive mesh with hanging nodes and redefine numerical solutions of the two-medium flow and the level set function on such mesh. The organization of this paper is as follows:

in Section 2 and Section 3, we review and use the RKDG methods with MGFM for solving the two-medium flow, adopt the RKDG methods [12] for solving the level set function, give the algorithms of advancing the adaptive procedures and redefine solutions of the flow and level set function on the mesh in detail. Extensive numerical tests for simulating the two-medium problems are present in Section 4 to verify these approaches' viabilities. Concluding remarks are then given in Section 5.

2 RKDG method with MGFM for two-medium flow simulations in 1D

2.1 RKDG method in 1D

In this section, we consider the one dimensional hyperbolic conservation laws:

$$\frac{\partial}{\partial t}U+\frac{\partial}{\partial x}f(U)\equiv\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho\mu\\ E\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho\mu\\ \rho\mu^2+p\\ \mu(E+p)\end{pmatrix}=0. \tag{2.1}$$

Here, ρ is the density, μ is the x-direction velocity, p is the pressure and E (total energy) $\equiv\rho e+\frac{1}{2}\rho\mu^2$, where e is the specific internal energy per unit mass. For closure of the system, the equation of state (EOS) is required. In this paper, the fluid is assumed to be inviscid, non-heat-conducting and compressible. The γ-law used for gases is given as:

$$\rho e=\frac{p}{\gamma-1}, \tag{2.2}$$

and Tait's EOS used for the water medium [10, 18] is expressed as:

$$\rho e=\frac{p+N\bar{B}}{N-1}, \tag{2.3}$$

where $\bar{B}=B-A$, $N=7.15$, $A=10^5Pa$, $B=3.31\times10^8Pa$ and $\rho_0=1000kg/m^3$.

For simplicity, the original mesh is distributed into several cells $I_i=[x_i-\frac{\Delta x_i}{2},x_i+\frac{\Delta x_i}{2}]$ with the cell sizes $|I_i|=\Delta x_i$, cell centers x_i, $i=1,\cdots,N$. The DG solution as well as the test function space is given by $V_h^k=\{p:p|_{I_i}\in P^k(I_i)\}$ and is the polynomial space of degree at most k on I_i. We adopt a local orthogonal basis over I_i, $\{v_l^{(i)}(x),l=0,1,\cdots,k\}$, such as:

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

Then the numerical solution $U^h(x,t)$ can be written as $U^h(x,t)=\sum_{l=0}^k U_i^{(l)}(t)v_l^{(i)}(x)$, for $x\in I_i$ and the degrees of freedom $U_i^{(l)}(t)$ are the moments defined by $U_i^{(l)}(t)=\frac{1}{\int_{I_i}(v_l^{(i)}(x))^2dx}$

$\int_{I_i} U^h(x,t)v_l^{(i)}(x)dx, l=0,\cdots,k$. In order to determine the approximate solution, we evolve the degrees of freedom $U_i^{(l)}(t)$:

$$\frac{d}{dt}U_i^{(l)}(t)=-\frac{1}{\int_{I_i}(v_l^{(i)}(x))^2dx}\Bigg(-\int_{I_i}f(U^h(x,t))\frac{d}{dx}v_l^{(i)}(x)dx+\hat{f}(U_{i+1/2}^{x,-},U_{i+1/2}^{x,+})v_l^{(i)}\Big(x_i+\frac{\Delta x_i}{2}\Big)$$
$$-\hat{f}(U_{i-1/2}^{x,-},U_{i-1/2}^{x,+})v_l^{(i)}\Big(x_i-\frac{\Delta x_i}{2}\Big)\Bigg),\quad l=0,\cdots,k, \tag{2.4}$$

where $U_{i+1/2}^{x,\pm}=U^h(x_i+\frac{\Delta x_i}{2},t)^{x,\pm}$ are the left and right limits of the discontinuous solution $U^h(x,t)$ at the cell interface $x_i+\frac{\Delta x_i}{2}$, and $\hat{f}(U^-,U^+)$ is a monotone flux for the scalar case and an exact or approximate Riemann solver for the system. In this paper, we use a Gauss-Lobatto quadrature formula for solving the element integral in (2.4).

Now we would like to give the way of deciding $U_{i+1/2}^{x,\pm}$ precisely. We first use the KXRCF method [16] as an indicator to detect the "troubled cell". We divide the boundary of the target cell I_i into two parts: ∂I_i^- and ∂I_i^+, where the flow is into ($\mu\cdot n<0$) and out of ($\mu\cdot n>0$) I_i, respectively. Then the target cell I_i is identified as the "troubled cell" on condition that:

$$\frac{|\int_{\partial I_i^-}(U^h(x,t)|_{I_i}-U^h(x,t)|_{I_{i_\ell}})ds|}{h^{\frac{k+1}{2}}|\partial I_i^-|\cdot||U^h(x,t)|_{I_i}||}>1, \tag{2.5}$$

where h is the radius of the circumscribed circle in I_i, I_{i_ℓ} is neighbor of I_i on side of ∂I_i^-.

If there are strong shocks or contact discontinuities in the troubled cell, such methods might generate spurious oscillations in this cell, and a limiter is needed to control spurious oscillations. For the purpose of constructing the slope-limiting operator on one dimensional adaptive mesh to deal with such difficulties and control the numerical oscillations, we present the procedure which is similar to [6, 8, 25, 30] and try to specify it in detail as follows. For the sake of clarity, we denote $U_i^{(*)}=U_i^{(*)}(t)$. Consider the adaptive mesh depicted in Fig. 1, we reconstruct the freedoms of the solution on such detected troubled cell:

$$U_{i+1/2}^{x,-}=U_i^{(0)}+\tilde{U}_i=U_i^{(0)}+\sum_{l=1}^{k}U_i^{(l)}v_l^{(i)}\Big(x_i+\frac{\Delta x_i}{2}\Big), \tag{2.6}$$

$$U_{i-1/2}^{x,+}=U_i^{(0)}-\tilde{\tilde{U}}_i=U_i^{(0)}-\Bigg(-\sum_{l=1}^{k}U_i^{(l)}v_l^{(i)}\Big(x_i-\frac{\Delta x_i}{2}\Big)\Bigg). \tag{2.7}$$

These are modified by the standard minmod limiter:

$$\tilde{U}_i^{(mod)}=m\Bigg(\tilde{U}_i,\alpha_1\frac{x_i+\frac{\Delta x_i}{2}-x_i}{x_{i_1}-x_i}(U_{i_1}^{(0)}-U_i^{(0)}),\ \alpha_2\frac{x_i+\frac{\Delta x_i}{2}-x_i}{x_i-x_{i_2}}(U_i^{(0)}-U_{i_2}^{(0)})\Bigg),$$
$$\tilde{\tilde{U}}_i^{(mod)}=m\Bigg(\tilde{\tilde{U}}_i,\alpha_1\frac{x_i-(x_i-\frac{\Delta x_i}{2})}{x_{i_1}-x_i}(U_{i_1}^{(0)}-U_i^{(0)}),\ \alpha_2\frac{x_i-(x_i-\frac{\Delta x_i}{2})}{x_i-x_{i_2}}(U_i^{(0)}-U_{i_2}^{(0)})\Bigg),$$

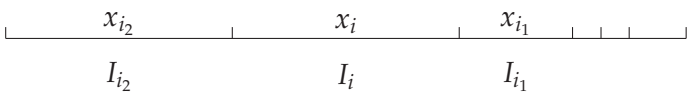

Figure 1: The troubled cell I_i and its two neighboring cells in 1D.

where $\alpha_1=2$, $\alpha_2=2$ and m is given by

$$m(a_1,\cdots,a_n)=\begin{cases} s\cdot\min_{1\le j\le n}|a_j|, & \text{if } sign(a_1)=\cdots=sign(a_n)=s,\\ 0, & \text{otherwise,}\end{cases} \tag{2.8}$$

or the TVB modified minmod function:

$$\tilde{m}(a_1,\cdots,a_n)=\begin{cases} a_1, & \text{if } |a_1|\le M\Delta x_i^2,\\ m(a_1,\cdots,a_n), & \text{otherwise,}\end{cases} \tag{2.9}$$

where $M>0$ is a constant, which is depended on the problem.

Then we would like to deal with the problem of choosing the proper numerical monotone flux $\hat{f}(U^-,U^+)$ specified above. The HLLC flux (a modification of the HLL flux) [33,34] is used here. The HLLC flux is based on the approximate Riemann solver with only three constant states separated by two waves while resolving the contact and shear waves.

Finally the overall solutions for the computational domain are obtained by computing different temporary time step for each medium flow and then choosing the smaller one as the overall $\triangle t=\min_{i=1}^{N}\{\frac{CFL\cdot\Delta x_i}{|\mu_i|+c_i}\}$. And at each inner time step, the ordinary differential equation (ODE) (2.4) $U_t=L(U)$ is solved by using a Runge-Kutta time discretization [31,32], such as a total variation diminishing (TVD) Runge-Kutta method:

$$\begin{cases} U^{(1)}=U^n+\triangle tL(U^n),\\ U^{(2)}=\frac{3}{4}U^n+\frac{1}{4}U^{(1)}+\frac{1}{4}\triangle tL(U^{(1)}),\\ U^{n+1}=\frac{1}{3}U^n+\frac{2}{3}U^{(2)}+\frac{2}{3}\triangle tL(U^{(2)}).\end{cases} \tag{2.10}$$

2.2 RKDG method for level set advection equation and MGFM in 1D

The level set advection equation in one dimension is

$$\phi_t+\mu\phi_x=0. \tag{2.11}$$

If we set $\eta=\phi_x$ and $H(\eta)=\mu\eta$, (2.11) is equivalent to the conservation law:

$$\eta_t+H(\eta)_x=0. \tag{2.12}$$

For simplicity, the mesh is specified as before. The DG solution [12] as well as the test function space is given by $V_h^{k-1}=\{p:p|_{I_i}\in P^{k-1}(I_i)\}$ and the polynomial space of degree at most $k-1$ on the cell I_i. We adopt the same local orthogonal basis over I_i, $\{v_l^{(i)}(x),l=0,1,\cdots,k-1\}$ as before. Then the numerical solution $\eta^h(x,t)$ in the space V_h^{k-1} can be written as $\eta^h(x,t)=\sum_{l=0}^{k-1}\eta_i^{(l)}(t)v_l^{(i)}(x)$, for $x\in I_i$ and the degrees of freedom $\eta_i^{(l)}(t)$ are the moments defined by

$$\eta_i^{(l)}(t)=\frac{1}{\int_{I_i}(v_l^{(i)}(x))^2dx}\int_{I_i}\eta^h(x,t)v_l^{(i)}(x)dx,\quad l=0,\cdots,k-1.$$

In order to determine the approximate solution, we evolve the degrees of freedom $\eta_i^{(l)}(t)$:

$$\frac{d}{dt}\eta_i^{(l)}(t)=-\frac{1}{\int_{I_i}(v_l^{(i)}(x))^2dx}\left(-\int_{I_i}H(\eta^h(x,t))\frac{d}{dx}v_l^{(i)}(x)dx+\hat{H}(\eta_{i+1/2}^-,\eta_{i+1/2}^+)v_l^{(i)}\left(x_i+\frac{\Delta x_i}{2}\right)\right.$$
$$\left.-\hat{H}(\eta_{i-1/2}^-,\eta_{i-1/2}^+)v_l^{(i)}\left(x_i-\frac{\Delta x_i}{2}\right)\right),\quad l=0,\cdots,k-1, \tag{2.13}$$

where $\eta_{i+1/2}^{\pm}=\eta^h(x_i+\frac{\Delta x_i}{2},t)^{\pm}$ are the left and right limits of the discontinuous solution $\eta^h(x,t)$ at the cell interface $x_i+\frac{\Delta x_i}{2}$. In this paper, we use the Lax-Friedrichs flux $\hat{H}(\eta^-,\eta^+)=H(\frac{\eta^-+\eta^+}{2})-\frac{1}{2}\alpha(\eta^+-\eta^-)$, where $\alpha=\max_\eta|H'(\eta)|$ with the maximum taken over the interval of $[\min(\eta^-,\eta^+),\max(\eta^-,\eta^+)]$. The advection equation for the level set function is updated together with the two-medium flow and $\triangle t$ is same as specified in (2.10). For solving the ODE $\eta_t=L(\eta)$, we also use the third order TVD Runge-Kutta method [31] in (2.10). The DG method for (2.12) is satisfied by the derivative $\eta=\phi_x$ and this determines ϕ for the target cell I_i up to a constant. One way that specified in [12] to get the missing constant is used here by requiring that:

$$\int_{I_i}(\phi_t+H(\phi_x))dx=0. \tag{2.14}$$

Then the numerical solution $\phi^h(x,t)$ in the space V_h^k can be obtained in this way. The limiting procedure is not used to solve for such level set advection equation.

In the MGFM-based algorithm, to define the ghost fluid states, a two-medium Riemann problem is defined and exactly or approximately solved to predict the interface status (Readers can refer to [17, 20] for detailed implementation). We would like to use the interfacial treatment via the technique of the modified ghost fluid method [18–20] coupled to the RKDG method for the single medium flow and introduce such method briefly as below:

The two nonlinear characteristics intersecting at the interface for (2.1) are given as:

$$\begin{cases}\frac{dp_I}{dt}+\rho_{IL}c_{IL}\frac{d\mu_I}{dt}=0,\ \text{along}\ \frac{dx}{dt}=\mu_I+c_{IL},\\ \frac{dp_I}{dt}-\rho_{IR}c_{IR}\frac{d\mu_I}{dt}=0,\ \text{along}\ \frac{dx}{dt}=\mu_I-c_{IR},\end{cases} \tag{2.15}$$

where ρ_{IL}, ρ_{IR}, c_{IL} and c_{IR} are the density and speed of sound to the two sides of the material interface, μ_I and p_I are the x-direction velocity and pressure at the interface. Then by using a discretization procedure, we get

$$\begin{cases} \frac{p_I-p_{IL}}{\hat{\rho}_{IL}\hat{c}_{IL}}+(\mu_I-\mu_{IL})=0, \\ \frac{p_I-p_{IR}}{\hat{\rho}_{IR}\hat{c}_{IR}}-(\mu_I-\mu_{IR})=0, \end{cases} \tag{2.16}$$

where $\hat{\rho}_{IL}\hat{c}_{IL}$ and $\hat{\rho}_{IR}\hat{c}_{IR}$ are the approximations of $\rho_{IL}c_{IL}$ and $\rho_{IR}c_{IR}$, respectively. And μ_{IL}, μ_{IR}, p_{IL} and p_{IR} can be evaluated along the different characteristic lines. Then we use (2.16) to solve a shock impacting on the interface. Since a reflected shock is not correctly treated by the algorithm based on the original GFM. In such a situation, we have

$$\begin{cases} \frac{p_I-p_L}{\rho_I^L\rho_L(p_I-p_L)/(\rho_I^L-\rho_L)}+(\mu_I-\mu_L)=0, \\ \frac{p_I-p_R}{\rho_I^R\rho_R(p_I-p_R)/(\rho_I^R-\rho_R)}-(\mu_I-\mu_R)=0, \end{cases} \tag{2.17}$$

where subscript L and R represent the status behind the incident shock and the status ahead of the transmitted shock, respectively. So we can use (2.17) to predict the interface condition. Specifically, for an interface located between cell i and $i+1$ at a time step t^n, we can solve equations via iteration and predict the interface condition for evaluating at the next time step t^{n+1}:

$$\begin{cases} \frac{p_I-p_{i-1}^n}{\rho_I^L\rho_{i-1}^n(p_I-p_{i-1}^n)/(\rho_I^L-\rho_{i-1}^n)}+(\mu_I-\mu_{i-1}^n)=0, \\ \frac{p_I-p_{i+2}^n}{\rho_I^R\rho_{i+2}^n(p_I-p_{i+2}^n)/(\rho_I^R-\rho_{i+2}^n)}-(\mu_I-\mu_{i+2}^n)=0. \end{cases} \tag{2.18}$$

It provides the exact solution when a shock wave is reflected and can work very well for two-medium flow simulation.

In such a MGFM-based algorithm, the level set technique is usually employed to capture the moving interface. A band of several grid cells as ghost cells are defined in the vicinity of the interface. At the ghost cells, ghost fluid and the real fluid co-exist. Once the ghost fluid cells and ghost fluid are defined for each medium, we can employ the adaptive RKDG method to solve for the single medium containing both the real fluid and ghost fluid cells.

2.3 The algorithm of adaptive method in 1D

Step 1.1. Partition the computing field into uniform cells as $I_i=[x_i-\frac{\Delta x_i}{2},x_i+\frac{\Delta x_i}{2}]$, $i=1,\cdots,N$. Use the initial conditions to get the degrees of freedom $U_i^{(l)}(t_0)$ and $\phi_i^{(l)}(t_0)$, $l=0,\cdots,k$,$i=1,\cdots,N$. And set the adaptive mesh level $R_i(t_0)=0$,$i=1,\cdots,N$ at first.

Step 1.2. Suppose at time t_n, the degrees of freedom $U_i^{(l)}(t_n)$, $\phi_i^{(l)}(t_n)$, $l=0,\cdots,k$,$i=1,\cdots,N$ and the adaptive mesh level $R_i(t_n)$,$i=1,\cdots,N$ are given. We apply KXRCF method as an indicator to detect "troubled cells".

Step 1.2.1. Suppose I_{i0} is a "troubled cell": if $R_{i0}(t_n) < R$, divide it into two equal size small children cells, increase such children cells' mesh level by 1, and redefine the degrees of freedom of $U^h(x,t_n)$ and $\phi^h(x,t_n)$ on these cells (specified in the following Subsection 2.5); if $R_{i0}(t_n) = R$, do nothing.

Step 1.2.2. If all children cells are not identified as "troubled cells", merge them into their parent cell and decrease such parent cell's mesh level by 1, and redefine the degrees of freedom of $U^h(x,t_n)$ and $\phi^h(x,t_n)$ on this cell (specified in the following Subsection 2.5).

By doing so, we obtain a new adaptive mesh with different mesh level $R_i(t_{n+1})$, $i = 1,\cdots,N$.

Step 1.3. Redefine the new degrees of freedom of $U^h(x,t_n)$ and $\phi^h(x,t_n)$ over the whole computation region on the new adaptive mesh.

Step 1.4. Evolve the solutions of the two-medium flow and level set function from t_n to t_{n+1} by the methodologies specified in Subsection 2.1, Subsection 2.2 and Subsection 2.3 to obtain the degrees of freedom $U_i^{(l)}(t_{n+1})$ and $\phi_i^{(l)}(t_{n+1})$, $l = 0,\cdots,k, i = 1,\cdots,N$.

Step 1.4.1. Compute the time step Δt.

Step 1.4.2. Obtain the new location of the material interface via computing the level set function by using the RKDG methods [12].

Step 1.4.3. Compute for different single medium flow by using RKDG methods coupled with MGFM which offers the technique of interface treatment.

Step 1.4.4. Obtain the new time level's solutions according to the new location of the interface.

Step 1.5. If t_{n+1} is less than the final time, go to Step 1.2..

2.4 Redefine $U^h(x,t)$ and $\phi^h(x,t)$ on the adaptive mesh

If the flagged cell A ("troubled cell") is restricted into two smaller cells B_1 and B_2, which are depicted in Fig. 2, the numerical conservative variable of $U^h(x,t)$ and the numerical level set function $\phi^h(x,t)$ denoted on cell A should be remapped to the new cells of B_1 and B_2. The methodology can be expressed as:

$$(U_{B_j}^{(0)}(t),\cdots,U_{B_j}^{(k)}(t))^T = M^{-1}(b_0,\cdots,b_k)^T, \quad M = (m_{sq})_{(k+1)\times(k+1)},$$
$$m_{sq} = \int_{I_{B_j}} v_s^{(A)}(x) v_q^{(B_j)}(x)dx, \quad b_s = \int_{I_{B_j}} U_A^h(x,t) v_s^{(B_j)}(x)dx,$$
$$(\phi_{B_j}^{(0)}(t),\cdots,\phi_{B_j}^{(k)}(t))^T = M^{-1}(b_0,\cdots,b_k)^T, \quad b_s = \int_{I_{B_j}} \phi_A^h(x,t) v_s^{(B_j)}(x)dx, \ s,q = 0,\cdots,k, \ j = 1,2.$$

If the children cells B_1 and B_2 are not detected as "troubled cells" any more, and should be merged into the parent cell A, the numerical conservative variable of $U^h(x,t)$ and the numerical level set function $\phi^h(x,t)$ denoted on A should be remapped from

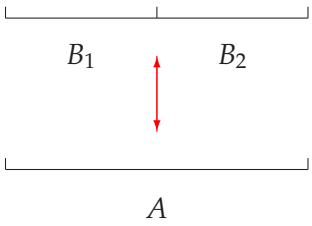

Figure 2: The topological map of parent and children cells in 1D.

those variables specified on cells B_1 and B_2. And the methodology adopted here can be expressed as below:

$$(U_A^{(0)}(t),\cdots,U_A^{(k)}(t))^T=M^{-1}(b_0,\cdots,b_k)^T,\quad M=(m_{sq})_{(k+1)\times(k+1)},$$
$$m_{sq}=\int_{I_A}v_s^{(A)}(x)v_q^{(A)}(x)dx,\quad b_s=\sum_{j=1}^{2}\int_{I_{B_j}}U_{B_j}^h(x,t)v_s^{(A)}(x)dx,$$
$$(\phi_A^{(0)}(t),\cdots,\phi_A^{(k)}(t))^T=M^{-1}(b_0,\cdots,b_k)^T,\quad b_s=\sum_{j=1}^{2}\int_{I_{B_j}}\phi_{B_j}^h(x,t)v_s^{(A)}(x)dx,\quad s,q=0,\cdots,k.$$

3 RKDG method with MGFM for two-medium flow simulations in 2D

3.1 RKDG method in 2D

In this section, we consider the two dimensional hyperbolic conservation laws:

$$\frac{\partial}{\partial t}U+\frac{\partial}{\partial x}f(U)+\frac{\partial}{\partial y}g(U)\equiv\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho\mu\\ \rho\nu\\ E\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho\mu\\ \rho\mu^2+p\\ \rho\mu\nu\\ \mu(E+p)\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho\nu\\ \rho\mu\nu\\ \rho\nu^2+p\\ \nu(E+p)\end{pmatrix}=0.\quad(3.1)$$

Here, ρ is the density, μ is the x-direction velocity, ν is the y-direction velocity, p is the pressure and E (total energy)$\equiv\rho e+\frac{1}{2}\rho(\mu^2+\nu^2)$, where e is the specific internal energy per unit mass.

It is assumed that the mesh is distributed into several cells $I_i=[x_i-\frac{\Delta x_i}{2},x_i+\frac{\Delta x_i}{2}]\times[y_i-\frac{\Delta y_i}{2},y_i+\frac{\Delta y_i}{2}]$, with cell sizes $|I_i|=\Delta x_i\Delta y_i$ and cell centers (x_i,y_i), $i=1,\cdots,N$. We now give the new test function space $V_h^k=\{p:p|_{I_i}\in P^k(I_i)\}$ as the polynomial space of degree at most k on the cell I_i. We adopt a local orthogonal basis over I_i, $\{v_l^{(i)}(x,y),\ l=0,1,\cdots,K;\ K=\frac{1}{2}(k+1)(k+2)-1\}$, such as: $v_0^{(i)}(x,y)=1$, $v_1^{(i)}(x,y)=\frac{x-x_i}{\Delta x_i}$, $v_2^{(i)}(x,y)=\frac{y-y_i}{\Delta y_i}$, $v_3^{(i)}(x,y)=(\frac{x-x_i}{\Delta x_i})^2-\frac{1}{12}$, $v_4^{(i)}(x,y)=\frac{(x-x_i)(y-y_i)}{\Delta x_i\Delta y_i}$, $v_5^{(i)}(x,y)=(\frac{y-y_i}{\Delta y_i})^2-\frac{1}{12},\cdots$. Then the numerical solution $U^h(x,y,t)$

in the space V_h^k can be written as $U^h(x,y,t)=\sum_{l=0}^K U_i^{(l)}(t)v_l^{(i)}(x,y)$, for $(x,y)\in I_i$ and the degrees of freedom $U_i^{(l)}(t)$ are the moments defined by

$$U_i^{(l)}(t)=\frac{1}{\int_{I_i}(v_l^{(i)}(x,y))^2dxdy}\int_{I_i}U^h(x,y,t)v_l^{(i)}(x,y)dxdy,\quad l=0,\cdots,K.$$

In order to determine the approximate solution, we evolve the degrees of freedom $U_i^{(l)}(t)$:

$$\begin{aligned}\frac{d}{dt}U_i^{(l)}(t)=&-\frac{1}{\int_{I_i}(v_l^{(i)}(x,y))^2dxdy}\Bigg\{-\int_{I_i}f(U^h(x,y,t))\frac{\partial}{\partial x}v_l^{(i)}(x,y)dxdy\\
&-\int_{I_i}g(U^h(x,y,t))\frac{\partial}{\partial y}v_l^{(i)}(x,y)dxdy+\int_{y_i-\frac{\Delta y_i}{2}}^{y_i+\frac{\Delta y_i}{2}}\Bigg[f(U^h(x_i+\frac{\Delta x_i}{2},y,t))v_l^{(i)}\Big(x_i+\frac{\Delta x_i}{2},y\Big)\\
&-f\Big(U^h\Big(x_i-\frac{\Delta x_i}{2},y,t\Big)\Big)v_l^{(i)}\Big(x_i-\frac{\Delta x_i}{2},y\Big)\Bigg]dy\\
&+\int_{x_i-\frac{\Delta x_i}{2}}^{x_i+\frac{\Delta x_i}{2}}\Bigg[g\Big(U^h\Big(x,y_i+\frac{\Delta y_i}{2},t\Big)\Big)v_l^{(i)}\Big(x,y_i+\frac{\Delta y_i}{2}\Big)\\
&-g\Big(U^h\Big(x,y_i-\frac{\Delta y_i}{2},t\Big)\Big)v_l^{(i)}\Big(x,y_i-\frac{\Delta y_i}{2}\Big)\Bigg]dx\Bigg\},\quad l=0,\cdots,K.\end{aligned}\tag{3.2}$$

In (3.2) the second and third integral terms can be computed either exactly or by a suitable numerical quadrature accurate to at least $\min\{\mathcal{O}(\Delta x_i^{2k+2}),\ \mathcal{O}(\Delta y_i^{2k+2})\}$. The fourth and fifth integral terms can also be computed by suitable numerical quadratures, but the flux functions f and g would need to be replaced by monotone numerical fluxes (or approximate Riemann solvers in the system case) because they are computed at cell interfaces in these four terms. The semi-discrete scheme (3.2) is discretized in time by a nonlinearly stable Runge-Kutta time discretization (2.10).

For the purpose of constructing the slope-limiting operator on two dimensional adaptive mesh, we also present the procedure which is similar to [6,8,25,30] and try to specify it in detail as follows. When it is the ordinary adaptive mesh, the cell I_i has some different size neighboring cells at the left side (I_{l_ℓ}, $\ell=1,\cdots,n_l$), right side (I_{r_ℓ}, $\ell=1,\cdots,n_r$), bottom side (I_{b_ℓ}, $\ell=1,\cdots,n_b$) and top side (I_{t_ℓ}, $\ell=1,\cdots,n_t$), respectively. These formulas are rewritten as:

$$\begin{aligned}\tilde{U}_i^{x,(mod)}=m\Bigg(&\tilde{U}_i^x,\min\Big(\alpha_1\frac{x_i+\frac{\Delta x_i}{2}-x_i}{x_{r_1}-x_i},\cdots,\alpha_{n_r}\frac{x_i+\frac{\Delta x_i}{2}-x_i}{x_{r_{n_r}}-x_i}\Big)\Big(\frac{\sum_{\ell=1}^{n_r}U_{r_\ell}^{(0)}|I_{r_\ell}|}{\sum_{\ell=1}^{n_r}|I_{r_\ell}|}-U_i^{(0)}\Big),\\
&\min\Big(\alpha_1\frac{x_i+\frac{\Delta x_i}{2}-x_i}{x_i-x_{l_1}},\cdots,\alpha_{n_l}\frac{x_i+\frac{\Delta x_i}{2}-x_i}{x_i-x_{l_{n_l}}}\Big)\Big(U_i^{(0)}-\frac{\sum_{\ell=1}^{n_l}U_{l_\ell}^{(0)}|I_{l_\ell}|}{\sum_{\ell=1}^{n_l}|I_{l_\ell}|}\Big)\Bigg),\end{aligned}$$

$$\tilde{\tilde{U}}_i^{x,(mod)} = m\left(\tilde{\tilde{U}}_i^x, \min\left(\alpha_1 \frac{x_i-(x_i-\frac{\Delta x_i}{2})}{x_{r_1}-x_i}, \cdots, \alpha_{n_r}\frac{x_i-(x_i-\frac{\Delta x_i}{2})}{x_{r_{n_r}}-x_i}\right)\left(\frac{\sum_{\ell=1}^{n_r} U_{r_\ell}^{(0)}|I_{r_\ell}|}{\sum_{\ell=1}^{n_r}|I_{r_\ell}|} - U_i^{(0)}\right),\right.$$
$$\left.\min\left(\alpha_1 \frac{x_i-(x_i-\frac{\Delta x_i}{2})}{x_i-x_{l_1}}, \cdots, \alpha_{n_l}\frac{x_i-(x_i-\frac{\Delta x_i}{2})}{x_i-x_{l_{n_l}}}\right)\left(U_i^{(0)} - \frac{\sum_{\ell=1}^{n_l} U_{l_\ell}^{(0)}|I_{l_\ell}|}{\sum_{\ell=1}^{n_l}|I_{l_\ell}|}\right)\right),$$

$$\tilde{U}_i^{y,(mod)} = m\left(\tilde{U}_i^y, \min\left(\alpha_1 \frac{y_i+\frac{\Delta y_i}{2}-y_i}{y_{t_1}-y_i}, \cdots, \alpha_{n_t}\frac{y_i+\frac{\Delta y_i}{2}-y_i}{y_{t_{n_t}}-y_i}\right)\left(\frac{\sum_{\ell=1}^{n_t} U_{t_\ell}^{(0)}|I_{t_\ell}|}{\sum_{\ell=1}^{n_t}|I_{t_\ell}|} - U_i^{(0)}\right),\right.$$
$$\left.\min\left(\alpha_1 \frac{y_i+\frac{\Delta y_i}{2}-y_i}{y_i-y_{b_1}}, \cdots, \alpha_{n_b}\frac{y_i+\frac{\Delta y_i}{2}-y_i}{y_i-y_{b_{n_b}}}\right)\left(U_i^{(0)} - \frac{\sum_{\ell=1}^{n_b} U_{b_\ell}^{(0)}|I_{b_\ell}|}{\sum_{\ell=1}^{n_b}|I_{b_\ell}|}\right)\right),$$

$$\tilde{\tilde{U}}_i^{y,(mod)} = m\left(\tilde{\tilde{U}}_i^y, \min\left(\alpha_1 \frac{y_i-(y_i-\frac{\Delta y_i}{2})}{y_{t_1}-y_i}, \cdots, \alpha_{n_t}\frac{y_i-(y_i-\frac{\Delta y_i}{2})}{y_{t_{n_t}}-y_i}\right)\left(\frac{\sum_{\ell=1}^{n_t} U_{t_\ell}^{(0)}|I_{t_\ell}|}{\sum_{\ell=1}^{n_t}|I_{t_\ell}|} - U_i^{(0)}\right),\right.$$
$$\left.\min\left(\alpha_1 \frac{y_i-(y_i-\frac{\Delta y_i}{2})}{y_i-y_{b_1}}, \cdots, \alpha_{n_b}\frac{y_i-(y_i-\frac{\Delta y_i}{2})}{y_i-y_{b_{n_b}}}\right)\left(U_i^{(0)} - \frac{\sum_{\ell=1}^{n_b} U_{b_\ell}^{(0)}|I_{b_\ell}|}{\sum_{\ell=1}^{n_b}|I_{b_\ell}|}\right)\right),$$

where all α_* are set to be 2 in our numerical simulation.

3.2 RKDG method for level set advection equation and MGFM in 2D

The level set advection equation in two dimensions is

$$\phi_t + \mu\phi_x + \nu\phi_y = 0. \tag{3.3}$$

If set $\eta = \phi_x$, $\zeta = \phi_y$ and $H(\eta,\zeta) = \mu\eta + \nu\zeta$, (3.3) is equivalent to the conservation laws:

$$\begin{cases} \eta_t + H(\eta,\zeta)_x = 0, \\ \zeta_t + H(\eta,\zeta)_y = 0. \end{cases} \tag{3.4}$$

The mesh is specified as before. We give the same test function space $V_h^{k-1} = \{p : p|_{I_i} \in P^{k-1}(I_i)\}$ as the polynomial space of degree at most $k-1$ on the cell I_i. We adopt the same local orthogonal basis over I_i, $\{v_l^{(i)}(x,y),\ l = 0,1,\cdots,\hat{K};\ \hat{K} = \frac{1}{2}k(k+1)-1\}$ as before. Then the numerical solutions $\eta^h(x,y,t)$ and $\zeta^h(x,y,t)$ in the space V_h^{k-1} can be written as $\eta^h(x,y,t) = \sum_{l=0}^{\hat{K}} \eta_i^{(l)}(t) v_l^{(i)}(x,y)$ and $\zeta^h(x,y,t) = \sum_{l=0}^{\hat{K}} \zeta_i^{(l)}(t) v_l^{(i)}(x,y)$, for $(x,y) \in I_i$ and the degrees of freedom $\eta_i^{(l)}(t)$ and $\zeta_i^{(l)}(t)$ are the moments defined by

$$\eta_i^{(l)}(t) = \frac{1}{\int_{I_i}(v_l^{(i)}(x,y))^2 dxdy}\int_{I_i} \eta^h(x,y,t) v_l^{(i)}(x,y) dxdy, \quad l = 0,\cdots,\hat{K},$$
$$\zeta_i^{(l)}(t) = \frac{1}{\int_{I_i}(v_l^{(i)}(x,y))^2 dxdy}\int_{I_i} \zeta^h(x,y,t) v_l^{(i)}(x,y) dxdy, \quad l = 0,\cdots,\hat{K}.$$

In order to determine the approximate solutions, we evolve the degrees of freedom $\eta_i^{(l)}(t)$ and $\zeta_i^{(l)}(t)$:

$$\begin{aligned}\frac{d}{dt}\eta_i^{(l)}(t)=&-\frac{1}{\int_{I_i}(v_l^{(i)}(x,y))^2dxdy}\Bigg(-\int_{I_i}H(\eta^h(x,y,t),\zeta^h(x,y,t))\frac{\partial}{\partial x}v_l^{(i)}(x,y)dxdy\\&+\int_{y_i-\frac{\Delta y_i}{2}}^{y_i+\frac{\Delta y_i}{2}}\Big(H\Big(\eta^h\Big(x_i+\frac{\Delta x_i}{2},y,t\Big),\zeta^h\Big(x_i+\frac{\Delta x_i}{2},y,t\Big)\Big)v_l^{(i)}\Big(x_i+\frac{\Delta x_i}{2},y\Big)\\&-H\Big(\eta^h\Big(x_i-\frac{\Delta x_i}{2},y,t\Big),\zeta^h\Big(x_i-\frac{\Delta x_i}{2},y,t\Big)\Big)v_l^{(i)}\Big(x_i-\frac{\Delta x_i}{2},y\Big)\Big)dy\Bigg),\quad l=0,\cdots,\hat{K},\end{aligned}\tag{3.5}$$

and

$$\begin{aligned}\frac{d}{dt}\zeta_i^{(l)}(t)=&-\frac{1}{\int_{I_i}(v_l^{(i)}(x,y))^2dxdy}\Bigg(-\int_{I_i}H(\eta^h(x,y,t),\zeta^h(x,y,t))\frac{\partial}{\partial y}v_l^{(i)}(x,y)dxdy\\&+\int_{x_i-\frac{\Delta x_i}{2}}^{x_i+\frac{\Delta x_i}{2}}\Big(H\Big(\eta^h\Big(x,y_i+\frac{\Delta y_i}{2},t\Big),\zeta^h\Big(x,y_i+\frac{\Delta y_i}{2},t\Big)\Big)v_l^{(i)}\Big(x,y_i+\frac{\Delta y_i}{2}\Big)\\&-H\Big(\eta^h\Big(x,y_i-\frac{\Delta y_i}{2},t\Big),\zeta^h\Big(x,y_i-\frac{\Delta y_i}{2},t\Big)\Big)v_l^{(i)}\Big(x,y_i-\frac{\Delta y_i}{2}\Big)\Big)dx\Bigg),\quad l=0,\cdots,\hat{K}.\end{aligned}\tag{3.6}$$

Then the third order TVD Runge-Kutta method (2.10) is used to solve for the ODE and the high order accurate RKDG method could be obtained both in space and time. The DG method for (3.4) is satisfied by requiring the derivatives $\eta=\phi_x$ and $\zeta=\phi_y$, and these determine ϕ for the target cell I_i up to a constant when $k=1$. But we also find (3.5) and (3.6) have more equations than the number of degrees of the freedom for the solutions when $k>1$ and then use least square sense:

$$\int_{I_i}(\phi_x-\eta)^2+(\phi_y-\zeta)^2dxdy=\min_{\forall\varphi\in P^k(I_i)}\int_{I_i}(\varphi_x-\eta)^2+(\varphi_y-\zeta)^2dxdy,\tag{3.7}$$

which is specified in [12] to get the missing constant by requiring that:

$$\int_{I_i}(\phi_t+H(\phi_x,\phi_y))dxdy=0.\tag{3.8}$$

Thus, the numerical solution $\phi^h(x,y,t)$ in the space V_h^k can be obtained in this way.

The MGFM-based RKDG method is extended to two dimensional two-medium flow and a two-medium Riemann problem in the normal direction of the interface is defined by following exactly the same way as described in [18–20]. The defined Riemann problem is also solved by using the exact or approximate Riemann problem solver. The predicted interface density, normal velocity and pressure are then used to replace the flow density, normal velocity and pressure at the cells of interface located and extended to the ghost fluid cells.

3.3 The algorithm of adaptive method in 2D

Step 2.1. Partition the computing field into uniform cells as $I_i=[x_i-\frac{\Delta x_i}{2},x_i+\frac{\Delta x_i}{2}]\times[y_i-\frac{\Delta y_i}{2},y_i+\frac{\Delta y_i}{2}]$, $i=1,\cdots,N$. Use the initial conditions to get the degrees of freedom $U_i^{(l)}(t_0)$ and $\phi_i^{(l)}(t_0)$, $l=0,\cdots,K,i=1,\cdots,N$. And set the adaptive mesh level $R_i(t_0)=0,i=1,\cdots,N$ at first.

Step 2.2. Suppose at time t_n, the degrees of freedom $U_i^{(l)}(t_n)$, $\phi_i^{(l)}(t_n)$, $l=0,\cdots,K,i=1,\cdots,N$ and the adaptive mesh level $R_i(t_n),i=1,\cdots,N$ are given. We apply KXRCF method as an indicator to detect "troubled cells".

Step 2.2.1. Suppose I_{i0} is a "troubled cell": if $R_{i0}(t_n)<R$, divide it into four equal size small children cells, increase such children cells' mesh level by 1, and redefine the degrees of freedom of $U^h(x,y,t_n)$ and $\phi^h(x,y,t_n)$ on these cells (specified in the following Subsection 3.5); if $R_{i0}(t_n)=R$, do nothing.

Step 2.2.2. If all children cells are not identified as "troubled cells", merge them into their parent cell and decrease such parent cell's mesh level by 1, and redefine the degrees of freedom of $U^h(x,y,t)$ and $\phi^h(x,y,t)$ on this cell (specified in the following Subsection 3.5).

By doing so, we obtain a new adaptive mesh with different mesh level $R_i(t_{n+1}),i=1,\cdots,N$.

Step 2.3. Redefine the new degrees of freedom of $U^h(x,y,t_n)$ and $\phi^h(x,y,t_n)$ over the whole computation region on the new adaptive mesh.

Step 2.4. Evolve the solutions of the two-medium flow and level set function from t_n to t_{n+1} by the methodologies specified in Subsection 3.1, Subsection 3.2 and Subsection 3.3 to obtain the degrees of freedom $U_i^{(l)}(t_{n+1})$ and $\phi_i^{(l)}(t_{n+1})$, $l=0,\cdots,K$, $i=1,\cdots,N$. The latter procedures are similar to Step 1.4.1.–Step 1.4.4.

Step 2.5. If t_{n+1} is less than the final time, go to Step 2.2.

3.4 Redefine $U^h(x,y,t)$ and $\phi^h(x,y,t)$ on the adaptive mesh

If the flagged cell A ("troubled cell") is restricted into four smaller cells B_1, B_2, B_3 and B_4, which are depicted in Fig. 3, the numerical conservative variable of $U^h(x,y,t)$ and the numerical level set function $\phi^h(x,y,t)$ denoted on cell A should be remapped to such new cells. The methodology can be expressed as:

$$
\begin{aligned}
&(U_{B_j}^{(0)}(t),\cdots,U_{B_j}^{(K)}(t))^T=M^{-1}(b_0,\cdots,b_K)^T,\quad M=(m_{sq})_{(K+1)\times(K+1)},\\
&m_{sq}=\int_{I_{B_j}} v_s^{(A)}(x,y)v_q^{(B_j)}(x,y)dxdy,\quad b_s=\int_{I_{B_j}} U_A^h(x,y,t)v_s^{(B_j)}(x,y)dxdy,\\
&(\phi_{B_j}^{(0)}(t),\cdots,\phi_{B_j}^{(K)}(t))^T=M^{-1}(b_0,\cdots,b_K)^T,\\
&b_s=\int_{I_{B_j}} \phi_A^h(x,y,t)v_s^{(B_j)}(x,y)dxdy,\quad s,q=0,\cdots,K,\ j=1,2,3,4.
\end{aligned}
$$

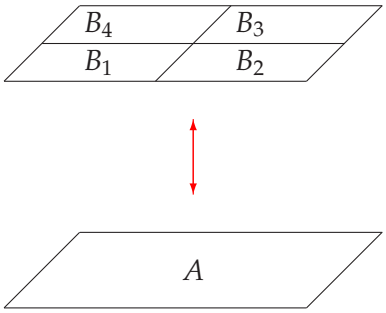

Figure 3: The topological map of parent and children cells in 2D.

If the children cells B_1, B_2, B_3 and B_4 are not detected as "troubled cells" any more, and should be merged into the parent cell A, the numerical conservative variable of $U^h(x,y,t)$ and the numerical level set function $\phi^h(x,y,t)$ denoted on cell A should be remapped from such children cells. And the methodology can be expressed as below:

$$
\begin{aligned}
&(U_A^{(0)}(t),\cdots,U_A^{(K)}(t))^T = M^{-1}(b_0,\cdots,b_K)^T, \quad M=(m_{sq})_{(K+1)\times(K+1)},\\
&m_{sq}=\int_{I_A} v_s^{(A)}(x,y)v_q^{(A)}(x,y)dxdy, \quad b_s=\sum_{j=1}^{4}\int_{I_{B_j}} U_{B_j}^h(x,y,t)v_s^{(A)}(x,y)dxdy,\\
&(\phi_A^{(0)}(t),\cdots,\phi_A^{(K)}(t))^T = M^{-1}(b_0,\cdots,b_K)^T,\\
&b_s=\sum_{j=1}^{4}\int_{I_{B_j}} \phi_{B_j}^h(x,y,t)v_s^{(A)}(x,y)dxdy, \quad s,q=0,\cdots,K.
\end{aligned}
$$

4 Numerical tests

In this Section, the results of the numerical tests for the gas-gas and gas-water flows by the adaptive RKDG methods with MGFM described in Section 2 and Section 3 are presented. For simplicity, the CFL number is set to be 0.18 for third order RKDG methods in one and two dimensions. In these tests, the units for the density, velocity, pressure, length and time are kg/m^3, m/s, Pa, m and s, respectively. All the computations are run on the computer cluster, 2850 Xeon CPU@3.4 GHz with 4 GB ram. In all of the 1D cases, the initial coarsest uniform mesh is set as equal 100 cells in the computing field. In 2D case, the initial uniform mesh is set as equal 35×30 cells in the computing field.

Example 4.1. The Euler equations (2.1) are considered in this problem and the following Riemann initial conditions are given as:

$$
(\rho,\mu,p,\gamma)^T=\begin{cases}(1.3333,0.3535\sqrt{10^5},1.5\times10^5,1.4)^T, & x\le0.05,\\ (1,0,10^5,1.4)^T, & 0.05<x\le0.5,\\ (3.1538,0,10^5,1.249)^T, & x>0.5.\end{cases} \tag{4.1}
$$

Table 1: Example 4.1. The computing times and the final cells subject to the different adaptive and uniform meshes.

adaptive level R	1	2	3
adaptive cells	116	133	164
time (second)	7.53	16.24	28.09
uniform cells	200	400	800
time (second)	11.45	36.01	120.29

The computing times and the final cells subject to the adaptive and uniform meshes are given in Table 1. The computed density, velocity and pressure at $t=0.0017$ against the exact solutions and the trajectories of the adaptive meshes with the TVB constant of $M=20$ are plotted in Fig. 4. The initial material interface is set at $x^*(0)=0.5$ and the initial mesh for adaptive method is 100 uniform cells. The final mesh number for adaptive method with adaptive level $R=3$ is 164. The location of the material interface is captured correctly in this test case. The computed results are oscillatory free at the neighborhood of the material interface for the density, velocity and pressure. From the trajectories of adaptive meshes, it is seen that meshes are refined along trajectories of discontinuity of the solution (shock, contact discontinuity) as expected. The computational results by two types of the adaptive RKDG methods for solving the two-medium flow and the level set advection equation are better than those with 200 cells and are comparable to those with 400 and 800 cells on uniform meshes.

Example 4.2. This is a gas-water shock tube problem with very high pressure in the gaseous medium. The initial conditions are:

$$(\rho,\mu,p,\gamma)^T=\begin{cases}(1270,0,8\times10^8,1.4)^T, & x\leq 0.5,\\ (1000,0,10^5,7.15)^T, & x>0.5.\end{cases} \tag{4.2}$$

The computing times and the final cells subject to the adaptive and uniform meshes are given in Table 2. The computed density, velocity and pressure at $t=0.00016$ against the exact solutions and the trajectories of the adaptive meshes with the TVB constant of $M=20$ are plotted in Fig. 5. The initial material interface is set at $x^*(0)=0.5$ and the initial mesh for adaptive method is 100 uniform cells. The final mesh number for

Table 2: Example 4.2. The computing times and the final cells subject to the different adaptive and uniform meshes.

adaptive level R	1	2	3
adaptive cells	112	141	167
time (second)	4.59	11.87	31.74
uniform cells	200	400	800
time (second)	27.98	62.18	143.66

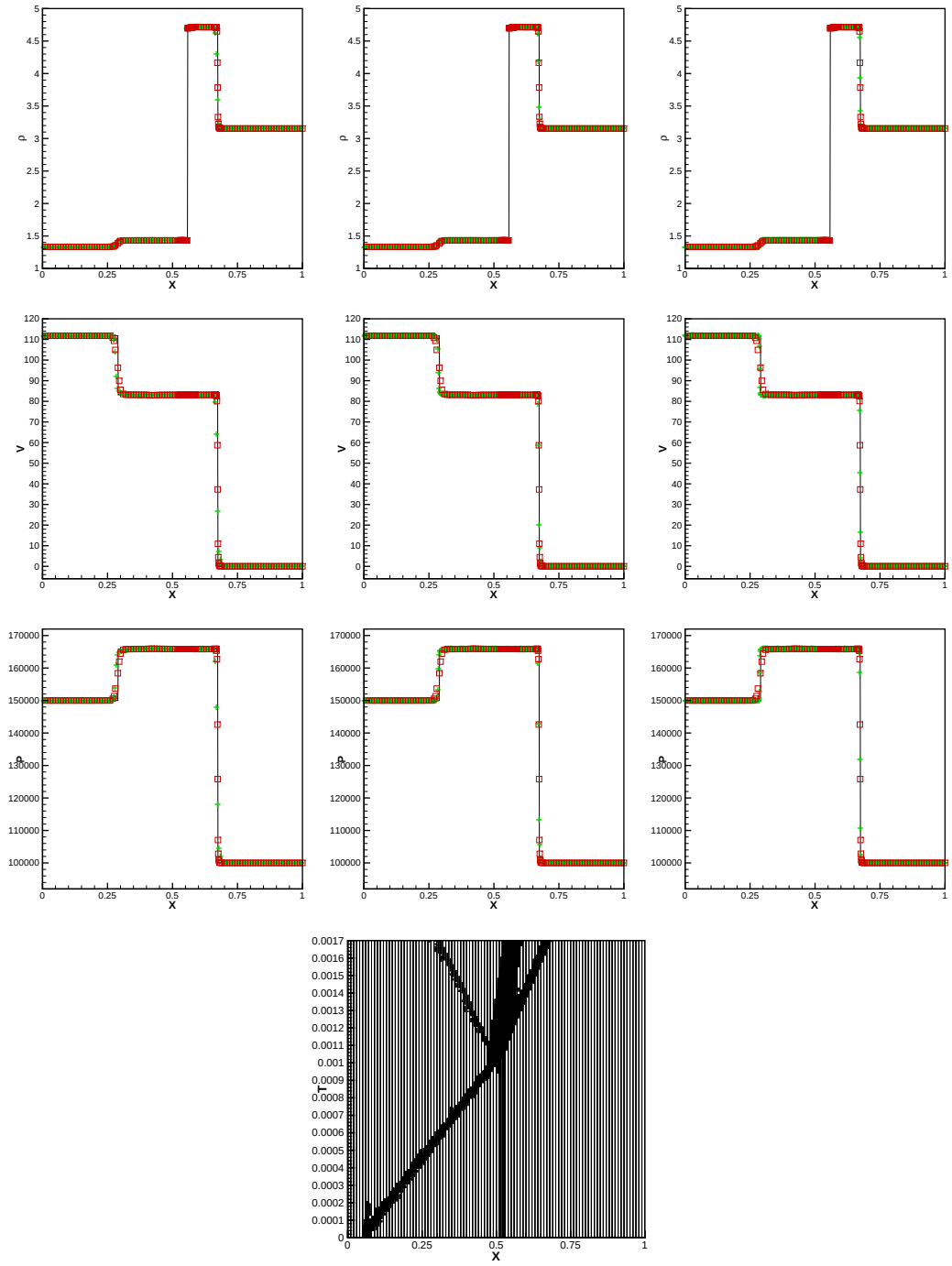

Figure 4: Example 4.1. Top to bottom: density, velocity, pressure and the trajectories of adaptive meshes. $t=0.0017$. Line: exact solution; squares: numerical solution on adaptive mesh; plus signs: numerical solution on uniform mesh. Left to right: adaptive mesh against uniform mesh with 200, 400 and 800 cells, respectively.

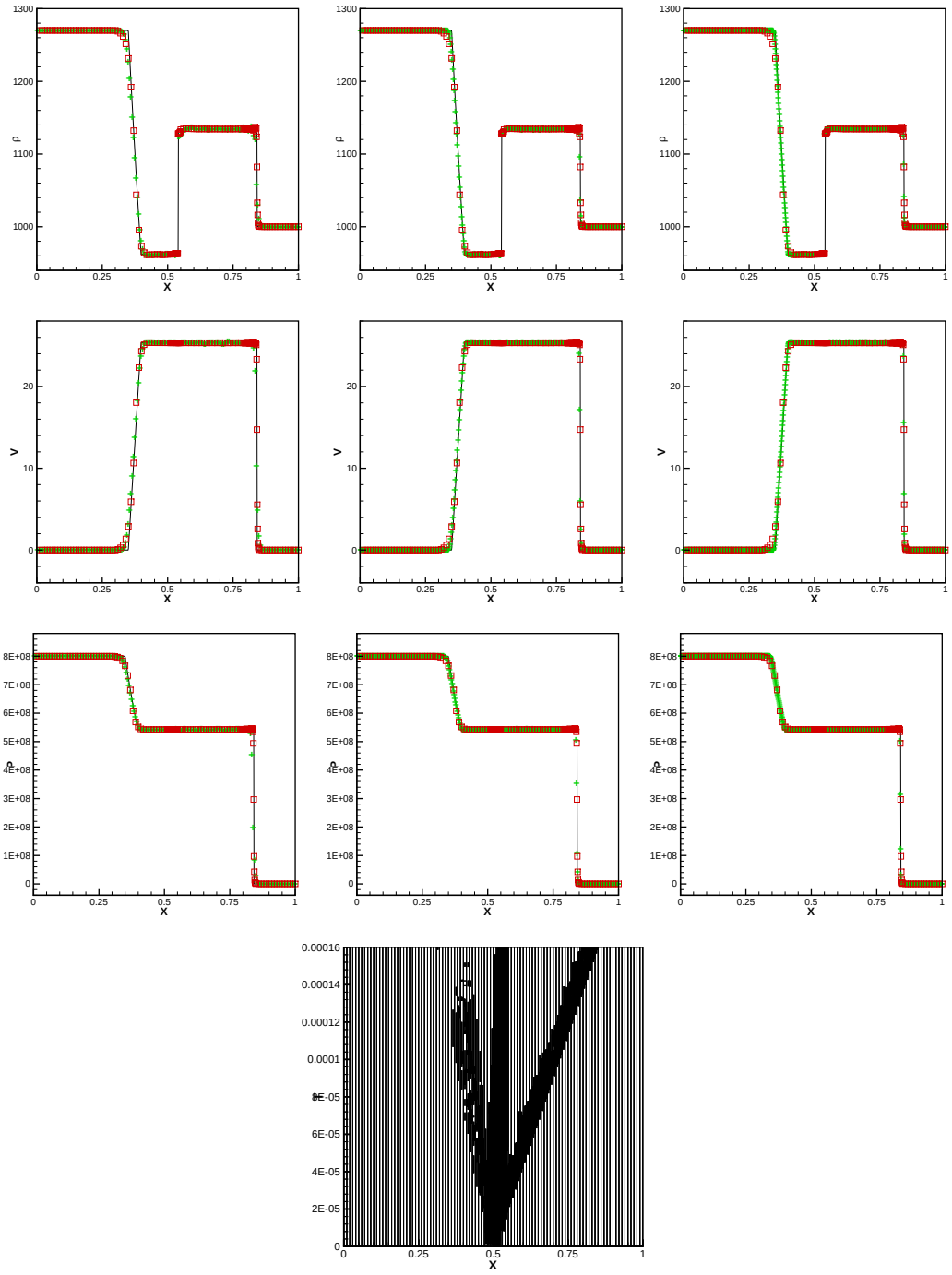

Figure 5: Example 4.2. Top to bottom: density, velocity, pressure and the trajectories of adaptive meshes. $t=0.00016$. Line: exact solution; squares: numerical solution on adaptive mesh; plus signs: numerical solution on uniform mesh. Left to right: adaptive mesh against uniform mesh with 200, 400 and 800 cells, respectively.

adaptive method with adaptive level $R=3$ is 167. We can verify that the location of the material interface is captured correctly. The computed results are oscillatory free at the neighborhood of the interface for the density, velocity and pressure. The computational results by adaptive method are better than those with 200 cells and are comparable to those with 400 and 800 cells.

Example 4.3. We greatly increase the energy of the explosive gaseous medium with the initial conditions given as:

$$(\rho,\mu,p,\gamma)^T=\begin{cases}(1630,0,7.81\times10^9,1.4)^T, & x\leq0.5,\\(1000,0,10^5,7.15)^T, & x>0.5.\end{cases} \tag{4.3}$$

The computing times and the final cells subject to the adaptive and uniform meshes are given in Table 3. The computed density, velocity and pressure at $t=0.0001$ against the exact solutions and the trajectories of the adaptive meshes with the TVB constant of $M=20$ are plotted in Fig. 6. The initial material interface is set at $x^*(0)=0.5$ and the initial mesh for adaptive method is 100 uniform cells. The final mesh number for adaptive method with adaptive level $R=3$ is 117. It is observed that the location of the material interface is captured precisely. The computed results are oscillatory free at the vicinities of the material interface for different physical variables. From the trajectories of adaptive meshes, we can also see that the meshes are refined along trajectories of discontinuity of the solution. It can be seen again that the computational results by adaptive method are better than those with 200 and 400 cells on uniform mesh and are comparable to those with 800 cells.

Table 3: Example 4.3. The computing times and the final cells subject to the different adaptive and uniform meshes.

adaptive level R	1	2	3
adaptive cells	106	114	117
time (second)	6.02	11.23	20.58
uniform cells	200	400	800
time (second)	16.02	63.32	235.51

Example 4.4. In this example, we study a planar underwater shock interacting with a gas bubble in an open domain. Similar problems have been studied in [11]. We examine an initial Mach 1.653 underwater shock wave makes impact on a gas bubble. The non-dimensional initial conditions are:

$$(\rho,\mu,\nu,p,\gamma)^T=\begin{cases}(1000,0,0,1,7.15)^T, & x>-1.2,\\(1176.3333,1.1692,0,9120,7.15)^T, & x\leq-1.2,\\(1,0,0,1,1.4)^T, & \sqrt{x^2+y^2}<1,\end{cases} \tag{4.4}$$

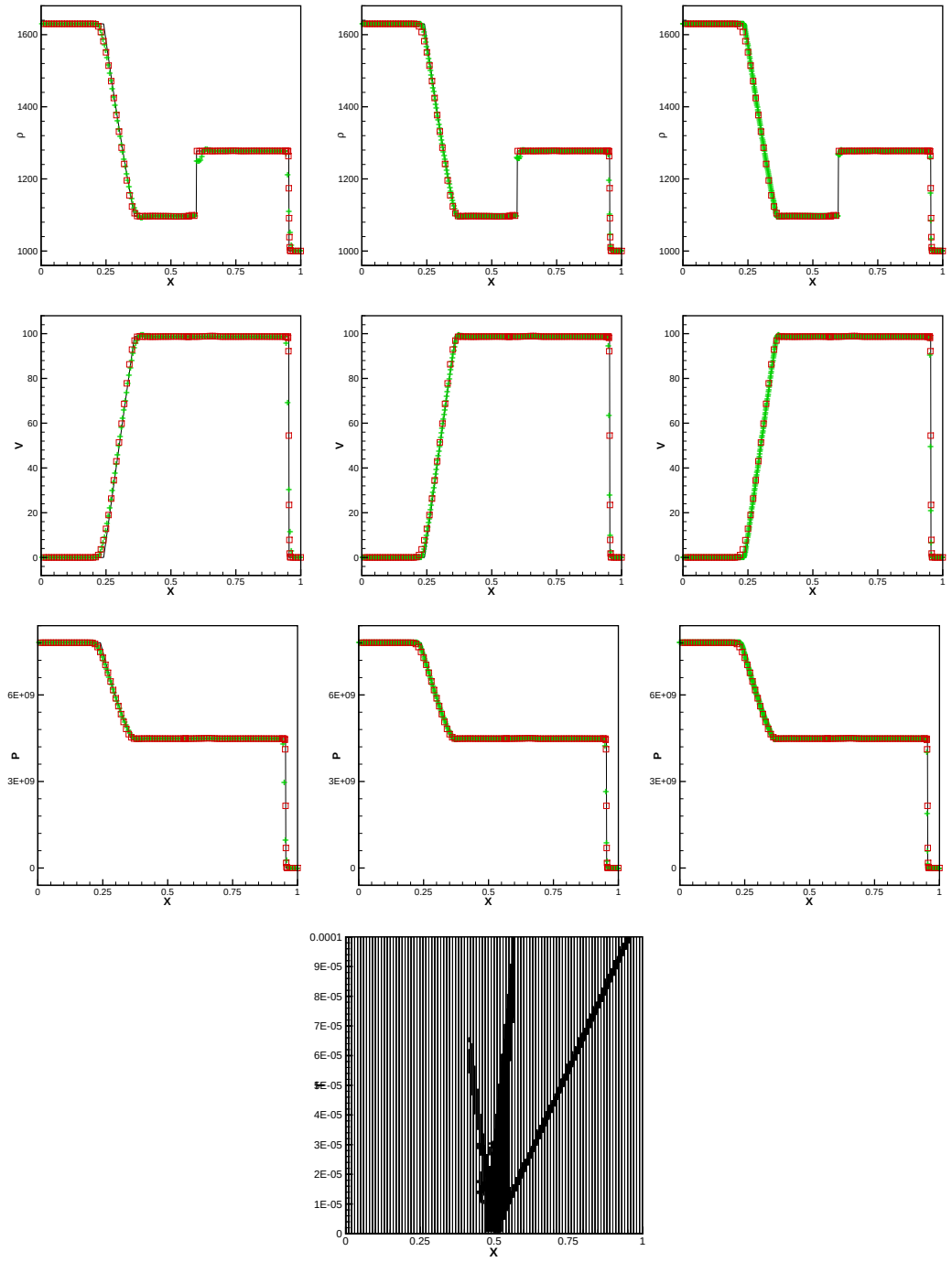

Figure 6: Example 4.3. Top to bottom: density, velocity, pressure and the trajectories of adaptive meshes. $t = 0.0001$. Line: exact solution; squares: numerical solution on adaptive mesh; plus signs: numerical solution on uniform mesh. Left to right: adaptive mesh against uniform mesh with 200, 400 and 800 cells, respectively.

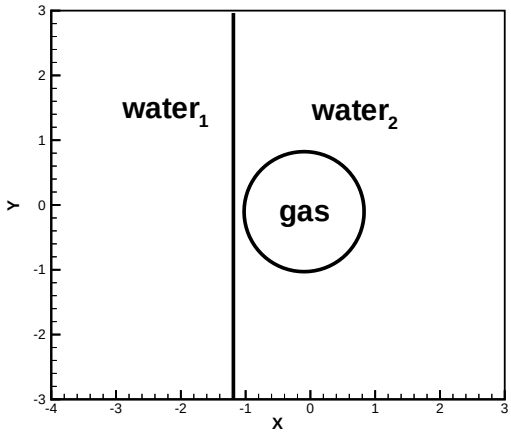

Figure 7: Example 4.4. The schematic diagram.

and the level set function $\phi=\sqrt{x^2+y^2}-1$, where $\phi\leq 0$ represents the gas and $\phi>0$ represents the water. We use the original uniformed mesh of 35×30 cells over the computational region $[-4,3]\times[-3,3]$ and the schematic diagram is shown in Fig. 7. The density is plotted at $t=0.06$, 0.19, 0.357, 0.481 and the adaptive mesh graphs to the RKDG method with TVB constant of $M=1$ are plotted in Figs. 8 and 9. Again, very complex physics occurs in this problem especially when a jet forms and bubble collapses in the very late stage. Our computations stopped just before the bubble collapses. In the earlier stage, the shock refraction on the bubble surface is regular initially but transits into an irregular type after the incident shock past over a critical angle [11].

5 Concluding remarks

In this paper, the adaptive RKDG methods with MGFM are applied to solve for the two-medium flow in one and two dimensions. We first use the KXRCF method [16] as an indicator to detect the troubled cell, then apply the modified TVB minmod-type limiter to reconstruct the freedoms of the solution inside of such cell, adopt RKDG methods for solving the level set function [12] and do the mesh adapting procedure. At each time step, a new adaptive rectangular mesh is obtained by splitting the troubled cell into smaller children cells (refining procedure) or merging all children cells into their parent cell (coarsening procedure) and obtain associated solutions of the two-medium flow and level set function on the new adaptive mesh for further computations. The proposed methodologies are able to provide a sharp real interface location and still with reasonable solution for the whole domain with very limited oscillations nearby the discontinuities in an adaptive manner. Extensive numerical results for two-medium flow are provided to show that the adaptive methodologies are stable and robust subject to many different initial conditions.

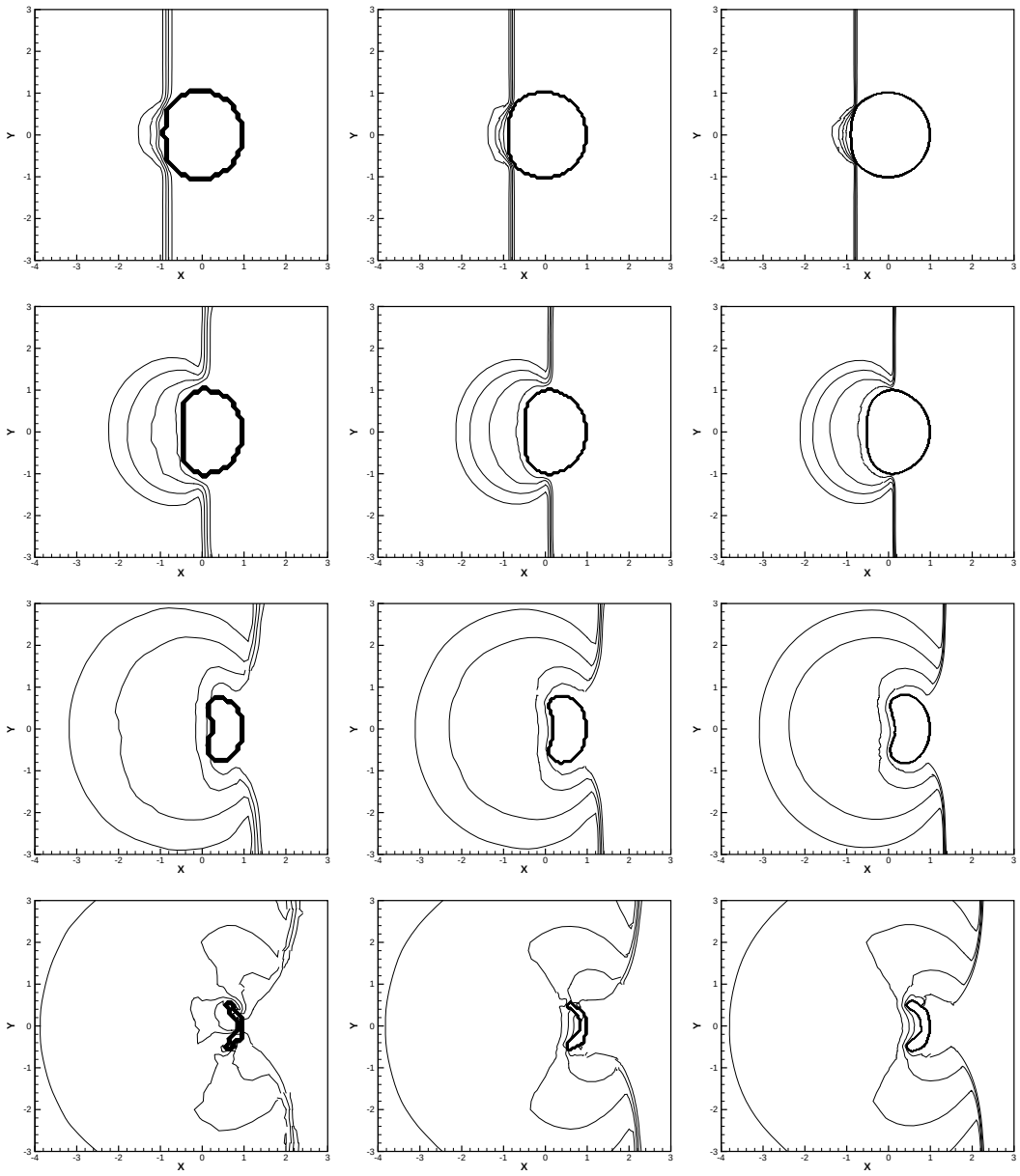

Figure 8: Example 4.4. Top to bottom: 30 equally spaced density contours from 38 to 1147, from 38 to 1147, from 40 to 1146 and from 38 to 1146. Top to bottom: $t=0.06$, 0.19, 0.357, 0.481. Left to right: adaptive mesh level $R=1$, $R=2$ and $R=3$. TVB constant: $M=1$. Original mesh: 35 $\times$ 30 cells.

Acknowledgments

The research was partially supported by NSFC grants 91203110, 11372005 and 11002071.

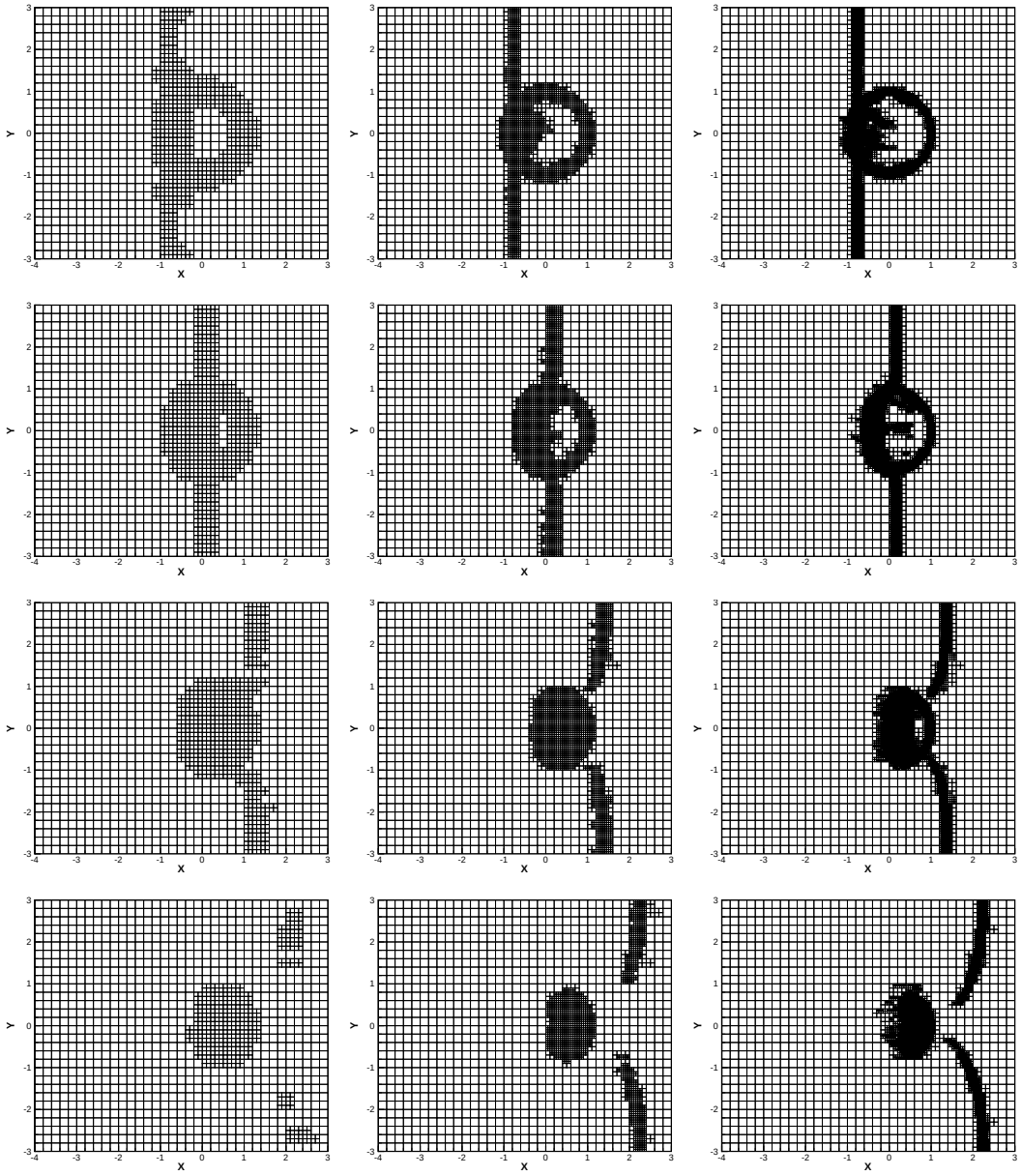

Figure 9: Example 4.4. Mesh. Top to bottom: $t=0.06$, 0.19, 0.357, 0.481. Left to right: adaptive mesh level $R=1$, $R=2$ and $R=3$. TVB constant: $M=1$. Original mesh: 35 $\times$ 30 cells.

References

[1] R. Abgrall, S. Karni. Computations of compressible multifluids. Journal of Computational Physics, 169: 594–623, 2001.

[2] G. Allaire, S. Clerc, S. Kokh. A five-equation model for the simulation of interfaces between

compressible fluids. Journal of Computational Physics, 181: 577–616, 2002.
[3] R. Biswas, K.D. Devine, J. Flaherty. Parallel, adaptive finite element methods for conservation laws. Applied Numerical Mathematics, 14: 255–283, 1994.
[4] B. Cockburn, S. Hou, C.-W. Shu. The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case. Mathematics of Computation, 54: 545–581, 1990.
[5] B. Cockburn, S.Y. Lin, C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one dimensional systems. Journal of Computational Physics, 84: 90–113, 1989.
[6] B. Cockburn, C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework. Mathematics of Computation, 52: 411–435, 1989.
[7] B. Cockburn, C.-W. Shu. The Runge-Kutta local projection P1-discontinuous Galerkin finite element method for scalar conservation laws. Mathematical Modelling and Numerical Analysis (M2AN), 25: 337–361, 1991.
[8] B. Cockburn, C.-W. Shu. The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, Journal of Computational Physics. 141: 199–224, 1998.
[9] K. Devine, J. Flaherty. Parallel adaptive hp-refinement techniques for conservation laws, Applied Numerical Mathematics. 20: 367–386, 1996.
[10] R.P. Fedkiw, T. Aslam, B. Merriman, S. Osher. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). Journal of Computational Physics, 152: 457–492, 1999.
[11] J. Grove, R. Menikoff. Anomalous reflection of a shock wave at a fluid interface. J. Fluid Mech., 219:313–336, 1990.
[12] C. Hu, C.-W. Shu. A discontinuous Galerkin finite element method for Hamilton-Jacobi equations, SIAM Journal on Scientific Computing. 21: 666–690, 1999.
[13] A.K. Kapila, R. Menikoff, J.B. Bdzil, S.F. Son, D.S. Stewart. Two-phase modeling of deflagration-to-detonation transition in granular materials: reduced equations. Physics of Fluids, 13: 3002–3024, 2001.
[14] B. Koren, M.R. Lewis, E.H. van Brummelen, B. van Leer. Riemann-problem and level-set approaches for homentropic two-fluid flow computations. Journal of Computational Physics, 229: 6220–6242, 2010.
[15] J.J. Kreeft, B. Koren. A new formulation of Kapilas five-equation model for compressible two-fluid flow, and its numerical treatment., Journal of Computational Physics, 181: 654–674, 2002.
[16] L. Krivodonova, J. Xin, J.F. Remacle, N. Chevaugeon, J.E. Flaherty. Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws. Applied Numerical Mathematics, 48: 323–338, 2004.
[17] T.G. Liu, B.C. Khoo, C.W. Wang. The ghost fluid method for compressible gas-water simulation. Journal of Computational Physics, 204: 193–221, 2005.
[18] T.G. Liu, B.C. Khoo, K.S. Yeo. The simulation of compressible multi-medium flow. Part I: a new methodology with applications to 1D gas-gas and gas-water cases. Computers and Fluids, 30: 291–314, 2001.
[19] T.G. Liu, B.C. Khoo, K.S. Yeo. The simulation of compressible multi-medium flow. Part II: applications to 2D underwater shock refraction. Computers and Fluids, 30: 315–337, 2001.
[20] T.G. Liu, B.C. Khoo, K.S. Yeo. Ghost fluid method for strong shock impacting on material interface. Journal of Computational Physics, 190: 651–681, 2003.

[21] J. Massoni, R. Saurel, B. Nkonga, R. Abgrall. Propositions de mthodes et modles Eulriens pour les problmes interfaces entre fluides compressibles en prsence de transfert de chaleur. International Journal of Heat and Mass Transfer, 45: 1287–1307, 2002.
[22] F. Petitpas, E. Franquet, R. Saurel, O.L. Metayer. A relaxation-projection method for compressible flows. Part II: Artificial heat exchanges for multiphase shocks. Journal of Computational Physics, 225: 2214–2248, 2007.
[23] J. Qiu, T. G. Liu and B.C. Khoo. Runge-Kutta Discontinuous Galerkin methods for two-medium flow simulations: one-dimensional case. Journal of Computational Physics, 222: 353–373, 2007.
[24] J. Qiu, T.G. Liu, B.C. Khoo. Simulations of compressible two-medium flow by Runge-Kutta discontinuous Galerkin methods with the ghost fluid method. Communications in Computational Physics, 3: 479-504, 2008.
[25] J. Qiu, C.-W. Shu. Runge-Kutta discontinuous Galerkin method using WENO limiters. SIAM Journal on Scientific Computing, 26: 907–929, 2005.
[26] J.J. Quirk, S. Karni. On the dynamics of a shock-bubble interaction. J. Fluid Mech., 318: 129–163, 1996.
[27] W.H. Reed, T.R. Hill. Triangular mesh methods for neutron transport equation. Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
[28] J.F. Remacle, J. Flaherty and M. Shephard. An adaptive discontinuous Galerkin technique with an orthogonal basis applied to compressible flow problems. SIAM Riview 45: 53–72, 2003.
[29] R. Saurel, E. Franquet, E. Daniel, O.L. Metayer. A relaxation-projection method for compressible flows. Part I: The numerical equation of state for the Euler equations. Journal of Computational Physics, 223: 822–845, 2007.
[30] C.-W. Shu. TVB uniformly high-order schemes for conservation laws. Mathematics of Computation, 49: 105–121, 1987.
[31] C.-W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, edited by A. Quarteroni, Editor, Lecture Notes in Mathematics, CIME subseries, Springer-Verlag, Berlin/New York; ICASE Report 97–65, 1997.
[32] C.-W. Shu, S. Osher. Efficient implementation of essentially non-oscillatory shock capturing schemes. Journal of Computational Physics, 77: 439–471, 1988.
[33] E.F. Toro. Riemann Solvers and Numerical Methods for Fluid Dynamics, a Practical Introduction, Springer, Berlin, 1997.
[34] E.F. Toro, M. Spruce, W. Speares. Restoration of the contact surface in the Harten-Lax-van Leer Riemann solver. Shock Waves., 4: 25–34, 1994.
[35] H. Zhu, J. Qiu. Adaptive Runge-Kutta discontinuous Galerkin methods using different indicators: One-dimensional case. J. Comput. Phys., 228: 6957–6976, 2009.
[36] H. Zhu, J. Qiu. An h-adaptive RKDG method with troubled-cell indicator for two- dimensional hyperbolic conservation laws. Adv. Comput. Math., 39: 445–463, 2013.
[37] J. Zhu, J. Qiu, T.G. Liu, B.C. Khoo. RKDG methods with WENO type limiters and conservative interfacial procedure for one-dimensional compressible multi-medium flow simulations. Applied Numerical Mathematics, 61: 554–580, 2011.