

A high order ENO conservative Lagrangian scheme for the compressible Euler equations

Juan Cheng¹ and Chi-Wang Shu²

Abstract

We develop a class of Lagrangian schemes for solving the Euler equations of compressible gas dynamics both in the Cartesian and in the cylindrical coordinates. The schemes are based on high order essentially non-oscillatory (ENO) reconstruction. They are conservative for the density, momentum and total energy, can maintain formal high order accuracy both in space and time and can achieve at least uniformly second order accuracy with moving and distorted Lagrangian meshes, are essentially non-oscillatory, and have no parameters to be tuned for individual test cases. One and two dimensional numerical examples in the Cartesian and cylindrical coordinates are presented to demonstrate the performance of the schemes in terms of accuracy, resolution for discontinuities, and non-oscillatory properties.

Keywords: Lagrangian scheme; high order accuracy; conservative scheme; ENO reconstruction; compressible Euler equations; ALE method

¹Institute of Applied Physics and Computational Mathematics, Beijing 100088, China. E-mail: cheng_juan@iapcm.ac.cn. Research is supported in part by NSFC grant 10572028. Additional support is provided by the National Basic Research Program of China under grant 2005CB321702, by the Foundation of National Key Laboratory of Computational Physics under grant 9140C6902010603 and by the National Hi-Tech Inertial Confinement Fusion Committee of China.

²Division of Applied Mathematics, Brown University, Providence, RI 02912. E-mail: shu@dam.brown.edu. Research is supported in part by NSFC grant 10671190 and by the Chinese Academy of Sciences during his visit to the University of Science and Technology of China and the Institute of Computational Mathematics and Scientific / Engineering Computing. Additional support is provided by ARO grant W911NF-04-1-0291 and NSF grant DMS-0510345.

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE 2007		2. REPORT TYPE		3. DATES COVERED 00-00-2007 to 00-00-2007	
4. TITLE AND SUBTITLE A high order ENO conservative Lagrangian scheme for the compressible Euler equations				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Brown University, Division of Applied mathematics, Providence, RI, 02912				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT We develop a class of Lagrangian schemes for solving the Euler equations of compressible gas dynamics both in the Cartesian and in the cylindrical coordinates. The schemes are based on high order essentially non-oscillatory (ENO) reconstruction. They are conservative for the density, momentum and total energy, can maintain formal high order accuracy both in space and time and can achieve at least uniformly second order accuracy with moving and distorted Lagrangian meshes, are essentially non-oscillatory, and have no parameters to be tuned for individual test cases. One and two dimensional numerical examples in the Cartesian and cylindrical coordinates are presented to demonstrate the performance of the schemes in terms of accuracy, resolution for discontinuities, and non-oscillatory properties.					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 49	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

1 Introduction

In numerical simulations of multidimensional fluid flow, there are two typical choices: a Lagrangian framework, in which the mesh moves with the local fluid velocity, and an Eulerian framework, in which the fluid flows through a grid fixed in space. More generally, the motion of the grid can also be chosen arbitrarily, this method is called the Arbitrary Lagrangian-Eulerian method (ALE; cf. [14, 3, 20, 16, 23]). Most ALE algorithms consist of three phases, a Lagrangian phase in which the solution and the grid are updated, a rezoning phase in which the nodes of the computational grid are moved to a more optimal position and a remapping phase in which the Lagrangian solution is transferred to the new grid.

In this paper, we focus on computational hydrodynamic methods for the Euler equations where the mesh moves with the flow velocity. Such methods, therefore, imply the use of distorted or nonuniform meshes. Particular examples include the Lagrangian methods, or the ALE methods which contain a Lagrangian phase.

Lagrangian methods are widely used in many fields for multi-material flow simulations such as astrophysics and computational fluid dynamics (CFD), due to their distinguished advantage of being able to capture the material interface sharply. Comparing with Eulerian methods, Lagrangian methods avoid a source of numerical error due to the advection terms in the conservation equations. For this reason, Lagrangian methods are frequently preferred in one-dimensional computations where mesh distortion plays no role. Even though the Euler equations are much simpler in the Lagrangian framework as they do not contain the advection terms, in two or more space dimensions they are actually more difficult to solve since the mesh moves with the fluid and can easily lose its quality. In the past years, many efforts have been made to develop Lagrangian methods. Some algorithms are developed from the nonconservative form of the Euler equations, for example, those discussed in [21, 4, 5, 6, 18, 35]. The other class of Lagrangian algorithms starts from the conservative form of the Euler equations which usually can guarantee exact conservation. See for example [3, 9, 8, 17, 32, 2] etc.

Most existing Lagrangian schemes for the Euler equations have first or at most second order accuracy. Among them many Lagrangian schemes of non conservative form are only first order accurate because of a first order error due to the nonconservative formulation of the momentum equation. On the other hand, some of the conservative Lagrangian schemes apply the linear interpolation strategy to achieve second order accuracy, meanwhile they usually use a flux limiter to control spurious oscillations which leads to a possible loss of this second order accuracy at some special points such as smooth extrema and sonic points.

Essentially non-oscillatory (ENO) schemes, first introduced by Harten and Osher [13] and Harten et al. [12], can achieve uniformly high-order accuracy with sharp, essentially non-oscillatory shock transitions. In the subsequent years, ENO schemes in the Eulerian formulation have accomplished successful applications in many fields especially with problems containing both shocks and complicated smooth flow structures, see for example [27]. Eulerian ENO schemes on unstructured meshes are developed in [1]. However, the application of the ENO methodology in the Lagrangian formulation does not seem to have been extensively explored.

In this paper, we develop a class of Lagrangian schemes for solving the Euler equations which are based on the high order ENO reconstruction both in the Cartesian and in the cylindrical coordinates. The schemes are conservative for the density, momentum and total energy, can maintain formal high order accuracy both in space and time and can achieve at least uniformly second order accuracy on moving and distorted Lagrangian meshes, are essentially non-oscillatory, and have no parameters to be tuned for individual test cases. They can also be extended to higher than second order accuracy by using curved meshes. Several one and two dimensional numerical examples in the Cartesian and cylindrical coordinates are presented which demonstrate the good performance of the schemes both in purely Lagrangian and in ALE calculations.

An outline of the rest of this paper is as follows. In Section 2, we describe the individual steps of the ENO Lagrangian scheme in one space dimension. In Section 3, we present one

dimensional numerical results. In Section 4, we extend the scheme to two space dimensions both in the Cartesian and in the cylindrical coordinates, while in Section 5 two dimensional numerical examples are given to verify the performance of the ENO Lagrangian method. In Section 6 we will give concluding remarks.

2 High order ENO conservative Lagrangian scheme – one space dimension

2.1 The compressible Euler equations in Lagrangian formulation

The Euler equations for unsteady compressible flow in the reference frame of a moving control volume can be expressed in integral form in the Cartesian coordinates as

$$\frac{d}{dt} \int_{\Omega(t)} \mathbf{U} d\Omega + \int_{\Gamma(t)} \mathbf{F} d\Gamma = 0 \quad (2.1)$$

where $\Omega(t)$ is the moving control volume enclosed by its boundary $\Gamma(t)$. The vector of the conserved variables \mathbf{U} and the flux vector \mathbf{F} are given by

$$\mathbf{U} = \begin{pmatrix} \rho \\ \mathbf{M} \\ E \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} (\mathbf{u} - \dot{\mathbf{x}}) \cdot \mathbf{n} \rho \\ (\mathbf{u} - \dot{\mathbf{x}}) \cdot \mathbf{n} \mathbf{M} + p \cdot \mathbf{n} \\ (\mathbf{u} - \dot{\mathbf{x}}) \cdot \mathbf{n} E + p \mathbf{u} \cdot \mathbf{n} \end{pmatrix} \quad (2.2)$$

where ρ is the density, \mathbf{u} is the velocity, $\mathbf{M} = \rho \mathbf{u}$ is the momentum, E is the total energy and p is the pressure, $\dot{\mathbf{x}}$ is the velocity of the control volume boundary $\Gamma(t)$, \mathbf{n} denotes the unit outward normal to $\Gamma(t)$. The system (2.1) represents the conservation of mass, momentum and energy.

The set of equations is completed by the addition of an equation of state (EOS) with the following general form

$$p = p(\rho, e) \quad (2.3)$$

where $e = \frac{E}{\rho} - \frac{1}{2} |\mathbf{u}|^2$ is the specific internal energy. Especially, if we consider the ideal gas, then the equation of state has a simpler form,

$$p = (\gamma - 1) \rho e$$

where γ is a constant representing the ratio of specific heat capacities of the fluid.

This paper focuses on solving the governing equations (2.1)-(2.2) in a Lagrangian framework, in which it is assumed that $\dot{\mathbf{x}} = \mathbf{u}$, and the vectors \mathbf{U} and \mathbf{F} then take the simpler form

$$\mathbf{U} = \begin{pmatrix} \rho \\ \mathbf{M} \\ E \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} 0 \\ p \cdot \mathbf{n} \\ p\mathbf{u} \cdot \mathbf{n} \end{pmatrix}. \quad (2.4)$$

2.2 The ENO conservative Lagrangian scheme in one space dimension

Here we develop a conservative Lagrangian finite volume scheme both on the staggered mesh and on the non-staggered mesh. We solve the conserved variables such as density, momentum and total energy directly. The non-staggered mesh means that all the conserved variables, in the form of cell averages, are stored at the cell's center and this cell is their common control volume. The staggered mesh implies that there are two sets of control volumes for the considered conserved variables, i.e., the density and total energy are cell-centered and their control volume is the cell they are located in, while the momentum is vertex-centered which uses a different control volume (also called a dual cell). We consider two types of dual cells on the staggered mesh, one is the domain surrounded by its two neighboring nodes, the other is the domain surrounded by the centers of its two neighboring cells. Since the discretization steps are similar, for simplicity, we only show the scheme on the second kind of the staggered mesh.

The spatial domain Ω is discretized into N computational cells $I_{i+1/2} = [x_i, x_{i+1}]$ of sizes $\Delta x_{i+1/2} = x_{i+1} - x_i$ with $i = 1, \dots, N$. For a given cell $I_{i+1/2}$, the location of the cell center is denoted by $x_{i+1/2}$. The fluid velocity u_i is defined at the vertex of the mesh. On the non-staggered mesh, all variables except the velocity are stored at the cell center $x_{i+1/2}$ in the form of cell averages. For example, the values of the cell averages for Cell $I_{i+1/2}$, denoted by $\bar{\rho}_{i+1/2}$, $\bar{M}_{i+1/2}$ and $\bar{E}_{i+1/2}$, are defined as follows

$$\bar{\rho}_{i+1/2} = \frac{1}{\Delta x_{i+1/2}} \int_{I_{i+1/2}} \rho dx, \quad \bar{M}_{i+1/2} = \frac{1}{\Delta x_{i+1/2}} \int_{I_{i+1/2}} M dx, \quad \bar{E}_{i+1/2} = \frac{1}{\Delta x_{i+1/2}} \int_{I_{i+1/2}} E dx.$$

On the staggered mesh, the difference from the non-staggered mesh is that the momentum \overline{M}_i is stored at the vertex x_i and its control volume is $I_i = [x_{i-1/2}, x_{i+1/2}]$ which is of the size $\Delta x_i = x_{i+1/2} - x_{i-1/2}$. Thus

$$\overline{\rho}_{i+1/2} = \frac{1}{\Delta x_{i+1/2}} \int_{I_{i+1/2}} \rho dx, \quad \overline{M}_i = \frac{1}{\Delta x_i} \int_{I_i} M dx, \quad \overline{E}_{i+1/2} = \frac{1}{\Delta x_{i+1/2}} \int_{I_{i+1/2}} E dx.$$

2.2.1 Spatial discretization

We first formulate the semi-discrete finite volume scheme of the governing equations (2.1) and (2.4) on the non-staggered mesh as

$$\frac{d}{dt} \begin{pmatrix} \overline{\rho}_{i+1/2} \Delta x_{i+1/2} \\ \overline{M}_{i+1/2} \Delta x_{i+1/2} \\ \overline{E}_{i+1/2} \Delta x_{i+1/2} \end{pmatrix} = - \begin{pmatrix} \hat{f}_D(\mathbf{U}_{i+1}^-, \mathbf{U}_{i+1}^+) - \hat{f}_D(\mathbf{U}_i^-, \mathbf{U}_i^+) \\ \hat{f}_M(\mathbf{U}_{i+1}^-, \mathbf{U}_{i+1}^+) - \hat{f}_M(\mathbf{U}_i^-, \mathbf{U}_i^+) \\ \hat{f}_E(\mathbf{U}_{i+1}^-, \mathbf{U}_{i+1}^+) - \hat{f}_E(\mathbf{U}_i^-, \mathbf{U}_i^+) \end{pmatrix} \quad (2.5)$$

where \hat{f}_D is the numerical flux of mass across the boundary of its control volume $I_{i+1/2}(t)$, which is consistent with the physical flux (2.4) in the sense that $\hat{f}_D(\mathbf{U}, \mathbf{U}) = 0$, \hat{f}_M is the numerical flux of momentum with $\hat{f}_M(\mathbf{U}, \mathbf{U}) = p$, and \hat{f}_E is the numerical flux of total energy with $\hat{f}_E(\mathbf{U}, \mathbf{U}) = pu$. \mathbf{U}_i^\pm and \mathbf{U}_{i+1}^\pm represent the left and right values of \mathbf{U} at the cell's boundary x_i and x_{i+1} respectively.

The conservative semi-discrete scheme has the following form on the staggered mesh

$$\frac{d}{dt} \begin{pmatrix} \overline{\rho}_{i+1/2} \Delta x_{i+1/2} \\ \overline{M}_i \Delta x_i \\ \overline{E}_{i+1/2} \Delta x_{i+1/2} \end{pmatrix} = - \begin{pmatrix} \hat{f}_D(\mathbf{U}_{i+1}^-, \mathbf{U}_{i+1}^+) - \hat{f}_D(\mathbf{U}_i^-, \mathbf{U}_i^+) \\ \hat{f}_M(\mathbf{U}_{i+1/2}^-, \mathbf{U}_{i+1/2}^+) - \hat{f}_M(\mathbf{U}_{i-1/2}^-, \mathbf{U}_{i-1/2}^+) \\ \hat{f}_E(\mathbf{U}_{i+1}^-, \mathbf{U}_{i+1}^+) - \hat{f}_E(\mathbf{U}_i^-, \mathbf{U}_i^+) \end{pmatrix}. \quad (2.6)$$

Since the method to determine the flux at the boundary of the control volume for the above two sets of meshes are similar, in the following we only describe the procedure on the non-staggered mesh.

The first step for establishing the scheme is to determine the fluxes ($\hat{f}_D, \hat{f}_M, \hat{f}_E$), and the first stage of this step is to identify the values of the primitive variables on each side of the boundary, that is \mathbf{U}_i^\pm , $i = 1, \dots, N$. The information we have is the cell average values of the conserved variables $\overline{\mathbf{U}}_{i+1/2} = (\overline{\rho}_{i+1/2}, \overline{M}_{i+1/2}, \overline{E}_{i+1/2})$, thus we will have to reconstruct these variables. The simplest reconstruction is to assume that all the variables are piecewise

constant, and equal to the given cell averages for each cell. Then we will have $\rho_i^- = \bar{\rho}_{i-1/2}$, $\rho_i^+ = \bar{\rho}_{i+1/2}$ etc., where ρ_i^- and ρ_i^+ are the values of density at the left side and the right side of the cell's boundary x_i . This reconstruction is very simple, but is only first-order accurate.

To obtain uniformly second or higher order accurate schemes, in this paper we will use the ENO idea [12] to reconstruct polynomial functions on each $I_{i+1/2}$ by using the information of the cell $I_{i+1/2}$ and its neighbors, such that they are second or higher order accurate approximations to the functions $\rho(x)$, $M(x)$ and $E(x)$ etc. on $I_{i+1/2}$. The procedure allows us to obtain arbitrary high order accurate approximation by a proper reconstruction. For simplicity, in this paper we will only discuss second and third order accurate schemes by performing the second and third order accurate reconstructions. It is easy to extend the procedure to a higher order reconstruction.

The method of local characteristic decomposition is used in the procedure of the ENO reconstruction. We refer to [31] for the details of the Roe-type characteristic decomposition that we have used in this paper. As to the details of how to conservatively reconstruct a polynomial by the ENO idea in each cell, we refer to [12]. The approximate values of each conserved variable at both sides of the cell's boundary are obtained from its reconstructed polynomial. Finally we obtain the values of the density ρ_i^- , ρ_i^+ , the momentum M_i^- , M_i^+ and the total energy E_i^- , E_i^+ at the left side and the right side of the cell's boundary x_i respectively.

Next, we will compute the fluxes given the primitive states at each side of a control volume's boundary.

In this paper we use the following three typical numerical fluxes:

- 1) The Godunov flux,
- 2) The L-F (Lax-Friedrichs) flux,
- 3) The HLLC (Harten-Lax-van Leer contact wave) flux.

The above three approximate fluxes have been originally developed for solving compressible Euler equations in the Eulerian formulation. Although these fluxes have been very

successful in the Eulerian formulation, their accuracy and robustness for solving the Euler equations in an Lagrangian framework are less explored.

In the following, we will describe the implementation of these three fluxes in our Lagrangian schemes.

1. The Godunov flux

To solve the exact Riemann problem at the cell's boundary x_i , we need to know the left and right states at the boundary. We would like to note that in a Lagrangian scheme the velocity at each side of x_i used here should be the relative fluid velocity, that is, $u_i^- - \dot{x}_i$ and $u_i^+ - \dot{x}_i$, where $u_i^- = M_i^-/\rho_i^-$, $u_i^+ = M_i^+/\rho_i^+$ and \dot{x}_i is the cell boundary's moving velocity which can be numerically determined as the Roe average $\frac{\sqrt{\rho_i^-}u_i^- + \sqrt{\rho_i^+}u_i^+}{\sqrt{\rho_i^-} + \sqrt{\rho_i^+}}$. The pressure values at the two sides of the vertex x_i are of the form $p_i^- = (\gamma - 1)[E_i^- - \frac{1}{2}(M_i^-)^2/\rho_i^-]$, $p_i^+ = (\gamma - 1)[E_i^+ - \frac{1}{2}(M_i^+)^2/\rho_i^+]$ if the ideal gas is considered. With the left state $\{\rho_i^-, u_i^- - \dot{x}_i, p_i^-\}$ and the right state $\{\rho_i^+, u_i^+ - \dot{x}_i, p_i^+\}$ at x_i , we can now obtain the pressure p_i^* and the relative velocity u_i' at x_i by the Riemann solver. The absolute velocity u_i^* at x_i should be $u_i' + \dot{x}_i$. Thus the fluxes \hat{f}_D , \hat{f}_M and \hat{f}_E in Scheme (2.5) have the following form

$$\begin{cases} \hat{f}_D(\mathbf{U}_i^-, \mathbf{U}_i^+) &= 0 \\ \hat{f}_M(\mathbf{U}_i^-, \mathbf{U}_i^+) &= p_i^* \\ \hat{f}_E(\mathbf{U}_i^-, \mathbf{U}_i^+) &= p_i^* u_i^*. \end{cases} \quad (2.7)$$

2. The L-F (Lax-Friedrichs) flux

The Lax-Friedrichs flux with the simple form is given by

$$\begin{cases} \hat{f}_D(\mathbf{U}_i^-, \mathbf{U}_i^+) &= \frac{1}{2}[0 - \alpha_i(\rho_i^+ - \rho_i^-)] \\ \hat{f}_M(\mathbf{U}_i^-, \mathbf{U}_i^+) &= \frac{1}{2}[(p_i^- + p_i^+) - \alpha_i(M_i^+ - M_i^-)] \\ \hat{f}_E(\mathbf{U}_i^-, \mathbf{U}_i^+) &= \frac{1}{2}[(p_i^- u_i^- + p_i^+ u_i^+) - \alpha_i(E_i^+ - E_i^-)] \end{cases} \quad (2.8)$$

where α_i is taken as an upper bound for the eigenvalues of the Jacobian, here in the Lagrangian formulation, $\alpha_i = \max(c_i^-, c_i^+)$, where c_i^\pm are the left and right values of the sound speed at x_i . In order to reduce the numerical viscosity, we perform the local

characteristic decomposition and then choose a different α_i for each characteristic field based on the size of the corresponding eigenvalue, rather than using the same α_i for all components as in (2.8).

3. The HLLC flux

We use the version of the HLLC flux described in [19] (see also [34]) for the ALE formulation which is defined by

$$\hat{\mathbf{F}}_i^{HLLC} = \begin{cases} \hat{\mathbf{F}}_i^-, & \text{if } S_i^- > 0, \\ \hat{\mathbf{F}}(\mathbf{U}_i^*), & \text{if } S_i^- \leq 0 < S_M, \\ \hat{\mathbf{F}}(\mathbf{U}_i^{**}), & \text{if } S_M \leq 0 < S_i^+, \\ \hat{\mathbf{F}}_i^+, & \text{if } S_i^+ < 0, \end{cases} \quad (2.9)$$

where

$$\hat{\mathbf{F}}_i^- = \begin{pmatrix} 0 \\ p_i^- \\ p_i^- u_i^- \end{pmatrix}, \quad \hat{\mathbf{F}}_i^+ = \begin{pmatrix} 0 \\ p_i^+ \\ p_i^+ u_i^+ \end{pmatrix},$$

and

$$\mathbf{U}_i^* = \begin{pmatrix} \rho_i^* \\ M_i^* \\ E_i^* \end{pmatrix} = \frac{1}{S_i^- - S_M} \begin{pmatrix} (S_i^- - v_i^-)\rho_i^- \\ (S_i^- - v_i^-)M_i^- + (p^* - p_i^-) \\ (S_i^- - v_i^-)E_i^- - p_i^- v_i^- + p^* S_M \end{pmatrix}, \quad (2.10)$$

$$\mathbf{U}_i^{**} = \begin{pmatrix} \rho_i^{**} \\ M_i^{**} \\ E_i^{**} \end{pmatrix} = \frac{1}{S_i^+ - S_M} \begin{pmatrix} (S_i^+ - v_i^+)\rho_i^+ \\ (S_i^+ - v_i^+)M_i^+ + (p^* - p_i^+) \\ (S_i^+ - v_i^+)E_i^+ - p_i^+ v_i^+ + p^* S_M \end{pmatrix}, \quad (2.11)$$

$$\hat{\mathbf{F}}(\mathbf{U}_i^*) = \begin{pmatrix} S_M \rho_i^* \\ S_M M_i^* + p^* \\ S_M E_i^* + (S_M + \dot{x}_i) p^* \end{pmatrix}, \quad \hat{\mathbf{F}}(\mathbf{U}_i^{**}) = \begin{pmatrix} S_M \rho_i^{**} \\ S_M M_i^{**} + p^* \\ S_M E_i^{**} + (S_M + \dot{x}_i) p^* \end{pmatrix}, \quad (2.12)$$

$$p^* = \rho_i^- (v_i^- - S_i^-)(v_i^- - S_M) + p_i^- \quad (2.13)$$

and $v_i^- = u_i^- - \dot{x}_i$, $v_i^+ = u_i^+ - \dot{x}_i$. S_M is defined as

$$S_M = \frac{\rho_i^+ v_i^+ (S_i^+ - v_i^+) - \rho_i^- v_i^- (S_i^- - v_i^-) + p_i^- - p_i^+}{\rho_i^+ (S_i^+ - v_i^+) - \rho_i^- (S_i^- - v_i^-)}. \quad (2.14)$$

The signal velocities S_i^- and S_i^+ are defined as

$$S_i^- = \min[v_i^- - c_i^-, (u_i - \dot{x}_i) - c_i], \quad S_i^+ = \max[v_i^+ + c_i^+, (u_i - \dot{x}_i) + c_i] \quad (2.15)$$

where u_i and c_i are the Roe's average variables for the velocity and the speed of sound. Since we are considering the Euler equations in the Lagrangian formulation, here we have $\dot{x}_i = u_i$.

Each of the above approximate fluxes has its own special features. The Godunov flux solves exactly the Riemann problem at the cell boundary and thus it has the least numerical viscosity among all the first order upwind fluxes. In particular, it has no numerical viscosity for the first equation hence it can maintain the mass of each cell unchanged during the time marching. But it also has the disadvantage of high computational cost. The L-F flux has relatively more numerical viscosity but it has a very simple form and is more robust in applications. The HLLC flux's viscosity and cost are between the Godunov flux and the L-F flux. It has good performance in many applications. We remark that both the L-F flux and the HLLC flux apply the numerical viscosity in all the equations including the mass equation, causing a possible change in the cell mass during the time evolution. We do, however, find in the following tests that the L-F flux and HLLC flux perform well on capturing the contact discontinuities in our Lagrangian schemes. In the actual simulation, especially in the ALE calculation, we can choose the most suitable flux depending on the requirement of the individual problem.

2.2.2 The determination of the vertex velocity

In the Lagrangian formulation, the grid moves with the fluid velocity which is defined at the vertex, thus we would need to know the velocity at the vertex to move the grid. Since the velocity is a derived quantity, we would need to obtain it from the conserved variables. In the following we describe how to determine the vertex's velocity in our schemes.

For the Godunov flux, since we solve exactly the Riemann problem at each vertex as a cell's boundary, we naturally obtain the velocity by the Riemann solver there.

For the L-F flux and the HLLC flux, the velocity at the cell's vertex is defined as the

Roe's average of velocities from both sides,

$$u_i = \frac{\sqrt{\rho_i^-} u_i^- + \sqrt{\rho_i^+} u_i^+}{\sqrt{\rho_i^-} + \sqrt{\rho_i^+}}. \quad (2.16)$$

2.2.3 Time discretization

The time marching for the semi-discrete scheme (2.5) or (2.6) is implemented by a class of TVD Runge-Kutta type methods [29]. Since the mesh changes with the time advancing in the Lagrangian simulation, the velocity, the position of each vertex and the size of each cell need to be updated at each Runge-Kutta stage. Thus the form of the Runge-Kutta method (we take the third-order case here as an example) in our Lagrangian schemes is as follows.

Step 1,

$$\begin{aligned} x_i^{(1)} &= x_i^{(0)} + u_i^{(0)} \Delta t^n, & \Delta x_{i+1/2}^{(1)} &= x_{i+1}^{(1)} - x_i^{(1)}, \\ \bar{\mathbf{U}}_{i+1/2}^{(1)} \Delta x_{i+1/2}^{(1)} &= \bar{\mathbf{U}}_{i+1/2}^{(0)} \Delta x_{i+1/2}^{(0)} + \Delta t^n \mathbf{L}(\bar{\mathbf{U}}_{i+1/2}^{(0)}); \end{aligned}$$

Step 2,

$$\begin{aligned} x_i^{(2)} &= \frac{3}{4} x_i^{(0)} + \frac{1}{4} [x_i^{(1)} + u_i^{(1)} \Delta t^n], & \Delta x_{i+1/2}^{(2)} &= x_{i+1}^{(2)} - x_i^{(2)}, \\ \bar{\mathbf{U}}_{i+1/2}^{(2)} \Delta x_{i+1/2}^{(2)} &= \frac{3}{4} \bar{\mathbf{U}}_{i+1/2}^{(0)} \Delta x_{i+1/2}^{(0)} + \frac{1}{4} [\bar{\mathbf{U}}_{i+1/2}^{(1)} \Delta x_{i+1/2}^{(1)} + \Delta t^n \mathbf{L}(\bar{\mathbf{U}}_{i+1/2}^{(1)})]; \end{aligned}$$

Step 3,

$$\begin{aligned} x_i^{(3)} &= \frac{1}{3} x_i^{(0)} + \frac{2}{3} [x_i^{(2)} + u_i^{(2)} \Delta t^n], & \Delta x_{i+1/2}^{(3)} &= x_{i+1}^{(3)} - x_i^{(3)}, \\ \bar{\mathbf{U}}_{i+1/2}^{(3)} \Delta x_{i+1/2}^{(3)} &= \frac{1}{3} \bar{\mathbf{U}}_{i+1/2}^{(0)} \Delta x_{i+1/2}^{(0)} + \frac{2}{3} [\bar{\mathbf{U}}_{i+1/2}^{(2)} \Delta x_{i+1/2}^{(2)} + \Delta t^n \mathbf{L}(\bar{\mathbf{U}}_{i+1/2}^{(2)})]; \end{aligned}$$

where

$$\begin{aligned} x_i^{(0)} &= x_i^n, & x_i^{n+1} &= x_i^{(3)}; & u_i^{(0)} &= u_i^n, & u_i^{n+1} &= u_i^{(3)}; \\ \Delta x_{i+1/2}^{(0)} &= \Delta x_{i+1/2}^n, & \Delta x_{i+1/2}^{n+1} &= \Delta x_{i+1/2}^{(3)}; \\ \bar{\mathbf{U}}_{i+1/2}^{(0)} &= \bar{\mathbf{U}}_{i+1/2}^n, & \bar{\mathbf{U}}_{i+1/2}^{n+1} &= \bar{\mathbf{U}}_{i+1/2}^{(3)}. \end{aligned}$$

\mathbf{L} is the numerical spatial operator representing the right hand of the scheme (2.5). Here the variables with the superscripts n and $n + 1$ represent the values of the corresponding variables at the n -th and $(n + 1)$ -th time steps respectively.

Consistently with the order of the spatial discretization in the scheme (2.5), we apply the Runge-Kutta method of the same order for its time marching.

At the end of this section, we list the explicit form of our first order Lagrangian scheme as an example,

$$\begin{pmatrix} \bar{\rho}_{i+1/2}^{n+1} \Delta x_{i+1/2}^{n+1} - \bar{\rho}_{i+1/2}^n \Delta x_{i+1/2}^n \\ \bar{M}_{i+1/2}^{n+1} \Delta x_{i+1/2}^{n+1} - \bar{M}_{i+1/2}^n \Delta x_{i+1/2}^n \\ \bar{E}_{i+1/2}^{n+1} \Delta x_{i+1/2}^{n+1} - \bar{E}_{i+1/2}^n \Delta x_{i+1/2}^n \end{pmatrix} = -\Delta t^n \begin{pmatrix} \hat{f}_D(\mathbf{U}_{i+1}^{n-}, \mathbf{U}_{i+1}^{n+}) - \hat{f}_D(\mathbf{U}_i^{n-}, \mathbf{U}_i^{n+}) \\ \hat{f}_M(\mathbf{U}_{i+1}^{n-}, \mathbf{U}_{i+1}^{n+}) - \hat{f}_M(\mathbf{U}_i^{n-}, \mathbf{U}_i^{n+}) \\ \hat{f}_E(\mathbf{U}_{i+1}^{n-}, \mathbf{U}_{i+1}^{n+}) - \hat{f}_E(\mathbf{U}_i^{n-}, \mathbf{U}_i^{n+}) \end{pmatrix} \quad (2.17)$$

where the time step Δt^n is determined as

$$\Delta t^n = \lambda \min_{i=1, \dots, N} (\Delta x_i^n / c_i^n)$$

with the Courant number λ chosen as $\lambda = 0.6$ in our computation.

3 Numerical results in one space dimension

In this section, we perform some numerical experiments in one space dimension. Purely Lagrangian computation and the ideal gas with $\gamma = 1.4$ are used to do the following tests unless otherwise stated. We have used the Godunov flux, the L-F flux and the HLLC flux respectively and the results are mostly similar, so only those with the HLLC flux are listed to save space.

3.1 Accuracy test

We first test the accuracy of our schemes on a problem with smooth solutions. The initial condition we choose for the accuracy test is:

$$\rho(x, 0) = 2 + \sin(2\pi x), \quad u(x, 0) = 1 + 0.1 \sin(2\pi x), \quad p(x, 0) = 1, \quad x \in [0, 1]$$

Table 3.1: Errors of the first order scheme on 1D non-staggered meshes using N_x initially uniform mesh cells.

N_x	Norm	Density	order	Momentum	order	Energy	order
100	L_1	0.11E-1		0.15E-1		0.29E-1	
	L_∞	0.34E-1		0.46E-1		0.78E-1	
200	L_1	0.55E-2	0.93	0.78E-2	0.95	0.15E-1	0.93
	L_∞	0.19E-1	0.89	0.25E-1	0.86	0.42E-1	0.88
400	L_1	0.28E-2	0.97	0.40E-2	0.97	0.77E-2	0.96
	L_∞	0.97E-2	0.93	0.13E-1	0.92	0.22E-1	0.93
800	L_1	0.14E-2	0.98	0.20E-2	0.98	0.39E-2	0.98
	L_∞	0.50E-2	0.96	0.69E-2	0.96	0.11E-1	0.96

with a periodic boundary condition. Since we do not know the exact solution, we take the numerical results by using the fifth order Eulerian WENO scheme [15] with 8000 grids as the reference “exact” solution. In Tables 3.1-3.2, we summarize the errors and numerical rate of convergence of our first and second order Lagrangian schemes at $t = 1$ on the 1D non-staggered meshes. We can clearly see from Tables 3.1 and 3.2 that the first and second order schemes achieve the designed order of accuracy, at least in the L_1 norm. However, we do observe a degeneracy of the L_∞ error for the second order scheme in Table 3.2, which is related to an accuracy degeneracy phenomenon of ENO schemes, originally discussed in [24] for Eulerian formulated schemes. This degeneracy is more serious for the third order scheme, hence we have used the modified third order ENO scheme via the introduction of a biasing factor introduced in [26]. The effect of using this factor in the stencil determination procedure is to bias towards a linearly stable stencil in smooth regions. Table 3.3 shows the error results of the modified ENO scheme by using a factor of 2, according to the suggestion in [26]. From this table, we can see the modified ENO scheme produces the correct third order accuracy. The following third order non-oscillatory tests are all performed by the modified third order ENO scheme, verifying its non-oscillatory property for problems with discontinuities.

Table 3.2: Errors of the second order ENO scheme on 1D non-staggered meshes using N_x initially uniform mesh cells.

N_x	Norm	Density	order	Momentum	order	Energy	order
100	L_1	0.16E-2		0.22E-2		0.43E-2	
	L_∞	0.52E-2		0.10E-1		0.17E-1	
200	L_1	0.48E-3	1.76	0.61E-3	1.86	0.12E-2	1.81
	L_∞	0.25E-2	1.09	0.42E-2	1.25	0.72E-2	1.21
400	L_1	0.13E-3	1.93	0.16E-3	1.94	0.32E-3	1.94
	L_∞	0.93E-3	1.39	0.17E-2	1.32	0.28E-2	1.35
800	L_1	0.34E-4	1.90	0.42E-4	1.91	0.84E-4	1.91
	L_∞	0.35E-3	1.40	0.65E-3	1.38	0.11E-2	1.38

Table 3.3: Errors of the modified third order ENO scheme on 1D non-staggered meshes using N_x initially uniform mesh cells.

N_x	Norm	Density	order	Momentum	order	Energy	order
100	L_1	0.52E-4		0.69E-4		0.13E-3	
	L_∞	0.27E-3		0.47E-3		0.69E-3	
200	L_1	0.66E-5	2.99	0.88E-5	2.98	0.17E-4	2.99
	L_∞	0.34E-4	2.96	0.60E-4	2.96	0.88E-4	2.96
400	L_1	0.82E-6	3.00	0.11E-5	3.00	0.21E-5	3.00
	L_∞	0.43E-5	2.99	0.75E-5	2.99	0.11E-4	2.99
800	L_1	0.10E-6	3.00	0.14E-6	3.00	0.26E-6	3.00
	L_∞	0.54E-6	3.00	0.94E-6	3.00	0.14E-5	3.00

3.2 Non-oscillatory tests

Example 3.1 (Lax problem). The first non-oscillatory test is the Riemann problem proposed by Lax. Its initial condition is as follows

$$(\rho_L, u_L, p_L) = (0.445, 0.698, 3.528), \quad (\rho_R, u_R, p_R) = (0.5, 0, 0.571).$$

Figure 3.1 shows the results with 200 initially uniform cells on the non-staggered mesh and on the staggered mesh at $t = 1$.

Comparing with the exact solution, we observe satisfactory non-oscillatory results for the non-staggered mesh in the left pictures of Figure 3.1, while there are obvious oscillations for the staggered mesh in the right pictures of Figure 3.1. Notice that the spurious oscillations appear even for the first order scheme, indicating that the problem is not due to the high order reconstruction. In order to explore further the possible reason for these oscillations, we study a simpler test case with the initial condition

$$\rho(x, 0) = 2 + \sin(2\pi x), \quad u(x, 0) = 1, \quad p(x, 0) = 1, \quad x \in [0, 1].$$

and a periodic boundary condition. The exact solution is easily given as

$$\rho(x, t) = 2 + \sin(2\pi(x - t)), \quad u(x, t) = 1, \quad p(x, t) = 1.$$

Figure 3.2 shows the numerical results of the first order schemes on the non-staggered mesh and on the staggered mesh at $t = 1$ respectively. We can see that the final velocity and pressure on the non-staggered mesh can keep constant while those on the staggered mesh can not. The reason is that, on the staggered mesh, the density and momentum are defined in different control volumes, causing possible large errors to the velocity which is a derived quantity. This error is particularly prominent for discontinuous solutions, resulting in spurious oscillations and even instability.

We therefore conclude that the non-staggered mesh is more suitable for conservative Lagrangian schemes. In the following we will only discuss schemes based on the non-staggered meshes.

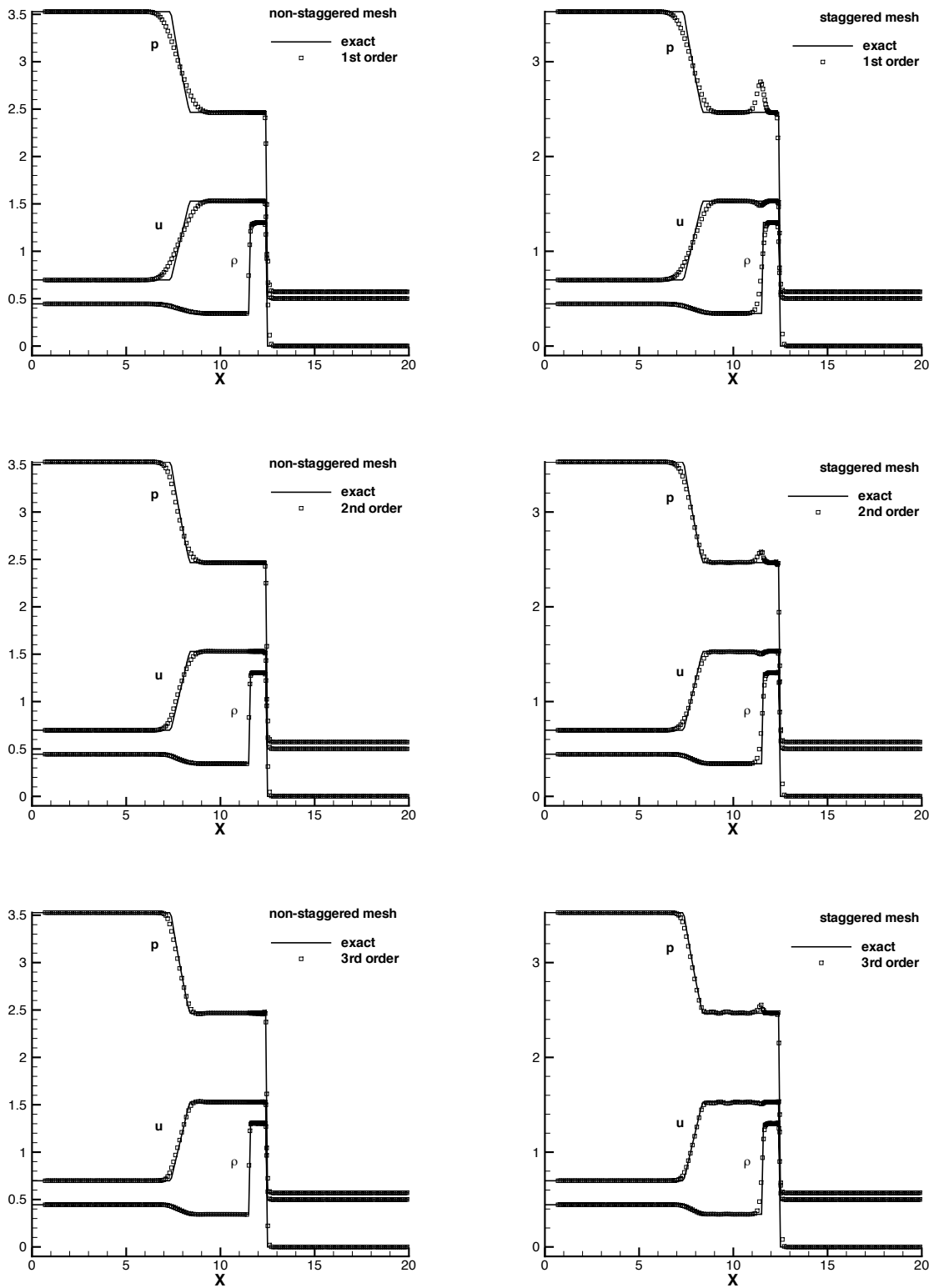


Figure 3.1: The results of the Lax problem on a 200 initially uniform cells. Left: non-staggered mesh; Right: staggered mesh. Top: first order; Middle: second order; Bottom: third order.

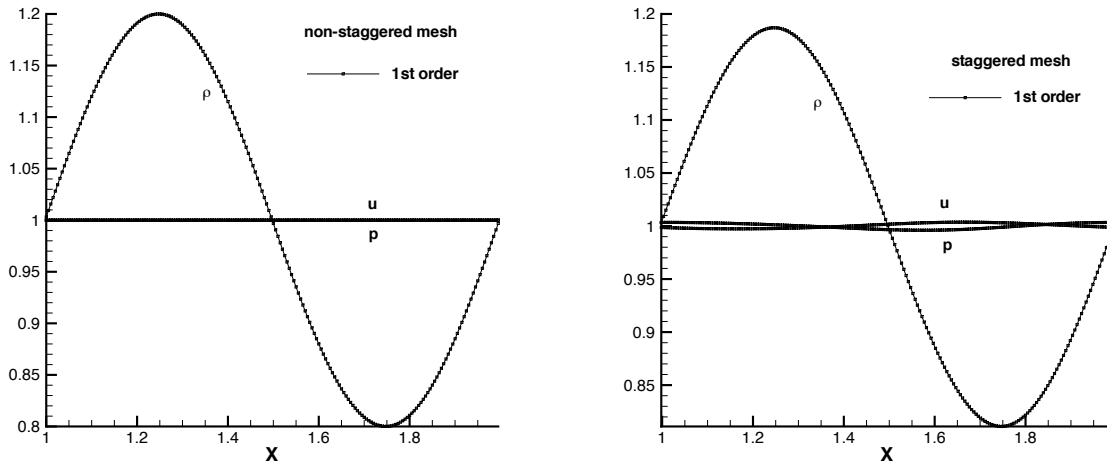


Figure 3.2: Density, velocity and pressure of the first order schemes. Left: on the non-staggered mesh; Right: on the staggered mesh.

Example 3.2 (The Noh problem [22]). The computational domain is $[0, 1]$. The initial state of the fluid is uniform with $(\rho, u, e) = (1, -1, 0)$. The shock is generated in a perfect ideal gas ($\gamma = 5/3$) by bringing the cold gas to rest at a rigid wall ($x = 0$). The analytic post shock density is 4 and the shock speed is $1/3$. The left pictures in Figure 3.3 show the computational densities with 200 initially uniform cells against the exact density at $t = 0.6$. We observe the typical errors near the left boundary for all orders of accuracy. The shock resolution is clearly better for the higher order schemes.

Example 3.3 (Two interacting blast waves). The initial data are

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

The reflective boundary condition is applied at both $x = 0$ and $x = 1$. In the right pictures of Figure 3.3, the computed densities with 400 initially uniform cells at the final time $t = 0.038$ are plotted against the reference “exact” solution, which is computed using a fifth order Eulerian WENO scheme [15] with 16000 grid points. We can see the very satisfactory resolution in the results of high order scheme with relatively coarse grids which demonstrates the advantage of the Lagrangian scheme. Meanwhile, we observe some overshoots in these

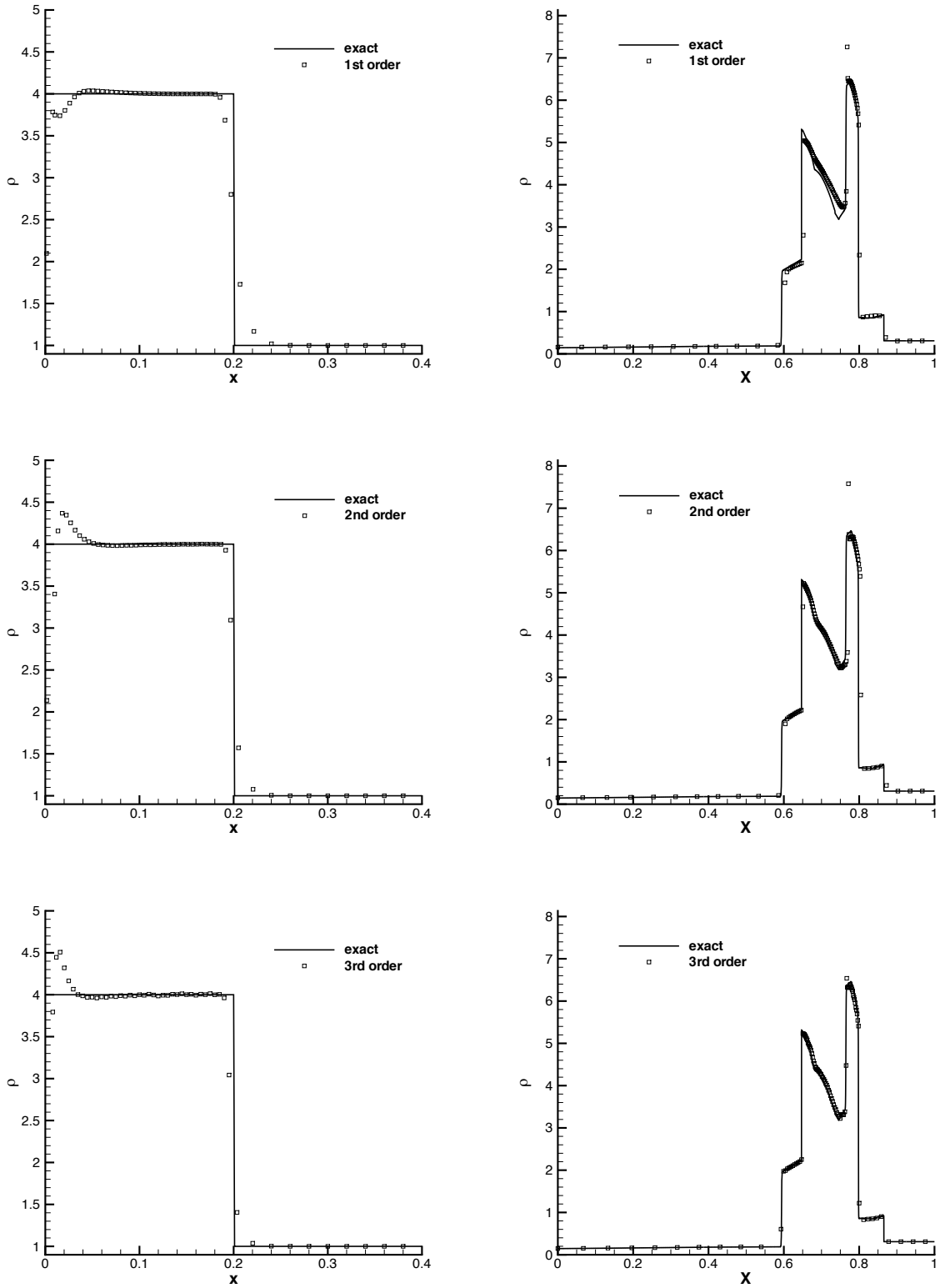


Figure 3.3: The density results. Left: the Noh problem at $t = 0.6$; Right: the blast wave problem at $t = 0.038$. Top: first order; Middle: second order; Bottom: third order.

figures and in some examples later. Apparently such overshoots are caused by the Lagrangian framework rather than by the high order ENO reconstruction, since they are more serious for the lower order schemes.

Example 3.4 (Leblanc shock tube problem). In this extreme shock tube problem, the computational domain is $[0, 9]$ filled with an ideal perfect gas with $\gamma = 5/3$. The initial condition consists of large ratio jumps for the energy and density and is given by the following data

$$\begin{aligned}(\rho, u, e) &= (1, 0, 0.1), \quad 0 \leq x < 3 \\(\rho, u, e) &= (0.001, 0, 10^{-7}), \quad 3 < x \leq 9.\end{aligned}$$

It is very difficult for a scheme to obtain accurate positions of the contact and shock discontinuities in such a severe test case [33]. The internal energy results of our schemes are shown in Figure 3.4 (the left pictures) with 1000 initially uniform cells at $t = 6$. By comparing with the exact solution, we can see that the shape and the position of the contact discontinuity and the shock can be maintained better when the high order ENO schemes are used. For this test case, the first order scheme with the HLLC flux does not run stably, hence the result with the L-F flux for the first order scheme is plotted in Figure 3.4.

Example 3.5 (Shock entropy wave interactions [30]). On a computational domain $[-10, 5]$, the initial condition is:

$$\begin{aligned}(\rho, u, e) &= (3.85714, 2.629369, 10.33333), \quad x < -4 \\(\rho, u, e) &= (1 + \epsilon \sin(kx), 0, 1), \quad x \geq -4\end{aligned}$$

where ϵ and k are the amplitude and wave number of the entropy wave. In our test, we take $\epsilon = 0.2$ and $k = 5$. The final time is $t = 1.8$. This problem is very suitable for testing the advantage of a high order scheme when the solution contains both shocks and complex smooth region structures.

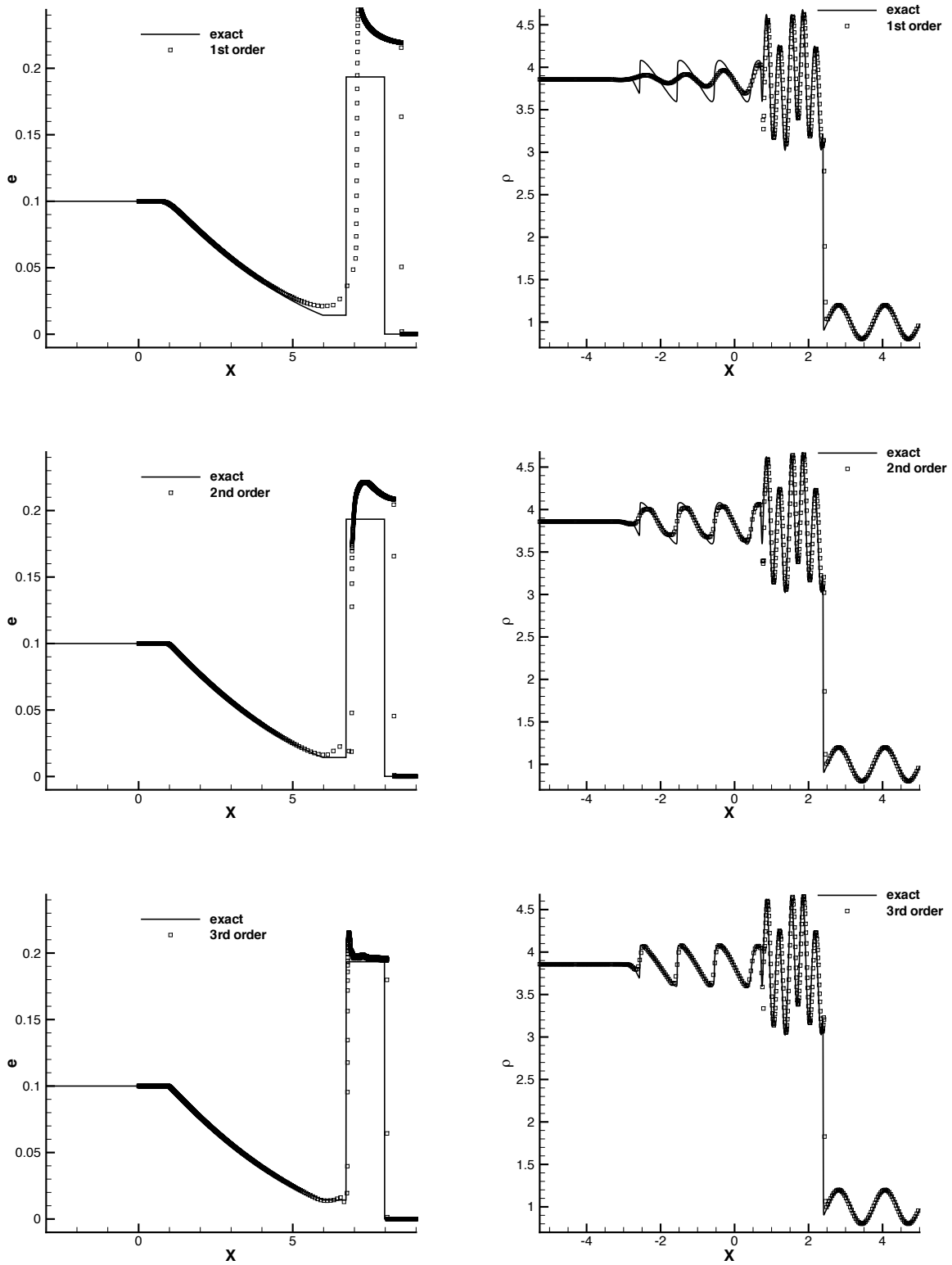


Figure 3.4: Left: the internal energy of the Leblanc problem; Right: the density of the shock entropy wave interactions problem. Top: first order; Middle: second order; Bottom: third order.

In Figure 3.4 (the right pictures), the computed density with 400 cells is plotted against the reference “exact” solution, which is obtained using the fifth order Eulerian WENO scheme [15] with 2000 grid points. We observe that the fine structure in the density profile makes the higher order schemes perform much better than the lower order methods.

4 High order ENO conservative Lagrangian scheme - two space dimensions

4.1 The scheme in the Cartesian coordinates

The 2D spatial domain Ω is discretized into $M \times N$ computational cells. $I_{i+1/2,j+1/2}$ is a quadrilateral cell constructed by the four vertices $\{(x_{i,j}, y_{i,j}), (x_{i+1,j}, y_{i+1,j}), (x_{i+1,j+1}, y_{i+1,j+1}), (x_{i,j+1}, y_{i,j+1})\}$. $S_{i+1/2,j+1/2}$ is denoted to be the area of the cell $I_{i+1/2,j+1/2}$ with $i = 1, \dots, M$, $j = 1, \dots, N$. For a given cell $I_{i+1/2,j+1/2}$, the location of the cell center is denoted by $(x_{i+1/2,j+1/2}, y_{i+1/2,j+1/2})$. The fluid velocity $(u_{i,j}, v_{i,j})$ is defined at the vertex of the mesh. On the non-staggered mesh, all the variables except velocity are stored at the cell center of $I_{i+1/2,j+1/2}$ in the form of cell averages. For example, the values of the cell averages for the cell $I_{i+1/2,j+1/2}$ denoted by $\bar{\rho}_{i+1/2,j+1/2}$, $\bar{M}_{i+1/2,j+1/2}^x$, $\bar{M}_{i+1/2,j+1/2}^y$ and $\bar{E}_{i+1/2,j+1/2}$ are defined as follows

$$\bar{\rho}_{i+1/2,j+1/2} = \frac{1}{S_{i+1/2,j+1/2}} \iint_{I_{i+1/2,j+1/2}} \rho dx dy, \quad \bar{M}_{i+1/2,j+1/2}^x = \frac{1}{S_{i+1/2,j+1/2}} \iint_{I_{i+1/2,j+1/2}} M_x dx dy,$$

$$\bar{M}_{i+1/2,j+1/2}^y = \frac{1}{S_{i+1/2,j+1/2}} \iint_{I_{i+1/2,j+1/2}} M_y dx dy, \quad \bar{E}_{i+1/2,j+1/2} = \frac{1}{S_{i+1/2,j+1/2}} \iint_{I_{i+1/2,j+1/2}} E dx dy$$

where ρ , M_x , M_y and E are the density, x -momentum, y -momentum and total energy, respectively.

4.1.1 Spatial discretization

The conservative semi-discrete scheme for the equations (2.1) and (2.4) has the following form on the 2D non-staggered mesh

$$\begin{aligned} \frac{d}{dt} \begin{pmatrix} \bar{p}_{i+1/2,j+1/2} S_{i+1/2,j+1/2} \\ \bar{M}_x^x{}_{i+1/2,j+1/2} S_{i+1/2,j+1/2} \\ \bar{M}_y^y{}_{i+1/2,j+1/2} S_{i+1/2,j+1/2} \\ \bar{E}_{i+1/2,j+1/2} S_{i+1/2,j+1/2} \end{pmatrix} &= - \int_{\partial I_{i+1/2,j+1/2}} \hat{\mathbf{F}} dl \\ &= - \int_{\partial I_{i+1/2,j+1/2}} \begin{pmatrix} \hat{f}_D(\mathbf{U}_n^-, \mathbf{U}_n^+) \\ \hat{f}_{M_x}(\mathbf{U}_n^-, \mathbf{U}_n^+) \\ \hat{f}_{M_y}(\mathbf{U}_n^-, \mathbf{U}_n^+) \\ \hat{f}_E(\mathbf{U}_n^-, \mathbf{U}_n^+) \end{pmatrix} dl. \end{aligned} \quad (4.1)$$

Here $\mathbf{n} = (n_x, n_y)$ is the outward unit normal of the quadrilateral boundary $\partial I_{i+1/2,j+1/2}$. $\mathbf{U}^\pm = (\rho^\pm, M_x^\pm, M_y^\pm, E^\pm)$ are the values of mass, x-momentum, y-momentum and total energy at two sides of the boundary. $\mathbf{U}_n^\pm = (\rho^\pm, M_n^\pm, E^\pm)$, where M_n^\pm are the left and right component values of the momentum which is vertical to the cell boundary, i.e. $M_n^\pm = (M_x^\pm, M_y^\pm) \cdot \mathbf{n}$. \hat{f}_D , \hat{f}_{M_x} , \hat{f}_{M_y} and \hat{f}_E are the numerical fluxes of mass, x -momentum, y -momentum and total energy across the cell boundary respectively. Here in the Lagrangian formulation, we have

$$\begin{cases} \hat{f}_D(\mathbf{U}_n, \mathbf{U}_n) &= 0, \\ \hat{f}_{M_x}(\mathbf{U}_n, \mathbf{U}_n) &= pn_x \\ \hat{f}_{M_y}(\mathbf{U}_n, \mathbf{U}_n) &= pn_y \\ \hat{f}_E(\mathbf{U}_n, \mathbf{U}_n) &= pu_n \end{cases} \quad (4.2)$$

where $u_n = \mathbf{u} \cdot \mathbf{n}$ is the normal velocity at the cell boundary.

Suppose the cell boundary $\partial I_{i+1/2,j+1/2}$ consists of M edges. The line integral in Eq. (4.1) is discretized by a q -point Gaussian integration formula,

$$\int_{\partial I_{i+1/2,j+1/2}} \hat{\mathbf{F}} dl \approx \sum_{m=1}^M \sum_{k=1}^q \omega_k \hat{\mathbf{F}}(\mathbf{U}_n(G_k, t)) \Delta l^m \quad (4.3)$$

where Δl^m is the length of the boundary edge m and G_k are the Gaussian quadrature points at the edge. Here $\hat{\mathbf{F}}(\mathbf{U}_n(G_k, t))$ is a numerical flux. For example the L-F flux is given by

$$\hat{\mathbf{F}}(\mathbf{U}_n(G_k, t)) = \frac{1}{2} [(\hat{\mathbf{F}}(\mathbf{U}_n^-(G_k, t)) + \hat{\mathbf{F}}(\mathbf{U}_n^+(G_k, t))) - \alpha(\mathbf{U}^+(G_k, t) - \mathbf{U}^-(G_k, t))] \quad (4.4)$$

where α has the same meaning as that in the one dimensional case.

We use the high order ENO reconstruction with Roe-type characteristic decomposition [31] to obtain \mathbf{U}^\pm and \mathbf{U}_n^\pm at the boundary and also use sufficiently high order quadrature to construct schemes up to the expected high-order spatial accuracy, for example the four-point Gauss-Lobatto integral formula is used, which has $G_1 = P_1$, $G_2 = \frac{1}{2}(P_1 + P_2) - \frac{\sqrt{5}}{10}(P_2 - P_1)$, $G_3 = \frac{1}{2}(P_1 + P_2) + \frac{\sqrt{5}}{10}(P_2 - P_1)$, $G_4 = P_2$ and $\omega_1 = \omega_4 = \frac{1}{12}$, $\omega_2 = \omega_3 = \frac{5}{12}$ for the line with endpoints P_1 and P_2 . We have discussed in detail the high order ENO reconstruction needed in our framework in [7], in the context of remapping. Therefore we do not repeat the details here and refer the readers to [7]. We do mention here, however, that we have found in numerical tests that the following WENO procedure is more robust than the ENO procedure for the third order case, hence this WENO procedure is used in the third order numerical tests. In this procedure, the coefficients of the reconstruction polynomial are chosen as the weighted averages of those determined by the final three possible stencils introduced in [7]. To be more specific, we use density as an example. To determine the coefficients $\{a_{mn}, m+n \leq 2\}$ of the quadratic polynomial reconstruction function inside the cell $I_{i+1/2,j+1/2}$,

$$\rho_{i+1/2,j+1/2}(x, y) = \sum_{m+n \leq 2} a_{mn} (x - x_{i+1/2,j+1/2})^m (y - y_{i+1/2,j+1/2})^n,$$

suppose the coefficients of the reconstruction polynomials of the three candidate stencils are $a_{mn}^i, i = 1, 2, 3$, then we choose $a_{mn} = \sum_{i=1}^3 w^i a_{mn}^i$ where w^i is the weight chosen as $w^i = (1/\sum_{m+n=2} |a_{mn}^i|^2)/c$ with $c = \sum_{i=1}^3 (1/\sum_{m+n=2} |a_{mn}^i|^2)$. This crude WENO reconstruction, which does not increase the accuracy of each candidate stencil but is very easy to compute, performs quite nicely in our numerical experiments.

The three numerical fluxes introduced in the one dimensional case are also applied here. The form of these fluxes in two dimensions is similar to that in one dimension except that the left and right values at the cell's boundary are chosen as \mathbf{U}_n^\pm in two dimensions rather than \mathbf{U}^\pm in one dimension.

4.1.2 The determination of the vertex velocity

Considering a vertex (i, j) shared by four edges which are given a serial number $k = 1, 2, 3, 4$, we define the direction of each edge to be the direction of the incremental index i or j , for example the direction of the edge with two endpoints $(i-1, j)$ and (i, j) is from $(i-1, j)$ to (i, j) . Along each edge k we can obtain the left value of velocity $(u^{k-}, v^{k-}) = (M_x^{k-}/\rho^{k-}, M_y^{k-}/\rho^{k-})$ and the right velocity $(u^{k+}, v^{k+}) = (M_x^{k+}/\rho^{k+}, M_y^{k+}/\rho^{k+})$ at this vertex in the procedure of the flux computation since the vertex (i, j) is one of the Gaussian quadrature points for our choice. We then split the left and right velocities into vertical and tangential components along the edge k . Denote (n_x^k, n_y^k) to be the clockwise unit normal of the edge k and denote w_t^{k-} and w_t^{k+} to be their tangential components and w_n^{k-} and w_n^{k+} to be their vertical components. Then the tangential velocity of the vertex (i, j) along the edge k is defined as

$$w_t^k = \frac{1}{2}(w_t^{k-} + w_t^{k+}), \quad k = 1, 2, 3, 4. \quad (4.5)$$

As to the vertical velocity, for the Godunov flux, we get it by the Riemann solver here and for the L-F flux and the HLLC flux, we get it by the Roe average of the vertical velocities from its two sides as in the one dimensional case, that is

$$w_n^k = \frac{\sqrt{\rho^-}w_n^{k-} + \sqrt{\rho^+}w_n^{k+}}{\sqrt{\rho^-} + \sqrt{\rho^+}}, \quad k = 1, 2, 3, 4, \quad (4.6)$$

where ρ^\pm are the densities from the left and right cells of the edge k respectively.

Thus by the formulas (4.5) and (4.6), we can get four x -velocities and y -velocities at the vertex (i, j) which have the following form,

$$w_x^k = w_n^k n_x^k - w_t^k n_y^k, \quad w_y^k = w_n^k n_y^k + w_t^k n_x^k, \quad k = 1, 2, 3, 4. \quad (4.7)$$

Finally the velocity at the vertex (i, j) is obtained as follows,

$$u_{i,j} = \frac{1}{4}(w_x^1 + w_x^2 + w_x^3 + w_x^4), \quad v_{i,j} = \frac{1}{4}(w_y^1 + w_y^2 + w_y^3 + w_y^4). \quad (4.8)$$

4.1.3 Time discretization

The time discretization is also similar to that in one dimension. We only list the first order Lagrangian scheme as a representative here to save space

$$\begin{pmatrix} \bar{\rho}_{i+1/2,j+1/2}^{n+1} S_{i+1/2,j+1/2}^{n+1} - \bar{\rho}_{i+1/2,j+1/2}^n S_{i+1/2,j+1/2}^n \\ \bar{M}_{i+1/2,j+1/2}^{x,n+1} S_{i+1/2,j+1/2}^{n+1} - \bar{M}_{i+1/2,j+1/2}^{x,n} S_{i+1/2,j+1/2}^n \\ \bar{M}_{i+1/2,j+1/2}^{y,n+1} S_{i+1/2,j+1/2}^{n+1} - \bar{M}_{i+1/2,j+1/2}^{y,n} S_{i+1/2,j+1/2}^n \\ \bar{E}_{i+1/2,j+1/2}^{n+1} S_{i+1/2,j+1/2}^{n+1} - \bar{E}_{i+1/2,j+1/2}^n S_{i+1/2,j+1/2}^n \end{pmatrix} = -\Delta t^n \sum_{m=1}^M \sum_{k=1}^q \omega_k \hat{\mathbf{F}}(\mathbf{U}_n(G_k, t)) \Delta l^m \quad (4.9)$$

where $S_{i+1/2,j+1/2}^n$ and $S_{i+1/2,j+1/2}^{n+1}$ are the areas of Cell $I_{i+1/2,j+1/2}$ at the n -th and $(n+1)$ -th time steps respectively. $S_{i+1/2,j+1/2}^{n+1}$ is determined by the following simple formulas,

$$\begin{aligned} x_{i,j}^{n+1} &= u_{i,j}^n \Delta t^n + x_{i,j}^n, & y_{i,j}^{n+1} &= v_{i,j}^n \Delta t^n + y_{i,j}^n, \\ S_{i,j}^{n+1} &= \frac{1}{2} [(x_{i+1,j+1}^{n+1} - x_{i,j}^{n+1})(y_{i,j+1}^{n+1} - y_{i+1,j}^{n+1}) + (x_{i,j+1}^{n+1} - x_{i+1,j}^{n+1})(y_{i+1,j+1}^{n+1} - y_{i,j}^{n+1})], \\ & i = 1, \dots, M, \quad j = 1, \dots, N. \end{aligned} \quad (4.10)$$

The time step Δt^n is chosen as follows

$$\Delta t^n = \lambda \min_{i=1, \dots, M, j=1, \dots, N} (\Delta l_{i+1/2,j+1/2}^n / c_{i+1/2,j+1/2}^n) \quad (4.11)$$

where $\Delta l_{i+1/2,j+1/2}^n$ is the shortest edge length of the cell $I_{i+1/2,j+1/2}$, and $c_{i+1/2,j+1/2}^n$ is the sound speed within this cell. The Courant number λ in the following tests is set to be 0.5 unless otherwise stated.

4.2 The scheme in the cylindrical coordinates

We seek to study the flow governed by the axisymmetric compressible Euler equations which have the following integral form in the Lagrangian formulation,

$$\begin{cases} \frac{d}{dt} \iint_{\Omega(t)} \rho r dx dr & = 0 \\ \frac{d}{dt} \iint_{\Omega(t)} M_x r dx dr & = - \int_{\Gamma(t)} p n_x r dl \\ \frac{d}{dt} \iint_{\Omega(t)} M_r r dx dr & = - \int_{\Gamma(t)} p n_r r dl + \iint_{\Omega(t)} (p + \rho u_\theta^2) dx dr \\ \frac{d}{dt} \iint_{\Omega(t)} M_\theta r dx dr & = - \iint_{\Omega(t)} \rho u_\theta u_r dx dr \\ \frac{d}{dt} \iint_{\Omega(t)} E r dx dr & = - \int_{\Gamma(t)} p u_n r dl \end{cases} \quad (4.12)$$

where ρ is the density, p is the pressure, M_x, M_r, M_θ are the momentum components in the axial, radial and azimuthal directions, and u_x, u_r, u_θ are the velocity components in the above mentioned directions. $\mathbf{n} = (n_x, n_r)$ is the unit outward normal to the boundary $\Gamma(t)$ in the $x-r$ coordinates. $u_n = (u_x, u_r) \cdot \mathbf{n}$ is the normal velocity. E is the total energy.

The values of the cell averages for the cell $I_{i+1/2, j+1/2}$, denoted by $\bar{\rho}_{i+1/2, j+1/2}$, $\bar{M}_{i+1/2, j+1/2}^x$, $\bar{M}_{i+1/2, j+1/2}^r$, $\bar{M}_{i+1/2, j+1/2}^\theta$ and $\bar{E}_{i+1/2, j+1/2}$, are defined as follows

$$\begin{aligned}\bar{\rho}_{i+1/2, j+1/2} &= \frac{1}{V_{i+1/2, j+1/2}} \iint_{I_{i+1/2, j+1/2}} \rho r dx dr, & \bar{M}_{i+1/2, j+1/2}^x &= \frac{1}{V_{i+1/2, j+1/2}} \iint_{I_{i+1/2, j+1/2}} M_x r dx dr, \\ \bar{M}_{i+1/2, j+1/2}^r &= \frac{1}{V_{i+1/2, j+1/2}} \iint_{I_{i+1/2, j+1/2}} M_r r dx dr, & \bar{M}_{i+1/2, j+1/2}^\theta &= \frac{1}{V_{i+1/2, j+1/2}} \iint_{I_{i+1/2, j+1/2}} M_\theta r dx dr, \\ \bar{E}_{i+1/2, j+1/2} &= \frac{1}{V_{i+1/2, j+1/2}} \iint_{I_{i+1/2, j+1/2}} E r dx dr.\end{aligned}$$

where $V_{i+1/2, j+1/2} = \iint_{I_{i+1/2, j+1/2}} r dx dr$ is the volume of the cell $I_{i+1/2, j+1/2}$.

The conservative semi-discrete scheme for the equations (4.12) has the following form

$$\begin{aligned}& \frac{d}{dt} \begin{pmatrix} \bar{\rho}_{i+1/2, j+1/2} V_{i+1/2, j+1/2} \\ \bar{M}_{i+1/2, j+1/2}^x V_{i+1/2, j+1/2} \\ \bar{M}_{i+1/2, j+1/2}^y V_{i+1/2, j+1/2} \\ \bar{M}_{i+1/2, j+1/2}^\theta V_{i+1/2, j+1/2} \\ \bar{E}_{i+1/2, j+1/2} V_{i+1/2, j+1/2} \end{pmatrix} \\ &= - \int_{\partial I_{i+1/2, j+1/2}} \hat{\mathbf{F}} dl + \begin{pmatrix} 0 \\ 0 \\ \iint_{I_{i+1/2, j+1/2}} (p + \rho u_\theta^2) dx dr \\ - \iint_{I_{i+1/2, j+1/2}} \rho u_\theta u_r dx dr \\ 0 \end{pmatrix}\end{aligned}\tag{4.13}$$

where

$$\int_{\partial I_{i+1/2, j+1/2}} \hat{\mathbf{F}} dl = \int_{\partial I_{i+1/2, j+1/2}} \begin{pmatrix} \hat{f}_D(\mathbf{U}_n^-, \mathbf{U}_n^+) \\ \hat{f}_{M^x}(\mathbf{U}_n^-, \mathbf{U}_n^+) \\ \hat{f}_{M^r}(\mathbf{U}_n^-, \mathbf{U}_n^+) \\ \hat{f}_{M^\theta}(\mathbf{U}_n^-, \mathbf{U}_n^+) \\ \hat{f}_E(\mathbf{U}_n^-, \mathbf{U}_n^+) \end{pmatrix} dl\tag{4.14}$$

and

$$\begin{cases} \hat{f}_D(\mathbf{U}_n, \mathbf{U}_n) &= 0, \\ \hat{f}_{M^x}(\mathbf{U}_n, \mathbf{U}_n) &= p n_x r \\ \hat{f}_{M^r}(\mathbf{U}_n, \mathbf{U}_n) &= p n_r r \\ \hat{f}_{M^\theta}(\mathbf{U}_n, \mathbf{U}_n) &= 0 \\ \hat{f}_E(\mathbf{U}_n, \mathbf{U}_n) &= p u_n r. \end{cases}\tag{4.15}$$

The calculation of the first term on the right hand side of Eq. (4.13) is similar to that in the Cartesian coordinates introduced in the above subsection. The calculation of the second term is performed by a suitable Gaussian integral in the corresponding cell to guarantee its high-order accurate approximation.

We use the same method as that used in the Cartesian coordinates to decide the velocity components (u_x, u_r) at the vertex in the x and r directions (since the grid moves just in the x - r coordinates, we only need to know u_x, u_r).

We also use the Runge-Kutta method to discretize the time derivatives in (4.13). The method to calculate the time step is the same as that in the Cartesian coordinates.

5 Numerical results in two space dimensions

It is much more difficult to simulate a 2D problem than to simulate a 1D one in the Lagrangian framework, mainly because of the mesh distortion in multi-dimensions. In this section, although we have run most examples using the first, second and third order schemes with the Godunov flux, the HLLC flux and the L-F flux respectively, we did unfortunately observe that for some very tough problems simulations with some of the fluxes such as the Godunov flux cannot run in a stable fashion. Since the L-F flux shows its best robustness in our simulation of 2D problems among these fluxes, we will only show the results performed by the L-F flux as representatives, although the results may not be the best for individual test cases among those using these three fluxes.

5.1 Numerical results in the Cartesian coordinates

5.1.1 Accuracy test

In the Cartesian coordinates, we choose the two-dimensional vortex evolution problem [28] as our accuracy test function. The vortex problem is described as follows: The mean flow is $\rho = 1$, $p = 1$ and $(u, v) = (1, 1)$ (diagonal flow). We add to this mean flow an isentropic vortex perturbations in (u, v) and the temperature $T = p/\rho$, no perturbation in the entropy

$$S = p/\rho^\gamma.$$

$$(\delta u, \delta v) = \frac{\epsilon}{2\pi} e^{0.5(1-r^2)}(-\bar{y}, \bar{x}), \quad \delta T = -\frac{(\gamma-1)\epsilon^2}{8\gamma\pi^2} e^{(1-r^2)}, \quad \delta S = 0$$

where $(-\bar{y}, \bar{x}) = (x-5, y-5)$, $r^2 = \bar{x}^2 + \bar{y}^2$, and the vortex strength is $\epsilon = 5$.

The computational domain is taken as $[0, 10] \times [0, 10]$, and periodic boundary conditions are used.

The convergence results for the first, second and third order ENO Lagrangian schemes at $t = 1$ are listed in Tables 5.1-5.3 respectively. In Tables 5.1 and 5.2, we can see the desired first and second order accuracy. However in Table 5.3 we cannot observe the expected third order accuracy. Further exploration indicates that this accuracy degeneracy cannot be cured by the modified ENO scheme via the introduction of a biasing factor in the stencil-choosing process. It represents a fundamental problem in our way of formulating the Lagrangian schemes. In a Lagrangian simulation, each cell represents a material particle, thus its shape may change with the movement of fluid, that means the cell with a quadrilateral shape initially may not keep its shape as a quadrilateral at a later time. It usually becomes a curved quadrilateral, while during our Lagrangian simulation the mesh is always supposed to be quadrilateral which is determined by the movement of its four vertices. This approximation of the mesh will bring second order error into the scheme. Thus for a Lagrangian scheme in multi-dimensions, it can be at most second order accurate if curved meshes are not used. We will not explore curved meshes in this paper, hence our ‘‘third order’’ scheme is indeed only second order accurate. However, in the following examples, we indeed often find better resolution in the fluid field by the third order scheme compared with that obtained by the lower order schemes, despite its formal second order accuracy.

5.1.2 Non-oscillatory tests

Example 5.1 (The Saltzman problem [10]). This is a well known difficult test case to validate the robustness of a Lagrangian scheme when the mesh is not aligned with the fluid flow. The problem consists of a rectangular box whose left end is a piston. The piston moves

Table 5.1: Errors of the first order scheme on 2D Cartesian coordinates for the vortex problem using $N_x \times N_y$ initially uniform mesh cells.

$N_x = N_y$	Norm	Density	order	Momentum	order	Energy	order
20	L_1	0.95E-2		0.27E-1		0.45E-1	
	L_∞	0.11E+0		0.20E+0		0.59E+0	
40	L_1	0.56E-2	0.78	0.14E-1	0.88	0.26E-1	0.81
	L_∞	0.46E-1	1.27	0.12E+0	0.78	0.27E+0	1.13
80	L_1	0.30E-2	0.88	0.77E-2	0.92	0.14E-1	0.91
	L_∞	0.26E-1	0.84	0.62E-1	0.92	0.14E+0	0.91
160	L_1	0.16E-2	0.94	0.40E-2	0.95	0.71E-2	0.95
	L_∞	0.13E-1	0.92	0.32E-1	0.95	0.75E-1	0.94

Table 5.2: Errors of the second order ENO scheme on 2D Cartesian coordinates for the vortex problem using $N_x \times N_y$ initially uniform mesh cells.

$N_x = N_y$	Norm	Density	order	Momentum	order	Energy	order
20	L_1	0.57E-2		0.13E-1		0.24E-1	
	L_∞	0.62E-1		0.14E+0		0.35E+0	
40	L_1	0.17E-2	1.74	0.42E-2	1.66	0.75E-2	1.70
	L_∞	0.31E-1	1.01	0.53E-1	1.40	0.13E+0	1.41
80	L_1	0.48E-3	1.84	0.12E-2	1.83	0.21E-2	1.83
	L_∞	0.92E-2	1.75	0.20E-1	1.42	0.39E-1	1.79
160	L_1	0.13E-3	1.89	0.31E-3	1.92	0.57E-3	1.90
	L_∞	0.28E-2	1.72	0.59E-2	1.74	0.13E-1	1.53

Table 5.3: Errors of the third order ENO scheme on 2D Cartesian coordinates for the vortex problem using $N_x \times N_y$ initially uniform mesh cells.

$N_x = N_y$	Norm	Density	order	Momentum	order	Energy	order
20	L_1	0.32E-2		0.61E-2		0.12E-1	
	L_∞	0.38E-1		0.67E-1		0.20E+0	
40	L_1	0.81E-3	1.99	0.14E-2	2.07	0.29E-2	2.02
	L_∞	0.12E-1	1.65	0.23E-1	1.54	0.65E-1	1.60
80	L_1	0.19E-3	2.10	0.33E-3	2.13	0.67E-3	2.12
	L_∞	0.25E-2	2.24	0.48E-2	2.25	0.13E-1	2.31
160	L_1	0.46E-4	2.05	0.79E-4	2.05	0.16E-3	2.05
	L_∞	0.74E-3	1.78	0.12E-2	1.95	0.32E-2	2.04

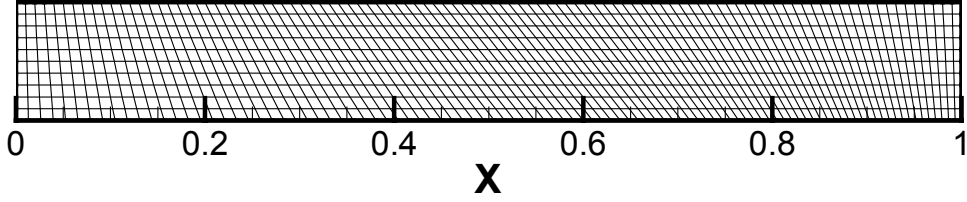


Figure 5.1: The initial mesh of the Saltzman problem.

into the box with a constant velocity of 1.0. The initial mesh is 100 cells in the x -direction and 10 cells in the y -direction which is defined by

$$x(i, j) = (i - 1)\Delta x + (11 - j) \sin(0.01\pi(i - 1))\Delta y, \quad y(i, j) = (j - 1)\Delta y$$

where $\Delta x = \Delta y = 0.01$. The initial mesh is displayed in Figure 5.1. Notice that the initial mesh is deliberately distorted to set it as a more demanding test case. The working fluid is described by an ideal gas with $\gamma = 5/3$. The initial conditions involve a stationary gas with a unity density and an internal energy of 10^{-4} . Reflective boundary conditions are used on the right, upper and lower boundaries. For this test case, it is necessary to first use a smaller Courant number in order to maintain stability. The Courant number λ is set to be 0.01 initially and returns to be 0.5 after $t = 0.01$. The analytic post shock density is 4.0 and the shock speed is 1.333. The purely Lagrangian numerical results are shown in Figures 5.2-5.4 for the time $t = 0.6$. At this time, the shock is expected to be located at $x = 0.8$. We can observe that our high order schemes preserve one-dimensional solution quite well except for the region near the top and bottom wall boundaries where the result is affected by the boundary conditions. Also we can see that the higher order schemes give better shock resolution for this example. Our results for this example compares well with those obtained in the literature.

Example 5.2 (The Sedov blast wave problem in a Cartesian coordinate system [25]). The Sedov blast wave problem models the expanding wave by an intense explosion in a perfect

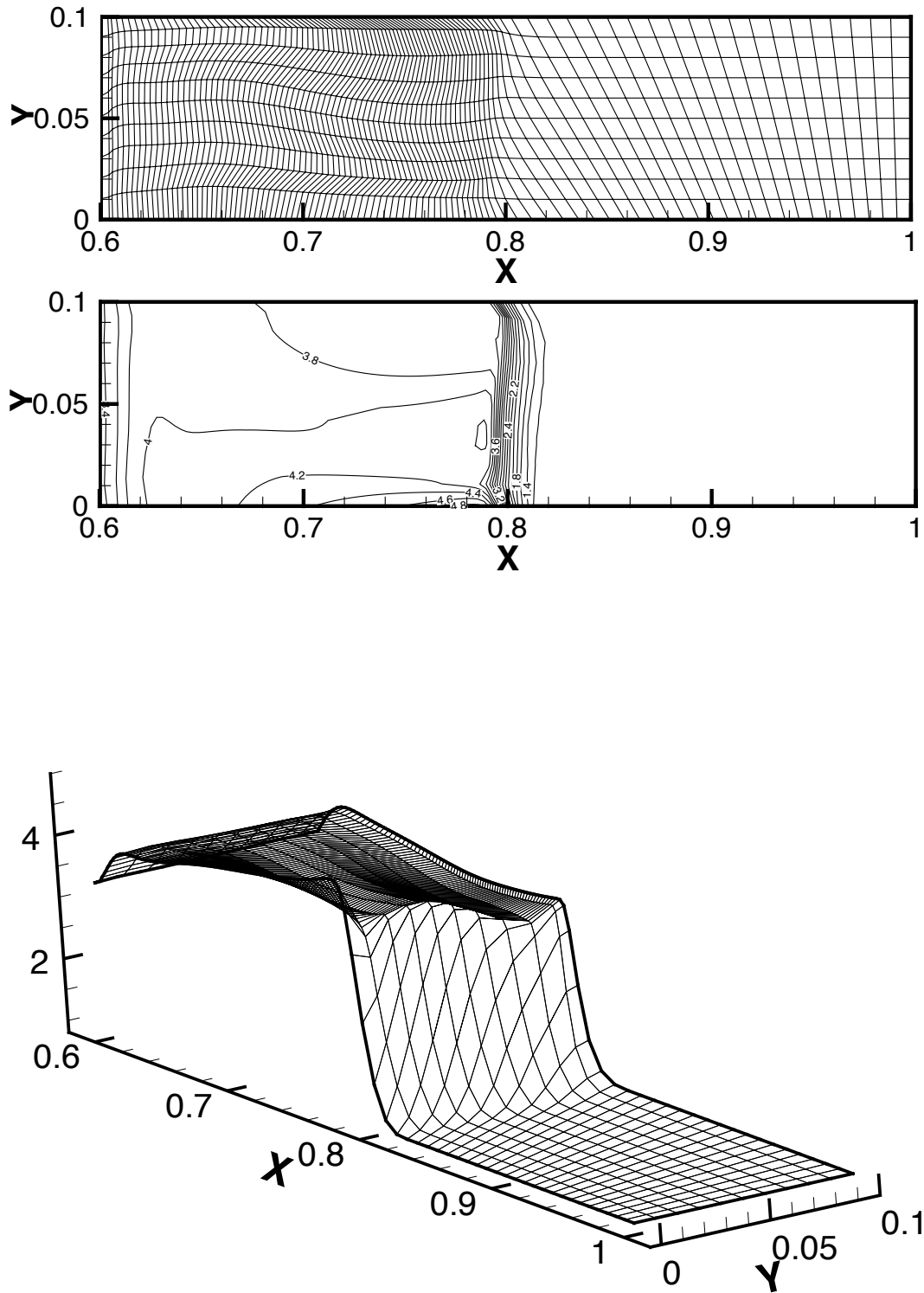


Figure 5.2: The first order results of the Saltzman problem at $t = 0.6$. Top: the mesh; Middle: density contour plots from 1.2 to 4.8 with 19 equally spaced contours; Bottom: density 3D surface plot.

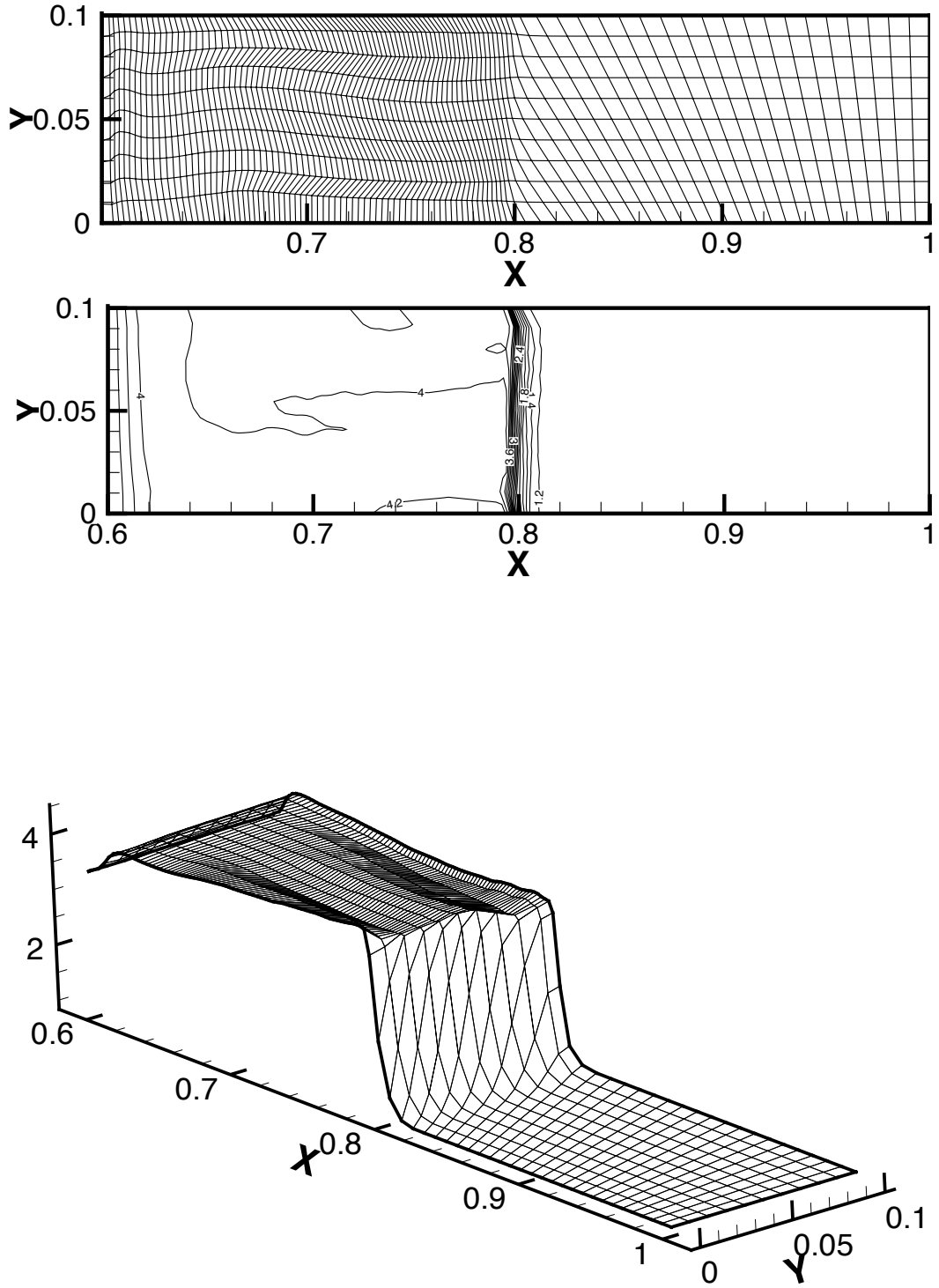


Figure 5.3: The second order results of the Saltzman problem at $t = 0.6$. Top: the mesh; Middle: density contour plots from 1.2 to 4.8 with 19 equally spaced contours; Bottom: density 3D surface plot.

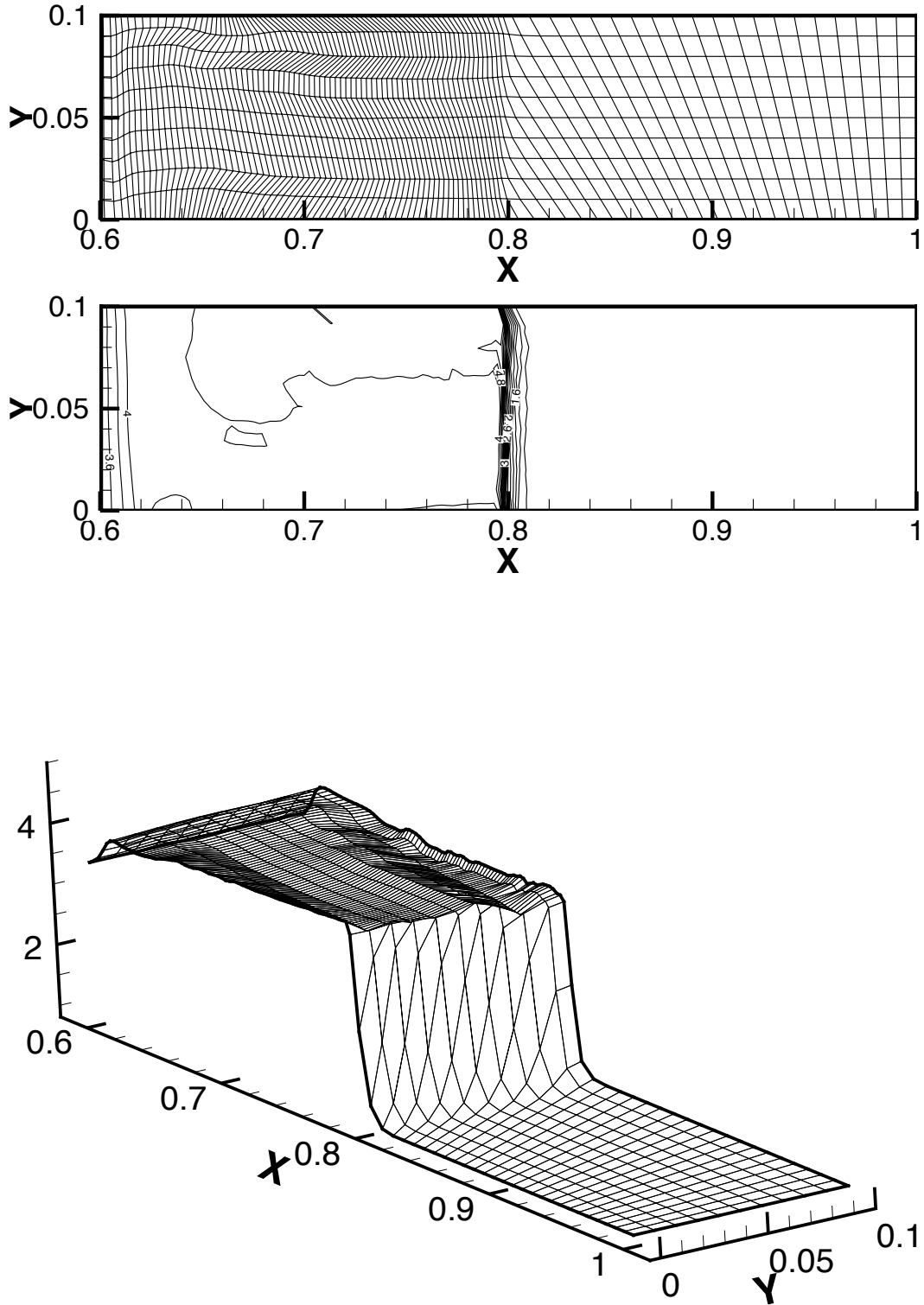


Figure 5.4: The third order results of the Saltzman problem at $t = 0.6$. Top: the mesh; Middle: density contour plots from 1.2 to 4.8 with 19 equally spaced contours; Bottom: density 3D surface plot.

gas. The simulation is first performed on a Cartesian grid whose initial uniform grid consists of 30×30 rectangular cells with a total edge length of 1.1 in both directions. The initial density is unity and the initial velocity is zero. The specific internal energy is zero except in the first zone where it has a value of 182.09. The analytical solution gives a shock at radius unity at time unity with a peak density of 6. Figure 5.5 shows the results by the purely Lagrangian calculations at the time $t = 1$. We can clearly see that the high order ENO scheme obtains higher peak density than the lower order one.

Example 5.3 (The Dukowicz problem). In this and the next examples, we will test the performance of our scheme in the ALE calculations. At each time step, we first use our Lagrangian scheme to update the solution and mesh, then we rezone the Lagrangian mesh to a more optimal position and finally remap the Lagrangian solutions to the new grid. The conservative remapping method is also based on the ENO methodology and is described in detail in [7]. The Dukowicz problem is a two dimensional shock refraction problem on an uneven mesh designed by Dukowicz and Meltz [10].

The computational domain consists of two adjacent regions with different densities but equal pressures. The left region is a 36×30 mesh with a vertical left boundary and a right boundary aligned at 30° to the horizontal direction. The right region is a 40×30 mesh uniformly slanted at 30° to the horizontal direction. See Figure 5.6. The initial conditions of the two regions are $\rho_L = 1, u_L = 0, p_L = 1$ and $\rho_R = 1, u_R = 0, p_R = 1.5$ respectively. The upper and lower boundaries are reflective and the left boundary is a piston, which moves from the left with velocity 1.48. The problem is run to a time of 1.3, just before the shock would leave the right region. The exact solution to the problem is shown in Figure 5.7 which is only valid away from the boundary as it is obtained under the assumption of an infinite medium. In this test, when the area of the minimum cell is less than 4×10^{-4} , we rezone the meshes by keeping the vertices at the left and right boundaries unchanged and redistributing the points in the x direction evenly. The numerical results using the ALE calculations are

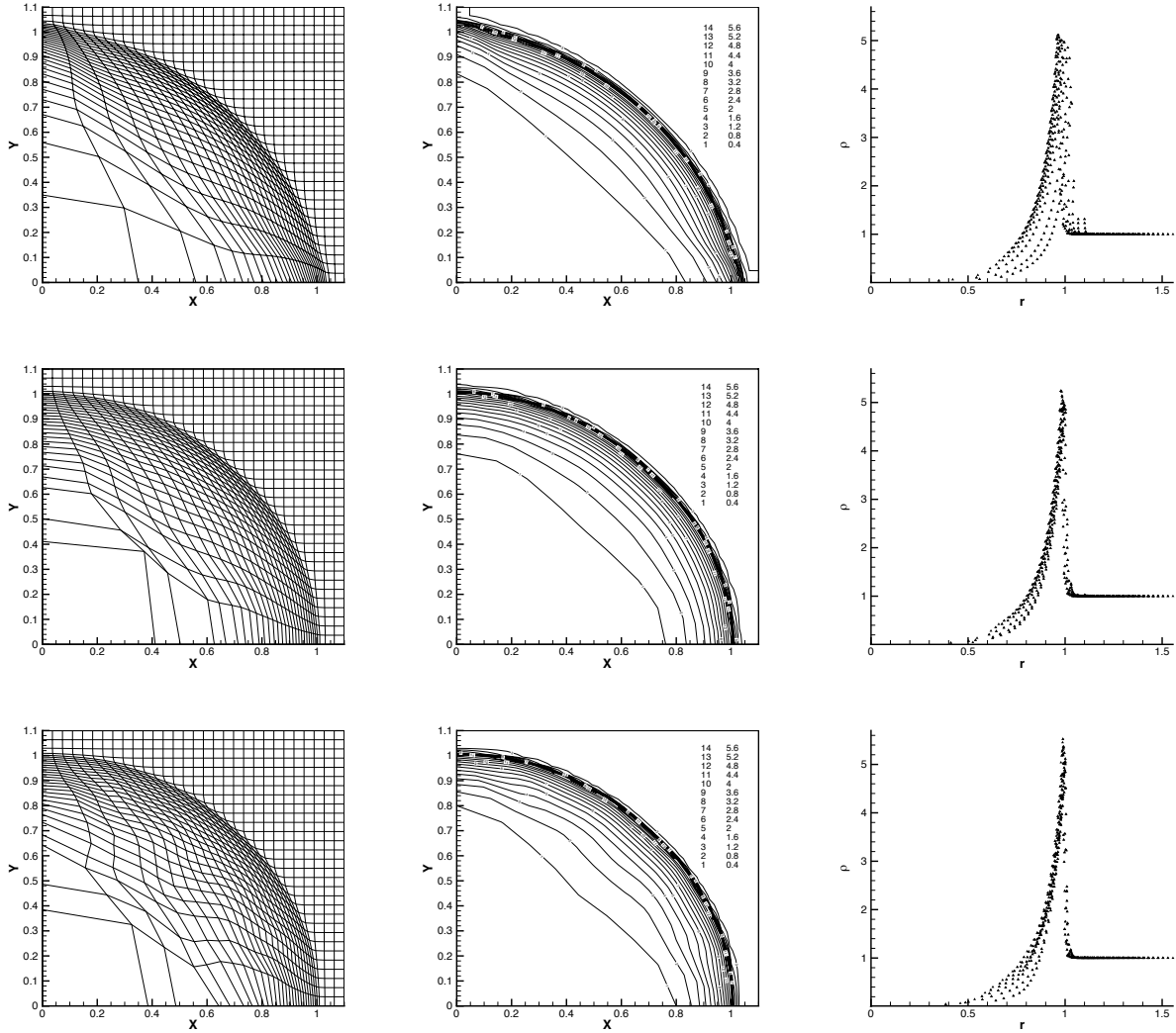


Figure 5.5: The Sedov problem at $t = 1$. Top: the first order results; Middle: the second order results; Bottom: the third order results. Left: the mesh; Middle: density contours; Right: density as a function of the radius.

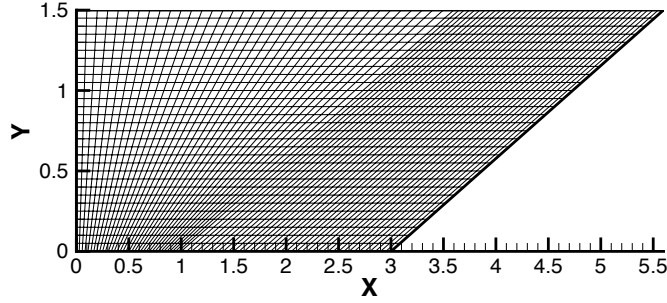


Figure 5.6: The initial computational grid.

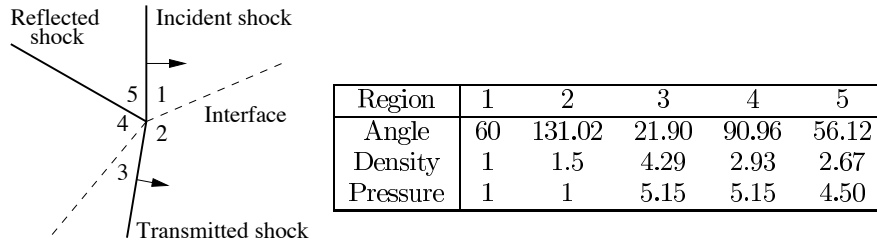


Figure 5.7: The exact solution for the Dukowicz problem.

shown at time $t = 1.3$ in Figures 5.8. The density contour plots in Figure 5.8 give consistent results with the exact solution. The results of the second and third order schemes show the interface along with the incident and the transmitted shocks clearly. The reflective shock does not show up clearly due to the small difference in density across it.

Example 5.4 (Double Mach reflection). The computational domain for this problem is chosen to be $[0, 4] \times [0, 1]$. The reflecting wall lies at the bottom of the computational domain starting $x = 1/6$. Initially a right-moving Mach 10 shock is positioned at $x = 1/6$, $y = 0$ and makes a 60° angle with the x -axis. For the bottom boundary, the exact post-shock condition is imposed for the part from $x = 0$ to $x = 1/6$ and a reflective boundary condition is used for the rest. At the top boundary of our computational domain, the flow values are set to describe the exact motion of the Mach 10 shock. In this example, we let the mesh return to its original state after each time step. We should note that initially the Courant number for the third order scheme takes a smaller value at 0.45 and after time=0.01 it returns to the usual value 0.5. The density results at the time $t = 0.2$ on a 240×60 uniform grid are shown in Figure 5.9, which demonstrate that our schemes also perform well using

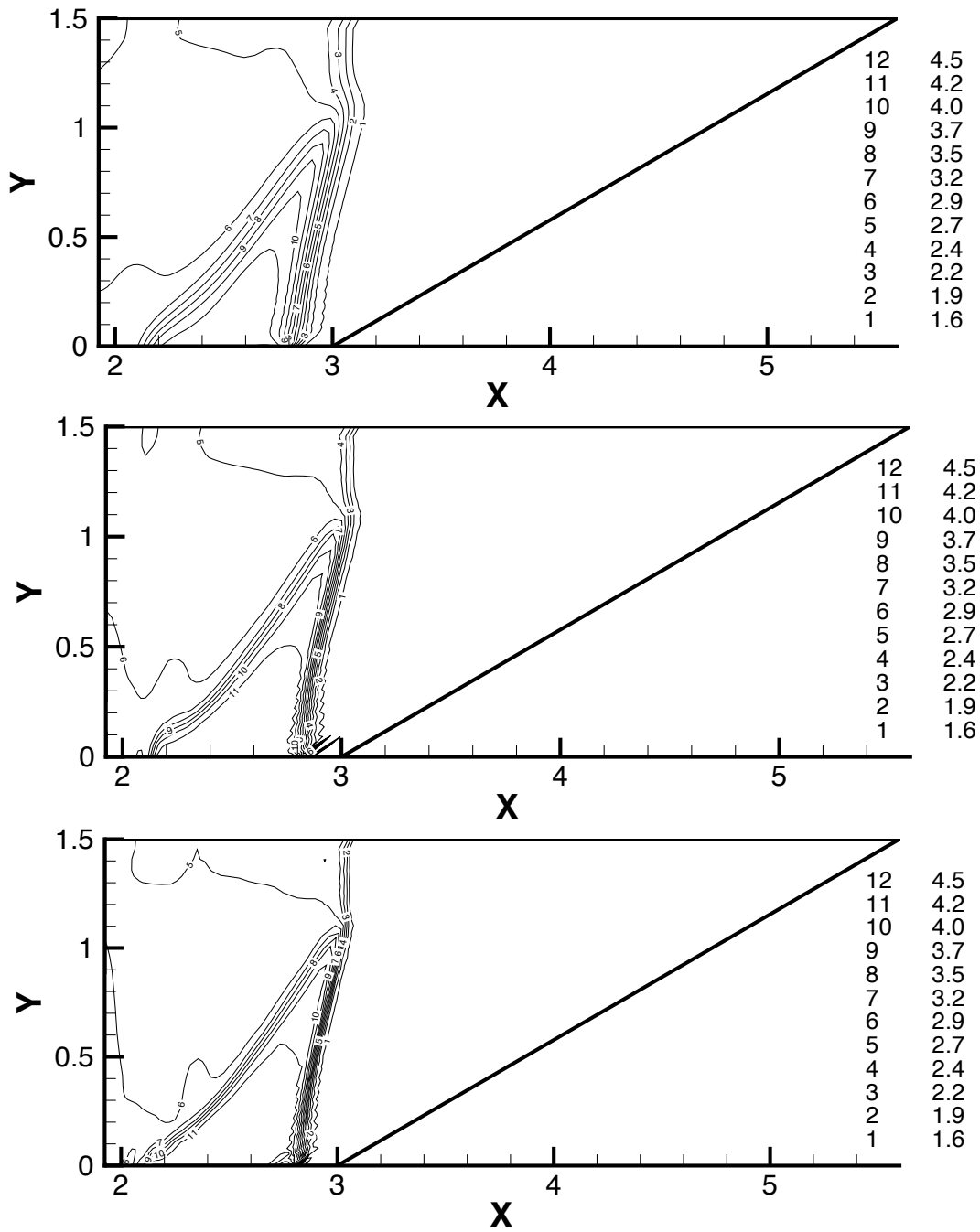


Figure 5.8: The density contour plot of the Dukowicz problem by using the ALE method. Top: first order; Middle: second order; Bottom: third order.

Table 5.4: Errors of the first order scheme on 2D cylindrical coordinates for the vortex problem using $N_x \times N_r$ initially uniform mesh cells.

$N_x = N_r$	Norm	Density	order	Momentum	order	Energy	order
20	L_1	0.17E-2		0.45E-2		0.92E-2	
	L_∞	0.31E-1		0.16E+0		0.19E+0	
40	L_1	0.88E-3	0.93	0.23E-2	0.95	0.48E-2	0.93
	L_∞	0.18E-1	0.78	0.84E-1	0.90	0.11E+0	0.78
80	L_1	0.44E-3	0.99	0.12E-2	0.99	0.24E-2	0.99
	L_∞	0.94E-2	0.95	0.43E-1	0.98	0.56E-1	0.95
160	L_1	0.22E-3	1.00	0.58E-3	1.00	0.12E-2	1.00
	L_∞	0.47E-2	0.99	0.21E-1	1.00	0.28E-1	0.99

the Eulerian mesh.

5.2 Numerical results in the cylindrical coordinates

5.2.1 Accuracy test

In the cylindrical coordinates, we test a problem with non-trivial velocities in the x and θ directions. The longitudinal vortex located in the r - θ coordinates moves along the symmetric x -coordinate with a constant velocity. Here we let $u_x = 2, u_r = 0$. u_θ is the azimuthal velocity of the vortex which has the same definition as that introduced in the previous subsection. To avoid the influence of the coordinate singularity at $r = 0$, we perform our accuracy tests on the computational domain $[0, 10] \times [1, 11]$. The results are presented in Tables 5.4-5.6 which show a satisfactory convergence performance in the cylindrical coordinates. In particular, we happily observe the desired third order accuracy both in L_1 and L_∞ norms in the results of our third order scheme, which is due to the mesh being able to keep its quadrilateral shape as the time marches in this test. The result also reinforces our previous claim about the reason of the accuracy degeneracy phenomenon due to the appearance of curved quadrilaterals, which is approximated by quadrilaterals in our numerical procedure, in the 2D Cartesian accuracy test.

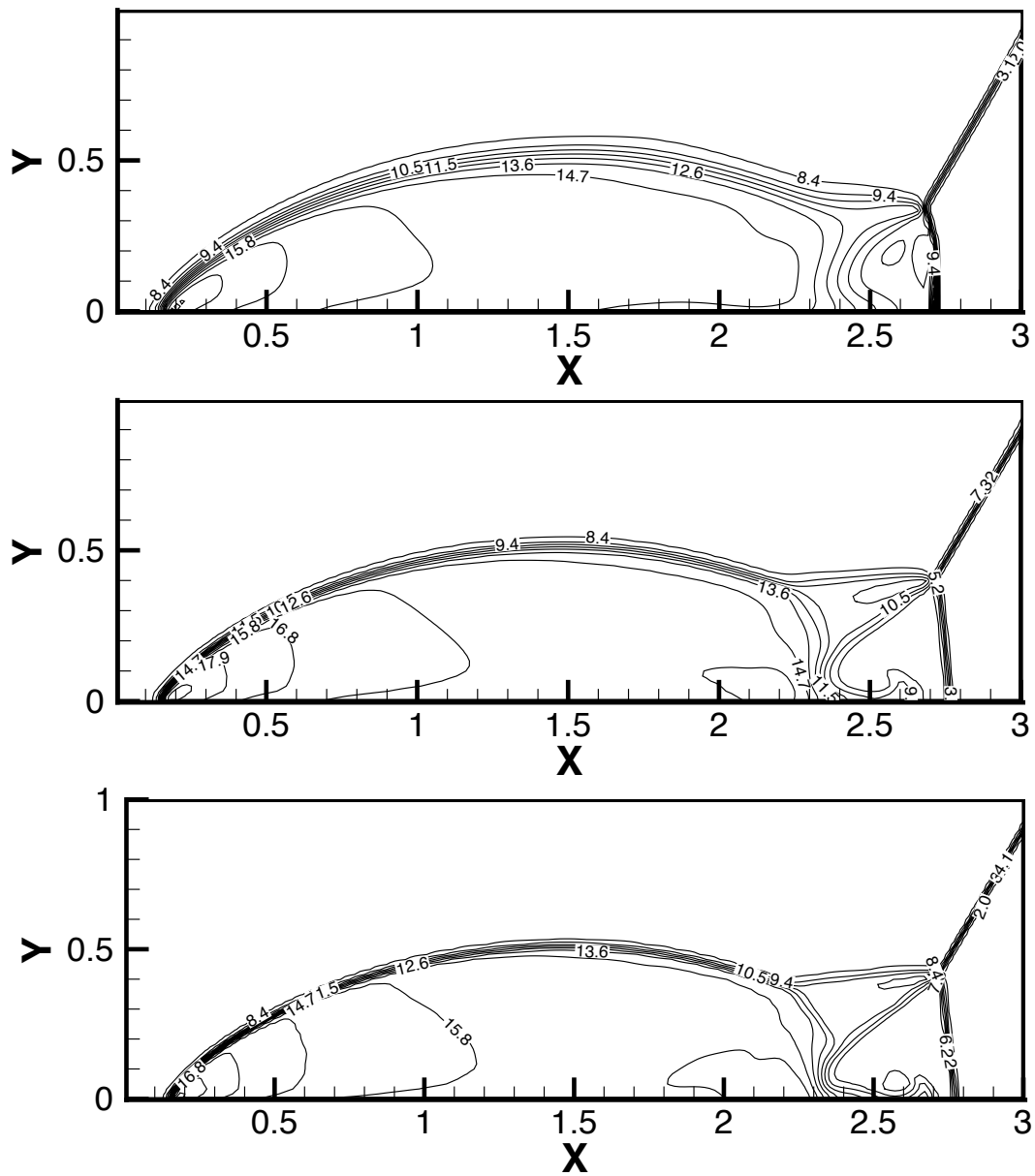


Figure 5.9: The results of the double Mach reflection problem with Eulerian meshes. Top: first order; Middle: second order; Bottom: third order. Density contour plots from 2 to 20 with 19 equally spaced contours.

Table 5.5: Errors of the second order ENO scheme on 2D cylindrical coordinates for the vortex problem using $N_x \times N_r$ initially uniform mesh cells.

$N_x = N_r$	Norm	Density	order	Momentum	order	Energy	order
20	L_1	0.61E-3		0.18E-2		0.33E-2	
	L_∞	0.13E-1		0.75E-1		0.77E-1	
40	L_1	0.12E-3	2.29	0.38E-3	2.21	0.68E-3	2.28
	L_∞	0.42E-2	1.60	0.21E-1	1.80	0.22E-1	1.81
80	L_1	0.29E-4	2.09	0.95E-4	2.01	0.16E-3	2.10
	L_∞	0.12E-2	1.77	0.66E-2	1.69	0.75E-2	1.56
160	L_1	0.72E-5	2.01	0.24E-4	1.97	0.39E-4	2.01
	L_∞	0.29E-3	2.05	0.20E-2	1.71	0.21E-2	1.87

Table 5.6: Errors of the third order ENO scheme on 2D cylindrical coordinates for the vortex problem using $N_x \times N_r$ initially uniform mesh cells.

$N_x = N_r$	Norm	Density	order	Momentum	order	Energy	order
20	L_1	0.22E-4		0.12E-3		0.73E-4	
	L_∞	0.86E-3		0.46E-2		0.27E-2	
40	L_1	0.30E-5	2.89	0.13E-4	3.23	0.10E-4	2.80
	L_∞	0.15E-3	2.54	0.57E-3	3.00	0.53E-3	2.33
80	L_1	0.31E-6	3.29	0.13E-5	3.34	0.11E-5	3.19
	L_∞	0.26E-4	2.52	0.75E-4	2.93	0.12E-3	2.15
160	L_1	0.31E-7	3.32	0.14E-6	3.27	0.15E-6	2.96
	L_∞	0.26E-5	3.31	0.71E-5	3.41	0.16E-4	2.95

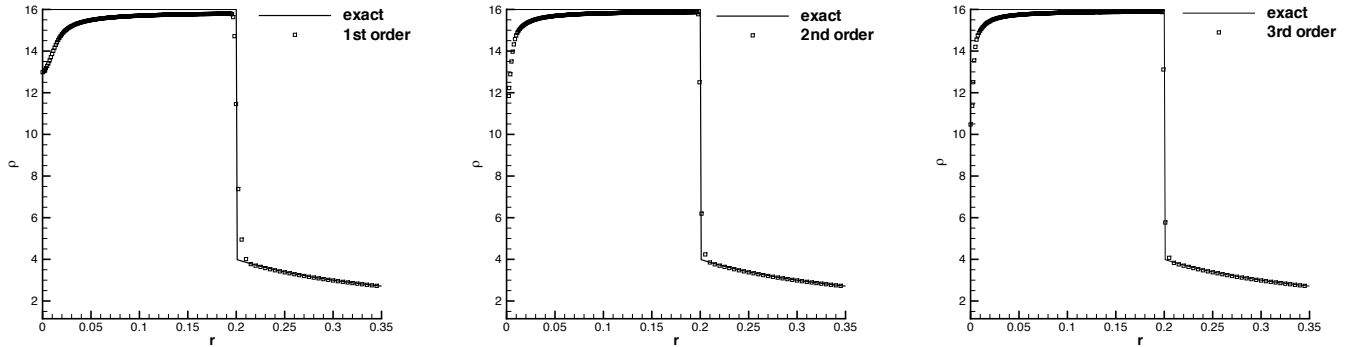


Figure 5.10: The density results of the Noh problem. Left: first order; Middle: second order; Right: third order.

5.2.2 Non-oscillatory tests

Example 5.5 (The Noh problem in a cylindrical coordinate system [22]). We test the Noh problem in a cylindrical coordinate system. The problem domain is $[0, 1] \times [0, 1]$. The initial state of the fluid is uniform with $(\rho, u, v, e) = (1, 0, -1, 0)$. The shock is generated in a perfect gas ($\gamma = 5/3$) by bringing the cold gas to rest at a rigid wall ($r = 0$). The analytical post shock density is 16 and the shock speed is $1/3$. Figure 5.10 shows the Lagrangian simulation results of our schemes at $t = 0.6$ with 10×200 grids. We observe good performance of our schemes, with an apparent advantage of the high order scheme over the low order one on the same mesh.

Example 5.6 (The Sedov problem in a cylindrical coordinate system with square grids [25]). We present the results of the Sedov blast wave in a cylindrical coordinate system as an example of a diverging shock wave. The initial computational domain is a 1.125×1.125 square consisting of 30×30 uniform cells. The initial density is unity and the initial velocity is zero. The specific internal energy is zero except in the first cell where it has a value of 1489.7. The analytical solution is a shock at radius unity at time unity with a peak density of 4. Figure 5.11 shows the results of our schemes by a purely Lagrangian computation. Here for the second order scheme the Courant number initially is taken as a smaller value at 0.45, and after time=0.01 it returns to the normal value 0.5. Figure 5.11 demonstrates that the

Lagrangian schemes also can produce good results for the Sedov problem in the cylindrical coordinates.

Example 5.7 (Interaction of a shock with longitudinal vortex [11]). The computational domain is $[-8, 4] \times [0, 5]$. At $t = 0$, there is a mean flow with a stationary shock at $x = 0$, that is

$$(\rho, p, u_x, u_r, u_\theta) = \begin{cases} (\rho_1, p_1, u_{x,1}, u_{r,1}, u_{\theta,1}) = (1, 1, \gamma^{1/2}M_1, 0, 0), & x < 0 \\ (\rho_2, p_2, u_{x,2}, u_{r,2}, u_{\theta,2}) = \left[\frac{(\gamma+1)M_1^2}{2+M_1^2(\gamma-1)}, \frac{2\gamma M_1^2 - (\gamma-1)}{(\gamma+1)}, M_2 \sqrt{\gamma \frac{p_2}{\rho_2}}, 0, 0 \right], & x > 0 \end{cases}$$

where M_1 is the Mach number at the upstream of the shock ($x < 0$) and $M_2 = \frac{2+M_1^2(\gamma-1)}{2\gamma M_1^2 - (\gamma-1)}$ is the Mach number at the downstream of the shock ($x > 0$).

Next, we superimpose an isentropic vortex with its axis along $r = 0$ on the upstream of the mean flow. The perturbation of azimuthal velocity u'_θ and temperature T' associated with the vortex are given by

$$u'_\theta = \frac{\epsilon r}{2\pi} e^{0.5(1-r^2)}, \quad T' = -\frac{(\gamma-1)\epsilon^2}{8\gamma\pi^2 r_0^2} e^{1-r^2} \quad (5.1)$$

where r_0 is the vortex core radius and ϵ is a non-dimensional circulation at $r = 1$. The axial and radial velocities u'_x, u'_r are zero. With no perturbation of the entropy $S = \log(p/\rho^\gamma)$ of the original mean flow, the final perturbed flow at $x < 0$ is as follows

$$\rho = \left(\frac{T_1 + T'}{p_1/\rho_1^\gamma} \right)^{1/(\gamma-1)}, \quad p = (T_1 + T')\rho, \quad u_x = u_{x,1}, \quad u_r = 0, \quad u_\theta = u'_\theta$$

where $T_1 = p_1/\rho_1$. In this test, we set $M_1 = 2, \epsilon = 7$ and $r_0 = 1$. The supersonic inflow, characteristic, symmetry and Neumann conditions are used at the left, right, bottom and upper boundaries respectively. The initial grid is uniform. After every three Lagrangian time steps, we take the rezoning and remapping steps to return the Lagrangian grid to the initial grid. The density results at two typical times of our schemes are given in Figures 5.12. From these figures we observe that the resolution of high order schemes on coarser meshes are comparable to that of low order schemes on finer meshes, and more details of the

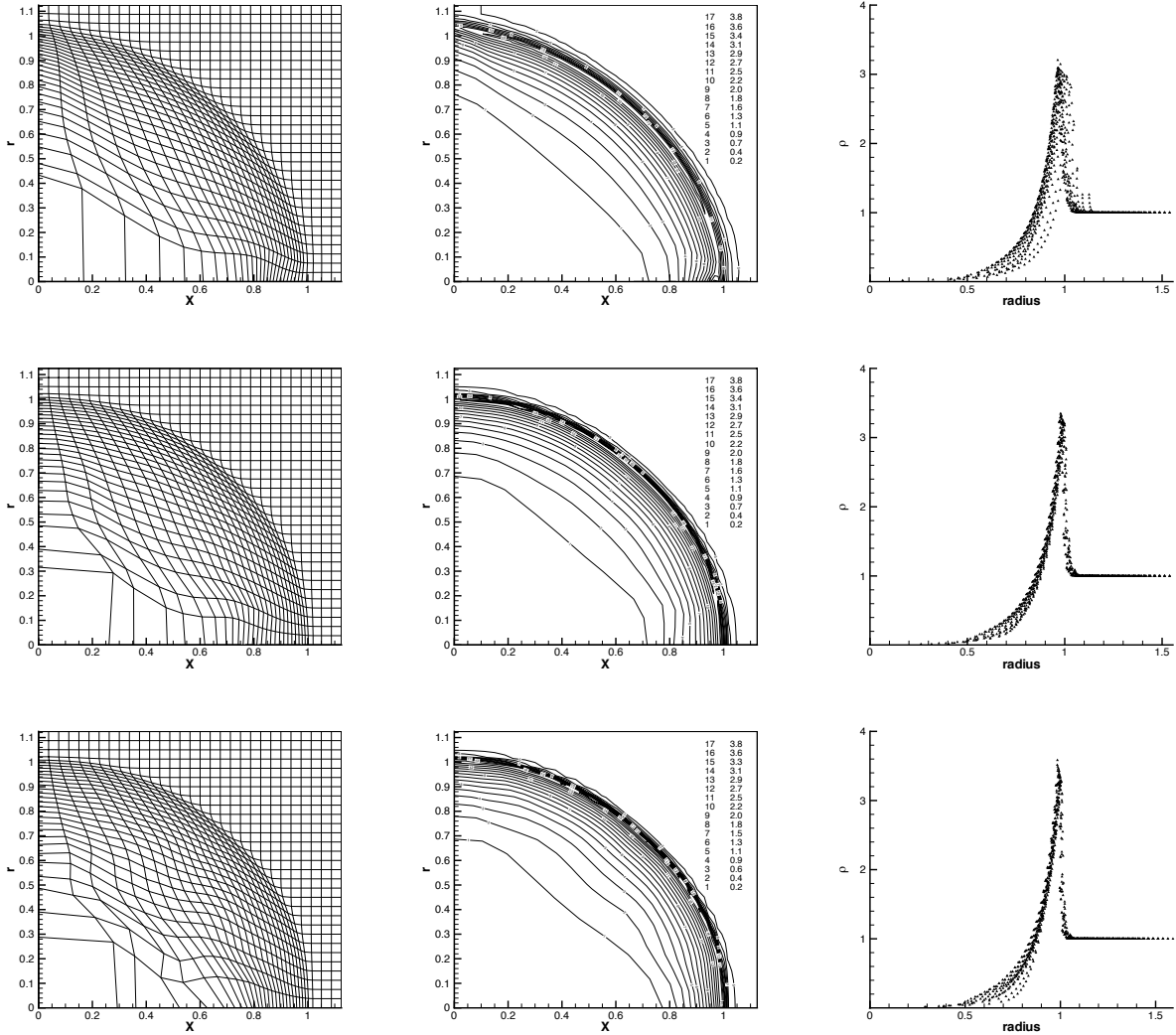


Figure 5.11: The Sedov problem at $t = 1$. Top: the first order results; Middle: the second order results; Bottom: the third order results. Left: the mesh; Middle: density contours; Right: density as a function of the radius.

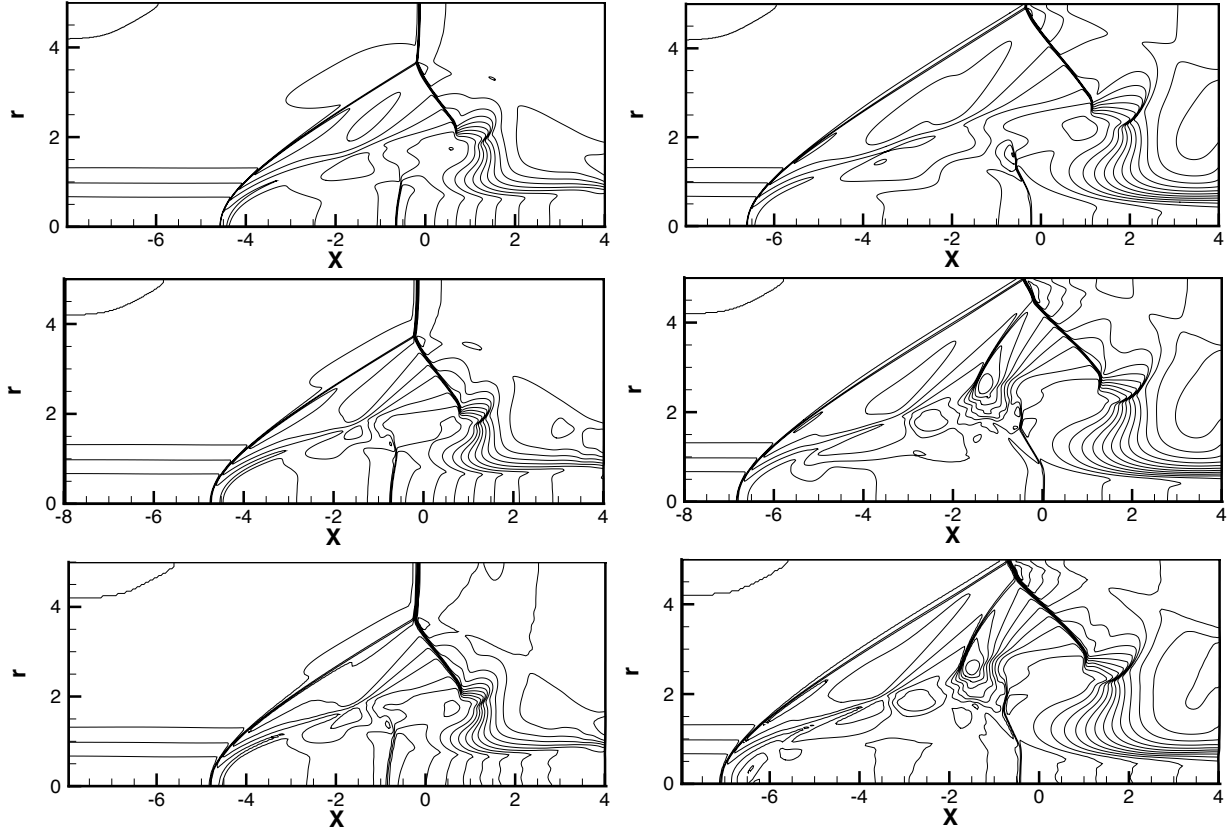


Figure 5.12: The density contours at $t = 7$ (Left) and $t = 11$ (Right). Contour levels from 0.4 to 3.2 with 15 equally spaced contours. Top: first order scheme with 1000×400 cells; Middle: second order scheme with 500×200 cells; Bottom: third order scheme with 250×100 cells.

fluid are captured by using higher order schemes. These results are also consistent with the results shown in [11].

6 Concluding remarks

In this paper we have described a class of new Lagrangian schemes for solving Euler equations which are based on high order essentially non-oscillatory (ENO) reconstruction both in the Cartesian coordinates and in the cylindrical coordinates. The schemes are conservative for density, momentum and total energy, maintain formal high order accuracy both in space and time and can achieve at least uniformly second order accuracy with moving and distorted Lagrangian meshes, are essentially non-oscillatory, and have no parameters to be tuned for

individual test cases. It can be extended to higher order accuracy by using curved meshes. Comparing with many current Lagrangian schemes, our ENO schemes overcome some of their disadvantages such as non-conservativity of momentum and total energy, low accuracy, and the existence of parameters which must be adjusted for individual test cases. One dimensional and two dimensional examples in the Cartesian as well as cylindrical coordinates have been presented which demonstrate the good performance of the schemes both in purely Lagrangian and in ALE calculations. Although we have only performed tests on quadrilateral grids, the strategy can be used on any polygon grid such as triangles. The investigation and improvement of these high order schemes in multi dimensions in terms of accuracy, resolution, and desirable properties such as symmetry preservation, constitute future work.

References

- [1] R. Abgrall, *On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation*, Journal of Computational Physics, 114, 1994, 45-58.
- [2] R. Abgrall, J. Breil, P.H. Maire and J. Ovardia, *A cell-centered Lagrangian scheme for two-dimensional compressible flow problems*, SIAM Journal on Scientific Computing, to appear.
- [3] D.J. Benson, *Computational methods in Lagrangian and Eulerian hydrocodes*, Computer Methods in Applied Mechanics and Engineering, 99, 1992, 235-394.
- [4] D.J. Benson, *Momentum advection on a staggered mesh*, Journal of Computational Physics, 100, 1992, 143-162.
- [5] J.C. Campbell and M.J. Shashkov, *A tensor artificial viscosity using a mimetic finite difference algorithm*, Journal of Computational Physics, 172, 2001, 739-765.

- [6] E.J. Caramana, D.E. Burton, M.J. Shashkov and P.P. Whalen, *The construction of compatible hydrodynamics algorithms utilizing conservation of total energy*, Journal of Computational Physics, 146, 1998, 227-262.
- [7] J. Cheng and C.-W. Shu, *A high order accurate conservative remapping method on staggered meshes*, Applied Numerical Mathematics, to appear.
- [8] B. Després and C. Mazeran, *Lagrangian gas dynamics in two dimensions and lagrangian systems*, Archive for Rational Mechanics and Analysis, 178, 2005, 327-372.
- [9] J.K. Dukowicz, M.C. Cline and F.L. Addessio, *A general topology Godunov method*, Journal of Computational Physics, 82, 1989, 29-63.
- [10] J.K. Dukowicz and B.J.A. Meltz, *Vorticity errors in multidimensional Lagrangian codes*, Journal of Computational Physics, 99, 1992, 115-134.
- [11] G. Erlebacher, M.Y. Hussaini and C.-W. Shu, *Interaction of a shock with a longitudinal vortex*, Journal of Fluid Mechanics, 337, 1997, 129-153.
- [12] A. Harten, B. Engquist, S. Osher and S.R. Chakravarthy, *Uniformly high order accurate essentially non-oscillatory schemes, III*, Journal of Computational Physics, 71, 1987, 231-303.
- [13] A. Harten and S. Osher, *Uniformly high-order accurate nonoscillatory schemes I*, SIAM Journal on Numerical Analysis, 24, 1987, 279-309.
- [14] C. Hirt, A. Amsden and J. Cook, *An arbitrary Lagrangian-Eulerian computing method for all flow speeds*, Journal of Computational Physics, 14, 1974, 227-253.
- [15] G. Jiang and C.-W. Shu, *Efficient implementation of weighted ENO schemes*, Journal of Computational Physics, 126, 1996, 202-228.

- [16] D.S. Kershaw, M.K. Prasad, M.J. Shaw and J.L. Milovich, *3D unstructured mesh ALE hydrodynamics with the upwind discontinuous finite element method*, Computer Methods in Applied Mechanics and Engineering, 158, 1998, 81-116.
- [17] B. Koobus and C. Farhat, *Second-order time-accurate and geometrically conservative implicit schemes for flow computations on unstructured dynamic meshes*, Computer Methods in Applied Mechanics and Engineering, 170, 1999, 103-129.
- [18] R. Loubère and M.J. Shashkov, *A subcell remapping method on staggered polygonal grids for arbitrary-Lagrangian-Eulerian methods*, Journal of Computational Physics, 209, 2005, 105-138.
- [19] H. Luo, J.D. Baum and R. Löhner, *On the computation of multi-material flows using ALE formulation*, Journal of Computational Physics, 194, 2004, 304-328.
- [20] L.G. Margolin, *Introduction to "An arbitrary Lagrangian-Eulerian computing method for all flow speeds"*, Journal of Computational Physics, 135, 1997, 198-202.
- [21] J. von Neumann and R.D. Richtmyer, *A method for the calculation of hydrodynamics shocks*, Journal of Applied Physics, 21, 1950, 232-237.
- [22] W.F. Noh, *Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux*, Journal of Computational Physics, 72, 1987, 78-120.
- [23] J.S. Peery and D.E. Carroll, *Multi-material ALE methods in unstructured grids*, Computer Methods in Applied Mechanics and Engineering, 187, 2000, 591-619.
- [24] A. Rogerson and E. Meiberg, *A numerical study of the convergence properties of ENO schemes*, Journal of Scientific Computing, 5, 1990, 151-167.
- [25] L.I. Sedov, *Similarity and Dimensional Methods in Mechanics*, Academic Press, New York, 1959.

- [26] C.-W. Shu, *Numerical experiments on the accuracy of ENO and modified ENO schemes*, Journal of Scientific Computing, 5, 1990, 127-149.
- [27] C.-W. Shu, *Preface to the republication of “Uniformly high order essentially non-oscillatory schemes, III”, by Harten, Engquist, Osher, and Chakravarthy*, Journal of Computational Physics, 131, 1997, 1-2.
- [28] C.-W. Shu, *Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws*, in *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, B. Cockburn, C. Johnson, C.-W. Shu and E. Tadmor (Editor: A. Quarteroni), Lecture Notes in Mathematics, volume 1697, Springer, Berlin, 1998, pp.325-432.
- [29] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock-capturing schemes*, Journal of Computational Physics, 77, 1988, 439-471.
- [30] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock capturing schemes II*, Journal of Computational Physics, 83, 1989, 32-78.
- [31] C.-W. Shu, T.A. Zang, G. Erlebacher, D. Whitaker and S. Osher, *High-order ENO schemes applied to two- and three-dimensional compressible flow*, Applied Numerical Mathematics, 9, 1992, 45-71.
- [32] R.W. Smith, *AUSM(ALE): a geometrically conservative arbitrary Lagrangian-Eulerian flux splitting scheme*, Journal of Computational Physics, 150, 1999, 268-286.
- [33] H.Z. Tang and T.T. Liu, *A note on the conservative schemes for the Euler equations*, Journal of Computational Physics, 218, 2006, 451-459.
- [34] E.F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Springer-Verlag, Berlin, 1999.

- [35] P.R. Woodward and P. Colella, *The numerical simulation of two-dimensional fluid flow with strong shocks*, Journal of Computational Physics, 54, 1984, 115-173.