An RKDG finite element method for the one-dimensional inviscid compressible gas dynamics equations in a Lagrangian coordinate*

Zhao Guo-Zhong(赵国忠)[a)†] Yu Xi-Jun(蔚喜军)[b)], and Zhang Rong-Pei(张荣培)[c)]

[a)] *Faculty of Mathematics, Baotou Teachers' College, Baotou* 014030, *China*
[b)] *Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing* 100088, *China*
[c)] *School of Science, Liaoning Shihua University, Fushun* 113001, *China*

(Received 4 April 2012; revised manuscript received 6 September 2012)

In this paper, Runge–Kutta Discontinuous Galerkin (RKDG) finite element method is presented to solve the one-dimensional inviscid compressible gas dynamic equations in a Lagrangian coordinate. The equations are discretized by the DG method in space and the temporal discretization is accomplished by the total variation diminishing Runge–Kutta method. A limiter based on the characteristic field decomposition is applied to maintain stability and non-oscillatory property of the RKDG method. For multi-medium fluid simulation, the two cells adjacent to the interface are treated differently from other cells. At first, a linear Riemann solver is applied to calculate the numerical flux at the interface. Numerical examples show that there is some oscillation in the vicinity of the interface. Then a nonlinear Riemann solver based on the characteristic formulation of the equation and the discontinuity relations is adopted to calculate the numerical flux at the interface, which suppresses the oscillation successfully. Several single-medium and multi-medium fluid examples are given to demonstrate the reliability and efficiency of the algorithm.

Keywords: compressible gas dynamic equations, RKDG finite element method, Lagrangian coordinate, multi-medium fluid

PACS: 02.30.Jr, 02.70.Dh, 02.60.Lj **DOI:** 10.1088/1674-1056/22/2/020202

1. Introduction

Nonlinear partial differential equations (NPDEs) play an important role in the study of nonlinear physical phenomena. Many researchers have devoted effort to developing the efficient method for searching the analytical and numerical solutions of NPDEs.[1–5] NPDEs are also a useful mathematical tool for describing the fluid dynamics phenomena. Due to the nonlinearity of NPDEs, many researchers are interested in developing the efficient numerical method for solving such PDES. In the numerical simulations of fluid dynamics, there are two typical choices: one is the Lagrangian framework, in which the mesh moves with the local fluid velocity, and another is the Eulerian framework, in which the fluid flows on a grid fixed in space. Compared with the Eulerian method, the Lagrangian method has several advantages. Firstly, it can distinctly capture contact discontinuities. Secondly, it can reduce a source of numerical error due to the advection terms in the conservative equations. Thirdly, it is adopted in many fields for multi-material flow simulations such as astrophysics, plasma physics, and nuclear physics, etc. In the past several years, many efforts have been made to develop Lagrangian-type methods.[6–13]

The first discontinuous Galerkin (DG) method was introduced in 1973 by Reed and Hill for numerical simulation of the neutron transport equation.[14] A major development of the DG method was carried out by Cockburn and Shu in a series of papers,[15–19] in which they established a framework for solving nonlinear time-dependent hyperbolic conservation laws. The temporal discretizations of the proposed method are accomplished by using explicit, nonlinearly stable high order Runge–Kutta method and spacial discretizations are achieved by using discontinuous Galerkin (DG) method with exact or approximate Riemann solvers as interface fluxes. The total variation bounded (TVB) limiter is applied to keep non-oscillatory properties for strong shocks. The scheme is termed RKDG method. The method has some attractive properties.[20,21] For example, it can be designed for any order of accuracy in space and time and it can handle complicated geometry and boundary conditions easily. Besides this, it is extremely local in data communications and is flexible to h-p adaptivity. The majority of the previous research is based on the Eulerian framework.[22–25]

The RKDG method has been widely applied and performed very well for solving the single-medium compressible flow. There are also some works using the RKDG finite element method to deal with the multi-medium fluid problem in the Eulerian framework. A relatively dominant difficulty for simulating compressible two-medium flow is the treatment of moving material interfaces and their vicinities.

*Project supported by the National Natural Science Foundation of China (Grant Nos. 11261035, 11171038, and 10771019), the Science Reaearch Foundation of Institute of Higher Education of Inner Mongolia Autonomous Region, China (Grant No. NJZZ12198), and the Natural Science Foundation of Inner Mongolia Autonomous Region, China (Grant No. 2012MS0102).

†Corresponding author. E-mail: zhaoguozhongbttc@sina.com

© 2013 Chinese Physical Society and IOP Publishing Ltd

Nonphysical oscillations usually occur in the vicinity of the material interface when a well-established numerical method for single-medium flow is directly applied to multi-medium flow. In the literature, there are some methods to overcome such difficulty. The ghost fluid method (GFM) is one of the applicable techniques to treat the material interface. RKDG finite element method for two-medium flow simulations in one and two dimensions with the original GFM and the modified GFM were investigated by Qiu *et al*.[26] Another important work of Qiu[27] is using the RKDG finite element method for two-medium flow computation with conservative treatment of the moving material interface. Chen *et al*.[21] computed multi-medium compressible flow by the RKDG finite element method with a new modified GFM. The majority of the above works are based on the Eulerian framework. As we know, the Lagrangian-type methods are also applicable techniques to treat the material interface, especially in a one-dimensional situation. The high resolution total variation diminishing method proposed by Harten and the finite-point method are used to tackle the multi-medium fluid simulation in a Lagrangian coordinate.[28,29] We discussed the single-medium fluid simulation in a unified coordinate.[30,31]

In this paper, an RKDG finite element method for one-dimensional inviscid compressible gas dynamic equations in a Lagrangian coordinate is developed. The main motivation of this work is to combine the advantages of the Lagrangian method and the DG method to construct an applicable algorithm for multi-medium fluid simulation. Some single material fluid and multi-medium fluid examples are discussed in this paper.

The paper is arranged as follows. In the second section, the RKDG method in a Lagrangian coordinate will be described in detail for single-medium and multi-medium fluids simulation. Some numerical examples will be given in the third section. In the last section, we will give some conclusions.

2. RKDG finite element method in a Lagrangian coordinate

Consider the gas dynamics equations for inviscid compressible fluid

$$u_t + f(u)_x = 0, \tag{1}$$

$$u(x,0) = u_0(x), \tag{2}$$

with

$$u = \begin{pmatrix} \rho \\ \rho v \\ \rho E \end{pmatrix}, \quad f(u) = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ v(\rho E + p) \end{pmatrix}. \tag{3}$$

Here ρ is the density, v is the velocity, E is the specific total energy, p is the pressure, which is related by

$$E = \frac{p}{\rho(\gamma-1)} + \frac{1}{2}v^2.$$

By introducing the Lagrangian coordinate ξ given by $\xi = \int_{x_0(t)}^{x} \rho(z,t)\,\mathrm{d}z$, where $x_0(t)$ denotes the Eulerian coordinate of a given fluid particle, the Euler equations (1) can be transformed into Lagrangian form

$$u_t + f(u)_\xi = 0, \tag{4}$$

$$u(\xi,0) = u_0(\xi), \tag{5}$$

with

$$u = \begin{pmatrix} \tau \\ v \\ E \end{pmatrix}, \quad f(u) = \begin{pmatrix} -v \\ p \\ pv \end{pmatrix}, \tag{6}$$

where $\tau = 1/\rho$ is the specific volume. The relation between the Lagrangian coordinate and the Eulerian coordinate is

$$\frac{\mathrm{d}x(\xi,t)}{\mathrm{d}t} = v(\xi,t). \tag{7}$$

2.1. Description of RKDG method

Assume the following mesh to cover the computational domain $[a,b]$, consisting of cells $I_j = [\xi_{j-1/2}, \xi_{j+1/2}]$, for $1 \le j \le N$, where

$$a = \xi_{1/2} < \xi_{3/2} < \cdots < \xi_{N+1/2} = b.$$

We denote

$$\xi_j = (\xi_{j-1/2} + \xi_{j+1/2})/2,$$
$$\Delta\xi_j = \xi_{j+1/2} - \xi_{j-1/2}, f^+_{j+1/2} = \lim_{\xi \to \xi^+_{j+1/2}} f(\xi),$$
$$f^-_{j+1/2} = \lim_{\xi \to \xi^-_{j+1/2}} f(\xi), \ 1 \le j \le N; \ h = \max_{1\le j\le N} \Delta\xi_j.$$

Assume the mesh is regular, namely there is a constant $c > 0$ independent of h such that

$$\Delta\xi_j \ge ch, \ 1 \le j \le N.$$

Multiplying Eqs. (4)–(6) by arbitrary smooth function w, m and integrating over I_j, after a simple formal integration by parts, the following weak forms are obtained

$$\int_{I_j} u_t w\,\mathrm{d}\xi + \int_{I_j} f(u) w_\xi\,\mathrm{d}\xi$$
$$- f_{j+1/2} w^-_{j+1/2} + f_{j-1/2} w^+_{j-1/2} = 0, \tag{8}$$

$$\int_{I_j} u(\xi,0) m\,\mathrm{d}\xi = \int_{I_j} u_0(\xi) m\,\mathrm{d}\xi. \tag{9}$$

The finite element space is defined as piecewise polynomials

$$V_h = V_h^k = \{v \in L^1(a,b) : v|_{I_j} \in P^k(I_j),$$
$$j = 1,\ldots,N\}, \tag{10}$$

where $P^k(I_j)$ denotes the set of polynomials of degree up to k defined on the cell I_j. The semi-discrete DG finite element

method for solving Eqs. (4)–(6) is: Find the unique functions $u_h \in V_h^k$, for all test functions w_h, $m_h \in V_h^k$, and $1 \le j \le N$, such that

$$\int_{I_j} (u_h)_t w_h \mathrm{d}\xi + \int_{I_j} f(u_h)(w_h)_\xi \mathrm{d}\xi - \hat{f}_{j+1/2}(w_h)^-_{j+1/2} + \hat{f}_{j-1/2}(w_h)^+_{j-1/2} = 0, \quad (11)$$

$$\int_{I_j} u_h(\xi,0) m_h \mathrm{d}\xi = \int_{I_j} u_0(\xi) m_h \mathrm{d}\xi. \quad (12)$$

Here, $\hat{f}_{j+1/2}$ are the numerical fluxes, which are the single-valued functions defined at the cell interfaces and dependent on the values of the numerical solutions u_h from both sides of the interface

$$\hat{f}_{j+1/2} = \hat{f}(u_h(\xi^-_{j+1/2},t), u_h(\xi^+_{j+1/2},t)).$$

In this paper, the numerical flux is taken as the Lax–Friedrichs (LF) flux

$$\hat{f}^{LF}_{j+1/2} = \frac{1}{2}[f(u_h(\xi^-_{j+1/2}) + f(u_h(\xi^+_{j+1/2},t)) - \alpha_{j+1/2}(u_h(\xi^+_{j+1/2},t) - u_h(\xi^-_{j+1/2},t)], \quad (13)$$

with

$$\alpha_{j+1/2} \equiv \alpha = \max_j\{\rho^\pm_{j+1/2} c^\pm_{j+1/2}\},$$

where c represents the sound velocity and satisfies $c = \sqrt{\gamma p/\rho}$. We assume that $\phi^l_j(\xi) = P_l(2(\xi-\xi_j)/\Delta\xi_j), 0 \le l \le k$, are the basis functions for finite element space V_h^k, where the first three polynomials are

$$P_0(\eta) = 1, \ P_1(\eta) = \eta, \ P_2(\eta) = \eta^2 - \frac{1}{3}. \quad (14)$$

Thus, the numerical solutions in every cell can be expressed as

$$u_h(\xi,t) = \sum_{l=0}^{k} u^l_j(t)\phi^l_j(\xi). \quad (15)$$

After space discretization, the ordinary differential equations (ODEs) about the undetermined coefficients $u^l_j(t), l = 0,\ldots,k$, are obtained

$$\frac{\mathrm{d}u_h}{\mathrm{d}t} = L_h(u_h), \quad \forall t \in [0,T], \quad u_h(0) = u_{h0}. \quad (16)$$

Let δt denote the time step. The above ODEs are solved by total variation diminishing (TVD) Runge–Kutta method.[32] The second order TVD Runge–Kutta formula is

$$u^{(1)} = u^n + \delta t L_h(u^n,t^n),$$
$$u^{n+1} = 0.5u^n + 0.5u^{(1)} + 0.5\delta t L_h(u^{(1)},t^n+\delta t).$$

The third order TVD Runge–Kutta formula is

$$u^{(1)} = u^n + \delta t L_h(u^n,t^n),$$
$$u^{(2)} = 0.75u^n + 0.25u^{(1)} + 0.25\delta t L_h(u^{(1)},t^n+\delta t),$$
$$u^{n+1} = \frac{1}{3}u^n + \frac{2}{3}u^{(2)} + \frac{2}{3}\delta t L_h(u^{(2)},t^n+\frac{1}{2}\delta t).$$

The numerical simulations are done with the quadratic element ($k = 2$) combined with the third order TVD Runge–Kutta method.

In order to maintain the stability and non-oscillatory property of the RKDG method for solving conservation laws with strong shocks, a limiter must be applied. The limiter adopted is described below in detail. We define

$$u^-_{j+1/2} = u^0_j + \tilde{u}_j, u^+_{j-1/2} = u^0_j - \tilde{\tilde{u}}_j.$$

Here

$$\tilde{u}_j = \sum_{l=1}^{k} u^l_j(t)\phi^l_j(\xi_{j+1/2}), \quad \tilde{\tilde{u}}_j = -\sum_{l=1}^{k} u^l_j(t)\phi^l_j(\xi_{j-1/2}).$$

Then $\tilde{u}_j$ and $\tilde{\tilde{u}}_j$ are modified by either the standard total variation diminishing (TVD) minmod limiter

$$\tilde{u}^{(\mathrm{mod})}_j = m(\tilde{u}_j, \Delta_+ u^{(0)}_j, \Delta_- u^{(0)}_j),$$
$$\tilde{\tilde{u}}^{(\mathrm{mod})}_j = m(\tilde{\tilde{u}}_j, \Delta_+ u^{(0)}_j, \Delta_- u^{(0)}_j),$$

where m is given by

$$m(a_1,a_2,\ldots,a_n) = \begin{cases} s\cdot\min_{1\le j\le n}|a_j|, & \text{if } \mathrm{sign}(a_1) = \mathrm{sign}(a_2) = \cdots = \mathrm{sign}(a_n) = s, \\ 0, & \text{otherwise}, \end{cases} \quad (17)$$

or the TVB modified minmod function

$$\tilde{m}(a_1,a_2,\ldots,a_n) = \begin{cases} a_1, & \text{if } |a_1| \le Mh^2, \\ m(a_1,a_2,\ldots,a_n), & \text{otherwise}, \end{cases} \quad (18)$$

where $M > 0$ is a constant depending on the solution of the given problem. In this paper, a limiter based on the characteristic field decomposition is used in each time step. The Jacobin matrix of Eqs. (4)–(6) is

$$\boldsymbol{A} = \begin{pmatrix} 0 & -1 & 0 \\ -p/\tau & -(\gamma-1)v/\tau & (\gamma-1)/\tau \\ -pv/\tau & p-(\gamma-1)v^2/\tau & (\gamma-1)/\tau v \end{pmatrix}. \quad (19)$$

The matrix consisting of the right eigenvectors of the Jacobin matrix is

$$\boldsymbol{R} = \begin{pmatrix} 1 & 1 & 1 \\ \rho c & 0 & -\rho c \\ \rho cv - p & p/(\gamma-1) & -\rho cv - p \end{pmatrix}. \quad (20)$$

The matrix consisting of the left eigenvectors of the Jacobin matrix is

$$\boldsymbol{L}=\frac{\gamma-1}{2\gamma p}\begin{pmatrix} p/(\gamma-1) & v+\gamma p/(\gamma-1)\rho c & -1 \\ 2p & -2v & 2 \\ p/(\gamma-1) & v-\gamma p/(\gamma-1)\rho c & -1 \end{pmatrix}. \quad (21)$$

The eigenvalues are $\lambda_1=-\rho c,\lambda_2=0,\lambda_3=\rho c$, respectively. We give the details of limiting the vector $\tilde{u}_j$ in the following three steps. Firstly, transform the three vectors $\tilde{u},\Delta_+u_j^{(0)}$, and $\Delta_-u_j^{(0)}$ to the characteristic field by left-multiplying with the matrix $\boldsymbol{L}$. Secondly, apply the scalar limiter to each component of the transformed vectors $\boldsymbol{L}\tilde{u},\boldsymbol{L}\Delta_+u_j^{(0)}$, and $\boldsymbol{L}\Delta_-u_j^{(0)}$. Finally, transform the limited version of the vector back to the original space by left-multiplying with the matrix $\boldsymbol{R}$.

In order to identify the location of every fluid we studied, the equation

$$\frac{\mathrm{d}x(\xi,t)}{\mathrm{d}t}=v(\xi,t) \quad (22)$$

should be discretized. Integrating Eq. (22) in time interval $[t^n,t^{n+1}]$, we can obtain

$$x(\xi,t^{n+1})=x(\xi,t^n)+\int_{t^n}^{t^{n+1}}v(\xi,t)\,\mathrm{d}t. \quad (23)$$

By using the appropriate numerical integration formula, the location of the fluid in time t^{n+1} can be obtained. In all the following numerical experiments, the formula

$$x(\xi,t^{n+1})=x(\xi,t^n)+\frac{\delta t}{2}(v(\xi,t^n)+v(\xi t^{n+1})) \quad (24)$$

is used. If the fluid we examined is located within the cells, then the fluid velocity is already known. If the fluid we examined is located at the interface of the cells, the Roe average is used to derive the velocity, which is

$$v_{j+1/2}=\frac{\sqrt{\rho^-_{i+1/2}}v^-_{i+1/2}+\sqrt{\rho^+_{i+1/2}}v^+_{i+1/2}}{\sqrt{\rho^-_{i+1/2}}+\sqrt{\rho^+_{i+1/2}}}. \quad (25)$$

2.2. Description of treatment of interface cell

In general, algorithms proposed for solving two-medium compressible flow consist of two parts. One is the method for solving the single-medium flow and the other is to treat the interface of the two fluids. In Refs. [33], [34], and [35], the third order essential non-oscillatory (ENO) methods are used together with the GFM for treating the interface, while the Godunov type or monotone upwind schemes for conservation laws (MUSCL) are used in Refs. [36]–[42]. In this paper, our intention is to investigate using the RKDG finite element method for multi-medium flow simulations in a one-dimensional case. Similar to the above-mentioned algorithms developed for multi-medium compressible flows, the proposed method also consists of two parts. One is the usual RKDG algorithm applicable for the flow field away from the material interface. The other is the newly developed DG technique for treating the moving interface. For calculation convenience, we place one cell point on the initial interface of the two fluids. Assume the interface is located at the node x_{N_1}. Considering the "compactness" advantage of the RKDG method, we only need to take care of the cells $[x_{N_1-1},x_{N_1}]$ and $[x_{N_1},x_{N_1+1}]$. The main difficulty is to give the flux at the node x_{N_1} appropriately. In this situation the Lax–Friedrichs flux is not applicable. We use the approximate Riemann solver to derive the flux at the interface. One is the linear Riemann solver and the other is the nonlinear Riemann solver. The related Riemann problem is the gas dynamic equation in the Lagrangian coordinate with the following initial conditions

$$(\rho,u,p,\gamma)=\begin{cases}(\rho^-_{N_1},u^-_{N_1},p^-_{N_1},\gamma_1), & \text{if } \eta\le 0,\\ (\rho^+_{N_1},u^+_{N_1},p^+_{N_1},\gamma_2), & \text{if } \eta\ge 0.\end{cases} \quad (26)$$

Our linear Riemann solver is based on the characteristic formulations

$$\frac{\partial u}{\partial t}+\frac{1}{\rho c}\frac{\partial p}{\partial t}+(\rho c)\left(\frac{\partial u}{\partial \eta}+\frac{1}{\rho c}\frac{\partial p}{\partial \eta}\right)=0, \quad (27)$$

$$\frac{\partial u}{\partial t}-\frac{1}{\rho c}\frac{\partial p}{\partial t}-(\rho c)\left(\frac{\partial u}{\partial \eta}-\frac{1}{\rho c}\frac{\partial p}{\partial \eta}\right)=0. \quad (28)$$

Along the characteristic, the above two formulas can be approximated by

$$(u^*-u^{\mathrm{L}})+\frac{1}{\rho^{\mathrm{L}}c^{\mathrm{L}}}(p^*-p^{\mathrm{L}})=0, \quad (29)$$

$$(u^*-u^{\mathrm{R}})-\frac{1}{\rho^{\mathrm{R}}c^{\mathrm{R}}}(p^*-p^{\mathrm{R}})=0, \quad (30)$$

where u^* and p^* are the corresponding velocity and pressure in the contact discontinuous region.[43] From the above two formulas we can get the u^* and p^*

$$p^*=\frac{u^{\mathrm{L}}-u^{\mathrm{R}}+\frac{1}{\rho^{\mathrm{R}}c^{\mathrm{R}}}p^{\mathrm{R}}+\frac{1}{\rho^{\mathrm{L}}c^{\mathrm{L}}}p^{\mathrm{L}}}{\frac{1}{\rho^{\mathrm{L}}c^{\mathrm{L}}}+\frac{1}{\rho^{\mathrm{R}}c^{\mathrm{R}}}}, \quad (31)$$

$$u^*=u^{\mathrm{L}}-\frac{1}{\rho^{\mathrm{L}}c^{\mathrm{L}}}(p^*-p^{\mathrm{L}}). \quad (32)$$

The fluxes on the node x_{N_1} can be written as

$$\widehat{f}^*_{N_1}=[-u^*,p^*,p^*u^*]^{\mathrm{T}}. \quad (33)$$

The linear Riemann solver is easy to use and the numerical results will be given in the following parts. There are some oscillations in the press and velocity in the vicinity of the interface. In order to suppress such oscillations, we must use the more accurate Riemann solvers. Here we use the nonlinear Riemann solver which is based on characteristic formulations as well as conservation laws across any discontinuous wave. There is no need to find the analytic solution completely. We only need to find the velocity and press in the contact discontinuous region.

For left waves, left shock waves, and left rarefaction waves, there is

$$(u^* - u^{\mathrm{L}}) + \frac{p^* - p^{\mathrm{L}}}{a^{\mathrm{L}}} = 0. \tag{34}$$

For right waves, there is

$$(u^* - u^{\mathrm{R}}) - \frac{p^* - p^{\mathrm{R}}}{a^{\mathrm{R}}} = 0. \tag{35}$$

In the above two formulas, there is

$$a^k = \begin{cases} -\dfrac{p^* - p^k}{\dfrac{2c^k}{\gamma_k - 1}\left[1 - \left(\dfrac{p^*}{p^k}\right)^{(\gamma_k - 1)/2\gamma_k}\right]}, & \text{if } p^* < p^k, \\ \rho^k c^k \sqrt{\dfrac{\gamma_k + 1}{\gamma_k}\left(\dfrac{p^*}{p^k}\right) + \dfrac{\gamma_k - 1}{\gamma_k}}, & \text{if } p^* > p^k, \end{cases}$$
$$k = \mathrm{L}, \mathrm{R}. \tag{36}$$

We use the simple iteration to calculate p^* and u^*. First, we can get from Eqs. (34) and (35)

$$u^* = \frac{a^{\mathrm{L}} u^{\mathrm{L}} + a^{\mathrm{R}} u^{\mathrm{R}} - (p^{\mathrm{R}} - p^{\mathrm{L}})}{a^{\mathrm{L}} + a^{\mathrm{R}}}, \tag{37}$$

$$p^* = \frac{a^{\mathrm{L}} p^{\mathrm{R}} + a^{\mathrm{R}} p^{\mathrm{L}} - a^{\mathrm{L}} a^{\mathrm{R}} (u^{\mathrm{R}} - u^{\mathrm{L}})}{a^{\mathrm{L}} + a^{\mathrm{R}}}. \tag{38}$$

We can calculate p^* with the simple iteration, that is

$$\begin{aligned} p^{*,n+1} &= p^* \\ &= \frac{a^{L,n} p^{\mathrm{R}} + a^{R,n} p^{\mathrm{L}} - a^{L,n} a^{R,n} (u^{\mathrm{R}} - u^{\mathrm{L}})}{a^{L,n} + a^{R,n}}, \end{aligned} \tag{39}$$

where

$$a^{k,n} = a^k(p^{*,n}, p^k, \rho^k), \quad k = \mathrm{L}, \mathrm{R}. \tag{40}$$

The initial value for iteration can be taken as

$$a^{L,0} = \rho^{\mathrm{L}} c^{\mathrm{L}}, \quad a^{R,0} = \rho^{\mathrm{R}} c^{\mathrm{R}}. \tag{41}$$

When we get p^* and u^*, then the fluxes can be taken as

$$\widehat{f}^*_{N_1} = [-u^*, p^*, p^* u^*]^{\mathrm{T}}. \tag{42}$$

The numerical results for the two-medium fluid simulations with nonlinear Riemann solver can be found in the following section.

3. Numerical results

In this section, some numerical experiments for one-dimensional inviscid compressible gas dynamic equations are performed.

3.1. Theoretical analysis and error estimation for discontinuous Galerkin finite element method

We will give some theoretical analysis and error estimation for solving one-dimensional time-dependent conservation laws. The one-dimensional conservation law is

$$s_t + f(s)_\xi = 0. \tag{43}$$

Following the derivation of the RKDG finite element schemes for solving Eqs. (4)–(6), the semi-discrete discontinuous Galerkin (DG) finite element method for solving Eq. (43) is: Find the unique function $s_h \in V_h^k$, for all test functions $w_h \in V_h^k$ and $1 \le j \le N$, such that

$$\begin{aligned} &\int_{I_j} (s_h)_t w_h \,\mathrm{d}\xi + \int_{I_j} f(s_h)(w_h)_\xi \,\mathrm{d}\xi \\ &- \hat{f}_{j+1/2}(w_h)^-_{j+1/2} + \hat{f}_{j-1/2}(w_h)^+_{j-1/2} = 0. \end{aligned} \tag{44}$$

Here, $\hat{f}_{j+1/2}$ are the numerical fluxes, which satisfy the following three conditions: consistency, continuity, and monotonicity. To save space, we will only give some theoretical results. The details can be found in Refs. [15]–[17].

Proposition 1 The solution s_h to the semi-discrete DG scheme (44) satisfies the following cell entropy inequality

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{I_j} U(s_h) \,\mathrm{d}\xi + \hat{F}_{j+1/2} - \hat{F}_{j-1/2} \le 0, \tag{45}$$

for the square entropy $U(s) = s^2/2$, for some consistent entropy flux

$$\hat{F}_{j+1/2} = \hat{F}(s_h(\xi^-_{j+1/2}, t), s_h(\xi^+_{j+1/2}, t))$$

satisfying $\hat{F}(s,s) = F(s)$.

Proposition 2 For periodic or compacted boundary conditions, the solution s_h to the semi-discrete scheme (44) satisfies the following L^2 stability

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_a^b (s_h)^2 \,\mathrm{d}\xi \le 0. \tag{46}$$

Starting from the semi-discrete scheme (44), we first use forward Euler time discretization to get a full-discrete scheme. The scheme is: Find the unique function $s_h^{n+1} \in V_h^k$, for all test functions $w_h \in V_h^k$ and $1 \le j \le N$, such that

$$\begin{aligned} &\int_{I_j} \frac{s_h^{n+1} - s_h^n}{\delta t} w_h \,\mathrm{d}\xi + \int_{I_j} f(s_h)(w_h)_\xi \,\mathrm{d}\xi \\ &- \hat{f}_{j+1/2}(w_h)^-_{j+1/2} + \hat{f}_{j-1/2}(w_h)^+_{j-1/2} = 0. \end{aligned} \tag{47}$$

Here, s_h^{n+1} represents the numerical solution at time $t = t^{n+1}$ and δt represents the time step. At the beginning of every time evolution, the limiter should be used to pre-process the s_h^n.

Proposition 3 For periodic or compactly supported boundary conditions, the solution s_h^n of scheme (47), with the

pre-process by the limiter, is total variation diminishing in the means (TVDM), that is

$$TVM(s_h^n) \le TVM(s_h^{n+1}), \tag{48}$$

where $TVM(s_h^n) = \sum_i |\bar{s}_{i+1}^n - \bar{s}_i^n|$ and $\bar{s}_i^n$ represents the cell average of numerical solution s_h^n.

Proposition 4 Under the same conditions as those in Proposition 3, the solution s_h^n of the scheme (47), with the Euler forward time discretization replaced by TVD Runge–Kutta time discretization, is TVDM.

Proposition 5 The solution s_h of the semi-discrete scheme (44) for solving the linear equation

$$u_t + u_\xi = 0$$

with a smooth solution s satisfying the error estimate

$$\int_a^b (s - s_h)^2 \,\mathrm{d}\xi \le Ch^{2k+2}, \tag{49}$$

where C depends on s and its derivatives but is independent of h.

From the above five propositions, we can conclude that, for the linear problem, the semi-discrete scheme can achieve its optimal convergence order. Furthermore it can suppress the non-physical oscillations. All these advantages lead us to construct the RKDG schemes for solving the one-dimensional Euler equations in the Lagrangian coordinate.

3.2. Numerical tests for single-medium flows

Lagrangian computation and the ideal gas with $\gamma = 1.4$ are used in the following tests.

Example 1 This numerical example is used for the accuracy test. We solve the Euler equation in the Lagrangian coordinate with the following initial condition

$$\rho(x,0) = 1 + 0.2\sin(\pi x),\ u(x,0) = 1,\ p(x,0) = 1,$$

and 2-periodic boundary conditions. The exact solutions are

$$\rho(x,t) = 1 + 0.2\sin(\pi(x-t)),\ u(x,t) = 1,\ p(x,t) = 1.$$

The L^∞ error for ρ is defined as follows:

$$\|\text{error}\|_{L^\infty} = \max_{i=1}^{N}(|\rho(\xi_i,T) - \rho_h(\xi_i,T)|), \tag{50}$$

where ρ and ρ_h express the corresponding exact and numerical solutions for density. The numerical order of accuracy is calculated as follows:

$$\text{order} = \frac{\log\left(\dfrac{\|\text{error}\|_{L^\infty,2h}}{\|\text{error}\|_{L^\infty,h}}\right)}{\log 2}. \tag{51}$$

First, we compute the solution up to $t = 2$. The errors and numerical order of accuracy are shown in Table 1. We can see that the schemes behave very well. Furthermore, the numerical schemes can work for quite a long time. Figure 1 gives the comparisons of numerical and exact solutions at time $t = 100$ with different meshes.

Table 1. The errors and numerical orders of accuracy of density ρ at $t = 2$.

Grids	$\|\text{error}\|_{L^\infty}$	L_∞ order
10	5.94088×10^{-3}	
20	1.61615×10^{-3}	1.87812
40	4.09642×10^{-4}	1.98013
80	1.02712×10^{-4}	1.99576
160	2.56961×10^{-5}	1.99899
320	6.42515×10^{-6}	1.99975

Example 2 We consider the Riemann problems of the gas dynamics equation for a polytropic gas. Two sets of initial conditions are used. One is proposed by Sod:

$$(\rho_\mathrm{L}, u_\mathrm{L}, p_\mathrm{L}) = (1.0, 0, 1.0),$$
$$(\rho_\mathrm{R}, u_\mathrm{R}, p_\mathrm{R}) = (0.125, 0, 0.10).$$

The other is used by Lax:

$$(\rho_\mathrm{L}, u_\mathrm{L}, p_\mathrm{L}) = (0.445, 0.698, 3.528),$$
$$(\rho_\mathrm{R}, u_\mathrm{R}, p_\mathrm{R}) = (0.5, 0, 0.571).$$

The initial computational domain is $[0,1]$ and $[0,2]$ for Sod and Lax problems. The initial discontinuity is located at $x_0 = 0.5$ and $x_0 = 1$ for Sod and Lax problems respectively. The transmissive boundary condition is applied at both ends of the computational domain. The exact solution of this example consists of a left rarefaction wave, a right-travelling shock wave, and a stationary contact discontinuity. Figures 2 and 3 show the numerical results with 400 initially uniform cells at $t = 0.2$ and $t = 0.13$ for Sod and Lax problems respectively.

Example 3[44] This example is used to test the ability of the method to resolve slowly-moving or stationary contact discontinuity. Its initial condition is as follows:

$$(\rho_\mathrm{L}, u_\mathrm{L}, p_\mathrm{L}) = (1.0, -19.59745, 1000.0),$$
$$(\rho_\mathrm{R}, u_\mathrm{R}, p_\mathrm{R}) = (1.0, -19.59745, 0.01).$$

The initial discontinuity is located at $x_0 = 0.8$. The computational domain is $[0,2]$. The transmissive boundary condition is applied at both $x = 0$ and $x = 2$. The exact solution of this example consists of a left rarefaction wave, a right-travelling shock wave, and a stationary contact discontinuity. Figure 4 shows the numerical results with 400 initially uniform cells at $t = 0.012$.

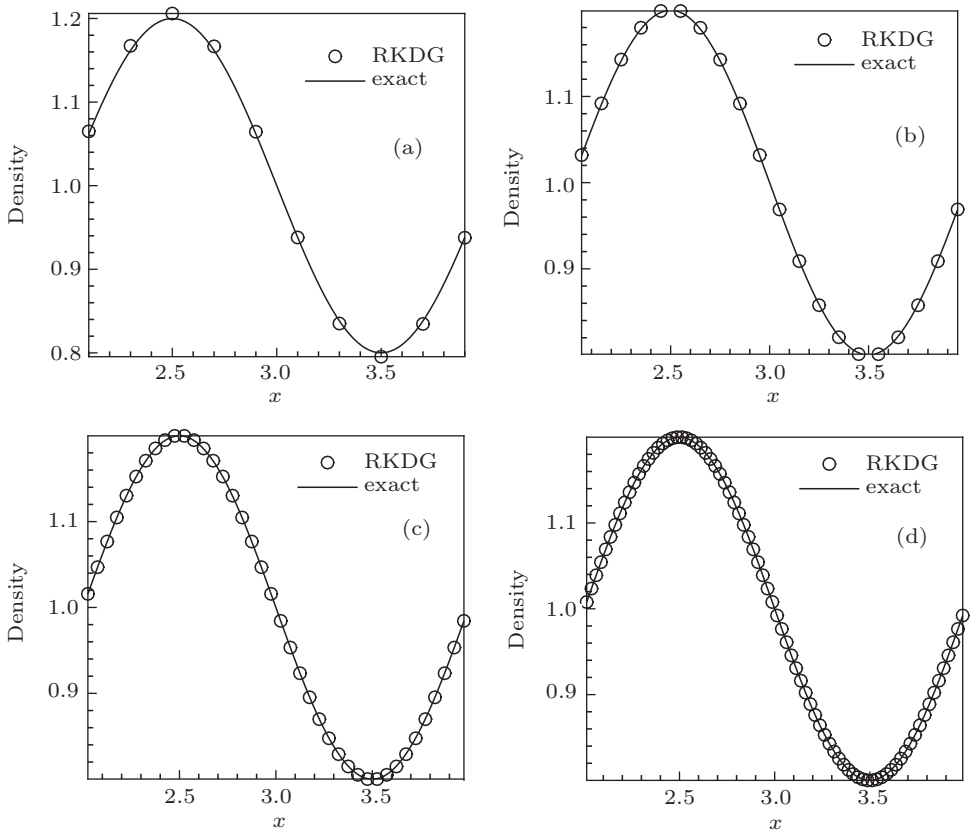


Fig. 1. Comparison between the RKDG solution and the exact solution at time 2.0 with different grids: (a) 10 cells, (b) 20 cells, (c) 40 cells, and (d) 80 cells.

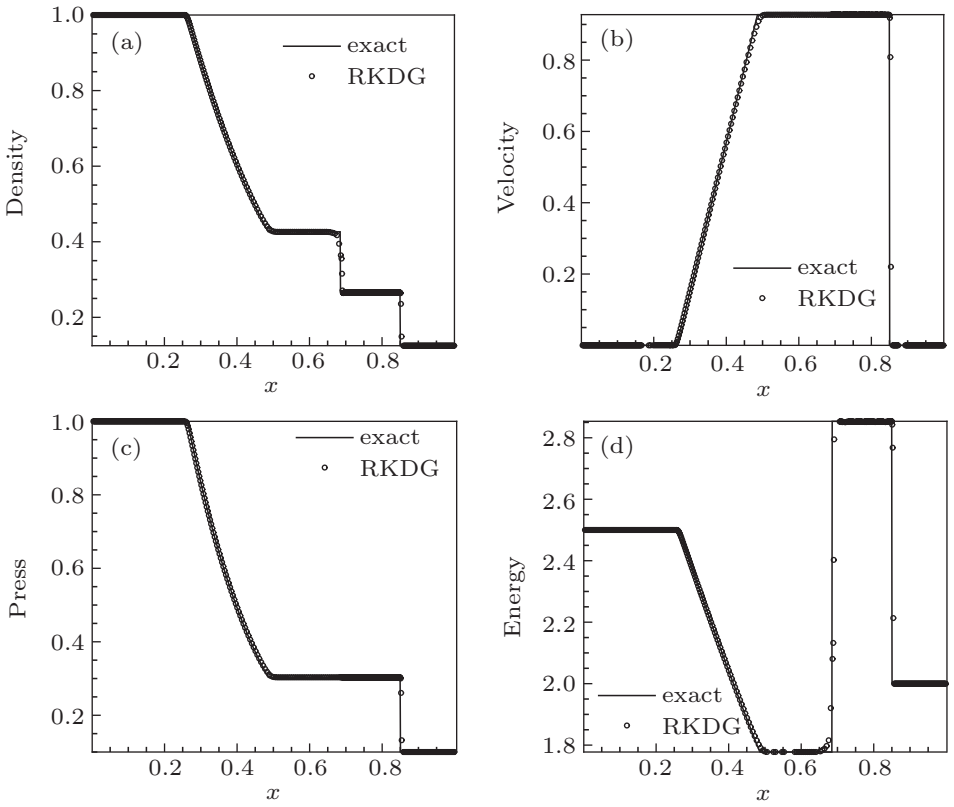


Fig. 2. Comparison between the RKDG solution and the exact solution at time 0.2 for Sod problem in example 2: (a) density, (b) velocity, (c) press, and (d) energy.

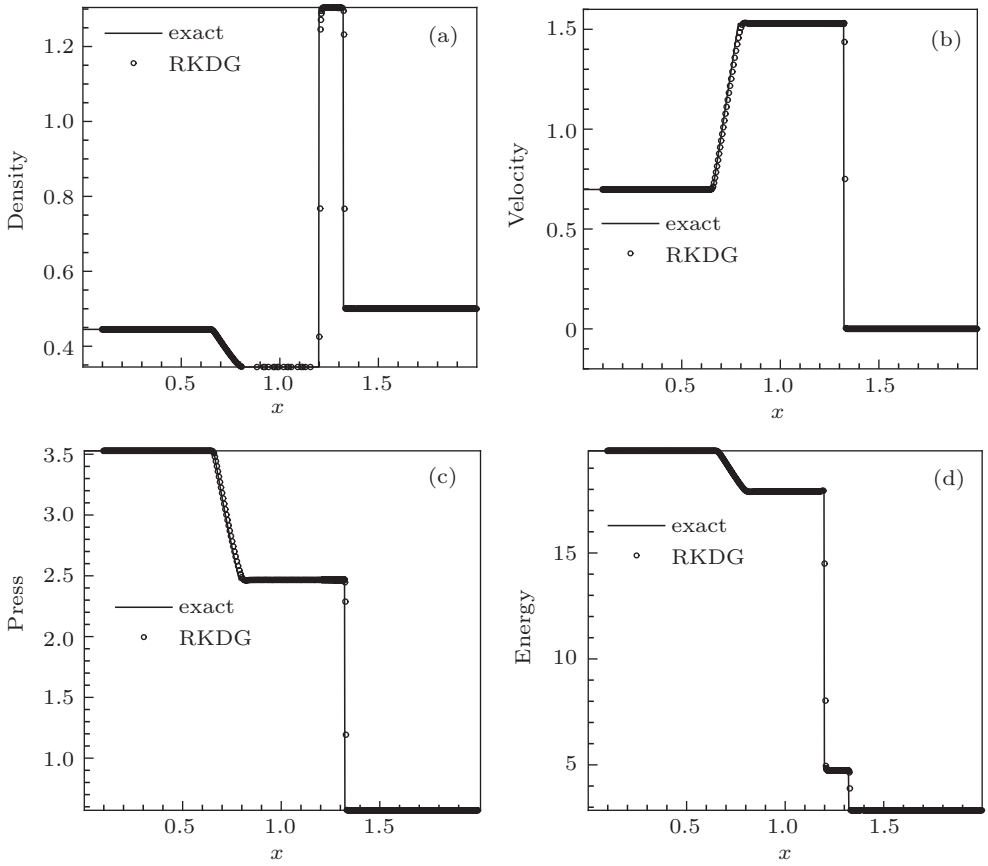


Fig. 3. Comparison between the RKDG solution and the exact solution at time 0.13 for Lax problem in example 2.

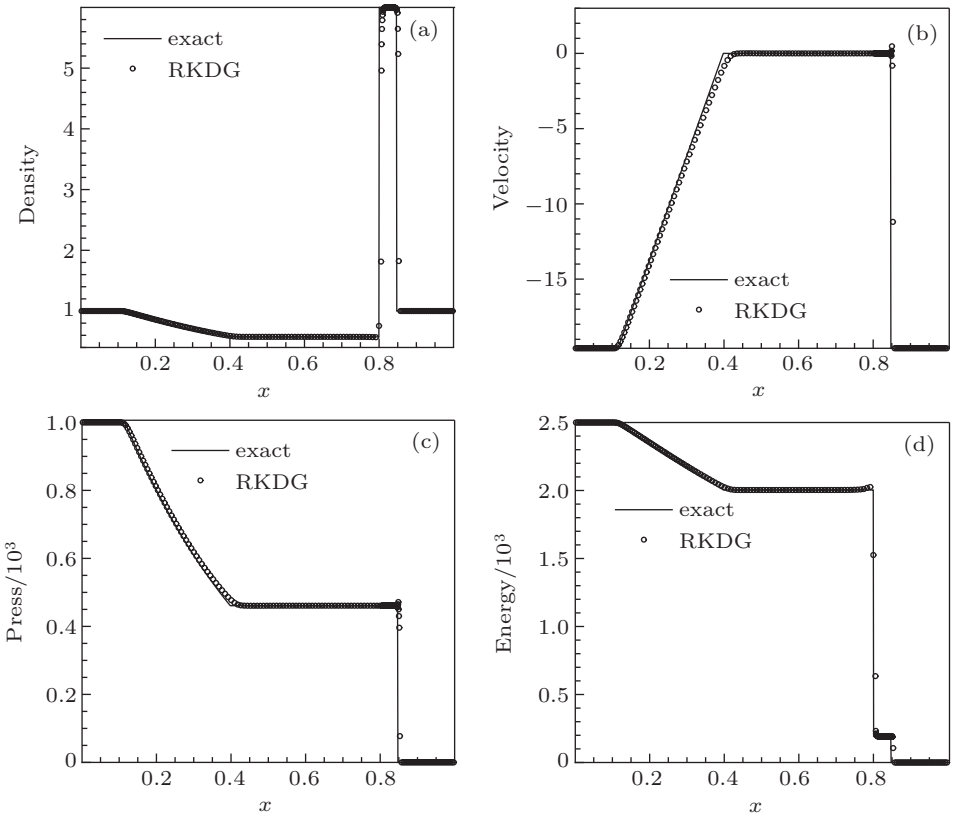


Fig. 4. Comparison between the RKDG solution and the exact solution at time 0.012 in example 3.

Example 4[44] (123 problem) The initial condition of this example is

$$(\rho_{\rm L}, u_{\rm L}, p_{\rm L}) = (1.0, -2.0, 0.4),$$
$$(\rho_{\rm R}, u_{\rm R}, p_{\rm R}) = (1.0, 2.0, 0.4).$$

The initial discontinuity is located at $x_0 = 0.5$. It has a solution consisting of two symmetric rarefaction waves and a trivial contact wave of zero speed. The star region between the nonlinear wave is close to vacuum, which makes this problem a suitable test for assessing the performance of numerical methods for low-density flows. The computational domain is $[0, 2]$. Figure 5 shows the numerical results with 400 initially uniform cells at $t = 0.15$. For this example, it is difficult to get the right internal energy through most of the numerical methods.[44] In Fig. 5, we compare the performance of the RKDG method in the Lagrangian coordinate and Runge–Kutta control volume discontinuous finite element method (RKCVDFEM)[45] with 400 uniform cells. To the best of our knowledge, only the high accuracy and high resolution numerical method can simulate the internal energy better. RKDG finite element method in the Lagrangian coordinate behaves better than most of the numerical methods in Ref. [44]. Figure 6 gives the numerical results with four schemes: Godunov scheme with HLL Riemann solver proposed by Harten, Lax and van Leer; Lax–Friedrichs scheme; Liou–Steffen scheme; weighted average flux (WAF) scheme. The derivation and the implementation of the above four schemes can be found in Ref. [44].

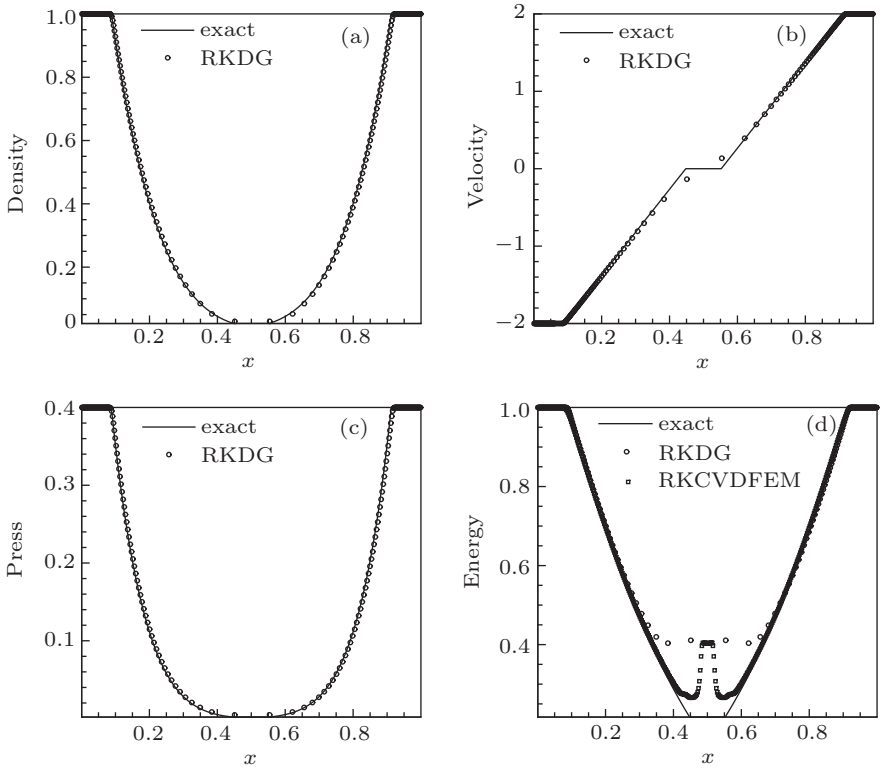


Fig. 5. Comparison between the RKDG solution and the exact solution at time 0.15 for 123 problem in example 4.

Example 5[7] (Two interacting blast waves problem) The initial conditions are

$$\rho = 1, \quad u = 1, \quad p = \begin{cases} 10^3, & 0 < x < 0.1, \\ 10^{-2}, & 0.1 < x < 0.9, \\ 10^2, & 0.9 < x < 1.0. \end{cases}$$

The reflective boundary condition is applied at both $x = 0$ and $x = 1$. Figure 7 shows the numerical results at $t = 0.038$ for different initial uniform cells. In Fig. 7, we use the numerical solution of RKCVDFEM with 2000 uniform cells as a referenced solution.

Comparing the numerical solution to the "exact solution" of Woodward and Colella in Ref. [9], we find that it shows all the important features of the various interactions and thus can be considered as a "converged" solution. Our numerical results can be compared with the numerical results in some references.[17,46]

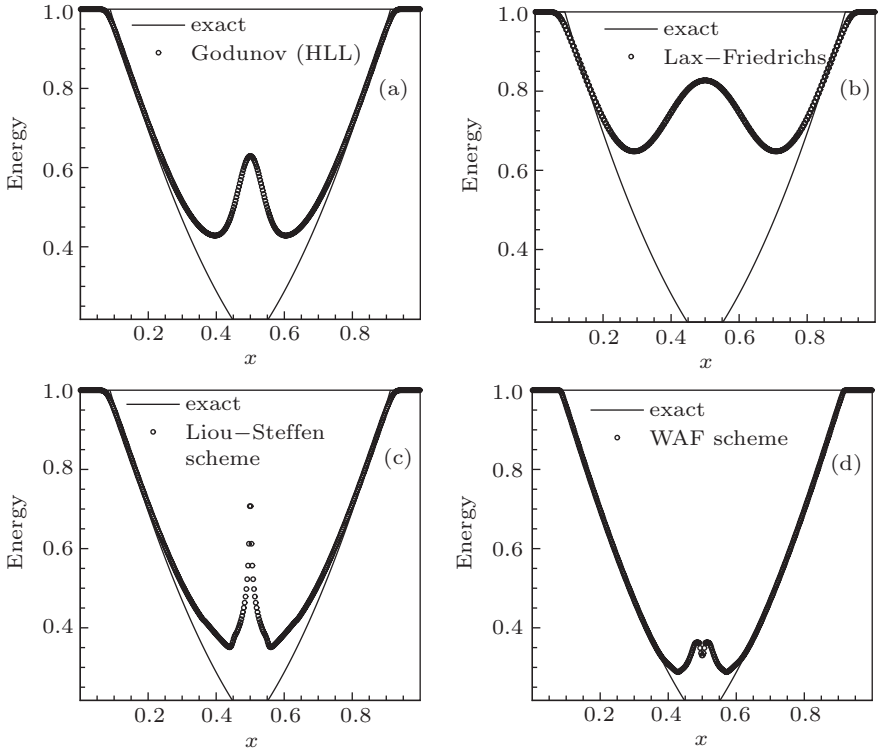


Fig. 6. Comparison between the exact solution and other numerical schemes at time 0.15 for 123 problem in example 4.

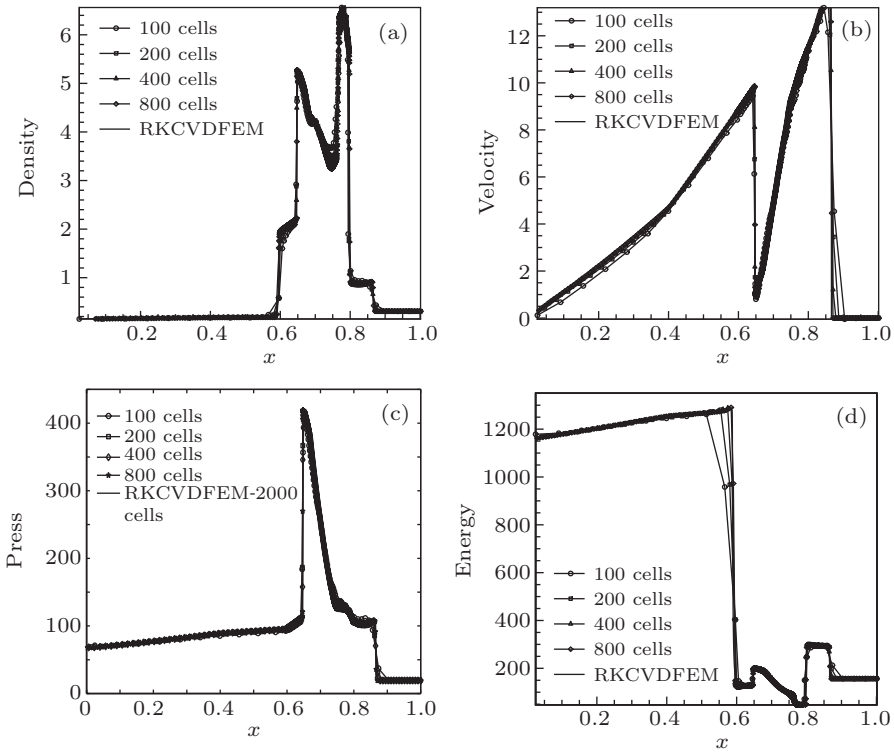


Fig. 7. Comparison between the RKDG solution and the exact solution at time 0.038 for two interacting blast waves problem in example 5.

Example 6 (Shock entropy wave interaction) To demonstrate the advantage of higher order methods, we use the Euler equations with initial condition

$$(\rho_{\mathrm{L}}, u_{\mathrm{L}}, p_{\mathrm{L}}) = (3.85714, 2.629369, 10.33333), \quad x < 4,$$
$$(\rho_{\mathrm{R}}, u_{\mathrm{R}}, p_{\mathrm{R}}) = (1.0 + \varepsilon \sin(kx), 0, 1.0), \quad x \geq -4,$$

where ε and k are the amplitude and wave number of the en-

tropy wave. In our test, we take $\varepsilon = 0.2$ and $k = 5$. The final time is $t = 1.8$. This problem is very suitable for testing the advantage of a high order scheme when the solution contains both shock and completely smooth region structures. The initial computational domain is $[-10, 5]$. The numerical results for density with different meshes are showed in Fig. 8.

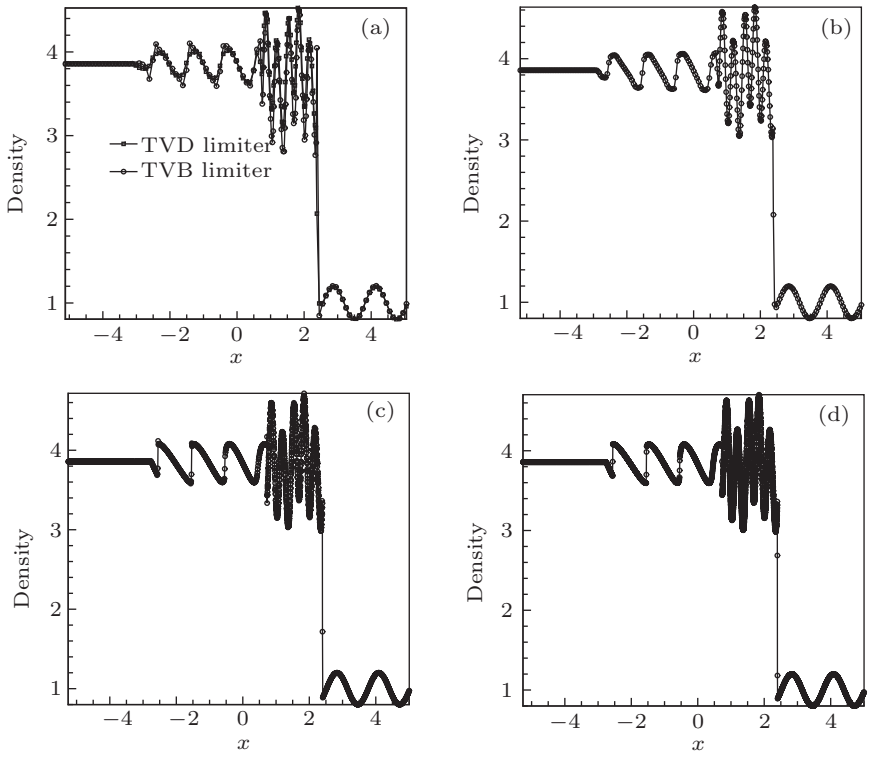


Fig. 8. Density at time 1.8 with different cells and limiters for shock entropy wave interaction problem in example 6: (a) 150 cells, (b) 300 cells and TVD limiter, (c) 1500 cells and TVB limiter ($M = 300$), (d) 3000 cells and TVB limiter ($M = 300$).

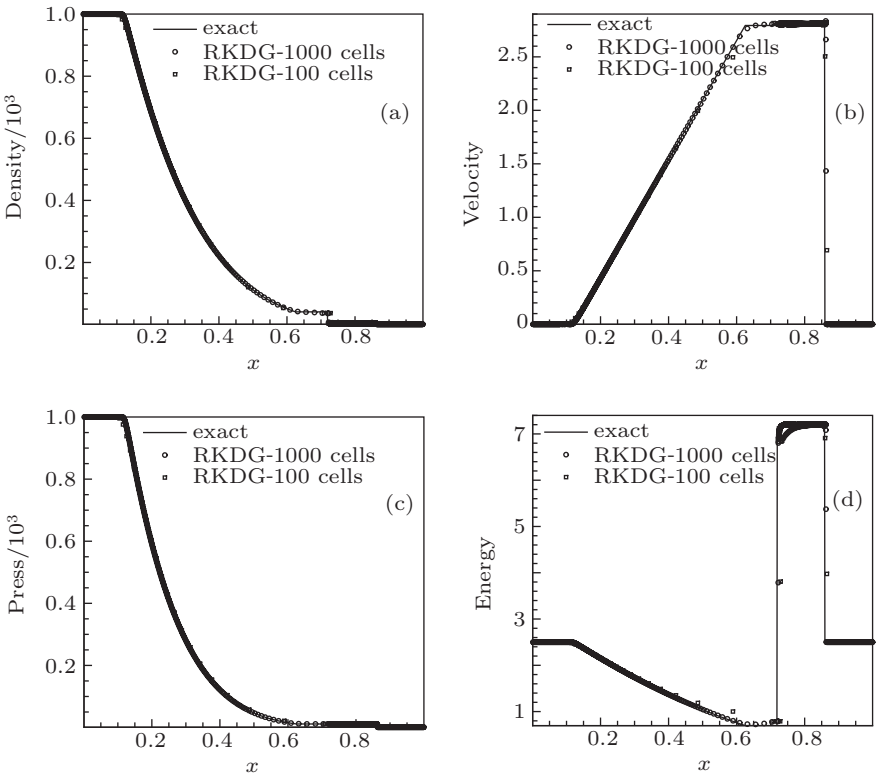


Fig. 9. Comparison between the RKDG solution and the exact solution at time 0.15 in example 7.

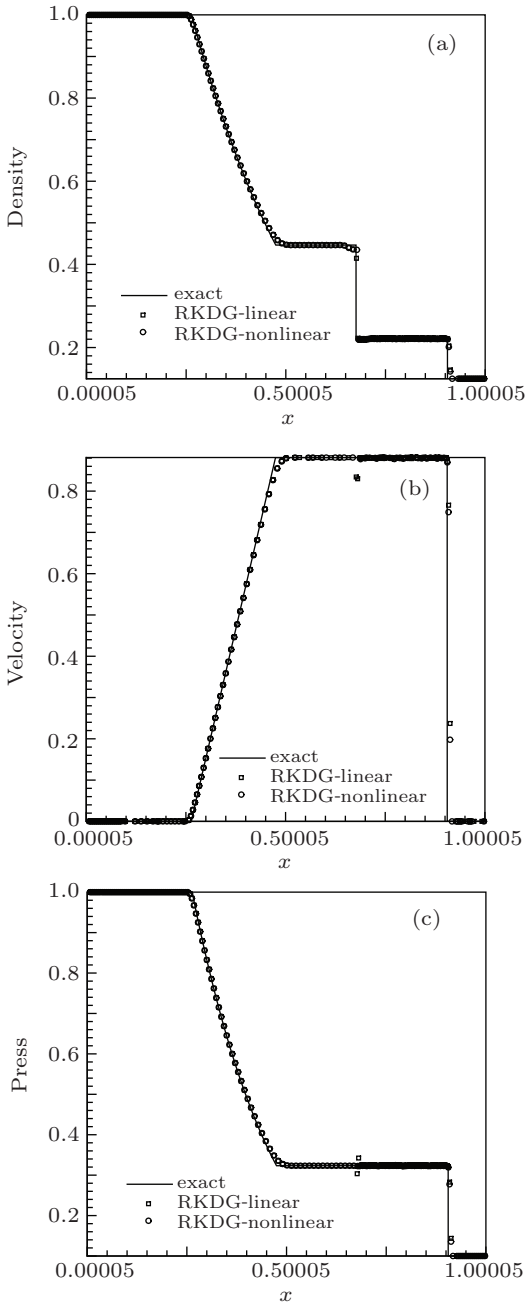


Fig. 10. Comparison between the RKDG solution and the exact solution at time 0.2 in example 8.

Example 7[47] This example to be considered is a Riemann problem subject to the following initial data

$$(\rho_{\rm L}, u_{\rm L}, p_{\rm L}) = (1000, 0, 1000),\ (\rho_{\rm R}, u_{\rm R}, p_{\rm R}) = (1, 0, 1).$$

The initial discontinuity is located at $x_0 = 0.3$. The computational domain is $[0,1]$. The transmissive boundary condition is applied at both $x = 0$ and $x = 1$. This is a problem with high density and high pressure ratios given initially. As a result, there is a very strong rarefaction wave generated in the high pressure region once the diaphragm is removed. Numerical investigation for the popular high resolution conservative schemes when applied to inviscid, compressible, perfect gas flows with an initial high density ratio as well as a high pressure ratio has been done by Tang in Ref. [47]. The results show that they work very inefficiently and may give inaccurate numerical results even over a very fine mesh when applied to such a problem. Figure 9 shows the numerical results with 1000 and 100 initially uniform cells and TVD limiter at $t = 0.15$. Our numerical methods with 1000 initially uniform cells work well.

3.3. Numerical tests for two-medium flows

The numerical results with RKDG finite element method combined with linear and nonlinear Riemann solvers are shown in this part.

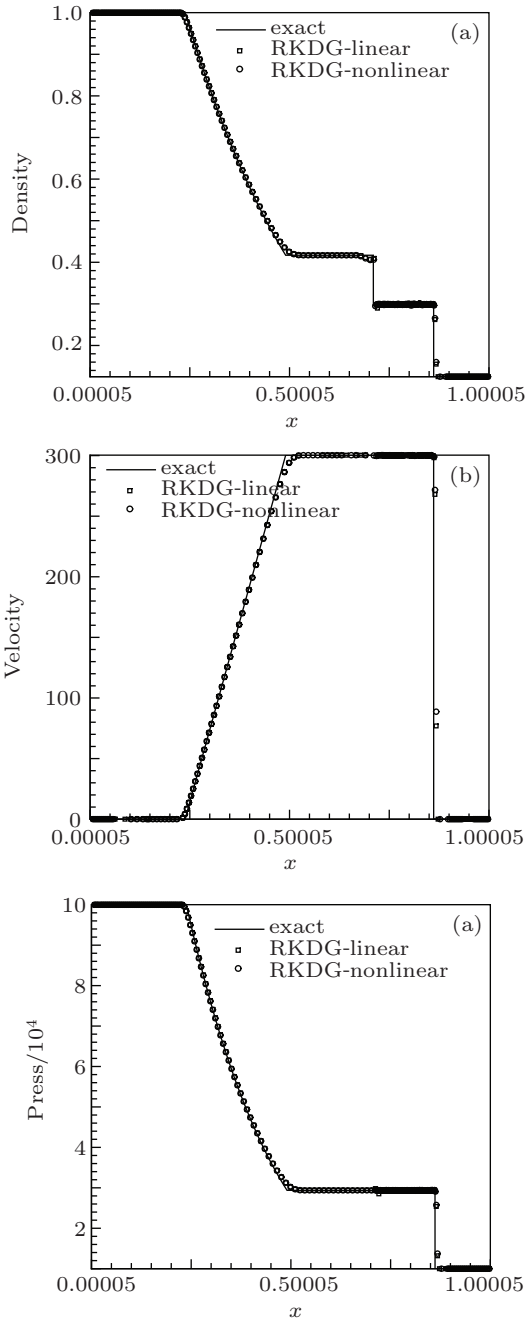


Fig. 11. Comparison between the RKDG solution and the exact solution at time 0.0007 for air-helium shock tube problem in example 9.

Example 8 We solve the one-dimensional nonlinear system of Euler equations (4). We use the following Riemann

initial condition:

$$(\rho_{\rm L}, u_{\rm L}, p_{\rm L}, \gamma_1) = (1.0,,0,1.0,1.4), \quad \text{for } x \le 0.5;$$
$$(\rho_{\rm R}, u_{\rm R}, p_{\rm R}, \gamma_2) = (0.125,0,0.1,1.9), \quad \text{for } x > 0.5.$$

The computed density, velocity, and press are plotted at $t = 0.2$ against the exact solution in Fig. 10.

Example 9[21] This is an air-helium shock tube problem with the initial conditions as

$$(\rho_{\rm L}, u_{\rm L}, p_{\rm L}, \gamma_1) = (1.0,0,1.0\times10^5,1.4), \quad \text{for } x \le 0.5;$$
$$(\rho_{\rm R}, u_{\rm R}, p_{\rm R}, \gamma_2) = (0.125,0,1.0\times10^4,1.2), \quad \text{for } x > 0.5.$$

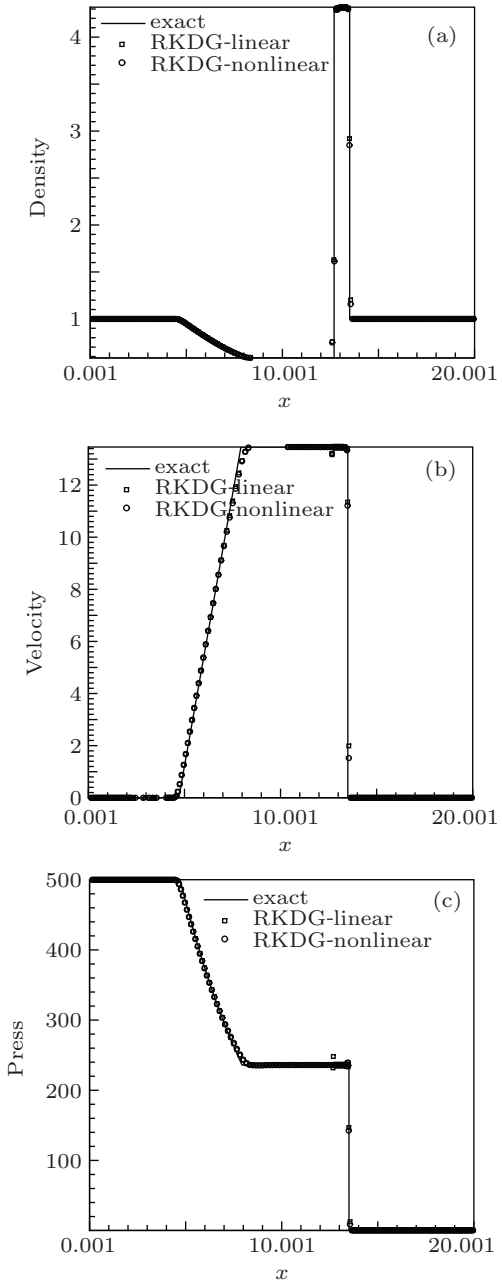


Fig. 12. Comparison between the RKDG solution and the exact solution at time 0.2 for stiff gas shock tube problem in example 9.

The computed density, velocity, and press are plotted at $t = 0.0007$ against the exact solution in Fig. 11.

Example 10 This is a stiff gas shock tube problem with the initial conditions as

$$(\rho_{\rm L}, u_{\rm L}, p_{\rm L}, \gamma_1) = (1.0,,0,500,1.4), \quad \text{for } x \le 0.5;$$
$$(\rho_{\rm R}, u_{\rm R}, p_{\rm R}, \gamma_2) = (1.0,0,0.2,1.6), \quad \text{for } x > 0.5.$$

The computed density, velocity, and press are plotted at $t = 0.2$ against the exact solution in Fig. 12.

In the above three examples, RKDG finite element method combined with the linear Riemann solver induces some oscillation in the contact discontinuity region. It cannot be adopted for practice work. However, RKDG finite element method combined with the nonlinear Riemann solver behaves better. It can suppress the oscillation and distinctly capture the contact discontinuity. We believe that the RKDG finite element method in the Lagrangian coordinate combined with more accurate Riemann solver will work better.

4. Conclusions

In this paper, the RKDG finite element method for solving the one-dimensional inviscid compressible gas dynamic equations in the Lagrangian coordinate is developed. Several single-medium and multi-medium examples are given to demonstrate the accuracy and reliability of the algorithm. The proposed method can capture the interface distinctly. Actually the interface is just the grid point. The schemes can be implemented easily. It has the potential to be used in more complex fluid interaction simulations.

References

[1] Zayed E M E and Arnous A H 2012 *Chin. Phys. Lett.* **29** 080203
[2] Liu Y Q, Chen D Y and Hu C 2012 *Chin. Phys. Lett.* **29** 080202
[3] Sun J Q, Luo S Y and Cai B G 2012 *Acta Phys. Sin.* **61** 140203 (in Chinese)
[4] Pang J, Jin L H and Zhao Q 2012 *Acta Phys. Sin.* **61** 140201 (in Chinese)
[5] Taogetusang and Bai Y M 2012 *Acta Phys. Sin.* **61** 130202 (in Chinese)
[6] Hirt C, Amsden A and Cook J 1974 *J. Comput. Phys.* **14** 227
[7] Cheng J and Shu C W 2007 *J. Comput. Phys.* **227** 1567
[8] Cheng J and Shu C W 2008 *Appl. Numer. Math.* **58** 1042
[9] Woodward P and Colella P 1984 *J. Comput. Phys.* **54** 115
[10] Colella P and Woodward P 1984 *J. Comput. Phys.* **54** 174
[11] Cheng J and Shu C W 2009 *Chin. J. Comput. Phys.* **26** 633 (in Chinese)
[12] Jia Z P and Yu X J 2007 *Chin. J. Comput. Phys.* **24** 543 (in Chinese)
[13] Shen Z J, Yan W and Lü G X 2010 *J. Comput. Phys.* **229** 4522
[14] Reed W H and Hill T R 1973 *Triangular Mesh Methods for the Neutron Transport Equation* (Los Alamos Scientific Laboratory Report LA-UR-73-479)
[15] Cockburn B and Shu C W 1991 *Math. Model. Numer. Anal.* (M2AN) **25** 337
[16] Cockburn B and Shu C W 1989 *Math. Comp.* **52** 411
[17] Cockburn B, Lin S Y and Shu C W 1990 *J. Comput. Phys.* **84** 90
[18] Cockburn B, Hou S and Shu C W 1990 *Math. Comp.* **54** 545
[19] Cockburn B and Shu C W 1998 *J. Comput. Phys.* **141** 199
[20] Cockburn B and Shu C W 2001 *J. Sci. Comput.* **16** 173
[21] Chen R S and Yu X J 2006 *Chin. J. Comput. Phys.* **23** 43 (in Chinese)
[22] Zhang R P, Yu X J and Zhao G Z 2011 *Chin. Phys.* B **20** 110205
[23] Zhang R P, Yu X J and Feng T 2012 *Chin. Phys.* B **21** 030202
[24] Zhang R P and Zhang L W 2012 *Chin. Phys.* B **21** 090206
[25] Qian X, Song S H, Gao E and Li W B 2012 *Chin. Phys.* B **21** 070206
[26] Qiu J X, Liu T G and Khoo B C 2008 *Commun. Comput. Phys.* **3** 479

[27] Qiu J X, Liu T G and Khoo B C 2007 *J. Comput. Phys.* **222** 353
[28] Li D Y, Shui H S, Dong S Q and Feng X S 1996 *Chin. J. Comput. Phys.* **13** 57 (in Chinese)
[29] Sun S K, Shen L J and Shen Z J 2010 *Chin. J. Comput. Phys.* **27** 317 (in Chinese)
[30] Zhao G Z and Yu X J 2012 *Acta Phys. Sin.* **61** 110208 (in Chinese)
[31] Zhao G Z, Yu X J and Zhang R P 2012 *Chin. J. Comput. Phys.* **29** 166 (in Chinese)
[32] Shu C W 1988 *SIAM J. Sci. Stat. Comput.* **9** 1073
[33] Fedkiw R P, Aslam T, Merriman B and Osher S 1999 *J. Comput. Phys.* **152** 457
[34] Caiden R, Fedkiw R P and Anderson C 2001 *J. Comput. Phys.* **166** 1
[35] Fedkiw R P 2002 *J. Comput. Phys.* **175** 200
[36] Abgrall R 1996 *J. Comput. Phys.* **125** 150
[37] Abgrall R and Karni S 2001 *J. Comput. Phys.* **169** 594
[38] Liu T G, Khoo B C and Yeo K S 2001 *Comput. Fluids* **30** 291
[39] Liu T G, Khoo B C and Yeo K S 2001 *Comput. Fluids* **30** 315
[40] Van Brummelen E H and Koren B 2003 *J. Comput. Phys.* **185** 289
[41] Liu T G, Khoo B C and Yeo K S 2003 *J. Comput. Phys.* **190** 651
[42] Liu T G, Khoo B C and Wang C W 2005 *J. Comput. Phys.* **204** 193
[43] Dai W L and Woodward P R 1998 *J. Comput. Phys.* **142** 182
[44] Toro E F 1997 *Riemann Solvers and Numerical Methods for Fluid Dynamics—A Practical Introduction* 2nd edn. (Berlin: Springer)
[45] Chen D W and Yu X J 2009 *Chin. J. Comput. Phys.* **26** 501 (in Chinese)
[46] Harten A, Engquist B, Osher S and Chakravarthy S 1987 *J. Comput. Phys.* **71** 231
[47] Tang H Z and Liu T G 2006 *J. Comput. Phys.* **218** 451