



# High order multi-moment constrained finite volume method. Part I: Basic formulation

Satoshi Ii<sup>a,c</sup>, Feng Xiao<sup>b,c,\*</sup>

<sup>a</sup> Department of Mechanical Engineering, The University of Tokyo, 7-3-1 Hongo Bunkyo-ku, Tokyo 113-8656, Japan

<sup>b</sup> LEM/LHD, Institute of Mechanics, Chinese Academy of Sciences, 15 Beisihuanxi Road, Beijing 100080, China

<sup>c</sup> Department of energy sciences, Tokyo Institute of Technology, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan

## ARTICLE INFO

### Article history:

Received 23 April 2008

Received in revised form 3 February 2009

Accepted 3 February 2009

Available online 20 February 2009

### Keywords:

Finite volume method

High-order accuracy

Multi-moment

Hyperbolic conservation laws

Compact stencil

Local reconstruction

## ABSTRACT

A new high-order finite volume method based on local reconstruction is presented in this paper. The method, so-called the multi-moment constrained finite volume (MCV) method, uses the point values defined within single cell at equally spaced points as the model variables (or unknowns). The time evolution equations used to update the unknowns are derived from a set of constraint conditions imposed on multi kinds of moments, i.e. the cell-averaged value and the point-wise value of the state variable and its derivatives. The finite volume constraint on the cell-average guarantees the numerical conservativeness of the method. Most constraint conditions are imposed on the cell boundaries, where the numerical flux and its derivatives are solved as general Riemann problems. A multi-moment constrained Lagrange interpolation reconstruction for the demanded order of accuracy is constructed over single cell and converts the evolution equations of the moments to those of the unknowns. The presented method provides a general framework to construct efficient schemes of high orders. The basic formulations for hyperbolic conservation laws in 1- and 2D structured grids are detailed with the numerical results of widely used benchmark tests.

© 2009 Elsevier Inc. All rights reserved.

## 1. Introduction

High-order schemes with local reconstructions use compact stencil for spatial discretization, and appear more attractive in flexibility and efficiency to locally refine the numerical solutions. Examples of the numerical algorithms that are conservative and make use of the locally defined degrees of freedom (DOFs) for high order reconstructions are the discontinuous Galerkin (DG) method [4–7], the spectral volume (SV) method [29–32] and, more recently the spectral difference (SD) method [17,33].

The DG method is a formulation based on the Galerkin finite element approach. The local degrees of freedom is increased by a high-order local discontinuous basis function constructed piecewisely over each mesh cell, and the Riemann problem is solved at cell boundaries by a class of numerical methods originally developed in the context of the finite difference or finite volume approaches. The cell-averaged value is then cast into a complete flux-form formulation to assure the conservation in the DG method. The local DOF in the SV method is increased by partitioning the control volume (or mesh cell) further into finer subdomains with geometrical regularity. The cell-averaged values for these subdomains are then updated as the new

\* Corresponding author. Address: LEM/LHD, Institute of Mechanics, Chinese Academy of Sciences, 15 Beisihuanxi Road, Beijing 100080, China. Tel./fax: +86 10 82543877.

E-mail address: [xiao@imech.ac.cn](mailto:xiao@imech.ac.cn) (F. Xiao).

unknowns. With the new DOFs defined on the regular decomposition in each mesh cell, high-order scheme can be constructed on a local base. The SV method essentially applies the finite volume formulation to all DOFs, so the numerical conservation for the unknowns is maintained.

Being more computationally efficient, the spectral difference (SD) method has been recently extended to unstructured grids [17,33]. The SD method originated from the conservative staggered-grid Chebyshev multi-domain method [14], where the unknowns of the state variable are defined at the Gauss points while the numerical fluxes are defined at the Gauss–Lobatto points within each mesh cell. Cell-wise Lagrange interpolating polynomials are constructed separately for the variable values and the flux values. The variable values at the Gauss points are interpolated to the Gauss–Lobatto points where the numerical fluxes are computed. The spatial derivative of the flux at the Gauss points is then computed by differentiating the flux interpolation function. Finally, the ordinary time evolution equation for the state variable is obtained by requiring the residual of the spatial discretization to be zero at the Gauss point. In the SD method, all unknowns are defined as the point values at the internal points within the mesh cell with the optimized locations. The numerical conservativeness can be enforced by combining the flux differencing at the internal point so as to balance the effective numerical flux across the cell surface. The SD is a point-based method, and is easily implemented on both structure and unstructured grids.

In a series of multi-moment schemes, developed from the underlying idea of the Constrained Interpolation Profile (CIP) method [43,45], more than one kind of moments are used as the model variables and individually updated by the governing equations that may have different forms but should be consistent to the original conservation law. For example, several conservative schemes, so-called the CIP-Conservative Semi-Lagrangian (CIP-CSL) methods for the scalar conservation law [26,44,35,36,10], can be constructed by including extra moments, such as the point value and the derivatives, in addition to the cell-averaged value. The point value and the derivatives that are defined at the cell surface are predicted by a semi-Lagrangian approach, while the cell average is updated by a finite volume formulation. For fluid dynamics, a class of so-called CIP/multi-moment finite volume method (CIP/MM FVM) have been devised by making use of the Volume-Integrated Average (VIA), Surface-Integrated Average (SIA) and Point Value (PV) as the model variables that must be memorized and predicted forward in time. In the CIP/MM FVM variants, the VIA moment is updated by a finite volume formulation of flux-form and is thus numerically conserved. A great flexibility is provided while predicting other moments. The multi-moment finite volume method as a general framework for fluid dynamic simulations has been so far implemented to various fluid flows, for example multi-fluid flows [37,38,40], compressible and incompressible flows [39,41,11] and geophysical flows [2,15,3].

For high-order schemes based on the multi-moments, we proposed in our previous paper [11] an accurate CIP/MM FVM using a characteristic semi-Lagrangian approach for both scalar and Euler conservation laws. A more general formulation that employs more derivative moments for arbitrary accuracy and uses the local derivative Riemann problems (DRP) [28,27] in the Eulerian representation is also proposed [42]. Another recent variant is found as the conservative interpolated differential operator method [12].

This paper presents a new high-order conservative scheme with more computational efficiency. Different from the existing methods, the new method, namely the multi-moment constrained finite volume (MCV) method, uses the point values defined within single cell at equally spaced points as the model variables (or unknowns). Instead of being directly treated as the model variables, the multi-moments are used to provide the constraint conditions that lead to the evolution equations to update the unknowns. The MCV method has exact conservation for the cell-averaged value, and high-order MCV schemes can be straightforwardly devised following a simple principle.

The basic idea and formulation of the MCV method are detailed in this paper. The rest of the paper is organized as follows: Section 2 describes the basic numerical formulation with the schemes from third- to sixth-orders illustrated in detail. The extension to 2D Cartesian coordinate grid is given in Section 3. Numerical tests are shown in Section 4, we have examined the numerical convergence rate by grid refinement tests for the presented schemes from third- to sixth-orders. Some widely used benchmark tests for both scalar and Euler conservation laws in 1D and 2D are also given to verify the numerical schemes. Finally, we end this paper with a short summary in Section 5.

## 2. Basic formulation for one dimension

Different from the multi-moment methods developed so far where the moments are directly treated as the model variables that need to be predicted at every time step, values at the points collocated within each mesh element are treated as the basic model variables in the present method. The evolution equations derived for different moments are employed as the constraints that yield the relations to update the unknowns at the collocating points within each cell.

We consider the 1D hyperbolic conservation laws as follows,

$$q_t + f(q)_x = 0, \quad (2.1)$$

where  $q$  is the vector of the conservative variables and  $f(q)$  is the vector of the flux function.

The computational domain is divided into non-overlapping cells,  $\delta x_i \in [x_{i-1/2}, x_{i+1/2}]$  ( $i = 1, \dots, N$ ), where  $x_{i+1/2}$  is the cell boundary. We define the moments, i.e. the cell-averaged value, the point-wise value and the derivatives of the field variable  $q(x,t)$  as,

$$\begin{aligned} \bar{q}_i(t) &\equiv \frac{1}{\Delta x_i} \int_{\delta x_i} q(x, t) dx, \\ q_{x_{ip}}^{[k]}(t) &\equiv \frac{\partial^k q}{\partial x^k}(x_{ip}, t); \quad \text{with } k = 0, 1, 2, \dots, K, \end{aligned} \tag{2.2}$$

where  $\Delta x_i = x_{i+1/2} - x_{i-1/2}$  is the volume for cell  $i$ , and  $x_{ip}$  represents a point within or at the boundary of cell  $i$ . We will denote  $q_{x_{ip}}^{[0]}$  by  $q_{ip}$  and  $q_{x_{ip}}^{[1]}$  by  $q_{x_{ip}}$  as a convention in this paper.

From (2.1), we have the evolution equations for different moments of physical field  $q(x, t)$  as,

$$\begin{aligned} \frac{d\bar{q}_i}{dt} &= -\frac{1}{\Delta x_i} (\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}}), \\ \frac{dq_{x_{ip}}^{[k]}}{dt} &= -\hat{f}_{x_{ip}}^{[k+1]}; \quad \text{with } k = 0, 1, 2, \dots, K, \end{aligned} \tag{2.3}$$

where  $\hat{f}$  and  $\hat{f}_x^{[k]}$  are the numerical formulations consistent to the flux function  $f(q)$  and its derivatives  $\partial_x^k f(q)$ .

The unknowns, or the degrees of freedom (DOFs), which should be predicted at every time step in the MCV scheme are the point values (PVs), or the PV moments in our context, defined within each single cell shown in Fig. 1. The cell-wise interpolation reconstruction is then locally built in terms of the unknowns. For example, in a 1D  $L$ th-order scheme,  $(L - 1)$ th-order polynomial  $\mathcal{P}_i(x)$  on cell  $i$  requires  $L$  PVs ( $q_{il}; l = 1, 2, \dots, L$ ), we write the interpolation polynomial in terms of the unknowns as,

$$\mathcal{P}_i(x) = \sum_{l=1}^L c_{il}(x) q_{il}, \tag{2.4}$$

where  $c_{il}(x)$  is the basis function. The points  $x_{il}$  for defining the unknowns  $q_{il}$  are not necessarily coincided with  $x_{ip}$  where the constraints in terms of the evolution equations for different moments are imposed.

It should be addressed that construction of (2.4) is essentially different from the conventional Lagrange interpolation though they share the same form. The values at the interior points are determined from the multi-moment constraints, thus we call it multi-moment constrained (MC) Lagrange polynomial. See the appendix of this paper for more details.

Given interpolation function (2.4), the constraints on different moments (2.3) are explicitly expressed in the unknowns, yielding a set of evolution equations to predict the unknowns,

$$\frac{d\bar{q}_i}{dt} = \frac{1}{\Delta x_i} \frac{d}{dt} \left( \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathcal{P}_i(x) dx \right) = \frac{1}{\Delta x_i} \sum_{l=1}^L \left( \frac{dq_{il}}{dt} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} c_{il}(x) dx \right) = -\frac{1}{\Delta x_i} (\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}}), \tag{2.5a}$$

$$\frac{dq_{x_{ip}}^{[k]}}{dt} = \frac{d}{dt} \left( \frac{\partial^k \mathcal{P}_i(x_{ip})}{\partial x^k} \right) = \sum_{l=1}^L \left( \frac{dq_{il}}{dt} \frac{\partial^k c_{il}(x_{ip})}{\partial x^k} \right) = -\hat{f}_{x_{ip}}^{[k+1]}; \quad k = 0, 1, 2, \dots, K. \tag{2.5b}$$

Eq. (2.5a) guarantees the exact conservativeness of the cell-integrated average value  $\bar{q}_i$ .

The reconstruction in an MCV method is built over each mesh cell, so the cell-wise interpolation function is continuous within each cell, but might be discontinuous across cell boundaries. Depending on the location of  $x_{ip}$ , the numerical formulations for the derivatives of the flux function in (2.5b) are computed in different ways.

Being one of the important features of the current method, we use the constraint conditions (2.5b) at cell boundaries ( $x_{i-1/2}$  and  $x_{i+1/2}$ ) to a largest possible extent.

When point  $x_{ip}$  is at cell boundary, the numerical flux needs to be evaluated by solving the Riemann problem. At a boundary point, the cell-wise reconstructions may lead to discontinuous values on the left and right sides of  $x_{ip}$ , denoted by  $q^-$  and  $q^+$ , respectively. The numerical fluxes are then computed by

$$\begin{aligned} \hat{f}_{ip} &= \text{Riemann}(f(q_{ip}^-), f(q_{ip}^+)), \\ \hat{f}_{x_{ip}}^{[k]} &= \text{Riemann}(f_x^{[k]}(q_{ip}^-, q_{x_{ip}}^-, \dots, q_{x_{ip}}^{[k]-}), f_x^{[k]}(q_{ip}^+, q_{x_{ip}}^+, \dots, q_{x_{ip}}^{[k]+})), \end{aligned} \tag{2.6}$$

where “Riemann( $\cdot, \cdot$ )” denotes a Riemann solver with the left-side and right-side values given from the reconstructions.

In practice, we make use of the approximate Riemann solvers for computational efficiency. Two simplest examples among many existing approximate Riemann solvers are the local Lax–Friedrich (LLF) flux and the Roe’s flux [19]. Computational cost can be further reduced if we replace the high-order derivative Riemann problems by linear and homogeneous derivative Riemann problems as detailed in [28,27] for the hyperbolic systems. As addressed in [28], since the first-instant plays a leading role in the interaction of the two states, the derivative Riemann problems with these simplifications provide a reasonable



Fig. 1. The locations of the unknowns in one dimension for the third-order (left), the fourth-order (middle) and the fifth-order (right) schemes.

accuracy. For the Euler equations in fact, our numerical experiments show that the following linearization to the flux functions gives adequate accuracy in terms of both numerical errors and convergence rate

$$f = Aq, \quad f_x = Aq_x, \dots, \quad f_x^{[k]} = Aq_x^{[k]}, \tag{2.7}$$

where  $A$  is the Jacobian matrix obtained by  $A = \partial f / \partial q$ . Provided the derivatives of the state variable  $q$  from the cell-wise reconstructions for both sides of a discontinuity, one can find the derivative flux of any high order at the expense of the conventional Riemann problem.

It should be noted that the local reconstruction can be also carried out in terms of flux function  $f$  itself, instead of the conservative variables  $q$ . For example, the numerical approximations for flux function  $f$  and its derivatives at cell boundary  $x_{ip}$  can be calculated by

$$\hat{f}_{x_{ip}}^{[k]} = \frac{1}{2} \left( f_{x_{ip}}^{[k]-} + f_{x_{ip}}^{[k]+} - R_{ip} |\Lambda_{ip}| R_{ip}^{-1} \left( q_{x_{ip}}^{[k]+} - q_{x_{ip}}^{[k]-} \right) \right) = \frac{1}{2} \left( f_{x_{ip}}^{[k]-} + f_{x_{ip}}^{[k]+} - R_{ip} \operatorname{sgn}(\Lambda_{ip}) R_{ip}^{-1} \left( f_{x_{ip}}^{[k]+} - f_{x_{ip}}^{[k]-} \right) \right), \tag{2.8}$$

where we make use of the relation:  $q = A^{-1}f = (R\Lambda R^{-1})^{-1}f = R\Lambda^{-1}R^{-1}f$ . The eigen matrices  $R$  and  $R^{-1}$ , as well as the eigen values in  $\Lambda$  are directly evaluated by the point values at  $x_{ip}$ . Without the linearization (2.7), the numerical simulations using (2.8) share a more uniform convergence rate for high order.

There are  $L$  unknowns in an  $L$ th-order scheme, thus moment constraint conditions of equal number must be specified. We only consider the cases of  $L \geq 3$  in this paper. When  $L$  is an odd number, we use one constraint on the cell-integrated average (2.5a) and  $2(K + 1)$  constraints on the point value and derivatives at the two ends of the cell, where  $K$  is determined by  $K = (L - 3)/2$ . When  $L$  is an even number, however, an extra moment is required to provide another constraint condition. We introduce the first-order derivative at the cell center ( $x_{ic} = (x_{i-1/2} + x_{i+1/2})/2$ ), and write (2.5b) as

$$\frac{dq_{x_{ic}}}{dt} = \frac{d}{dt} \left( \frac{\partial}{\partial x} \mathcal{P}_i(x_{ic}) \right) = \sum_{l=1}^L \left( \frac{dq_{ip}}{dt} \frac{\partial}{\partial x} c_{il}(x_{ic}) \right) = -\hat{f}_{x_{ic}}^{[2]}. \tag{2.9}$$

As shown in the following subsections, the second-order derivative of the flux function  $\hat{f}_{x_{ic}}^{[2]}$  in (2.9) is computed by an approximation in terms of the flux from the interpolation reconstruction  $\hat{f}_{ic} = f(\mathcal{P}_i(x_{ic}))$  as well as the flux and the derivatives of the flux function at cell boundaries.

Provided all the values and the derivatives of the flux function, the semi-discretized system of (2.5a) and (2.5b) finally yields the following ordinary differential equations in terms of the unknowns  $Q = [q_{i1}, q_{i2}, \dots, q_{iL}]$ ,

$$\frac{dQ}{dt} = R(Q), \tag{2.10}$$

where  $R(Q)$  stand for the approximations in terms of the discretized flux and the derivatives of the flux function. We use the third-order TVD Runge–Kutta method [22],

$$\begin{aligned} Q_1 &= Q_0 + \Delta t R(Q_0), \\ Q_2 &= \frac{3}{4} Q_0 + \frac{1}{4} Q_1 + \frac{1}{4} \Delta t R(Q_1), \\ Q_3 &= \frac{1}{3} Q_0 + \frac{2}{3} Q_2 + \frac{2}{3} \Delta t R(Q_2). \end{aligned} \tag{2.11}$$

for time integration.

In the rest of this section, we give the examples of the MCV schemes from third- to sixth-orders.

### 2.1. Third-order MCV scheme

In the third-order scheme, the unknowns are three PVs ( $q_{il}, l = 1, 2, 3$ ) at three equally spaced points within cell  $i$  as shown in Fig. 1,

$$q_{i1}(t) = q(x_{i1}, t), \quad q_{i2}(t) = q(x_{i2}, t), \quad q_{i3}(t) = q(x_{i3}, t), \tag{2.12}$$

where  $x_{i1} = x_{i-1/2}$ ,  $x_{i2} = (x_{i-1/2} + x_{i+1/2})/2$  and  $x_{i3} = x_{i+1/2}$ .

A cell-wise MC-Lagrange polynomial,

$$\mathcal{P}_i(x) = \sum_{l=1}^3 c_{il}(x) q_{il}, \tag{2.13}$$

is then constructed. We impose constraint conditions on three moments, i.e. one cell-averaged value and two PVs at cell surfaces as

$$\begin{aligned} \bar{q}_i &= \frac{1}{\Delta x_i} \int_{\delta x_i} \mathcal{P}_i(x) dx = \frac{(q_{i1} + 4q_{i2} + q_{i3})}{6}, \\ q_{i-\frac{1}{2}} &= \mathcal{P}_i(x_{i-\frac{1}{2}}) = q_{i1}, \\ q_{i+\frac{1}{2}} &= \mathcal{P}_i(x_{i+\frac{1}{2}}) = q_{i3}. \end{aligned} \tag{2.14}$$

The evolution equations of the moments (2.5a) and (2.5b) then read,

$$\begin{aligned} \frac{d}{dt} \bar{q}_i &= \frac{1}{6} \frac{d}{dt} (q_{i1} + 4q_{i2} + q_{i3}) = -\frac{1}{\Delta x_i} (\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}}), \\ \frac{d}{dt} q_{i-\frac{1}{2}} &= \frac{dq_{i1}}{dt} = -\hat{f}_{xi-\frac{1}{2}}, \\ \frac{d}{dt} q_{i+\frac{1}{2}} &= \frac{dq_{i3}}{dt} = -\hat{f}_{xi+\frac{1}{2}}. \end{aligned} \tag{2.15}$$

The constraint conditions (2.15) finally lead to the equations to directly update the unknowns as

$$\begin{aligned} \frac{dq_{i1}}{dt} &= -\hat{f}_{xi-\frac{1}{2}}, \\ \frac{dq_{i2}}{dt} &= -\frac{3}{2} \bar{f}_{xi} + \frac{1}{4} (\hat{f}_{xi-\frac{1}{2}} + \hat{f}_{xi+\frac{1}{2}}), \\ \frac{dq_{i3}}{dt} &= -\hat{f}_{xi+\frac{1}{2}}, \end{aligned} \tag{2.16}$$

where  $\bar{f}_{xi} = (\hat{f}_{i+1/2} - \hat{f}_{i-1/2})/\Delta x_i$  is the cell-averaged first-order derivative of the numerical flux. The numerical fluxes at cell boundary, say  $x_{i-1/2}$ , are obtained by solving the (derivative) Riemann problems,

$$\begin{aligned} \hat{f}_{i-\frac{1}{2}} &= \text{Riemann}(f_{i-\frac{1}{2}}(q_{i-\frac{1}{2}}^-), f_{i-\frac{1}{2}}(q_{i-\frac{1}{2}}^+)), \\ \hat{f}_{xi-\frac{1}{2}} &= \text{Riemann}(f_{xi-\frac{1}{2}}(q_{i-\frac{1}{2}}^-, q_{xi-\frac{1}{2}}^-), f_{xi-\frac{1}{2}}(q_{i-\frac{1}{2}}^+, q_{xi-\frac{1}{2}}^+)), \end{aligned} \tag{2.17}$$

or more simply by the flux linearization (2.7)

$$\begin{aligned} \hat{f}_{i-\frac{1}{2}} &= \text{Riemann}(A_{i-\frac{1}{2}}^- q_{i-\frac{1}{2}}^-, A_{i-\frac{1}{2}}^+ q_{i-\frac{1}{2}}^+), \\ \hat{f}_{xi-\frac{1}{2}} &= \text{Riemann}(A_{i-\frac{1}{2}}^- q_{xi-\frac{1}{2}}^-, A_{i-\frac{1}{2}}^+ q_{xi-\frac{1}{2}}^+). \end{aligned} \tag{2.18}$$

The left-side and right-side values and derivatives of the state variables are computed from the cell-wise reconstructions as

$$\begin{aligned} q_{i-\frac{1}{2}}^- &= \mathcal{P}_{i-1}(x_{i-\frac{1}{2}}) = q_{(i-1)3}, \\ q_{xi-\frac{1}{2}}^- &= \frac{\partial}{\partial x} \mathcal{P}_{i-1}(x_{i-\frac{1}{2}}) = \frac{-3q_{(i-1)1} + 4q_{(i-1)2} - q_{(i-1)3}}{\Delta x_{i-1}}, \\ q_{i-\frac{1}{2}}^+ &= \mathcal{P}_i(x_{i-\frac{1}{2}}) = q_{i1}, \\ q_{xi-\frac{1}{2}}^+ &= \frac{\partial}{\partial x} \mathcal{P}_i(x_{i-\frac{1}{2}}) = \frac{q_{i1} - 4q_{i2} + 3q_{i3}}{\Delta x_i}. \end{aligned} \tag{2.19}$$

For cell boundary  $x_{i+1/2}$ , the same procedure applies with the cell index shifted by one.

### 2.2. Fourth-order MCV scheme

In the fourth-order MCV, we put the four unknowns at equidistant positions over cell  $i$  as shown in Fig. 1,

$$q_{i1}(t) = q(x_{i1}, t), \quad q_{i2}(t) = q(x_{i2}, t), \quad q_{i3}(t) = q(x_{i3}, t), \quad q_{i4}(t) = q(x_{i4}, t), \tag{2.20}$$

where  $x_{i1} = x_{i-1/2}$ ,  $x_{i2} = (2x_{i-1/2} + x_{i+1/2})/3$ ,  $x_{i3} = (x_{i-1/2} + 2x_{i+1/2})/3$  and  $x_{i4} = x_{i+1/2}$ . As aforementioned, with  $L$  being an even number, a new constraint condition needs to be imposed on the first-order derivative at the cell center ( $x_{ic} = (x_{i-1/2} + x_{i+1/2})/2$ ), in addition to the cell-averaged value and two PVs at both cell boundaries. A polynomial of four DOFs,  $\mathcal{P}_i(x) = \sum_{l=1}^4 c_{il}(x)q_{il}$ , is then derived. From (2.5a) and (2.5b), we get the following linear system,

$$\begin{aligned}
\frac{d\bar{q}_i}{dt} &= \frac{d}{dt} \frac{(q_{i1} + 3q_{i2} + 3q_{i3} + q_{i4})}{8} = -\bar{f}_{xi}, \\
\frac{dq_{i-\frac{1}{2}}}{dt} &= \frac{dq_{i1}}{dt} = -\hat{f}_{xi-\frac{1}{2}}, \\
\frac{dq_{i+\frac{1}{2}}}{dt} &= \frac{dq_{i4}}{dt} = -\hat{f}_{xi+\frac{1}{2}}, \\
\frac{dq_{xic}}{dt} &= \frac{d}{dt} \frac{(q_{i1} - 27q_{i2} + 27q_{i3} - q_{i4})}{8\Delta x_i} = -\hat{f}_{xic}^{[2]},
\end{aligned} \tag{2.21}$$

which then yield the evolution equations to update the PV unknowns as

$$\begin{aligned}
\frac{dq_{i1}}{dt} &= -\hat{f}_{xi-\frac{1}{2}}, \\
\frac{dq_{i2}}{dt} &= -\frac{4}{3}\bar{f}_{xi} + \frac{1}{27}(4\hat{f}_{xi-\frac{1}{2}} + 5\hat{f}_{xi+\frac{1}{2}}) + \frac{4}{27}\Delta x_i \hat{f}_{xic}^{[2]}, \\
\frac{dq_{i3}}{dt} &= -\frac{4}{3}\bar{f}_{xi} + \frac{1}{27}(5\hat{f}_{xi-\frac{1}{2}} + 4\hat{f}_{xi+\frac{1}{2}}) - \frac{4}{27}\Delta x_i \hat{f}_{xic}^{[2]}, \\
\frac{dq_{i4}}{dt} &= -\hat{f}_{xi+\frac{1}{2}}.
\end{aligned} \tag{2.22}$$

The numerical fluxes at cell boundary are solved as the (derivative) Riemann problems in a similar way to that in the third-order MCV scheme. The computation of the second-order derivative of the numerical flux at the cell center  $\hat{f}_{xic}^{[2]}$ , however, needs a different procedure. In the present method, given five constraint conditions on flux, i.e.  $\hat{f}_{ic} = f(\mathcal{P}_i(x_{ic}))$ ,  $\hat{f}_{i-\frac{1}{2}}$ ,  $\hat{f}_{i+\frac{1}{2}}$ ,  $\hat{f}_{xi-\frac{1}{2}}$  and  $\hat{f}_{xi+\frac{1}{2}}$ ,  $\hat{f}_{xic}^{[2]}$  is evaluated from a fourth-order interpolation by

$$\hat{f}_{xic}^{[2]} = \frac{8(-2\hat{f}_{ic} + \hat{f}_{i-\frac{1}{2}} + \hat{f}_{i+\frac{1}{2}})}{\Delta x_i^2} + \frac{(\hat{f}_{xi-\frac{1}{2}} - \hat{f}_{xi+\frac{1}{2}})}{\Delta x_i}. \tag{2.23}$$

It is notified that the second-order derivative of the flux function at cell center ( $x_{ic}$ ) is computed not only from the value obtained from the interpolation reconstruction ( $\hat{f}_{ic}$ ), but also from the numerical fluxes and the derivative fluxes at the cell boundaries. It makes the current method substantially different from other existing algorithms.

### 2.3. Fifth-order MCV scheme

For the fifth-order MCV, the unknowns are the five PVs defined at the equidistant points,

$$x_{i1} = x_{i-\frac{1}{2}}, \quad x_{i2} = \frac{(3x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}})}{4}, \quad x_{i3} = \frac{(x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}})}{2}, \quad x_{i4} = \frac{(x_{i-\frac{1}{2}} + 3x_{i+\frac{1}{2}})}{4}, \quad x_{i5} = x_{i+\frac{1}{2}}, \tag{2.24}$$

over cell  $i$  as shown in Fig. 1. The interpolation is constructed by an MC-Lagrange polynomial  $\mathcal{P}_i(x) = \sum_{l=1}^5 c_{il}(x)q_{il}$  of 5 DOFs. The constraint conditions are imposed on 5 moments, i.e. the cell-averaged value, the PVs and first-order derivatives at the two cell boundaries. Using the interpolation polynomial  $\mathcal{P}_i(x)$ , we get the following evolution equations for the PV unknowns as,

$$\begin{aligned}
\frac{d\bar{q}_i}{dt} &= \frac{d}{dt} \frac{(7q_{i1} + 32q_{i2} + 12q_{i3} + 32q_{i4} + 7q_{i5})}{90} = -\bar{f}_{xi}, \\
\frac{dq_{i-\frac{1}{2}}}{dt} &= \frac{dq_{i1}}{dt} = -\hat{f}_{xi-\frac{1}{2}}, \\
\frac{dq_{i+\frac{1}{2}}}{dt} &= \frac{dq_{i5}}{dt} = -\hat{f}_{xi+\frac{1}{2}}, \\
\frac{dq_{xi-\frac{1}{2}}}{dt} &= \frac{d}{dt} \frac{(-25q_{i1} + 48q_{i2} - 36q_{i3} + 16q_{i4} - 3q_{i5})}{3\Delta x_i} = -\hat{f}_{xi-\frac{1}{2}}^{[2]}, \\
\frac{dq_{xi+\frac{1}{2}}}{dt} &= \frac{d}{dt} \frac{(3q_{i1} - 16q_{i2} + 36q_{i3} - 48q_{i4} + 25q_{i5})}{3\Delta x_i} = -\hat{f}_{xi+\frac{1}{2}}^{[2]}.
\end{aligned} \tag{2.25}$$

The numerical fluxes at cell boundaries, i.e.  $\hat{f}_{i\pm 1/2}$ ,  $\hat{f}_{xi\pm 1/2}$  and  $\hat{f}_{xi\pm 1/2}^{[2]}$ , can be evaluated by solving the derivative Riemann problems up to second-orders. After solving the linear system (2.25), the PV unknowns are updated in time.

### 2.4. Sixth-order MCV scheme

A sixth-order scheme requires six PVs located at the six points equally spaced over cell  $i$  to build a polynomial of six DOFs,  $\mathcal{P}_i(x) = \sum_{l=1}^6 c_{il}(x)q_{il}$ . In addition to the constraint conditions in fifth-order scheme, the first-order derivative moment at cell center ( $x_{ic}$ ) is added to provide a new constraint. The evolution equations for the moments (2.5a) and (2.5b) read

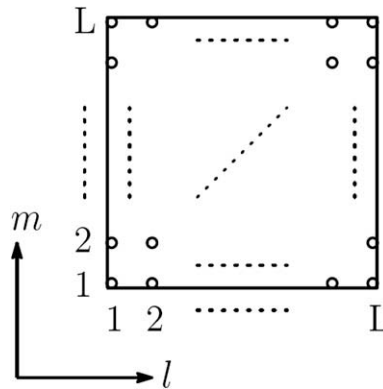
**Table 1**

The maximum CFL numbers for computational stability of the MCV schemes in 1D scalar advection test.

| Accuracy   | 3rd | 4th  | 5th | 6th  |
|------------|-----|------|-----|------|
| CFL number | 0.4 | 0.25 | 0.2 | 0.14 |

$$\begin{aligned}
 \frac{d\bar{q}_i}{dt} &= \frac{d}{dt} \frac{(19q_{i1} + 75q_{i2} + 50q_{i3} + 50q_{i4} + 75q_{i5} + 19q_{i6})}{288} = -\bar{f}_{xi}, \\
 \frac{dq_{i-\frac{1}{2}}}{dt} &= \frac{dq_{i1}}{dt} = -\hat{f}_{xi-\frac{1}{2}}, \\
 \frac{dq_{i+\frac{1}{2}}}{dt} &= \frac{dq_{i6}}{dt} = -\hat{f}_{xi+\frac{1}{2}}, \\
 \frac{dq_{xi-\frac{1}{2}}}{dt} &= \frac{d}{dt} \frac{(-137q_{i1} + 300q_{i2} - 300q_{i3} + 200q_{i4} - 75q_{i5} + 12q_{i6})}{12\Delta x_i} = -\hat{f}_{xi-\frac{1}{2}}^{[2]}, \\
 \frac{dq_{xi+\frac{1}{2}}}{dt} &= \frac{d}{dt} \frac{(-12q_{i1} + 75q_{i2} - 200q_{i3} + 300q_{i4} - 300q_{i5} + 137q_{i6})}{12\Delta x_i} = -\hat{f}_{xi+\frac{1}{2}}^{[2]}, \\
 \frac{dq_{xic}}{dt} &= \frac{d}{dt} \frac{(-9q_{i1} + 125q_{i2} - 2250q_{i3} + 2250q_{i4} - 125q_{i5} + 9q_{i6})}{384\Delta x_i} = -\hat{f}_{xic}^{[2]}.
 \end{aligned}
 \tag{2.26}$$

The equations for updating the PV unknowns,  $q_{i1}, \dots, q_{i6}$ , can be derived straightforwardly from the  $6 \times 6$  linear system (2.26).



**Fig. 2.** The points over cell  $ij$  to define the unknowns.

**Table 2**

Numerical errors and convergence rate for the 1D linear scalar equation.

| Accuracy | Mesh | $L_1$ error | $L_1$ order | $L_\infty$ error | $L_\infty$ order |
|----------|------|-------------|-------------|------------------|------------------|
| 3rd      | 10   | 1.29E-2     | -           | 2.00E-2          | -                |
|          | 20   | 1.69E-3     | 2.93        | 2.64E-3          | 2.92             |
|          | 40   | 2.14E-4     | 2.98        | 3.36E-4          | 2.97             |
|          | 80   | 2.68E-5     | 3.00        | 4.22E-5          | 2.99             |
| 4th      | 10   | 2.06E-4     | -           | 3.19E-4          | -                |
|          | 20   | 1.31E-5     | 3.98        | 2.06E-5          | 3.95             |
|          | 40   | 8.32E-7     | 3.98        | 1.31E-6          | 3.98             |
|          | 80   | 5.25E-8     | 3.99        | 8.24E-8          | 3.99             |
| 5th      | 10   | 5.21E-5     | -           | 8.05E-5          | -                |
|          | 20   | 1.67E-6     | 4.96        | 2.61E-6          | 4.95             |
|          | 40   | 5.34E-8     | 4.97        | 8.38E-8          | 4.96             |
|          | 80   | 1.66E-9     | 5.01        | 2.61E-9          | 5.00             |
| 6th      | 10   | 4.06E-7     | -           | 6.27E-7          | -                |
|          | 20   | 6.46E-9     | 5.97        | 1.00E-8          | 5.97             |
|          | 40   | 9.95E-11    | 6.02        | 1.56E-10         | 6.00             |
|          | 80   | 1.53E-12    | 6.02        | 2.48E-12         | 5.98             |

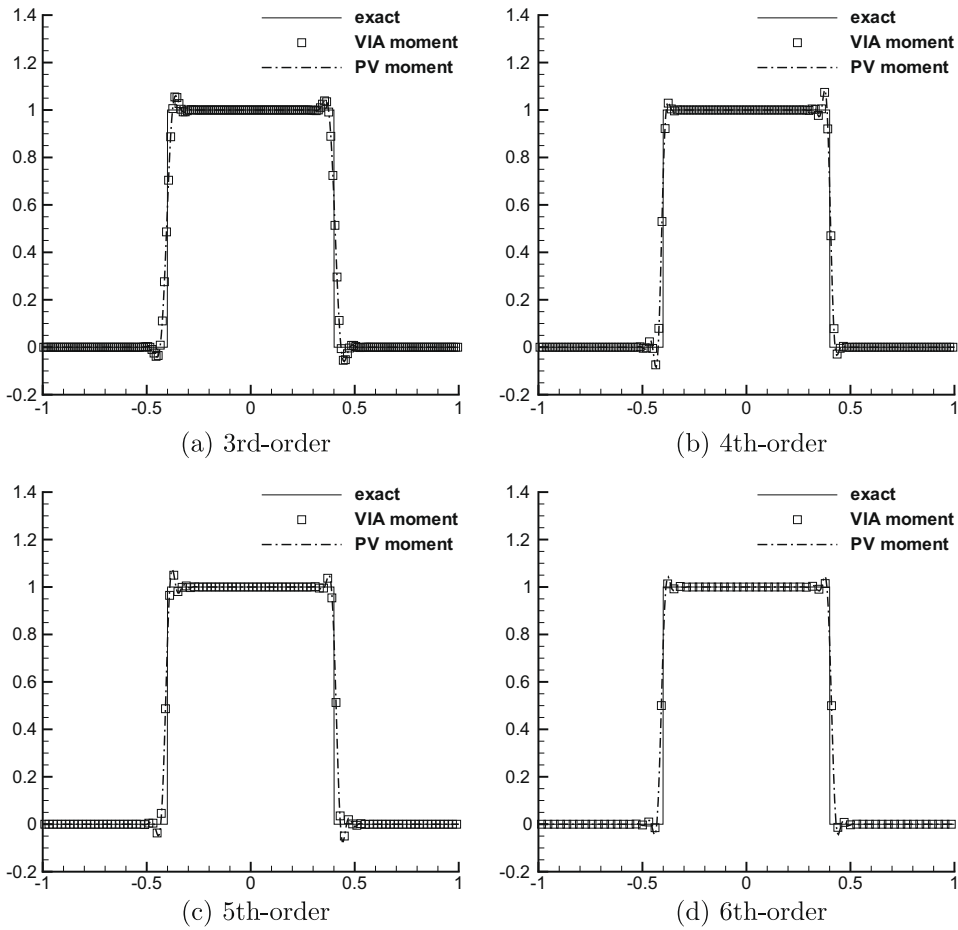


Fig. 3. Advection of a square wave after one period without limiting.

The second-order derivative of the numerical flux at cell center is evaluated from  $\hat{f}_{ic}$ ,  $\hat{f}_{i\pm 1/2}$ ,  $\hat{f}_{xi\pm 1/2}$  and  $\hat{f}_{xi\pm 1/2}^{[2]}$  as follows,

$$\hat{f}_{xic}^{[2]} = \frac{12(-2\hat{f}_{ic} + \hat{f}_{i-\frac{1}{2}} + \hat{f}_{i+\frac{1}{2}})}{\Delta x_i^2} + \frac{9(\hat{f}_{xi-\frac{1}{2}} - \hat{f}_{xi+\frac{1}{2}})}{4\Delta x_i} + \frac{\hat{f}_{xi-\frac{1}{2}}^{[2]} + \hat{f}_{xi+\frac{1}{2}}^{[2]}}{8}. \quad (2.27)$$

The numerical flux at the cell center are computed from the interpolation polynomial,  $\hat{f}_{ic} = f(\mathcal{P}_i(x_{ic}))$ . All other flux or the derivatives of the flux function are evaluated at the cell boundaries, thus should be computed from the Riemann solvers.

### 2.5. Some remarks

In the present MCV formulations, the points to define the unknown PVs are equally spaced. It is different from the Gauss–Lobatto points used in the conservative staggered-grid Chebyshev multi-domain spectral method [14] or the SD method [17,1], and the formula to convert (2.3) to the evolution equations for the PV unknowns, (2.5a) and (2.5b), are more efficient. In the schemes from third-order to sixth-order, the only numerical flux computed from the interpolation function at the cell center is  $\hat{f}_{ic}$  in (2.23) and (2.27), while other flux or the derivatives of flux are evaluated at cell boundaries. We increase the derivatives of flux at cell boundary when constructing higher schemes to provide constraints for the increased DOFs, rather than use more fluxes at interior points as that in the Chebyshev multi-domain spectral method [14]. Our numerical tests show that the MCV method has a satisfactory convergence rate for all schemes from third to sixth-orders presented in this paper. Moreover, since there is not any volume integration involved in the present method, it is quite computationally efficient.

As mentioned above, most of constraint conditions are imposed on the cell boundary as the Riemann and derivative Riemann problems. It implies that the increase in the DOFs for higher order scheme directly leads to an increase of the information exchanged across the cell boundary. Consequently, different from other existing schemes using local high-order reconstruction, the current MCV method allows larger CFL number for computational stability. We have carried out numerical experiments for 1D linear advection equation with smooth initial condition, and used the third-order Runge–Kutta



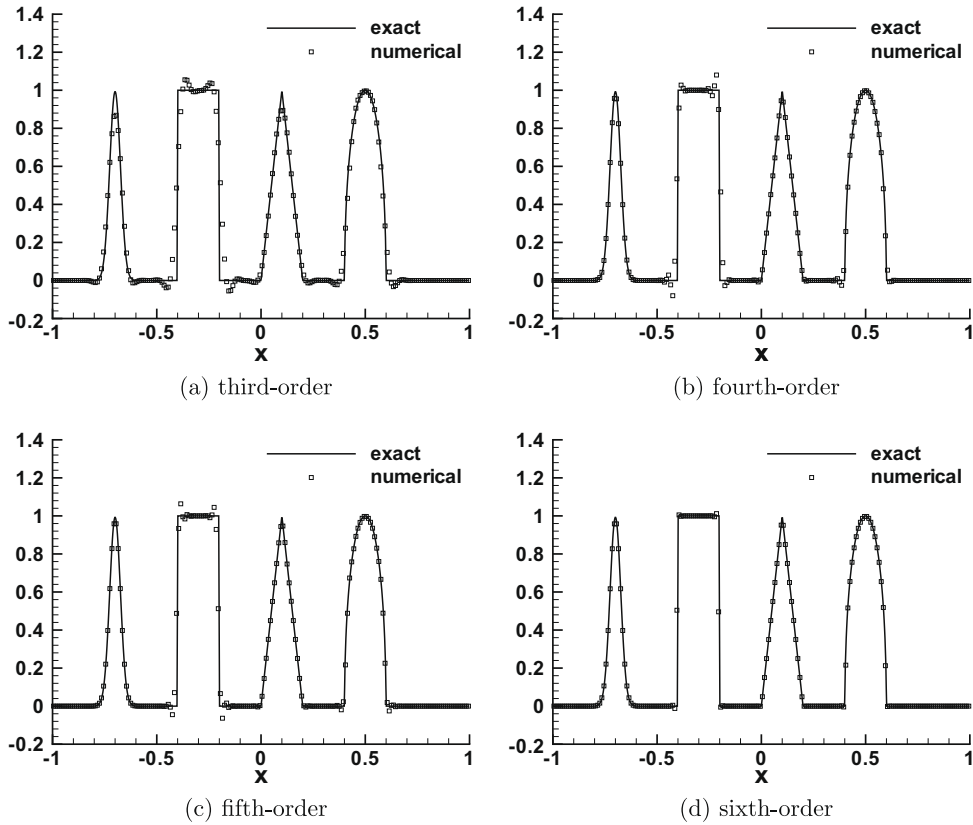


Fig. 4. Numerical results of the 1D advection of a complex profile without limiter at  $t = 2$ .

scheme (2.11) for time integration. Table 1 gives the maximum CFL numbers for stability of the MCV schemes from third-order to sixth-order.

Compared with the maximum stable CFL number for the DG scheme and the SV scheme of the same order as given in [29], the MCV method has a less restrictive CFL condition for computational stability.

2.6. The initial condition

In the numerical experiments presented in this paper, special care must be paid to specify the initial values of the unknowns, i.e. the PVs located at the equally spaced points over each mesh cell. It is trivial if the initial profile is analytically given everywhere. Otherwise the unknown point values must be provided from high-order interpolations or from the “physical observations” that have been assimilated by a numerical model of equivalent accuracy.

As a typical case, we describe here the process to determine the initial unknowns for a mesh cell where a jump discontinuity exists. Assuming a jump discontinuity exists in  $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$  and given the PVs at the cell boundaries, we decide the initial conditions for the unknowns according to the following rules.

- (i) The PVs at the boundaries of cell  $i$  are firstly specified as

$$q_{i\pm\frac{1}{2}} = q(x_{i\pm\frac{1}{2}}, 0).$$

- (ii) The cell-averaged value is approximated by the PVs at the cell boundaries by a simple algebraic average,

$$\bar{q}_i = \frac{q(x_{i-\frac{1}{2}}, 0) + q(x_{i+\frac{1}{2}}, 0)}{2}.$$

- (iii) The derivatives at the cell boundaries are initially assumed to be zero,

$$q_{xi\pm\frac{1}{2}}, q_{xi\pm\frac{1}{2}}^{[2]}, \dots = 0.$$

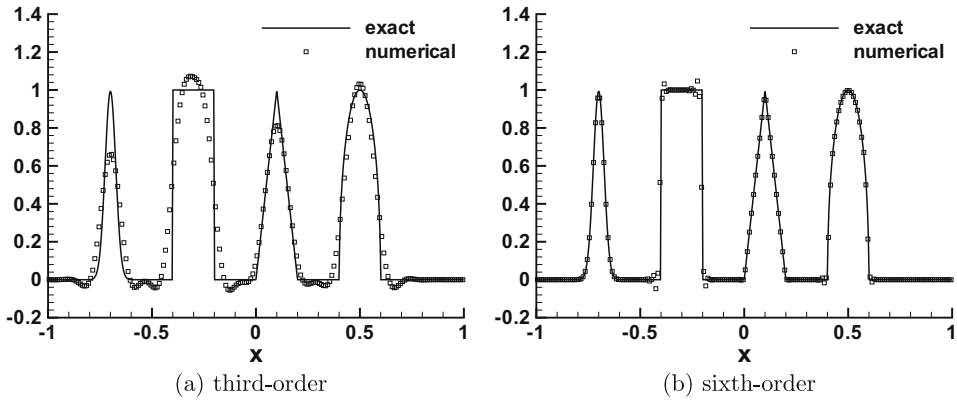


Fig. 5. Comparison between the third-order MCV and sixth-order MCV schemes in the advection problem at  $t = 20$  (after 10 periods).

(iv) When  $L$  is an even number, the derivative at the cell center is approximated by,

$$q_{xic} = \frac{\partial}{\partial x} \tilde{P}(x_{ic}),$$

where  $\tilde{P}(x)$  is the polynomial of  $(L - 1)$  DOFs where the derivative at the cell center is not involved. We estimate  $q_{xic}$  by a central difference in terms of the PVs at the two ends in this paper for simplicity, which slightly degrades the accuracy in the initial conditions, but only makes negligible difference in the numerical results.

From the above conditions (i)–(iv), the initial PV unknowns can be obtained by the multi-moment interpolation polynomial  $P_i(x) = \sum_{j=1}^L c_{ij}(x)q_{ij}$ .

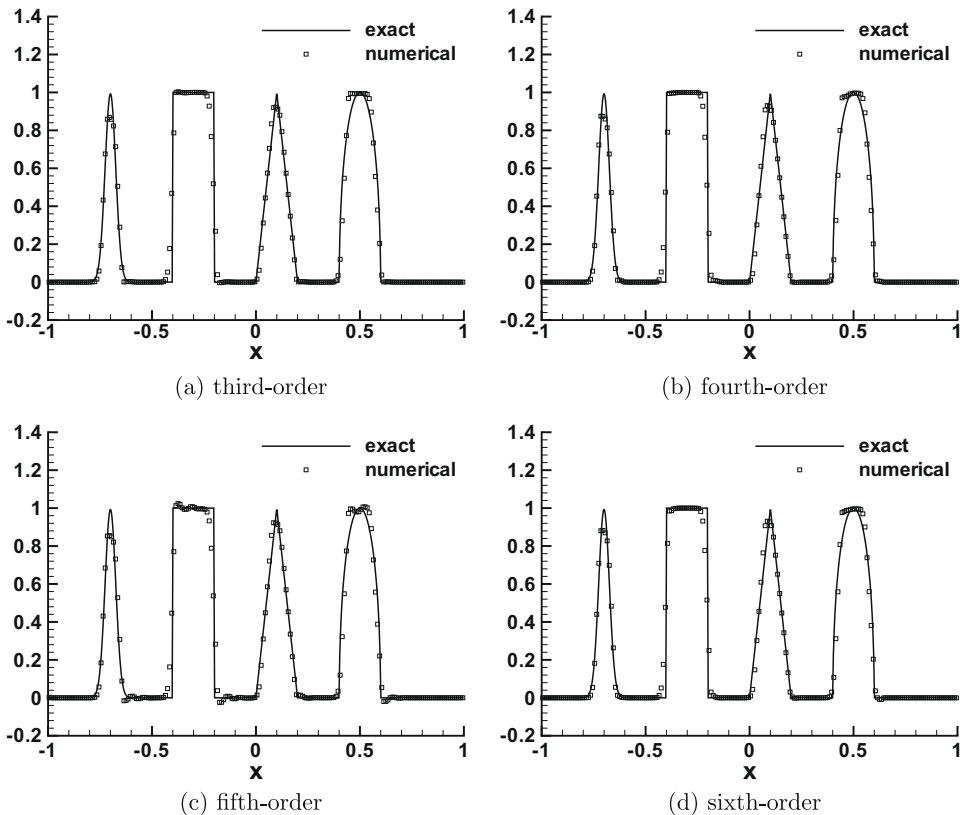


Fig. 6. Numerical results of the 1D advection problem with TVB limiter ( $M = 200$ ) at  $t = 2$ .

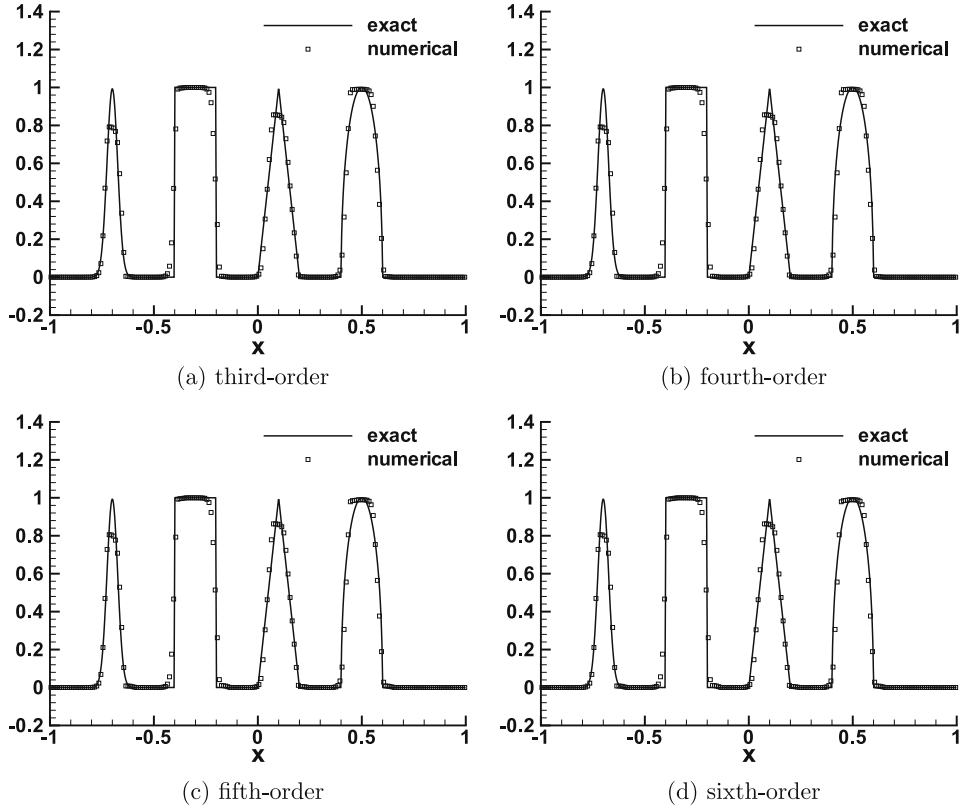


Fig. 7. Numerical results of the 1D advection problem with TVD limiter ( $M=0$ ) at  $t=2$ .

We give below the initial values of the PV unknowns for the MCV schemes from third-order to sixth-order.

- Third-order:

$$\begin{pmatrix} q_{i1}(0) \\ q_{i2}(0) \\ q_{i3}(0) \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ 1/2 & 1/2 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} q(x_{i-\frac{1}{2}}, 0) \\ q(x_{i+\frac{1}{2}}, 0) \end{pmatrix}, \quad (2.28)$$

- Fourth-order:

$$\begin{pmatrix} q_{i1}(0) \\ q_{i2}(0) \\ q_{i3}(0) \\ q_{i4}(0) \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ 2/3 & 1/3 \\ 1/3 & 2/3 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} q(x_{i-\frac{1}{2}}, 0) \\ q(x_{i+\frac{1}{2}}, 0) \end{pmatrix}, \quad (2.29)$$

- Fifth-order:

$$\begin{pmatrix} q_{i1}(0) \\ q_{i2}(0) \\ q_{i3}(0) \\ q_{i4}(0) \\ q_{i5}(0) \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ 27/32 & 5/32 \\ 1/2 & 1/2 \\ 5/32 & 27/32 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} q(x_{i-\frac{1}{2}}, 0) \\ q(x_{i+\frac{1}{2}}, 0) \end{pmatrix}, \quad (2.30)$$

- Sixth-order:

$$\begin{pmatrix} q_{i1}(0) \\ q_{i2}(0) \\ q_{i3}(0) \\ q_{i4}(0) \\ q_{i5}(0) \\ q_{i6}(0) \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ 2608/3125 & 517/3125 \\ 1881/3125 & 1244/3125 \\ 1244/3125 & 1881/3125 \\ 517/3125 & 2608/3125 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} q(x_{i-\frac{1}{2}}, 0) \\ q(x_{i+\frac{1}{2}}, 0) \end{pmatrix}. \quad (2.31)$$

**Table 3**Numerical errors and convergence rate for  $u_t + (u^2/2)_x = 0$  at  $t = 0.5/\pi$ .

| Accuracy | Mesh | $L_1$ error | $L_1$ order | $L_\infty$ error | $L_\infty$ order |
|----------|------|-------------|-------------|------------------|------------------|
| 3rd      | 20   | 5.43E-4     | –           | 2.83E-3          | –                |
|          | 40   | 9.91E-5     | 2.45        | 6.79E-4          | 2.06             |
|          | 80   | 1.49E-5     | 2.73        | 1.34E-4          | 2.34             |
|          | 160  | 2.11E-6     | 2.82        | 3.55E-5          | 1.92             |
|          | 320  | 2.83E-7     | 2.90        | 8.82E-6          | 2.01             |
| 4th      | 20   | 2.49E-5     | –           | 1.80E-4          | –                |
|          | 40   | 1.41E-6     | 4.14        | 1.39E-5          | 3.69             |
|          | 80   | 7.86E-8     | 4.17        | 9.21E-7          | 3.92             |
|          | 160  | 4.57E-9     | 4.10        | 6.56E-8          | 3.81             |
|          | 320  | 3.17E-10    | 3.85        | 6.78E-9          | 3.27             |
| 5th      | 20   | 3.44E-6     | –           | 2.77E-5          | –                |
|          | 40   | 2.27E-7     | 3.92        | 2.98E-6          | 3.22             |
|          | 80   | 1.35E-8     | 4.07        | 1.84E-7          | 4.02             |
|          | 160  | 5.94E-10    | 4.51        | 8.38E-9          | 4.46             |
|          | 320  | 2.18E-11    | 4.77        | 4.28E-10         | 4.29             |
| 6th      | 20   | 9.12E-8     | –           | 5.93E-7          | –                |
|          | 40   | 1.59E-9     | 5.84        | 1.75E-8          | 5.08             |
|          | 80   | 4.56E-11    | 5.12        | 5.98E-10         | 4.87             |
|          | 160  | 6.50E-13    | 6.13        | 1.69E-11         | 5.15             |
|          | 320  | 1.58E-14    | 5.36        | 3.87E-13         | 5.45             |

### 2.7. Limiting projection

To eliminate the spurious numerical oscillation, we apply the slope-limiting operator  $\Lambda\Pi_i$  proposed in [7] to the solutions at each Runge–Kutta step. The slope-limiting operator  $\Lambda\Pi_i$  is a linear approximation defined as

$$\Lambda\Pi_i\mathcal{P}_i(x) = \bar{q}_i + \tilde{q}_{xi}(x - x_{ic}), \quad (2.32)$$

where  $\tilde{q}_x$  is the limited gradient. It is required that the limiting procedure satisfies

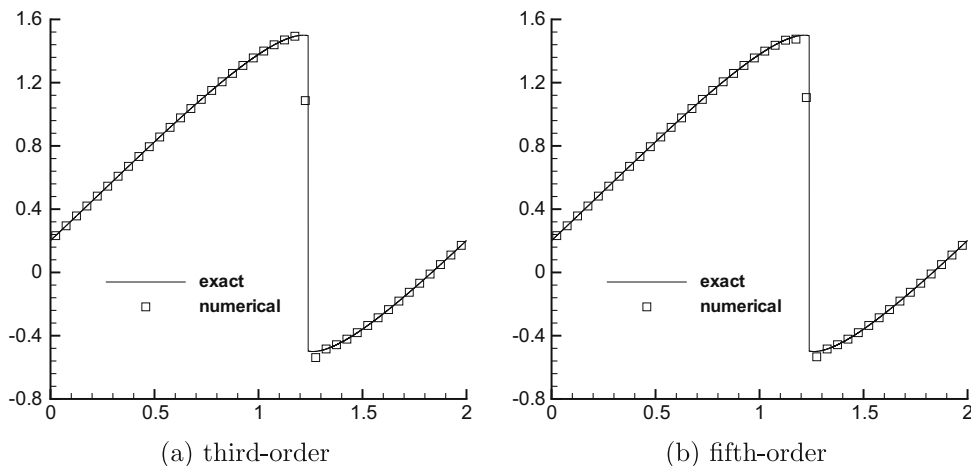
$$\int_{\delta x_i} \Lambda\Pi_i\mathcal{P}_i(x) dx = \int_{\delta x_i} \mathcal{P}_i(x) dx = \Delta x_i \bar{q}_i \quad (2.33)$$

for the numerical conservativeness.

The limited gradient  $\tilde{q}_{xi}$  is computed based on the TVD concept [9]. We apply the *superbee* limiter,

$$\tilde{q}_{xi} = \max\left(\min\left(\Delta q_{i-1/2}, \beta\Delta q_{i+1/2}\right), \min\left(\Delta q_{i+1/2}, \beta\Delta q_{i-1/2}\right)\right), \quad (2.34)$$

where  $\Delta q_{i-1/2} = q_{ic} - q_{(i-1)c}$ ,  $\Delta q_{i+1/2} = q_{(i+1)c} - q_{ic}$  and  $\beta$  is a positive to control the gradient sharpness, and is set to 1.8–2.0 in this paper.



**Fig. 8.** Numerical results for the nonlinear Burger's equation at  $t = 1.5/\pi$  on a 40-cell mesh. The reference ("exact") solution is computed 20,480-cell mesh by a conventional second-order finite volume method. The TVB parameter is  $M = 200$ .

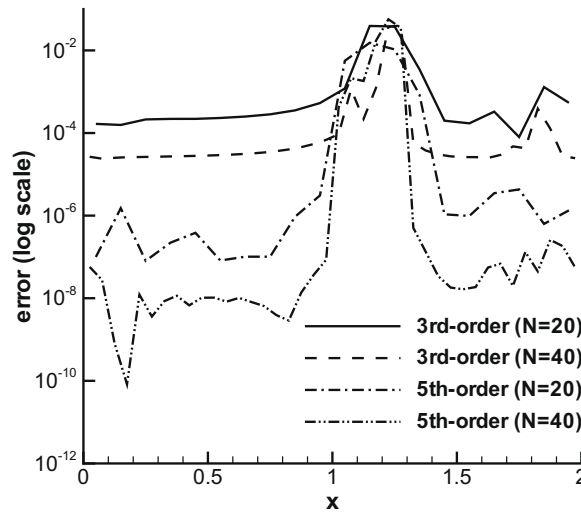


Fig. 9. Numerical errors of the third- and fifth-order MCV schemes at  $t = 1.5/\pi$ . The TVB parameter  $M = 200$ .

The *minmod* and *maxmod* functions are given respectively as

$$\text{minmod}(a_1, a_2) = \begin{cases} s \min(|a_1|, |a_2|), & \text{if } s = \text{sign}(a_1) = \text{sign}(a_2), \\ 0, & \text{otherwise,} \end{cases} \tag{2.35}$$

and

$$\text{maxmod}(a_1, a_2) = \begin{cases} s \max(|a_1|, |a_2|), & \text{if } s = \text{sign}(a_1) = \text{sign}(a_2), \\ 0, & \text{otherwise.} \end{cases} \tag{2.36}$$

As it is well-known that the TVD limiter might over-smooth the local extremes in the numerical solution, we introduce also the TVB limiter [21]. We slightly modified the formulation to get a smooth transition between the limited and unlimited regions,

$$q_{il} = \begin{cases} q_{il}, & \text{if } |-q_{i1} + q_{iL}| \leq M\Delta x_i^2 \cap (-q_{i1} + q_{iL})(-q_{(i-1)L} + q_{(i+1)1}) > 0, \\ \Lambda \Pi_i \mathcal{P}_i(x_{il}), & \text{otherwise,} \end{cases} \tag{2.37}$$

( $l = 1, \dots, L$ ). The parameter  $M$  is a switching parameter, and is somewhat problem-dependent as in other high-order schemes. As will be seen later in the numerical tests, the limiting procedure affects the local solutions depending on the TVB parameter which needs to be specified in advance in the current formulation. It should be a common challenging issue

**Table 4**  
Numerical errors and convergence rate for the 1D Euler equations.

| Accuracy | Mesh | $L_1$ error | $L_1$ order | $L_\infty$ error | $L_\infty$ order |
|----------|------|-------------|-------------|------------------|------------------|
| 3rd      | 10   | 2.59E-3     | -           | 3.99E-3          | -                |
|          | 20   | 3.38E-4     | 2.94        | 5.28E-4          | 2.92             |
|          | 40   | 4.28E-5     | 2.98        | 6.71E-5          | 2.98             |
|          | 80   | 5.37E-6     | 2.99        | 8.43E-6          | 2.99             |
| 4th      | 10   | 4.12E-5     | -           | 6.37E-5          | -                |
|          | 20   | 2.62E-6     | 3.98        | 4.12E-6          | 3.95             |
|          | 40   | 1.66E-7     | 3.98        | 2.61E-7          | 3.98             |
|          | 80   | 1.05E-8     | 3.98        | 1.65E-8          | 3.98             |
| 5th      | 10   | 1.04E-5     | -           | 1.61E-5          | -                |
|          | 20   | 3.35E-7     | 4.96        | 5.22E-7          | 4.95             |
|          | 40   | 1.06E-8     | 4.98        | 1.66E-8          | 4.97             |
|          | 80   | 3.39E-10    | 4.97        | 5.33E-10         | 4.96             |
| 6th      | 10   | 8.11E-8     | -           | 1.25E-7          | -                |
|          | 20   | 1.25E-9     | 6.02        | 1.98E-9          | 5.98             |
|          | 40   | 1.98E-11    | 5.98        | 3.09E-11         | 6.00             |
|          | 80   | 3.38E-13    | 5.87        | 6.01E-13         | 5.68             |

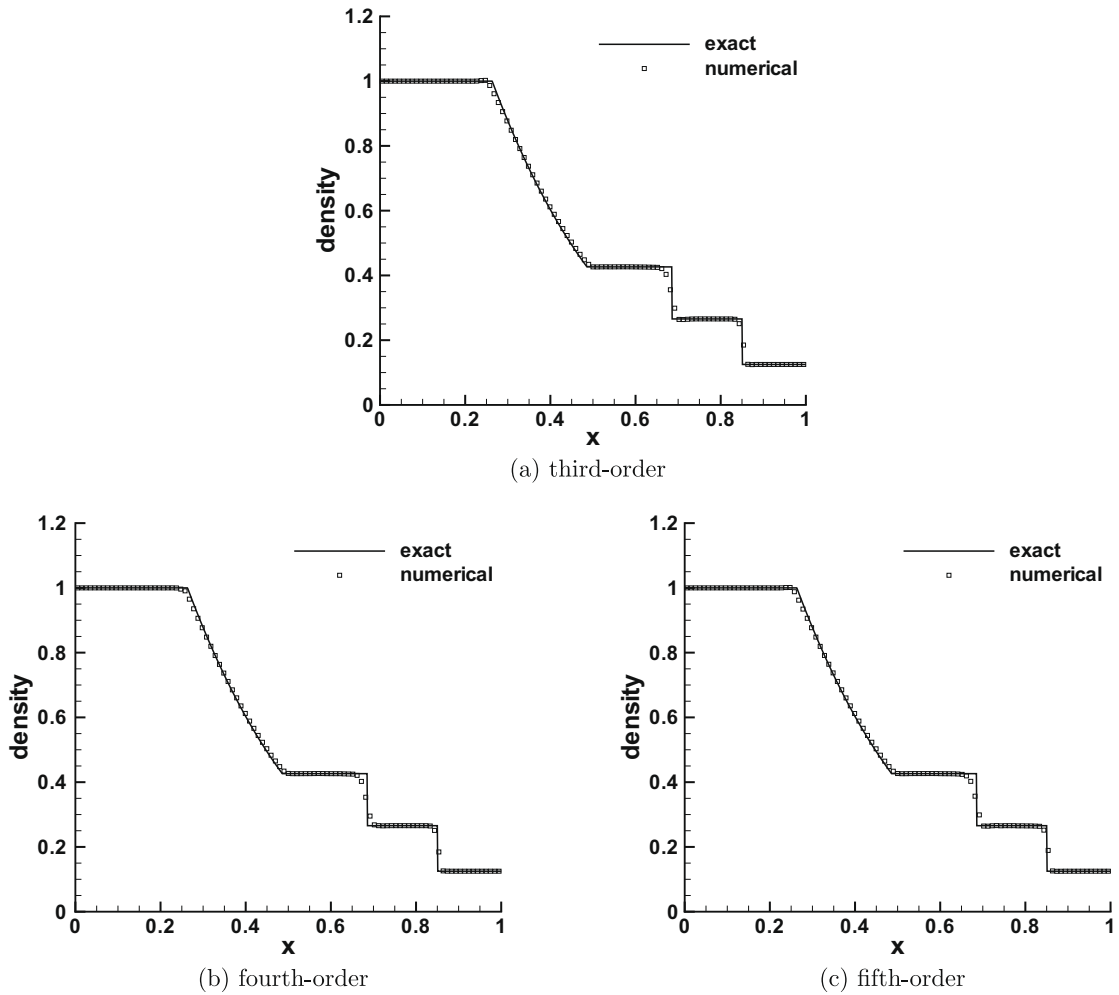


Fig. 10. Numerical results of Sod's problem at  $t = 0.2$ .

for all high-order schemes to maintain high order in smooth region while get rid of spurious oscillation at discontinuity. Some schemes with the WENO idea [18] may probably provide a solution.

### 3. Extension to 2D Cartesian grid

In Cartesian grid, it is straightforward to extend the MCV method to multi-dimensions. Due to the regularity of the mesh, we can directly implement the one-dimensional formulation to  $x$  and  $y$  directions, respectively.

We consider the 2D hyperbolic system,

$$q_t + f(q)_x + g(q)_y = 0, \tag{3.1}$$

with  $f$  and  $g$  being the flux functions in  $x$  and  $y$  directions, respectively.

The unknown PVs are defined at the points shown in Fig. 2. For an  $L$ th-order scheme, the unknowns over cell  $ij$  are  $q_{ijm}$  defined at points  $(x_{ijl}, y_{ijm})$ ;  $l = 1, \dots, L, m = 1, \dots, L$  with

$$x_{ij1} (= x_{i-1/2}), x_{ij2}, \dots, x_{ijl}, \dots, x_{ijL} (= x_{i+1/2}) \quad \text{and} \quad y_{ij1} (= y_{j-1/2}), y_{ij2}, \dots, y_{ijm}, \dots, y_{ijL} (= y_{j+1/2})$$

being equally spaced over  $[x_{i-1/2}, x_{i+1/2}]$  and  $[y_{j-1/2}, y_{j+1/2}]$ , respectively.

The 1D scheme is then implemented over the line segments  $\overline{x_{ij1}x_{ijL}} \cap y_{ijm}$ , ( $m = 1, \dots, L$ ) in  $x$  direction and  $x_{ijl} \cap \overline{y_{ij1}y_{ijL}}$ , ( $l = 1, \dots, L$ ) in  $y$  direction separately.

We introduce the averaged values along line segment  $\overline{x_{ij1}x_{ijL}} \cap y_{ijm}$  in  $x$  direction by

$$\bar{q}_{ijm}^{(x)}(t) \equiv \frac{1}{\Delta x_i} \int_{\Delta x_i} q(x, y_{ijm}, t) dx \tag{3.2}$$

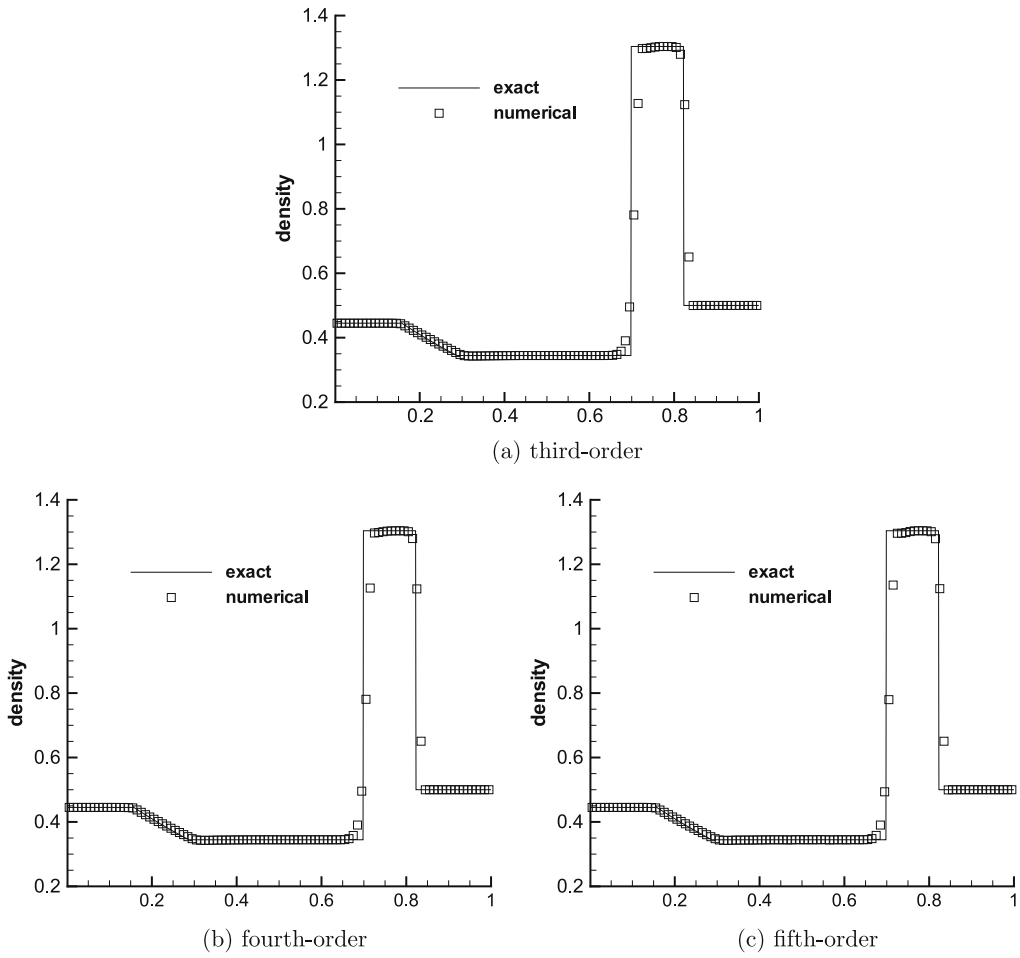


Fig. 11. Numerical results of Lax's problem at  $t = 0.13$  with TVB parameter  $M = 20$ .

and the point-wise value and derivatives at two ends of the line segment by

$$q_{xi\pm\frac{1}{2}jm}^{[k](x)}(t) \equiv \frac{\partial^k q}{\partial x^k}(x_{i\pm\frac{1}{2}}, y_{ijm}, t); \quad \text{with } k = 0, 1, 2, \dots, K. \tag{3.3}$$

Along line segment  $x_{ijl} \cap \overline{y_{ij1}y_{ijL}}$  in  $y$  direction, the moments are defined by

$$\overline{q}_{ijl}^{(y)}(t) \equiv \frac{1}{\Delta y_j} \int_{\delta y_j} q(x_{ijl}, y, t) dy \tag{3.4}$$

and

$$q_{yij\pm\frac{1}{2}l}^{[k](y)}(t) \equiv \frac{\partial^k q}{\partial y^k}(x_{ijl}, y_{j\pm\frac{1}{2}}, t); \quad \text{with } k = 0, 1, 2, \dots, K. \tag{3.5}$$

The unknown PVs along  $\overline{x_{ij1}x_{ijL}} \cap y_{ijm}$  in  $x$  direction is defined by

$$q_{ijm}^{(x)}(t) \equiv q(x_{ijl}, y_{ijm}, t); \quad \text{with } l = 0, 1, 2, \dots, L \tag{3.6}$$

and those along  $x_{ijl} \cap \overline{y_{ij1}y_{ijL}}$  in  $y$  direction by

$$q_{ijm}^{(y)}(t) \equiv q(x_{ijl}, y_{ijm}, t); \quad \text{with } m = 0, 1, 2, \dots, L. \tag{3.7}$$

We summarize the solution procedure of the 2D  $L$ th-order MCV scheme as follows,

- (i) Give the initial values for all PV moments over all cells,
 
$$q_{ijm}(0) = q(x_{ijl}, y_{ijm}, 0), \quad (l = 1, \dots, L, m = 1, \dots, L).$$

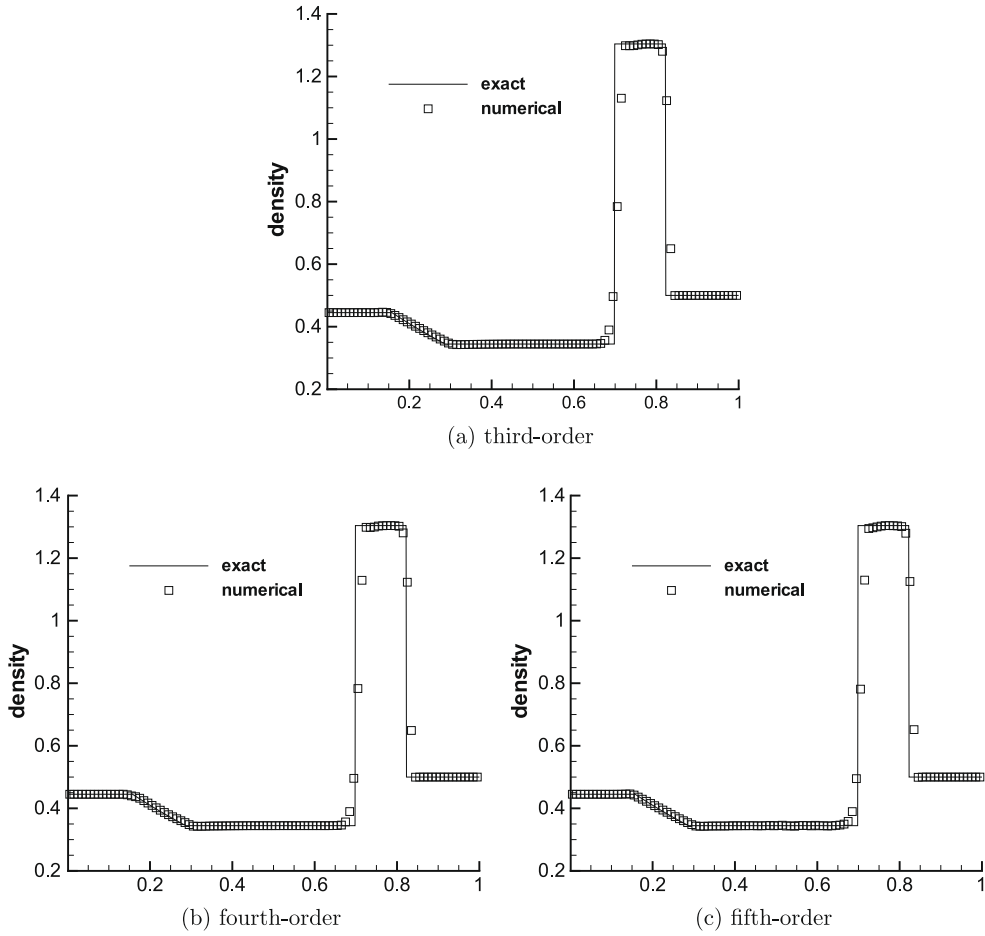


Fig. 12. Same as Fig. 11, but  $M = 100$ .

(ii) Construct 1D MC-Lagrange polynomial interpolation along each line in  $x$  and  $y$  directions. In  $x$  direction, the interpolation function reads

$$\mathcal{P}_{ijm}^{(x)}(x) = \sum_{l=1}^L c_{ijl}^{(x)}(x)q_{ijlm}^{(x)}, \quad (m = 1, \dots, L), \tag{3.8}$$

where the basis function  $c_{ijl}^{(x)}(x)$  ( $l = 1, \dots, L$ ) only depends on  $x$ . In  $y$  direction, the interpolation function is

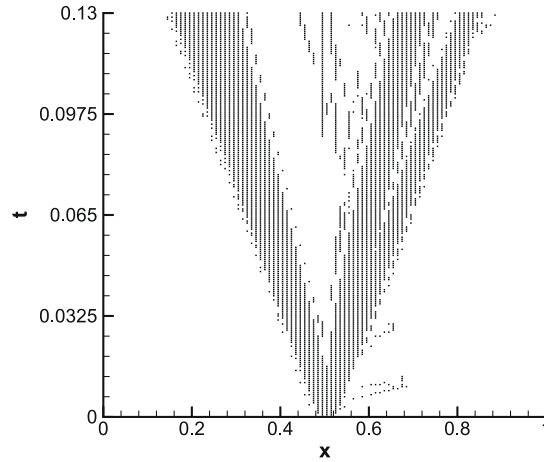
$$\mathcal{P}_{ijl}^{(y)}(y) = \sum_{m=1}^L c_{ijm}^{(y)}(y)q_{ijlm}^{(y)}, \quad (l = 1, \dots, L), \tag{3.9}$$

and the basis function  $c_{ijm}^{(y)}(y)$  ( $m = 1, \dots, L$ ) only depends on  $y$ .

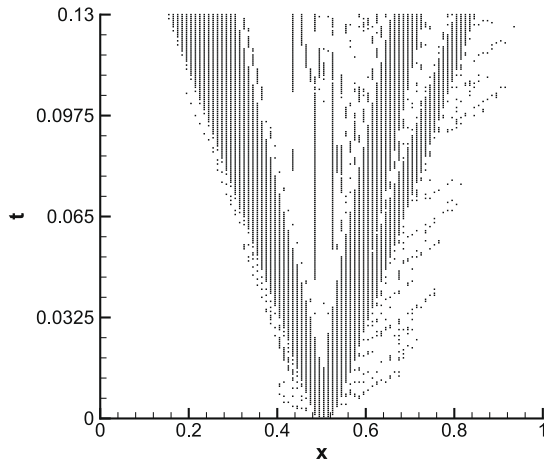
(iii) The numerical formulations of the flux and the derivatives of the flux function,  $\hat{f}_{i-1/2jm}, \hat{f}_{xi-1/2jm}^{[k]}$ ; ( $m = 1, \dots, L$ ) at cell boundary  $x_{i-1/2}$  are evaluated by,

$$\begin{aligned} \hat{f}_{i-1/2jm} &= \text{Riemann}\left(f_{i-1/2jm}^-(q_{i-1/2jm}^-), f_{i-1/2jm}^+(q_{i-1/2jm}^+)\right), \\ \hat{f}_{xi-1/2jm} &= \text{Riemann}\left(f_{xi-1/2jm}^-(q_{i-1/2jm}^-, q_{xi-1/2jm}^-), f_{xi-1/2jm}^+(q_{i-1/2jm}^+, q_{xi-1/2jm}^+)\right), \\ &\vdots \\ \hat{f}_{xi-1/2jm}^{[k]} &= \text{Riemann}\left(f_{xi-1/2jm}^{[k]-}(q_{i-1/2jm}^{[k]-}, q_{xi-1/2jm}^{[k]-}, \dots, q_{xi-1/2jm}^{[k]-}), f_{xi-1/2jm}^{[k]+}(q_{i-1/2jm}^{[k]+}, q_{xi-1/2jm}^{[k]+}, \dots, q_{xi-1/2jm}^{[k]+})\right), \end{aligned} \tag{3.10}$$

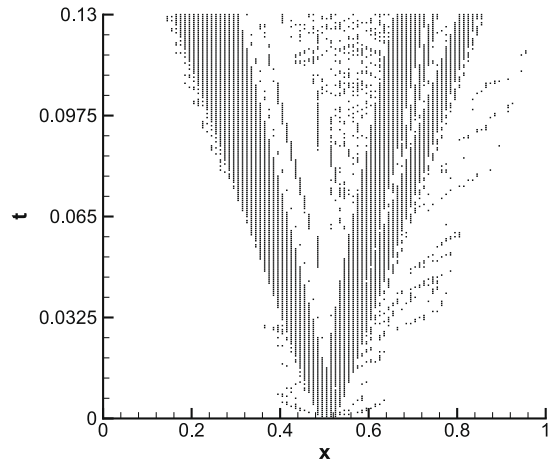




(a) third-order



(b) fourth-order



(c) fifth-order

Fig. 13. The cells where limiting is applied. TVB parameter is  $M = 20$ .

where the left-side and right-side values at the cell boundary are computed by

$$q_{i-\frac{1}{2}m}^- = \mathcal{P}_{i-1jm}^{(x)}(x_{i-\frac{1}{2}}), \quad \dots, \quad q_{xi-\frac{1}{2}m}^{[k]-} = \frac{\partial^k}{\partial x^k} \mathcal{P}_{i-1jm}^{(x)}(x_{i-\frac{1}{2}}),$$

$$q_{i-\frac{1}{2}m}^+ = \mathcal{P}_{ijm}^{(x)}(x_{i-\frac{1}{2}}), \quad \dots, \quad q_{xi-\frac{1}{2}m}^{[k]+} = \frac{\partial^k}{\partial x^k} \mathcal{P}_{ijm}^{(x)}(x_{i-\frac{1}{2}}),$$

( $m = 1, \dots, L$ ).

Analogously, the same procedure applies in the  $y$  direction at cell boundary  $y_{j-1/2}$  for  $\hat{g}_{ij-1/2l}^-, \hat{g}_{yij-1/2l}^{[k]-}$  ( $l = 1, \dots, L$ ) as

$$\hat{g}_{ij-\frac{1}{2}} = \text{Riemann}\left(\mathcal{g}_{ij-\frac{1}{2}}\left(q_{ij-\frac{1}{2}}^-\right), \mathcal{g}_{ij-\frac{1}{2}}\left(q_{ij-\frac{1}{2}}^+\right)\right),$$

$$\hat{g}_{yij-\frac{1}{2}} = \text{Riemann}\left(\mathcal{g}_{yij-\frac{1}{2}}\left(q_{ij-\frac{1}{2}}^-, q_{yij-\frac{1}{2}}^-\right), \mathcal{g}_{yij-\frac{1}{2}}\left(q_{ij-\frac{1}{2}}^+, q_{yij-\frac{1}{2}}^+\right)\right),$$

$\vdots$

$$\hat{g}_{yij-\frac{1}{2}}^{[k]} = \text{Riemann}\left(\mathcal{g}_{yij-\frac{1}{2}}^{[k]}\left(q_{ij-\frac{1}{2}}^-, q_{yij-\frac{1}{2}}^-, \dots, q_{yij-\frac{1}{2}}^{[k]-}\right), \mathcal{g}_{yij-\frac{1}{2}}^{[k]}\left(q_{ij-\frac{1}{2}}^+, q_{yij-\frac{1}{2}}^+, \dots, q_{yij-\frac{1}{2}}^{[k]+}\right)\right),$$

where

$$q_{ij-\frac{1}{2}}^- = \mathcal{P}_{ij-1l}^{(y)}(y_{j-\frac{1}{2}}), \quad \dots, \quad q_{yij-\frac{1}{2}}^{[k]-} = \frac{\partial^k}{\partial y^k} \mathcal{P}_{ij-1l}^{(y)}(y_{j-\frac{1}{2}}),$$

$$q_{ij-\frac{1}{2}}^+ = \mathcal{P}_{ijl}^{(y)}(y_{j-\frac{1}{2}}), \quad \dots, \quad q_{yij-\frac{1}{2}}^{[k]+} = \frac{\partial^k}{\partial y^k} \mathcal{P}_{ijl}^{(y)}(y_{j-\frac{1}{2}}),$$

( $l = 1, \dots, L$ ).

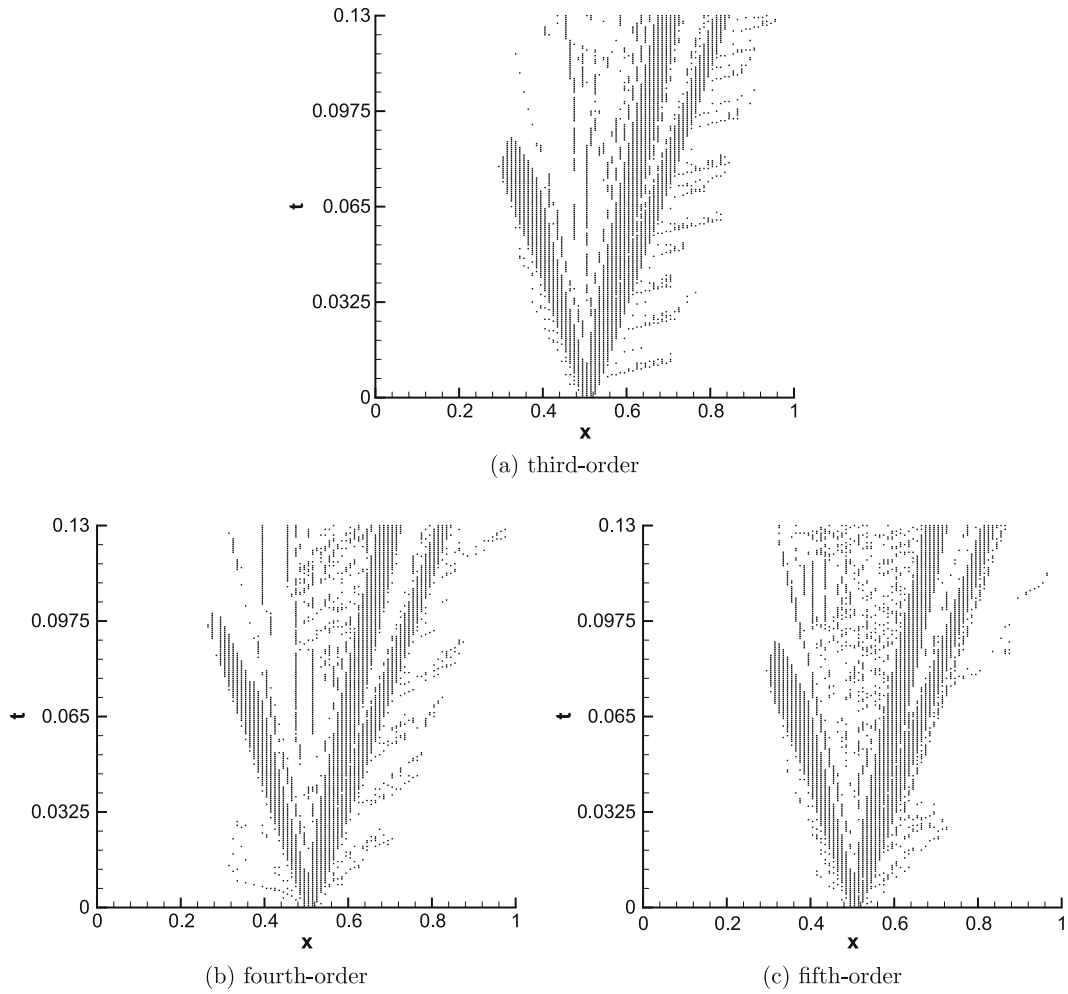


Fig. 14. Same as Fig. 13, but  $M = 100$ .

(iv) If  $L$  is an even number, the first-order derivatives in respect to  $x$  and  $y$  at the center points of line segments  $\overline{x_{ij1}x_{ijL}} \cap \overline{y_{ijm}}$ , ( $m = 1, \dots, L$ ) and  $\overline{x_{ijl}} \cap \overline{y_{ij1}y_{ijL}}$ , ( $l = 1, \dots, L$ ) need to be evaluated as the constraint conditions, which then requires the second-order derivatives of the flux functions. We use formulations in 1D, such as (2.23) and (2.27), and compute the second-order derivatives of the flux functions by

$$\begin{aligned} \hat{f}_{x_{icjcm}}^{[2]} &= \mathcal{T}(\hat{f}_{icjcm}, \hat{f}_{i\pm\frac{1}{2}jm}, \hat{f}_{xi\pm\frac{1}{2}jm}, \dots) \quad (m = 1, \dots, L), \\ \hat{g}_{y_{icjcl}}^{[2]} &= \mathcal{T}(\hat{g}_{icjcl}, \hat{g}_{ij\pm\frac{1}{2}l}, \hat{g}_{yij\pm\frac{1}{2}l}, \dots) \quad (l = 1, \dots, L). \end{aligned} \tag{3.12}$$

where  $\mathcal{T}$  denotes the numerical approximation to the second-order derivative in terms of the center point value, the flux and the derivatives of the flux at the cell boundary.

(v) From (2.5a) and (2.5b), the time evaluation equations in  $x$  direction for different moments are written as

$$\begin{aligned} \frac{dq_{ijm}^{(x)}}{dt} &= \frac{\hat{f}_{i+\frac{1}{2}jm} - \hat{f}_{i-\frac{1}{2}jm}}{\Delta x_i}, \\ \frac{dq_{xi\pm\frac{1}{2}jm}^{[k](x)}}{dt} &= -\hat{f}_{xi\pm\frac{1}{2}jm}^{[k+1]}; \quad k = 0, \dots, K, \\ \frac{dq_{x_{icjcm}}^{(x)}}{dt} &= -\hat{f}_{x_{icjcm}}^{[2]}, \end{aligned} \tag{3.13}$$

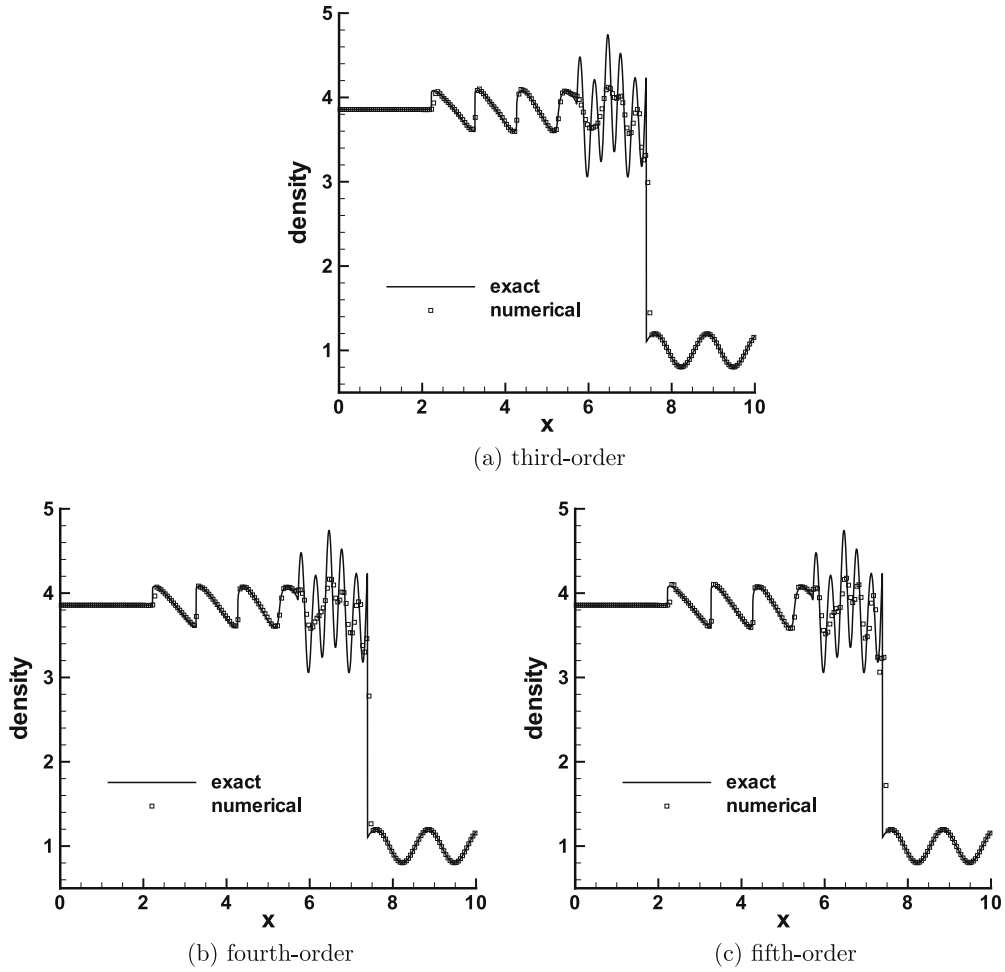


Fig. 15. Numerical results of the shock-turbulence interaction with  $M = 500$  at  $t = 1.8$ .

for  $m = 1, \dots, L$ , and those in  $y$  direction read

$$\begin{aligned} \frac{dq_{ijl}^{(y)}}{dt} &= \frac{\hat{g}_{ij+\frac{1}{2}l} - \hat{g}_{ij-\frac{1}{2}l}}{\Delta y_j}, \\ \frac{dq_{yij\pm\frac{1}{2}l}^{[k](y)}}{dt} &= -\hat{g}_{yij\pm\frac{1}{2}l}^{[k+1]}, \quad k = 0, \dots, K, \\ \frac{dq_{yicjcl}^{(y)}}{dt} &= -\hat{g}_{yicjcl}^{[2]}, \end{aligned} \tag{3.14}$$

for  $l = 1, \dots, L$ .

(vi) The evolution equations for the  $x$  and  $y$  components of the unknown PVs,  $q_{ijlm}^{(x)}$  and  $q_{ijlm}^{(y)}$  are separately cast as

$$\begin{aligned} \frac{dq_{ijlm}^{(x)}}{dt} &= S^{(x)}(\hat{f}_{xicjc}, \hat{f}_{i\pm\frac{1}{2}jm}, \hat{f}_{xi\pm\frac{1}{2}jm}, \dots), \\ \frac{dq_{ijlm}^{(y)}}{dt} &= S^{(y)}(\hat{g}_{yicjc}, \hat{g}_{ij\pm\frac{1}{2}l}, \hat{g}_{yij\pm\frac{1}{2}l}, \dots), \end{aligned} \tag{3.15}$$

for  $l = 1, \dots, L$  and  $m = 1, \dots, L$ . The operators  $S^{(x)}$  and  $S^{(y)}$  stand for the numerical formulations derived from the moment constraint conditions (3.13) and (3.14).

Finally, time evolution equations of the PV unknowns are obtained by summing up the  $x$  and  $y$  components by

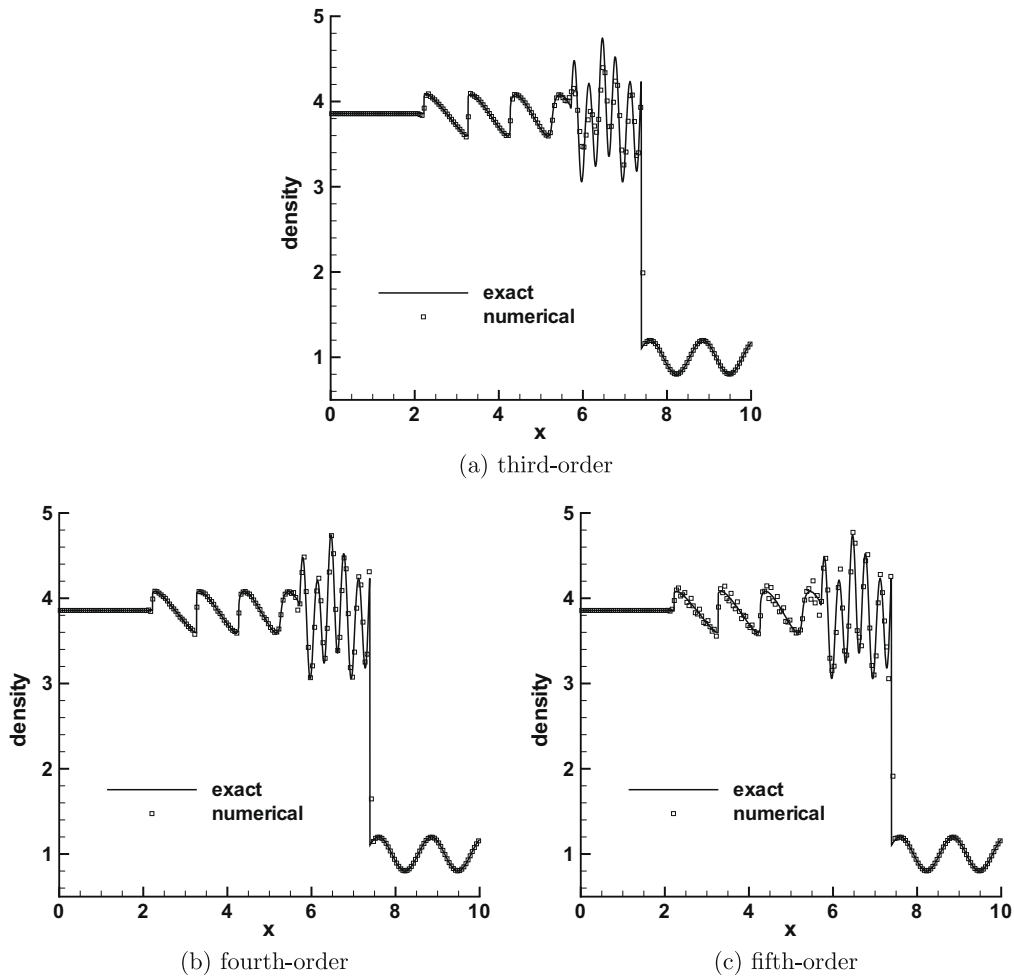


Fig. 16. Numerical results of the shock-turbulence interaction with  $M = 2500$  at  $t = 1.8$ .

$$\frac{dq_{ijlm}}{dt} = \frac{dq_{ijlm}^{(x)}}{dt} + \frac{dq_{ijlm}^{(y)}}{dt} = S^{(x)}(\hat{f}_{xicjc}^{[2]}, \hat{f}_{i\pm\frac{1}{2}jm}, \hat{f}_{xi\pm\frac{1}{2}jm}, \dots) + S^{(y)}(\hat{g}_{yicjc}^{[2]}, \hat{g}_{ij\pm\frac{1}{2}l}, \hat{g}_{yij\pm\frac{1}{2}l}, \dots), \tag{3.16}$$

for  $l = 1, \dots, L$  and  $m = 1, \dots, L$ .

It is easy to show that the volume-integrated average (VIA moment),

$$\bar{q}_{ij} = \left( \int_{\delta x_i} \int_{\delta y_j} q(x, y, t) dx dy \right) / (\Delta x_i \Delta y_j)$$

is numerically conserved. In fact, the total 2D VIA is the summation of the averages over the line segments,  $\bar{q}_{ij}^{(x)}(y), \bar{q}_{ij}^{(y)}(x)$  in  $x$  and  $y$  directions, i.e.

$$\frac{d\bar{q}_{ij}}{dt} = \frac{1}{\Delta y_j} \int_{\delta y_j} \frac{d\bar{q}_{ij}^{(x)}(y)}{dt} dy + \frac{1}{\Delta x_i} \int_{\delta x_i} \frac{d\bar{q}_{ij}^{(y)}(x)}{dt} dx. \tag{3.17}$$

The integrals  $\bar{q}_{ij}^{(x)}(y) = \int_{\delta x_i} q(x, y, t) dx / \Delta x_i$  and  $\bar{q}_{ij}^{(y)}(x) = \int_{\delta y_j} q(x, y, t) dy / \Delta y_j$  are then cast into a discretized form as

$$\frac{d\bar{q}_{ij}}{dt} = \sum_{m=1}^L \omega_m \frac{d\bar{q}_{ijm}^{(x)}}{dt} + \sum_{l=1}^L \omega_l \frac{d\bar{q}_{ijl}^{(y)}}{dt}, \tag{3.18}$$

where  $\omega$  are the weighting constants for the quadrature. From (3.13) and (3.14), the 2D VIA are evaluated by the total fluxes across the cell boundary by

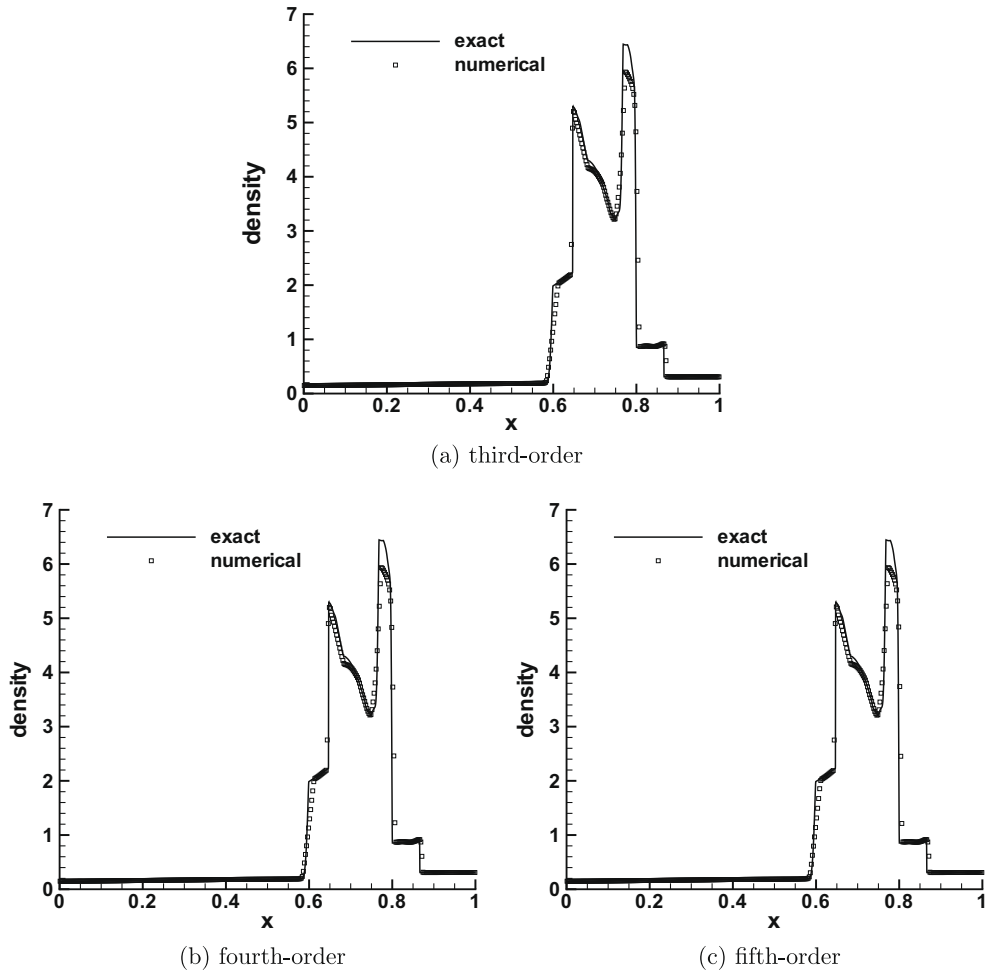


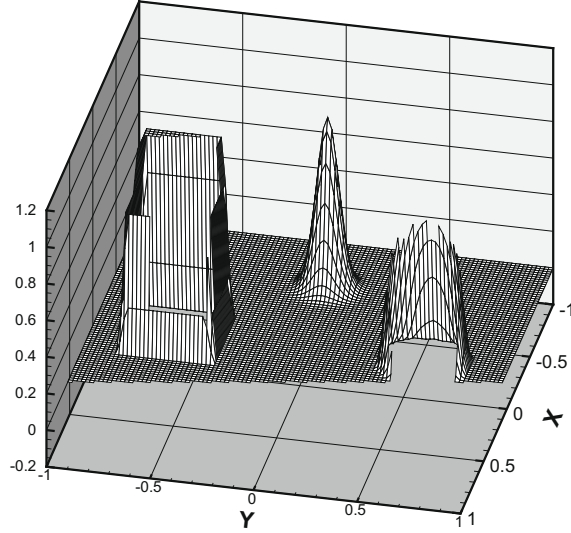
Fig. 17. Numerical results of two interacting blast waves at  $t = 0.038$ .

Table 5  
Numerical errors and convergence rate for the 2D linear scalar equation.

| Accuracy | Mesh | $L_1$ error | $L_1$ order | $L_\infty$ error | $L_\infty$ order |
|----------|------|-------------|-------------|------------------|------------------|
| 3rd      | 10   | 1.25E-2     | –           | 1.93E-2          | –                |
|          | 20   | 1.67E-3     | 2.90        | 2.61E-3          | 2.89             |
|          | 40   | 2.13E-4     | 2.97        | 3.34E-4          | 2.97             |
|          | 80   | 2.68E-5     | 2.99        | 4.21E-5          | 2.99             |
| 4th      | 10   | 1.86E-4     | –           | 2.88E-4          | –                |
|          | 20   | 1.27E-5     | 3.87        | 1.97E-5          | 3.87             |
|          | 40   | 8.19E-7     | 3.95        | 1.28E-6          | 3.94             |
|          | 80   | 5.25E-8     | 3.96        | 8.25E-8          | 3.96             |
| 5th      | 10   | 5.35E-5     | –           | 8.26E-5          | –                |
|          | 20   | 1.72E-6     | 4.96        | 2.71E-6          | 4.93             |
|          | 40   | 5.88E-8     | 4.87        | 9.24E-8          | 4.87             |
|          | 80   | 1.71E-9     | 5.10        | 2.69E-9          | 5.10             |

$$\frac{d\bar{q}_{ij}}{dt} = -\sum_{m=1}^L \omega_m \frac{\hat{f}_{i+\frac{1}{2}jm} - \hat{f}_{i-\frac{1}{2}jm}}{\Delta x_i} - \sum_{l=1}^L \omega_l \frac{\hat{g}_{ij+\frac{1}{2}l} - \hat{g}_{ij-\frac{1}{2}l}}{\Delta y_j} = -\frac{\bar{f}_{i+\frac{1}{2}j} - \bar{f}_{i-\frac{1}{2}j}}{\Delta x_i} - \frac{\bar{g}_{ij+\frac{1}{2}} - \bar{g}_{ij-\frac{1}{2}}}{\Delta y_j}, \tag{3.19}$$

where  $\bar{f}_{i\pm\frac{1}{2}j} = \sum_{m=1}^L \omega_m \hat{f}_{i\pm\frac{1}{2}jm}$  and  $\bar{g}_{ij\pm\frac{1}{2}} = \sum_{l=1}^L \omega_l \hat{g}_{ij\pm\frac{1}{2}l}$  are the numerical fluxes across the 2D cell boundaries in  $x$  and  $y$  directions. The total 2D VIA is then computed by a completely conservative formulation of flux-form.



## 4. Numerical examples

### 4.1. 1D Linear scalar equation

In this subsection numerical tests to advect 1D profiles are solved by the MCV schemes for the following scalar equation

$$q_t + q_x = 0. \quad (4.1)$$

#### 4.1.1. Accuracy test

First, we check the convergence rate of the present method via the grid refinement tests. Smooth distribution are initially given as

$$q(x, 0) = \sin(\pi x) \quad (-1 \leq x \leq 1). \quad (4.2)$$

We refined the grid for the schemes of different orders. The numerical errors at  $t = 2$  (after one period) are examined. Two kinds norms are used to measure the errors, i.e.  $L_1 = \sum_{i=1}^{imax} |q_i^e - q_i^n| / imax$  and  $L_\infty = \max_{1 \leq i \leq imax} (q_i^e - q_i^n)$ , where  $q^e$  is the exact solution and  $q^n$  the numerical one.

The convergence rates of the cell-averaged value from third-order to sixth-order schemes are shown in Table 2. It is observed that the expected convergence rate for different order schemes are obtained.

#### 4.1.2. Advection of a square wave

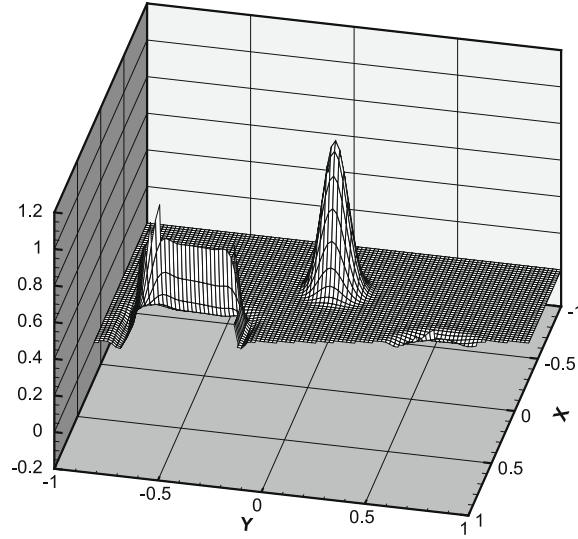
This numerical example is to test the capability of MCV schemes to capture a jump discontinuity. The key for a scheme to resolve a piecewise jump discontinuity essentially depends on the interpolation reconstruction. It is well-known that with increased wave number the Fourier series converges to such a function almost everywhere except at the jump discontinuity where the Gibbs phenomenon of a finite overshoot or undershoot occurs.

We computed the following square wave advected by (4.1),

$$q(x, 0) = \begin{cases} 1 & \text{for } |x| \leq 0.4 \\ 0 & \text{otherwise} \end{cases} \quad (-1 \leq x \leq 1). \quad (4.3)$$

The initial conditions of the unknowns for different order schemes are given in the way described in Section 2.6.

To see how the multi-moment reconstruction approaches the square wave, we ran the test without the limiting projection for one revolution. We use meshes of 200, 133, 100 and 66 cells for third-, fourth-, fifth- and sixth-order MCV schemes separately to maintain the total number of DOFs being roughly 400 for all cases. Here, we share the PV moment on cell boundary without limiting. The numerical solutions are shown in Fig. 3. The multi-moment reconstruction behaves in a manner quite similar to the spectral method. The sixth-order scheme gives the most converged numerical solution compared to other lower order schemes.

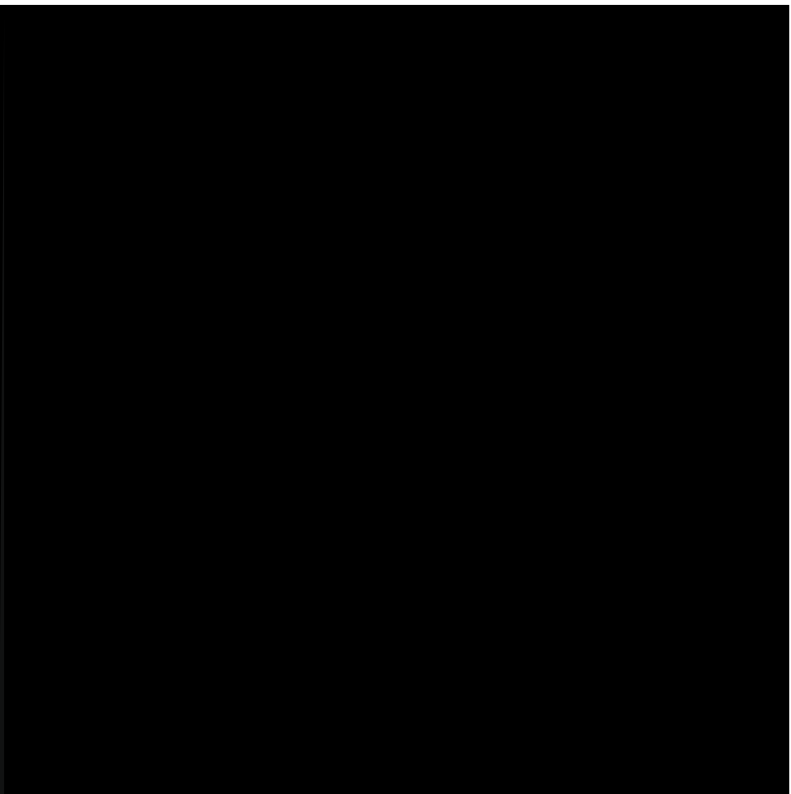
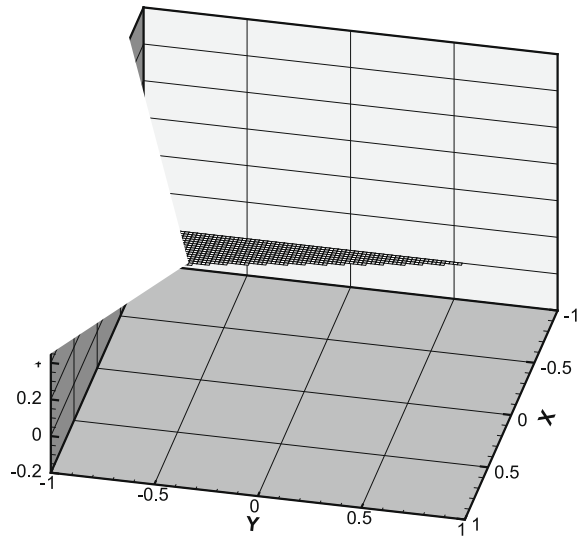


#### 4.1.3. Jiang and Shu's problem [13]

This example was used in [13] to evaluate the capability of the schemes in capturing both discontinuity and smooth solution. The initial distribution includes four types of profiles as

$$q(x, 0) = \begin{cases} \frac{1}{6}(G(x, \beta, z - \delta) + G(x, \beta, z + \delta) + 4G(x, \beta, z)) & \text{for } -0.8 \leq x \leq -0.6, \\ 1 & \text{for } -0.4 \leq x \leq -0.2, \\ 1 - |10(x - 0.1)| & \text{for } 0.0 \leq x \leq 0.2, \\ \frac{1}{6}(F(x, \alpha, a - \delta) + F(x, \alpha, a + \delta) + 4F(x, \alpha, a)) & \text{for } 0.4 \leq x \leq 0.6, \\ 0, & \text{otherwise,} \end{cases} \quad (4.4)$$

where the functions are defined by  $G(x, \beta, z) = e^{-\beta(x-z)^2}$  and  $F(x, \alpha, a) = \sqrt{\max(1 - \alpha^2(x-a)^2, 0)}$ . The constants are specified as  $a = 0.5$ ,  $z = -0.7$ ,  $\delta = 0.005$ ,  $\alpha = 10$  and  $\beta = \log 2 / (36\delta^2)$ . We computed over a 200-cell mesh with and without the use of the slope limiter.

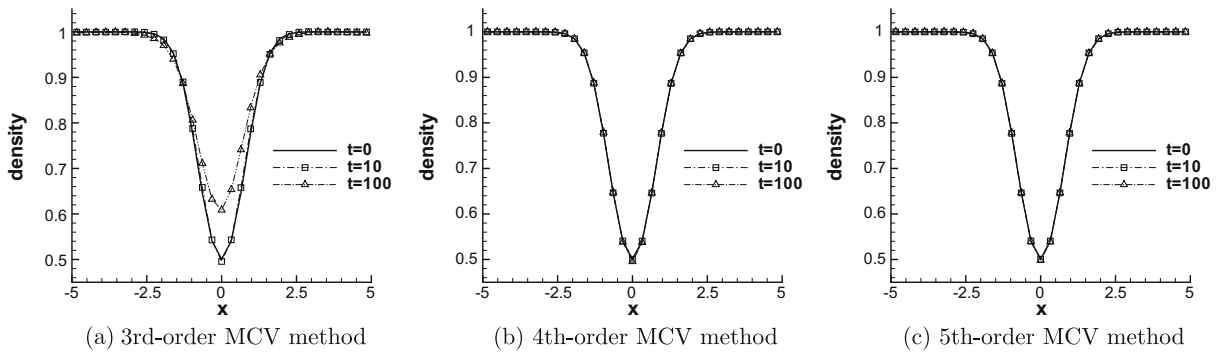




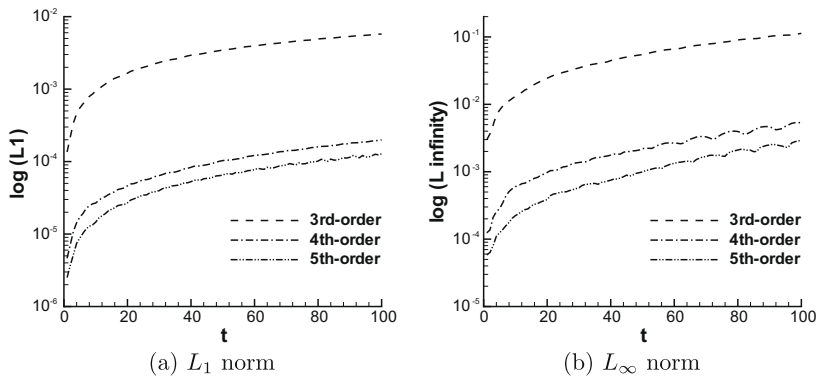
**Table 7**

Numerical errors and convergence rate for the vortex propagation test at  $t = 2$ .

| Accuracy | $h$  | $L_1$ error | $L_1$ order | $L_\infty$ error | $L_\infty$ order |
|----------|------|-------------|-------------|------------------|------------------|
| 3rd      | 1/2  | 1.63E-4     | –           | 1.10E-2          | –                |
|          | 1/4  | 2.72E-5     | 2.58        | 1.91E-3          | 2.53             |
|          | 1/8  | 3.73E-6     | 2.87        | 2.76E-4          | 2.79             |
|          | 1/16 | 4.80E-7     | 2.96        | 3.55E-5          | 2.96             |
| 4th      | 1/2  | 9.49E-6     | –           | 5.92E-4          | –                |
|          | 1/4  | 6.42E-7     | 3.89        | 5.13E-5          | 3.53             |
|          | 1/8  | 3.89E-8     | 4.04        | 3.16E-6          | 4.02             |
|          | 1/16 | 2.38E-9     | 4.03        | 1.78E-7          | 4.15             |
| 5th      | 1/2  | 6.78E-6     | –           | 3.41E-4          | –                |
|          | 1/4  | 2.52E-7     | 4.75        | 1.88E-5          | 4.18             |
|          | 1/8  | 8.25E-9     | 4.93        | 7.44E-7          | 4.66             |
|          | 1/16 | 2.67E-10    | 4.95        | 2.44E-8          | 4.93             |



**Fig. 21.** Density profiles of the 2D vortex propagation problem with  $N_x = N_y = 32$  on  $y = 0$  cross section at  $t = 10$  and  $t = 100$ .



**Fig. 22.** Time evolution of the numerical errors for 2D vortex propagation problem from the third-, fourth- and fifth-order MCV methods.

The numerical results at  $t = 2$  (after one period) without limiter is plotted in Fig. 4. It is clearly observed that a higher order scheme produces better results for both discontinuity and smooth region. The sixth-order MCV scheme gives the best results, not only for the smooth solution but also for the discontinuous jumps which were revolved within two mesh cells as shown in the previous test.

We continued the computations until  $t = 20$  (after 10 period) and compared the numerical results from the third-order and the sixth-order MCV schemes. The numerical results are shown in Fig. 5. Compared to the third-order one, which is more dissipative, the sixth-order scheme looks much superior for long term integration.

Numerical results with a TVB limiter for  $M = 200$  and  $M = 0$  are shown in Figs. 6 and 7, respectively. When the limiting projection applied to eliminate the spurious oscillation, the difference in the numerical results from the schemes of different

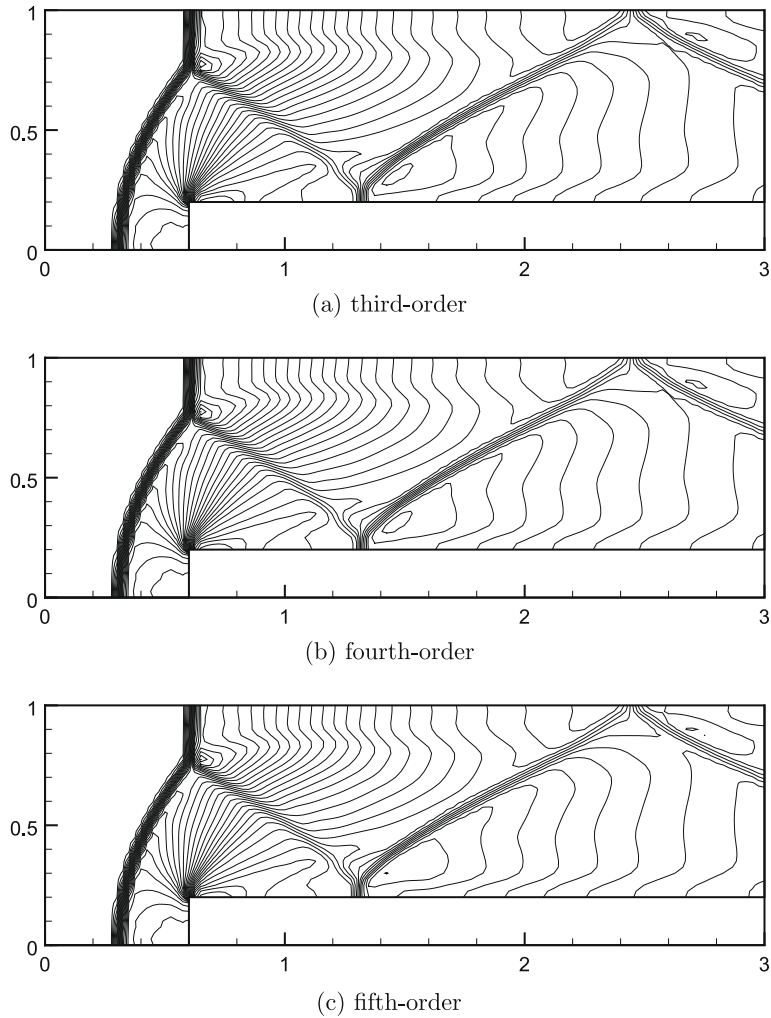


Fig. 23. Numerical results of Mach 3 wind tunnel with a step at  $t = 4.0$  with  $120 \times 40$  cells.

orders is not much significant. Furthermore, the parameter  $M$  in the TVB scheme affects the numerical results, but is problem-dependent. As in other high-order schemes, the limiting projection is still a topic worth further investigations.

4.2. 1D nonlinear inviscid Burger's equation

The 1D nonlinear Burger's equation without the effect of viscosity is written as,

$$u_t + \left(\frac{1}{2}u^2\right)_x = 0. \tag{4.5}$$

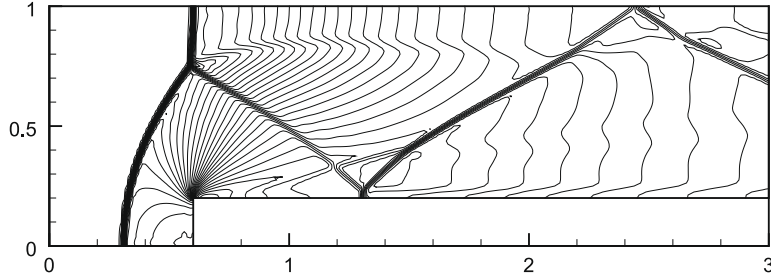
In this example, we compute the numerical flux at cell boundary  $x_{i-\frac{1}{2}}$  by

$$\hat{f}_{xi-\frac{1}{2}}^{[k]} = \frac{1}{2} \left( f_{xi-\frac{1}{2}}^{[k]-} + f_{xi-\frac{1}{2}}^{[k]+} - \text{sgn}(\alpha_{i-\frac{1}{2}}) (f_{xi-\frac{1}{2}}^{[k]+} - f_{xi-\frac{1}{2}}^{[k]-}) \right), \tag{4.6}$$

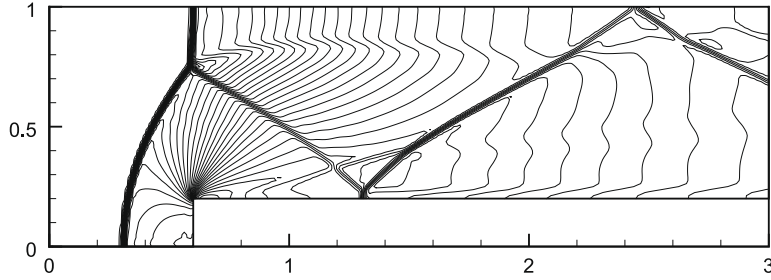
where  $\alpha_{i-\frac{1}{2}} = (\bar{u}_{i-1} + \bar{u}_i)/2$ . As an alternative, the multi-moment reconstruction is built in terms of the flux function  $f = (u^2/2)$  instead of the conservative variable  $u$ .

A smooth profile

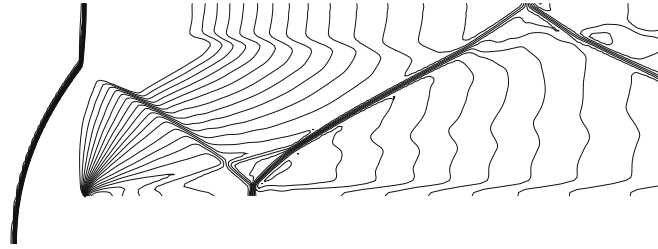
$$u(x, 0) = 0.5 + \sin(\pi x) \quad (0 \leq x \leq 2) \tag{4.7}$$



(a) third-order



(b) fourth-order



was used as the initial condition. As can be expected, shock waves will develop in the course of time due to the nonlinearity. In order to evaluate the accuracy, or the convergence rate in respect to grid refinement, we ran the computation up to  $t = 0.5\pi$  before the formation of the shocks. The mesh are gradually refined by 20, 40, 80, 160 and 320 cells. The numerical errors in  $L_1$  and  $L_\infty$  norms as well as the convergence rate for third-, fourth-, fifth- and sixth-order MCV schemes are shown in Table 3. It is observed that a roughly spectral decline in the numerical errors is obtained when a finer mesh is used.

We continued the computation on the 40-cell mesh until  $t = 1.5\pi$ . A TVB limiter is used to remove the numerical oscillation in the presence of the shock waves. The numerical results of 3rd and 5th schemes are plotted in Fig. 8. To further examine the numerical errors and the effects of the limiting projection in the case where a shock is developed, we show the error distributions of third- and fifth-order schemes in Fig. 9. It is found that the convergence due to both  $h$ -refinement and  $p$ -refinement are obtained. In the vicinity of discontinuity where the TVB limiter switches the reconstruction to lower order, however, the orders of the schemes on all meshes degrade. It reveals that the TVB limiter maintains to some extent the numerical accuracy of a high order scheme in the smooth region, but the determination of the TVB parameter remains a problem requiring further investigation.

#### 4.3. 1D Euler equations

The 1D Euler equations are written as

$$q = (\rho, \rho u, E)^T, \quad f(q) = (\rho u, \rho u^2 + p, u(E + p))^T, \quad (4.8)$$



order,

from  
the report

ing from the

$x \leq 1$ .

fifth-order

r being  $M = 20$

al outputs (0.6) re-

blotted out the spot

and  $M = 100$  cases.

ront of the expansion

$M = 100$  allows a less res-

the expansion (as in the

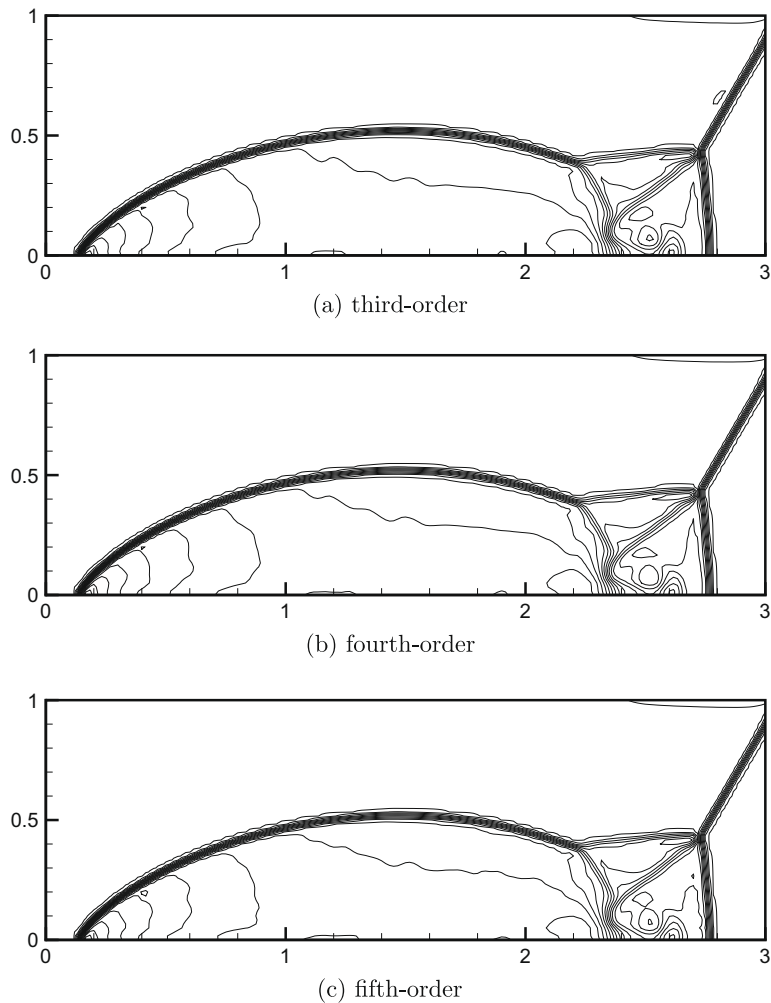


Fig. 27. Numerical results of double Mach reflection at  $t = 0.2$  with  $180 \times 60$  cells.

#### 4.3.4. Shock–turbulence interaction [24]

Second example in the Euler equations was proposed by Shu [24], where the initial conditions are

$$(\rho_0, u_0, p_0) = \begin{cases} (3.857148, 2.629369, 10.333333) & \text{for } 0 \leq x \leq 1 \\ (1 + 0.2 \sin(5x - 5), 0, 1) & \text{otherwise} \end{cases} \quad (0 \leq x \leq 10). \quad (4.12)$$

We computed this example with two values of  $M$ , i.e.  $M = 500$  and  $M = 2500$ .

The results of density at  $t = 1.8$  with 200 cells are displayed in Figs. 15 and 16. As expected, when  $M = 500$ , higher order scheme can well resolve the solution around smooth region. When  $M = 2500$ , the fourth-order MCV scheme reproduces better the smooth profile compared to the third-order scheme. However, the result from the fifth-order scheme has some numerical oscillations. This might partly due to the limiting procedure, and implies that higher order scheme needs other type of limiter such as those using the WENO reconstruction [16,13].

#### 4.3.5. Two interacting blast waves [34]

The problem including strong interacting shock waves is tested as well. The initial conditions with very large pressure difference are defined as

$$(\rho_0, u_0, p_0) = \begin{cases} (1, 0, 1000) & \text{for } 0 \leq x \leq 0.1 \\ (1, 0, 0.01) & \text{for } 0.1 \leq x \leq 0.9 \\ (1, 0, 100) & \text{otherwise} \end{cases} \quad (0 \leq x \leq 1). \quad (4.13)$$







#### 4.5. 2D Euler equations

The 2D Euler equations are given by

$$\begin{aligned} q &= (\rho, \rho u, \rho v, E)^T, \\ f(q) &= (\rho u, \rho u^2 + p, \rho u v, u(E + p))^T, \\ g(q) &= (\rho v, \rho u v, \rho v^2 + p, v(E + p))^T, \end{aligned} \tag{4.17}$$

where  $\rho$  is the density,  $(u, v)$  the velocity components in  $x$  and  $y$  directions,  $p$  the pressure and  $E$  the total energy given by

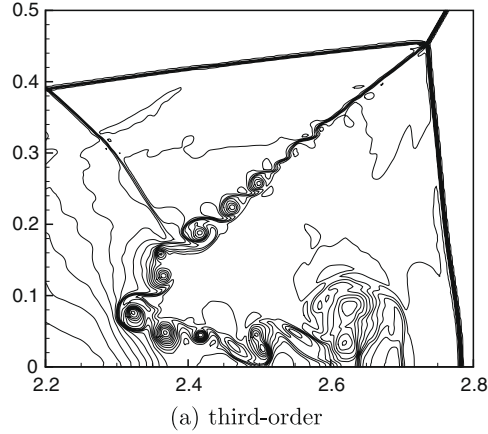
$$E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho (u^2 + v^2). \tag{4.18}$$

We apply Roe's approximate Riemann solver with Harten's entropy fix [9];

$$|\lambda_m^*| = \begin{cases} |\lambda_m| & \text{if } |\lambda_m| \geq \epsilon, \\ (\lambda_m^2 + \epsilon^2)/(2\epsilon), & \text{otherwise,} \end{cases} \tag{4.19}$$

where  $\lambda_m^*$  is the modified  $m$  component of the characteristics velocity and  $\epsilon$  is a small positive.





In order to visually illustrate the accuracy of the numerical solutions, we re-computed the test on the computational domain of  $[-5, 5]^2$  with periodical boundary conditions, so that the vortex returns to its initial position at an interval of  $t = 10$ . The cell number is set to  $N_x = N_y = 32$ . We plot the density profiles on  $y = 0$  for third-, fourth- and fifth-order MCV methods at  $t = 10$  (1 cycle) and  $t = 100$  (10 cycles) in Fig. 21. All the results at  $t = 10$  are perfectly preserved, however, the result of third-order MCV method looks dissipative after a long term run at  $t = 100$ . The results of the fourth- and fifth-order MCV methods at  $t = 100$ , on the other hand, maintain visually identical to the analytical solution (the initial profile). The time evolution of the numerical errors measured by  $L_1$  and  $L_\infty$  norms are shown in Fig. 22, which manifests again the numerical accuracy of the schemes.

#### 4.5.3. Mach 3 wind tunnel with a step [34]

This problem was originally reported in [34] to verify 2D shock capturing schemes. The uniform compressible flow of Mach 3 attacks a step from left in a tunnel with a size of  $[0, 3] \times [0, 1]$ . An inflow condition is imposed on the left boundary, and an outflow condition on the right boundary. The upper and lower boundaries are reflective walls. We run the computation until  $t = 4$  with a TVB *superbee* limiter where parameters  $M = 5$  and  $\beta = 1.8$  in (2.37) and (2.34) are used.

The numerical results with the mesh number increased from  $120 \times 40$  to  $960 \times 320$  from the third-, fourth- and fifth-order MCV schemes are shown in Figs. 23–26. In our simulation, the non-physical Mach stem problem caused by the singular point at the step corner does not appear even without any modifications. It is also noted that the fifth-order MCV can better resolves the vortex structures generated at the joint point of the three shocks, as shown in Fig. 26. Our results are competitive to the results of the high-order DG method [7].

#### 4.5.4. Double Mach Reflection [34]

This is also a widely used test problem to validate schemes for capturing both shock and vortex structures. The computation domain is set as  $[0, 3.2] \times [0, 1]$ . Compressible flow of Mach 10 moves obliquely and is reflected on the bottom wall at the section  $x \geq 1/6$ . The left and top boundary conditions are imposed by the post-shock conditions and right boundary is

specified as outflow condition. We use the uniform grid with the meshes of different resolutions. The computation is carried out up to  $t = 0.2$ .

The numerical results are shown in Figs. 27–30, and the blew-up solutions for the vortex structures on the fine meshes are shown in Figs. 31 and 32. With increased mesh resolution, the vortex structures and instability along the slip lines are resolved adequately. It is observed that the fifth-order scheme gives the best solution, as shown in Fig. 31. Solutions with a  $1440 \times 480$  mesh in Fig. 32 reproduce the complicated flow structures quite similar to that of the ninth-order WENO scheme given in [20].

**5. Concluding remarks**

We have proposed and tested a conservative and high-order finite volume method, so-called multi-moment constrained finite volume (MCV) method, for the 1D and 2D hyperbolic conservation laws. The unknowns are the point values defined within single cell at equally spaced points. The time evolution equations used to update the unknowns are derived from a set of constraint conditions imposed on different kinds of moments, i.e. the cell-averaged value and the point-wise value of the state variable and its derivatives. The constraint on the cell-average guarantees the numerical conservativeness of the method. Most constraint conditions are imposed on the cell boundaries, where the numerical flux and its derivatives are solved as general Riemann problems. A multi-moment Lagrange interpolation reconstruction converts the evolution equations of the moments to those of the unknowns.

The equally spaced points are used to defined the unknowns in the present method, yielding a more simple and efficient formulation to derive the evolution equations from the moment constraint conditions. The MCV method allows larger CFL numbers for computational stability compared to other high-order schemes using local reconstructions. We have given the MCV schemes from third to sixth-orders in this paper. The numerical tests show that the MCV method is able to produce competitive results and is promising as a high-order method for CFD applications. The basic idea reported in this paper can be easily extended to unstructured mesh. We will report the formulation on unstructured mesh in a separate paper.

**Appendix A. A The multi-moment constrained Lagrange interpolation**

In an MCV scheme, multi-moment constrained Lagrange polynomial (MC-Lagrange polynomial) is used for reconstruction. Different from the conventional Lagrange interpolation, the values at the interior points are determined from constraints in terms of multi-moments.

We use a local coordinate system  $\xi \in [0, 1]$  that is connected to the real mesh cell  $x \in [x_{i-1/2}, x_{i+1/2}]$  by

$$\xi = \frac{x - x_{i-1/2}}{\Delta x_i}, \tag{A.1}$$

where  $\Delta x_i = x_{i+1/2} - x_{i-1/2}$ .

Consider a set of sample points  $(\xi_1(=0), \xi_2, \dots, \xi_L(=1))$  equally spaced in  $[0, 1]$ , we denote the samples of function  $q(\xi)$  by  $q_1, q_2, \dots, q_L$ . An MC-Lagrange polynomial is in the form of Lagrange polynomial,

$$\mathcal{P}(\xi) = \sum_{l=1}^L c_l q_l, \tag{A.2}$$

where

$$c_l = \prod_{j=1, j \neq l}^L \frac{\xi - \xi_j}{\xi_l - \xi_j} \tag{A.3}$$

is the basis function of the Lagrange interpolation.

Given the constraints in terms of the multi-moments as follows

$$\begin{cases} \mathcal{P}(\xi_1) = q_1, \\ \mathcal{P}(\xi_L) = q_L, \\ \int_0^1 \mathcal{P}(\xi) d\xi = \bar{q}, \\ \frac{\partial^k}{\partial \xi^k} \mathcal{P}(\xi_1) = \frac{\partial^k}{\partial \xi^k} q(\xi_1) = q_{\xi_1}^{[k]}, \\ \frac{\partial^k}{\partial \xi^k} \mathcal{P}(\xi_L) = \frac{\partial^k}{\partial \xi^k} q(\xi_L) = q_{\xi_L}^{[k]}, \\ k = 1, 2, \dots \end{cases} \tag{A.4}$$

the values at the cell ends, i.e.  $q_1$  and  $q_L$  are determined from the first two equations in (A.4) in the way same as the conventional Lagrange interpolation. The values at the interior points  $(q_2, q_3, \dots, q_{L-1})$  are found from the rest relationships in (A.4), which is essentially different from sampling the data at the discrete points  $(q(\xi_2), q(\xi_3), \dots, q(\xi_{L-1}))$  as in the conventional Lagrange interpolation.

We give two examples of the MC-Lagrange polynomial as below,

• Quadratic MC-Lagrange polynomial.

Three DOFs in a quadratic MC-Lagrange polynomial ( $q_1, q_2, q_3$ ) are uniquely determined by following constraints,

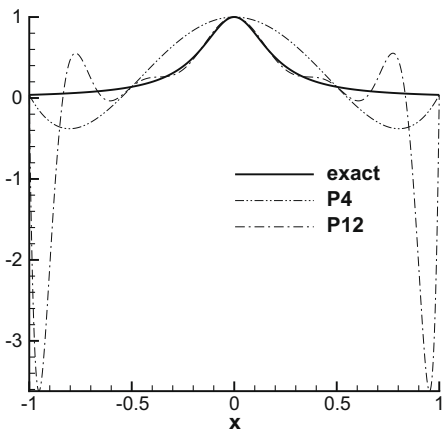
$$\begin{cases} \mathcal{P}(\xi_1) = q_1, \\ \mathcal{P}(\xi_3) = q_3, \\ \int_0^1 \mathcal{P}(\xi) d\xi = \bar{q}. \end{cases} \tag{A.5}$$

The only interior point value is

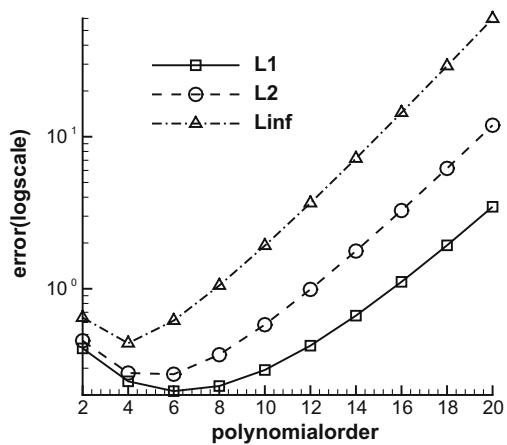
$$q_2 = (6\bar{q} - q_1 - q_3)/4. \tag{A.6}$$

• Quartic MC-Lagrange polynomial.

Five constraint conditions are given as

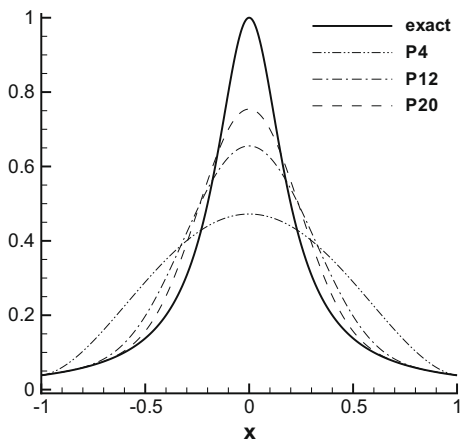


(a) 4th and 12th-order Lagrange interpolations

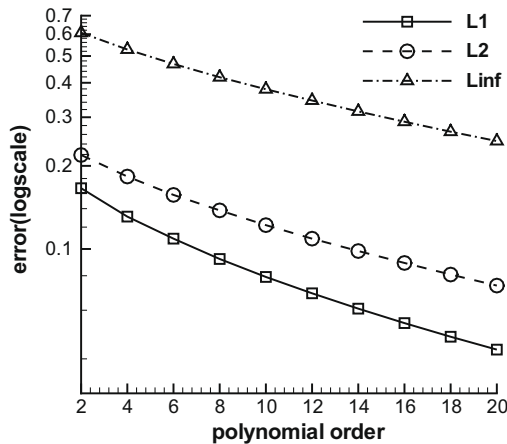


(b) Errors for polynomials of different orders

Fig. 33. Interpolation for fitting  $q(x) = 1/(1 + 25x^2)$  by conventional Lagrange reconstruction.



(a) Interpolation with 4th, 12th and 20th-order MC-Lagrange polynomials



(b) Errors for polynomials of different orders

Fig. 34. Interpolation for fitting  $q(x) = 1/(1 + 25x^2)$  by MC-Lagrange reconstruction.

$$\begin{cases} \mathcal{P}(\xi_1) = q_1, \\ \mathcal{P}(\xi_5) = q_5, \\ \int_0^1 \mathcal{P}(\xi) d\xi = \bar{q}, \\ \frac{\partial}{\partial \xi} \mathcal{P}(\xi_1) = \frac{\partial}{\partial \xi} q(\xi_1) = q_{\xi_1}, \\ \frac{\partial}{\partial \xi} \mathcal{P}(\xi_5) = \frac{\partial}{\partial \xi} q(\xi_5) = q_{\xi_5}. \end{cases} \quad (\text{A.7})$$

The three interior point values are

$$\begin{cases} q_2 = \frac{51}{128} q_1 + \frac{27}{512} q_{\xi_1} - \frac{95}{256} q_5 + \frac{21}{512} q_{\xi_5} + \frac{135}{128} \bar{q}, \\ q_3 = -\frac{3}{8} q_1 - \frac{1}{32} q_{\xi_1} - \frac{7}{16} q_5 + \frac{1}{32} q_{\xi_5} + \frac{15}{8} \bar{q}, \\ q_4 = -\frac{61}{128} q_1 - \frac{21}{512} q_{\xi_1} + \frac{81}{256} q_5 - \frac{27}{512} q_{\xi_5} + \frac{135}{128} \bar{q}. \end{cases} \quad (\text{A.8})$$

Imposing the multi-moment constraints, such as the cell-average and the derivatives of the function, in the process of determining interpolation function, the MC-Lagrange polynomial is very stable and less oscillating. To show this, we approximate function

$$q(x) = 1/(1 + 25x^2) \quad (\text{A.9})$$

which makes the conventional Lagrange interpolation fail due the well-known Runge phenomenon.

The VIA moment as well as the PV moment and derivatives of different orders at the domain boundary are used as the multi-moment constraints to determine the MC-Lagrange polynomial.

We include the results of the conventional Lagrange interpolation for comparison in Fig. 33. As well-known, the Lagrange interpolation that samples the values at equidistance points suffers a large oscillation at two ends of the domain, thus fails to converge to the original function, and even worse, the interpolation becomes unstable as one increases the order of the polynomial.

The MC-Lagrange polynomial, as shown in Fig. 34, demonstrates a well regulated behavior and converges to the original function at almost a linear rate.

Finally, we should note that the multi-moment constraining is a general concept. We by no means limit ourselves to the Lagrange interpolation and constraints discussed here, and other alternatives are worthy of further exploration.

## References

- [1] K. Van den Abeele, C. Lacor, Z.J. Wang, On the connection between the spectral volume and the spectral difference method, *J. Comput. Phys.* 227 (2007) 877–885.
- [2] R. Akoh, S. li, F. Xiao, A CIP/multi-moment finite volume method for shallow water equations with source terms, *Int. J. Numer. Method Fluid* 56 (2008) 2245–2270.
- [3] C.G. Chen, F. Xiao, Shallow water model on spherical-cubic grid by multi-moment finite volume method, *J. Comput. Phys.* 227 (2008) 5019–5044.
- [4] B. Cockburn, C.W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws II: General framework, *Math. Comput.* 52 (1989) 411–435.
- [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, *J. Comput. Phys.* 84 (1989) 90–113.
- [6] B. Cockburn, S. Hou, C.W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case, *Math. Comput.* 54 (1990) 545–581.
- [7] B. Cockburn, C.W. Shu, The Runge–Kutta discontinuous Galerkin method for conservation laws V: The multidimensional systems, *J. Comput. Phys.* 141 (1998) 199–224.
- [8] E. Garnier, P. Sagaut, M. Deville, A class of explicit ENO filters with application to unsteady flows, *J. Comput. Phys.* 170 (2001) 184–204.
- [9] A. Harten, High resolution schemes for hyperbolic conservation laws, *J. Comput. Phys.* 49 (1983) 357–393.
- [10] S. li, M. Shimuta, F. Xiao, A 4th-order and single-cell-based advection scheme on unstructured grids using multi-moments, *Comput. Phys. Commun.* 173 (2005) 17–33.
- [11] S. li, F. Xiao, CIP/multi-moment finite volume method for Euler equations: A semi-Lagrangian characteristic formulation, *J. Comput. Phys.* 222 (2007) 849–871.
- [12] Y. Imai, T. Aoki, K. Takizawa, Conservative form of interpolated differential operator scheme for compressible and incompressible fluid dynamics, *J. Comput. Phys.* 227 (2008) 2263–2285.
- [13] G. Jiang, C.W. Shu, Efficient implementation of weighted ENO schemes, *J. Comput. Phys.* 126 (1996) 202–228.
- [14] D.A. Kopriva, J.H. Koliass, A conservative staggered-grid Chebyshev multidomain method for compressible flows, *J. Comput. Phys.* 125 (1996) 244–261.
- [15] X.L. Li, D.H. Chen, X.D. Peng, K. Takahashi, F. Xiao, A multi-moment finite volume shallow water model on Yin-Yang overset spherical grid, *Mon. Weather Rev.* 136 (2008) 3066–3086.
- [16] X.D. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, *J. Comput. Phys.* 115 (1994) 200–212.
- [17] Y. Liu, M. Vinokur, Z.J. Wang, Spectral difference method for unstructured grids I. Basic formulation, *J. Comput. Phys.* 216 (2006) 780–801.
- [18] J. Qiu, C.W. Shu, Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method: one-dimensional case, *J. Comput. Phys.* 193 (2003) 115–135.
- [19] P.L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* 43 (1981) 357–372.
- [20] J. Shi, Y.T. Zhang, C.W. Shu, Resolution of high order WENO schemes for complicated flow structures, *J. Comput. Phys.* 186 (2003) 690–696.
- [21] C.W. Shu, TVB uniformly high-order schemes for conservation laws, *Math. Comput.* 49 (1987) 105–121.
- [22] C.W. Shu, Total-variation-diminishing time discretizations, *SIAM J. Sci. Stat. Comput.* 9 (1988) 1073–1084.
- [23] C.W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, ICASE Report No. 97-65, 1997.
- [24] C.W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock capturing schemes, II, *J. Comput. Phys.* 83 (1989) 32–78.
- [25] G. Sod, A survey of several finite difference methods for systems of non-linear conservation laws, *J. Comput. Phys.* 27 (1978) 1–31.
- [26] R. Tanaka, T. Nakamura, T. Yabe, Constructing exactly conservative scheme in a non-conservative form, *Comput. Phys. Commun.* 126 (2000) 232–243.
- [27] V.A. Titarev, E.F. Toro, ADER: arbitrary high order Godunov approach, *J. Sci. Comput.* 17 (2002) 609–618.

- [28] E.F. Toro, V.A. Titarev, Derivative Riemann solvers for systems of conservation laws and ADER methods, *J. Comput. Phys.* 212 (2006) 150–165.
- [29] Z.J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation, *J. Comput. Phys.* 178 (2002) 210–251.
- [30] Z.J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids II: Extension to two-dimensional scalar equation, *J. Comput. Phys.* 179 (2002) 665–697.
- [31] Z.J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids III: One-dimensional systems and partition optimization, *J. Sci. Comput.* 20 (2004) 137–157.
- [32] Z.J. Wang, Y. Liu, L. Zhang, Spectral (finite) volume method for conservation laws on unstructured grids IV: Extension to two-dimensional systems, *J. Comput. Phys.* 194 (2004) 716–741.
- [33] Z.J. Wang, Yen Liu, Georg May, Antony Jameson, Spectral difference method for unstructured grids II. Extension to the Euler equations, *J. Sci. Comput.* 32 (2007) 45–71.
- [34] P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, *J. Comput. Phys.* 54 (1984) 115–173.
- [35] F. Xiao, T. Yabe, Completely conservative and oscillation-less semi-Lagrangian schemes for advection transportation, *J. Comput. Phys.* 170 (2001) 498–522.
- [36] F. Xiao, T. Yabe, X. Peng, H. Kobayashi, Conservative and oscillation-less atmospheric transport schemes based on rational functions, *J. Geophys. Res.* 107 (2002) 4609–4619.
- [37] F. Xiao, Profile-modifiable conservative transport schemes and a simple multi integrated moment formulation for hydrodynamics, in: S. Armfield, P. Morgan, K. Srinivas (Eds.), *Computational Fluid Dynamics 2002*, Springer, 2003, p. 106.
- [38] F. Xiao, A. Ikebata, An efficient method for capturing free boundary in multi-fluid simulations, *Int. J. Numer. Method Fluid* 42 (2003) 187–210.
- [39] F. Xiao, Unified formulation for compressible and incompressible flows by using multi integrated moment method: one-dimensional inviscid compressible flow, *J. Comput. Phys.* 195 (2004) 629–654.
- [40] F. Xiao, A. Ikebata, T. Hasegawa, Numerical simulations of free-interface fluids by a multi integrated moment method, *Comput. Struct.* 83 (2005) 409–423.
- [41] F. Xiao, R. Akoh, S. li, Unified formulation for compressible and incompressible flows by using multi-integrated moments II: Multi-dimensional version for compressible and incompressible flows, *J. Comput. Phys.* 213 (2006) 31–56.
- [42] F. Xiao, S. li, CIP/multi-moment finite volume method with arbitrary order of accuracy, *JSCE J.* 12 (2007) 409–415.
- [43] T. Yabe, T. Aoki, A universal solver for hyperbolic-equations by cubic-polynomial interpolation. 1. One-dimensional solver, *Comput. Phys. Commun.* 66 (1991) 219–232.
- [44] T. Yabe, R. Tanaka, T. Nakamura, F. Xiao, Exactly conservative semi-Lagrangian scheme (CIP-CSL) in one dimension, *Mon. Weather Rev.* 129 (2001) 332–344.
- [45] T. Yabe, F. Xiao, T. Utsumi, The constrained interpolation profile method for multiphase analysis, *J. Comput. Phys.* 169 (2001) 556–593.