# Second order conservative Lagrangian DG schemes for compressible flow and their application in preserving spherical symmetry in two-dimensional cylindrical geometry

Wenjing Feng [a] , Juan Cheng [b] , Chi-Wang Shu [c]

**Abstract**

In this paper, we construct a second-order cell-centered Lagrangian discontinuous Galerkin (DG) scheme for the two-dimensional compressible Euler equations on quadrilateral meshes. This Lagrangian DG scheme is based on the physical coordinates rather than the fixed reference coordinates, hence it does not require studying the evolution of the Jacobian matrix for the flow mapping between the different coordinates. The conserved variables are solved directly, and the scheme can preserve the conservation property for mass, momentum and total energy. The strong stability preserving (SSP) Runge-Kutta (RK) method is used for the time discretization. Furthermore, there are two main contributions. Firstly, differently from the previous work, we design a new Lagrangian DG scheme which is truly second-order accurate for all the variables such as density, momentum, total energy, pressure and velocity, while the similar DG schemes in the literature may lose second-order accuracy for certain variables, as shown in numerical experiments. Secondly, as an extension and application, we develop a particular Lagrangian DG scheme in the cylindrical geometry, which is designed to be able to preserve one-dimensional spherical symmetry for all the linear polynomials in two-dimensional cylindrical coordinates when computed on an equal-angle-zoned initial grid. The distinguished feature is that it can maintain both the spherical symmetry and conservation properties, which is very important for many applications such as implosion problems. A series of numerical experiments in the two-dimensional Cartesian and cylindrical coordinates are given to demonstrate the good performance of the Lagrangian DG schemes in terms of accuracy, symmetry and non-oscillation.

**Keywords:** Lagrangian DG scheme; Second-order accuracy; Conservation; Spherical symmetry preservation; Compressible Euler equations

[a] School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, China and Graduate School, China Academy of Engineering Physics, Beijing 100088, China. E-mail: fengwenjing22@gscaep.ac.cn.

[b] Corresponding author. Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing 100088, China and HEDPS, Center for Applied Physics and Technology, and College of Engineering, Peking University, Beijing 100871, China. E-mail: cheng_juan@iapcm.ac.cn. Research is supported in part by NSFC grant 12031001 and National Key R&D Program of China No. 2022YFA1004500.

[c] Division of Applied Mathematics, Brown University, Providence, RI 02912, United States of America. E-mail: chi-wang_shu@brown.edu. Research is supported in part by NSF grant DMS-2309249.

# 1 Introduction

In many fields involving multi-material fluid flows such as astrophysics, inertial confinement fusion (ICF) and computational fluid dynamics (CFD), numerical simulations aim to track fluid motion as closely as possible, including locating and tracking material interfaces sharply, and achieving high resolution of the wave structure. For this reason, the Lagrangian method is regarded as the first choice. The Lagrangian method is based on the grid moving with local fluid velocity, where there is no mass exchange across the moving cell edges. Compared with the Eulerian method based on the fixed grid, the Lagrangian method has the significant advantage in capturing material interfaces and contact discontinuities automatically and sharply. Moreover, the Lagrangian method should be particularly careful to ensure that the cell mass is not changed during the movement of the grid, especially for high-dimensional applications.

In the past decades, there has been a lot of work devoted to developing the Lagrangian methods. Some work is based on the staggered-grid discretization, where velocity (momentum) is stored at the vertices, while density and internal energy are within the cells, such as [1, 5–7, 27, 37]. The artificial viscosity term is often needed to stablize the scheme. The other work is built on the cell-centered discretization [8, 9, 20, 33, 35, 36], where all the variables are defined at the cell centers, and evolved on the same control volume. Unlike the staggered schemes, the cell-centered schemes usually do not require additional artificial viscosity, as standard high resolution non-oscillatory techniques are directly applicable. Also, they are easier to achieve high-order accuracy, preserve the conservation property, and design the remapping algorithms when necessary.

As another aspect of the Lagrangian simulation, it is critical but challenging to preserve the spherical symmetry property of a scheme in two-dimensional cylindrical coordinates, when the original physical problems have this symmetry. In fact, in many applications such as the implosion problems, there are many three-dimensional cylindrical symmetric models such as sphere-shape capsules and cylinder-shape hohlraum. In order to reduce the computational cost, the Lagrangian methods in two-dimensional cylindrical coordinates are used to simulate these models. Thus, the preservation of physical symmetry needs to be considered in numerical simulations. If the spherical symmetry of a scheme is lost, a small spherical symmetry deviation caused by numerical errors may be amplified by physical or numerical instabilities which may produce unpredicted large errors.

The spherical symmetry-preserving property of the Lagrangian schemes in two-dimensional cylindrical coordinates has been widely studied. In the earlier years, there were the area-weighted schemes [2, 6, 34, 43] discretized on the planar area, and the true volume schemes [3] discretized on the genuine control volume. Unfortunately, these spherical symmetry-preserving Lagrangian schemes may not satisfy the conservation of momentum and total energy, and are only first-order accurate. In [10, 11, 13], the above difficulties have been resolved in a finite volume framework. It is worth mentioning that Cheng and Shu in [13] pioneeringly proposed a second-order cell-centered Lagrangian finite volume scheme based on the true control volume discretizations in two-dimensional cylindrical coordinates. By compatible discretizations of the source term in the momentum equation and maintaining the symmetry of the reconstructed polynomials in the local coordinates, the second-order scheme is designed to preserve one-dimensional spherical symmetry in two-dimensional cylindrical geometry using an equal-angel-zoned initial grid without losing the conservation of mass, momentum and total energy. Later, in [26], Ling et al. further extended this idea and developed a cell-centered Lagrangian scheme which can achieve positivity-preserving and symmetry-preserving properties simultaneously.

Note that most of the methods mentioned above are based on the finite volume framework. Thus, an important aspect worth exploring is the application of discontinuous Galerkin (DG) method in the Lagrangian simulations. The DG method was first introduced to solve the linear steady-state neutron transport equation in [39]. Later, it was developed into Runge-Kutta discontinuous Galerkin (RKDG) methods by Cockburn et al. in [14–18], in which the authors established a framework to easily solve nonlinear time-dependent problems such as the Burgers equation, Euler equations, and shallow water equations, etc. Compared with the finite volume methods, the DG methods have distinguished advantages in smaller stencil, easiness to be parallelized, flexibility on complex geometry and on $h$-$p$ adaptation.

In the past years, there has been some research on the Lagrangian DG methods. In [22, 32, 44, 45], the total Lagrangian DG schemes have been extensively discussed. The total Lagrangian schemes refer to the discretization on a fixed reference configuration, which is usually the initial one. The physical variables in different configurations are related by the deformation gradient tensor, i.e., the Jacobian matrix associated to the flow mapping. To ensure the consistency between the reference and the physical configurations, the deformation gradient tensor has to satisfy a constraint condition, namely the Piola compatibility condition [45]. In the total Lagrangian formulation, the variables solved directly are mostly the primitive variables, i.e., the specific volume, velocity, and specific total energy. Similar research can be found in [28–31]. Recently, in [46,47], the authors developed a cell-centered updated Lagrangian DG scheme, where the updated Lagrangian scheme refers to the discretization on the moving physical configuration of the flow. The variables solved directly are the conservative ones, i.e., the density, momentum and total energy. The basis functions are defined on the time-dependent moving mesh, rather than on a fixed frame. The second-order accuracy of pressure or internal energy for two-dimensional numerical examples is shown in these papers. However, we observe that the second-order accuracy for the conserved variables may be lost for this kind of schemes, see the numerical results in Section 5. Moreover, in [29], Liu et al. presented two different Lagrangian DG schemes in the cylindrical coordinates: the first one is the true volume DG scheme and the second one is the area-weighted DG scheme, in which the physical evolution equations for the specific volume, velocity and specific total energy are discretized in these two schemes. Unfortunately, these two schemes cannot maintain the conservation and spherical symmetry properties simultaneously. To the best knowledge of the authors, no previous work focused on the Lagrangian DG scheme that preserves both the spherical symmetry and conservation properties in the cylindrical coordinates.

In this paper, we will construct a new second-order cell-centered Lagrangian discontinuous Galerkin scheme for the two-dimensional compressible Euler equations on quadrilateral meshes. Here we only consider the pure Lagrangian scheme without involving the rezoning and remapping phases. This Lagrangian DG scheme belongs to the updated Lagrangian formulation, and the DG space is defined in the physical coordinates rather than the fixed reference coordinates, hence there is no need to study the evolution of the Jacobian matrix for the flow mapping between the different coordinates. To be more specific, we solve the conserved variables such as density, momentum and total energy directly, so the scheme can keep the conservation property naturally. The HLLC numerical flux is used, and it has a particularly simple form in the Lagrangian description [12]. To suppress spurious oscillations, we use the multi-resolution weighted essentially non-oscillatory (WENO) limiters [48]. The time discretization is based on the second-order strong stability preserving (SSP) Runge-Kutta (RK) method [41]. As an extension and application, we develop the above framework in the two-dimensional cylindrical geometry, and design a spherical symmetry-preserving conservative Lagrangian DG scheme. Furthermore, there are two main contributions.

(1) Differently from the previous work, the Lagrangian DG scheme in this paper is second-order accurate for all the variables such as density, momentum, total energy, pressure and velocity, while the existing updated DG schemes may lose second-order accuracy for certain variables. In fact, according to the framework proposed by Cockburn et al. in [14–18], the DG scheme is constructed as (2.6) without any adjustments, which can achieve arbitrarily high order accuracy. In the Lagrangian simulation, the grid moves with the fluid velocity, so the $\mathbf{u}-\mathbf{u}_g$ flux term in (2.6) is thrown away simply in the previous literature, where $\mathbf{u}$ is the fluid velocity and $\mathbf{u}_g$ is the grid velocity. This treatment leads to some flaws in accuracy, and we have seen that the second-order accuracy for certain variables may be lost in numerical experiments in Section 5. The simplest way to deal with this flaw is to recover the $\mathbf{u}-\mathbf{u}_g$ flux term, but it will lead to mass exchange between neighboring cells, which is not allowed in the pure Lagrangian method. To balance the above two issues, we add this $\mathbf{u}-\mathbf{u}_g$ flux term only to equations where the basis functions are not equal to 1, i.e., we assume that the $\mathbf{u}-\mathbf{u}_g$ flux term is equal to 0 only in the equations of cell averages. In this way, we will obtain a truly second-order Lagrangian DG scheme without mass exchange between neighboring cells. See Section 5 for the accuracy tests.

(2) Based on the above framework, we further design a second-order Lagrangian DG scheme solving the compressible Euler equations in two-dimensional cylindrical coordinates. This DG scheme is designed to preserve one-dimensional spherical symmetry when computed on an equal-angle-zoned initial grid without losing the conservation of mass, momentum and total energy. In fact, it is very difficult to construct a high-order DG scheme that preserves both the spherical symmetry and conservation. Not only the vertex velocity but also all the variables that occur in the DG scheme should be calculated symmetrically. Unlike the finite volume scheme, we want to ensure the the symmetry property of linear polynomials for all the conserved variables, rather than just the symmetry of the cell averages. Specifically, this DG scheme is constructed in three steps. Firstly, we use a true volume scheme to calculate the cell averages of the conserved variables, which guarantees the conservation property. And then we use an area-weighted scheme to calculate the slopes of the polynomials, which is easier to preserve the symmetry property. Finally, the mass matrix and source terms appearing in these schemes must be treated carefully to ensure the symmetry of all the numerical solution polynomials without losing the original second-order accuracy. Moreover, we point out that the WENO limiter is accomplished in the cell's local $\xi$-$\eta$ coordinates as shown in Fig. 3.1, which does not affect the spherical symmetry-preserving property of the scheme, as explained in [13].

Several two-dimensional numerical examples in the Cartesian and cylindrical coordinates are given to demonstrate the good performance of the Lagrangian DG schemes in terms of accuracy, symmetry and non-oscillation.

The rest of this paper is organized as follows. In Section 2, we describe all the details of the second-order conservative Lagrangian DG scheme for the compressible Euler equations in two-dimensional Cartesian coordinates. In Section 3, we design a second-order Lagrangian DG scheme which preserves both the spherical symmetry and conservation properties in two-dimensional cylindrical coordinates. In Section 4, we prove the symmetry-preserving property of the scheme proposed in Section 3. The two-dimensional numerical experiments are given in Section 5. We will end in Section 6 with concluding remarks.

# 2 Second-order conservative Lagrangian discontinuous Galerkin scheme for the Euler equations in the Cartesian coordinates

## 2.1 The compressible Euler equations in two-dimensional Cartesian coordinates

The Euler equations for unsteady compressible flow in two-dimensional Cartesian coordinates can be expressed as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = 0,\ (x, y) \in \Omega(t),\ t \geq 0, \tag{2.1}$$

where $\mathbf{U}$ is the vector of the conserved variables, $\mathbf{F}$ is the flux vector and $\Omega(t)$ is the moving control volume. Here $\mathbf{U}$ and $\mathbf{F}$ are given by

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho \mathbf{u} \\ E \end{pmatrix}, \quad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho \mathbf{u} \\ \rho \mathbf{u}\mathbf{u}^T + p\mathbf{I} \\ (E + p)\mathbf{u} \end{pmatrix}, \tag{2.2}$$

where $\rho$ is density, $\mathbf{u} = (u, v)^T$ is velocity, $\mathbf{M} = \rho \mathbf{u}$ is momentum, $E$ is total energy, $p$ is pressure and $\mathbf{I}$ is the $2 \times 2$ identity matrix. The set of equations is completed by adding an equation of state (EOS) with the general form $p = p(\rho, e)$, where $e = \frac{E}{\rho} - \frac{1}{2}|\mathbf{u}|^2$ is the specific internal energy. Especially, if we consider the perfect gas, then the equation of state is given as

$$p = (\gamma - 1)\rho e = (\gamma - 1)(E - \frac{1}{2}\rho |\mathbf{u}|^2), \tag{2.3}$$

where $\gamma$ is the specific gas constant.

In this paper, we focus on the Lagrangian framework, where the control volume moves with the fluid velocity. Therefore, the kinematic equation can be expressed as

$$\frac{d\mathbf{x}}{dt} = \mathbf{u}, \quad \mathbf{x}(0) = \mathbf{x}_0, \tag{2.4}$$

where $\mathbf{x} = (x, y)^T$, and $\mathbf{x}_0$ represents the coordinates at $t = 0$.

## 2.2 The second-order conservative Lagrangian DG scheme

In this subsection, we will construct a second-order conservative cell-centered Lagrangian DG scheme, where we solve the conserved variables such as density, momentum and total energy directly. Before we do that, we first introduce the notations used throughout the paper. We discretize the two-dimensional moving spatial domain $\Omega(t)$ into $M \times N$ cells. $I_{ij}(t)$ is a cell with the general quadrilateral shape constructed by the four vertices $\{(x_{i-\frac{1}{2},j-\frac{1}{2}}, y_{i-\frac{1}{2},j-\frac{1}{2}}),\ (x_{i+\frac{1}{2},j-\frac{1}{2}}, y_{i+\frac{1}{2},j-\frac{1}{2}}),\ (x_{i+\frac{1}{2},j+\frac{1}{2}}, y_{i+\frac{1}{2},j+\frac{1}{2}}),\ (x_{i-\frac{1}{2},j+\frac{1}{2}}, y_{i-\frac{1}{2},j+\frac{1}{2}})\}$. The edges of the cell $I_{ij}$ is denoted as $\partial I_{ij}$. In the Cartesian coordinates, we use rectangular cells at the initial time unless otherwise stated. The finite element space is defined as

$$V_h^k(t) = \{\varphi(x, y, t) \in L^2(\Omega(t)) : \varphi(x, y, t)|_{I_{ij}(t)} \in P^k(I_{ij}(t)),\ 1 \leq i \leq M,\ 1 \leq j \leq N\}, \tag{2.5}$$

where $P^k(I_{ij}(t))$ denotes the set of polynomials of degree up to $k$ defined in the cell $I_{ij}$, and the polynomial $\varphi \in V_h^k(t)$ is time-space dependent. According to [8, 9], in multi-dimensions, the pure Lagrangian scheme on linear grid can achieve at most second order accuracy, so we set $k = 1$ in this paper.

By using the Reynold's transport theorem and divergence theorem, we can get the semi-discrete Lagrangian DG scheme for solving (2.1): find $\mathbf{U} \in \mathbf{V}_h^k(t)$ such that for all the test functions $\varphi(x, y, t) \in V_h^k(t)$ and all

$1 \le i \le M,\ 1 \le j \le N$, we have

$$\begin{aligned}\frac{d}{dt}\iint_{I_{ij}(t)} (\mathbf{U}\varphi(x,y,t))\,dxdy = &\iint_{I_{ij}(t)} \left(\mathbf{U}\left(\mathbf{u}\cdot\nabla\varphi + \frac{\partial\varphi}{\partial t}\right)\right) dxdy - \int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds \\ &+ \iint_{I_{ij}(t)} (\mathbf{G}\cdot\nabla\varphi)\,dxdy - \int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{G}}\cdot\mathbf{n}\varphi\right) ds\end{aligned} \tag{2.6}$$

where $\mathbf{G}(\mathbf{U}) = (0, p\mathbf{I}, p\mathbf{u})^T$, $u_n^g = \mathbf{u}_g \cdot \mathbf{n}$, $\mathbf{u}_g$ is the grid velocity, and $\mathbf{n} = (n_x, n_y)^T$ denotes the unit outward normal to $\partial I_{ij}$. $\widehat{\mathbf{G}}\cdot\mathbf{n} = (0, \widehat{p}\mathbf{n}, \widehat{p\mathbf{u}}\cdot\mathbf{n})^T$, $\widehat{\mathbf{U}} = (\widehat{\rho}, \widehat{\rho\mathbf{u}}, \widehat{E})^T$ and $\widehat{\mathbf{u}}\cdot\mathbf{n}$ are the numerical fluxes, which will be described in detail later. We notice that there is the flux term $\int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds$ in the above scheme. In the Lagrangian framework, there should be no mass exchange between neighboring cells. Therefore, we consider that the flux term $\int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds$ is only computed in the equations where the basis functions are not equal to 1, i.e., if $\varphi = 1$, we assume that $\int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\right) ds = 0$. By doing so, the local conservation of the cell mass is valid, and the DG scheme is limited to second-order accuracy. As we know, we cannot get higher than second order accuracy with the pure Lagrangian scheme on the straight-line grid anyway [8, 9]. Moreover, it should be noted that we cannot simply throw away the flux term $\int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds$. Although it has been done in some references, this treatment may result in a loss of second-order accuracy. We have tested the accuracy of the scheme with/without $\int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds$ in the numerical experiments. The results show that the accuracy is second-order for all the variables such as density, momentum, total energy, pressure and velocity by using the scheme (2.6) with the flux term $\int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds$ $(\varphi \neq 1)$. At the same time, we observe that the scheme without the term $\int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds$ may lose second-order accuracy of the conserved variables, see Tables 5.3 and 5.5. Therefore, we can conclude that this term $\int_{\partial I_{ij}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds$ $(\varphi \neq 1)$ is necessary for keeping the second-order accuracy of the DG scheme (2.6). See Section 5 for more details.

Next, we explain other details in the design of the DG scheme, including numerical fluxes, grid velocity, time derivatives of basis functions and numerical quadrature. For the numerical fluxes, we use the approximate Riemann solvers, which have smaller computational cost compared to the exact Riemann solvers. To be more specific, we choose the HLLC Riemann solver [12] in the Lagrangian description for the numerical flux $\widehat{\mathbf{G}}\cdot\mathbf{n} = (0, \widehat{p}\mathbf{n}, \widehat{p\mathbf{u}}\cdot\mathbf{n})^T$. We first define $S_*$ and $p_*$ as follows:

$$\begin{aligned} S_* &= \frac{\rho^{out}u_n^{out}(S_+ - u_n^{out}) - \rho^{int}u_n^{int}(S_- - u_n^{int}) + p^{int} - p^{out}}{\rho^{out}(S_+ - u_n^{out}) - \rho^{int}(S_- - u_n^{int})}, \\ p_* &= \rho^{int}(u_n^{int} - S_-)(u_n^{int} - S_*) + p^{int}, \end{aligned} \tag{2.7}$$

where $(\rho^{int}, \mathbf{u}^{int}, p^{int})^T$ and $(\rho^{out}, \mathbf{u}^{out}, p^{out})^T$ are the variables inside and outside the cell $I_{ij}$ (inside the neighboring cell) along the cell edges $\partial I_{ij}$, respectively. $u_n^{int/out} = \mathbf{u}^{int/out}\cdot\mathbf{n}$, $c = \sqrt{\frac{\gamma p}{\rho}}$ is the sound speed, and $\tilde{u}_n$ and $\tilde{c}$ are the Roe averages of $u_n$ and $c$ respectively. The acoustic wavespeeds $S_-$ and $S_+$ can be taken as follows,

$$S_- = \min\{u_n^{int} - c^{int},\ \tilde{u}_n - \tilde{c}\},\ S_+ = \max\{u_n^{out} + c^{out},\ \tilde{u}_n + \tilde{c}\}.$$

Thus we have

$$\widehat{p} = p_*,\ \widehat{\mathbf{u}}\cdot\mathbf{n} = S_*,\ \widehat{p\mathbf{u}}\cdot\mathbf{n} = p_*S_*. \tag{2.8}$$

For the other fluxes $\widehat{\mathbf{U}} = (\widehat{\rho}, \widehat{\rho\mathbf{u}}, \widehat{E})^T$, we choose the upwind form, which has the following expression

$$(\widehat{\rho}, \widehat{\rho\mathbf{u}}, \widehat{E}) = \begin{cases} (\rho^{int}, \rho^{int}\mathbf{u}^{int}, E^{int}), & \text{if } S_* - u_n^g \ge 0, \\ (\rho^{out}, \rho^{out}\mathbf{u}^{out}, E^{out}), & \text{if } S_* - u_n^g < 0. \end{cases} \tag{2.9}$$

For the grid velocity $\mathbf{u}_g = (u_g, v_g)^T$, we need to derive an expression on the cell edges. In this paper, we only consider the straight-edge grid, so the grid moves with the fluid velocity defined at the vertex. We assume that $(u_1, v_1)$ and $(u_2, v_2)$ are the two vertex velocities of the cell edge $m$, which are defined in detail in the next subsection. In order to ensure that the grids have straight line edges during the movement, we can obtain the following expression of the grid velocity on the edge $m$,

$$u_g = u_1\frac{x-x_2}{x_1-x_2} + u_2\frac{x-x_1}{x_2-x_1}, \quad v_g = v_1\frac{y-y_2}{y_1-y_2} + v_2\frac{y-y_1}{y_2-y_1}, \tag{2.10}$$

where the point $(x, y)$ is located on the cell edge $m$ from $(x_1, y_1)$ to $(x_2, y_2)$.

For $\frac{\partial\varphi}{\partial t}$ in the scheme (2.6), we obtain it by using the Reynold's transport theorem. We set $\varphi = \phi_q^{ij}$ $(q = 1, 2, 3)$ in scheme (2.6), where $\phi_1^{ij}$, $\phi_2^{ij}$, $\phi_3^{ij}$ are three basis functions of the $V_h^1(t)$ DG space in the cell $I_{ij}$. For simplicity, we choose $\phi_1^{ij} = 1$ and the other two basis functions are orthogonal to 1, which have the following expressions,

$$\phi_1^{ij}(x, y, t) = 1,\ \phi_2^{ij}(x, y, t) = x - x_c^{ij}(t),\ \phi_3^{ij}(x, y, t) = y - y_c^{ij}(t),\ (x, y) \in I_{ij}, \tag{2.11}$$

where $x_c^{ij}(t) = \frac{1}{S_{ij}(t)}\iint_{I_{ij}(t)} x dxdy$, $y_c^{ij}(t) = \frac{1}{S_{ij}(t)}\iint_{I_{ij}(t)} y dxdy$, and $S_{ij}$ represents the area of the cell $I_{ij}(t)$. Using the Reynold's transport theorem, we can get

$$\begin{aligned}
\frac{\partial\phi_1}{\partial t} &= 0,\\
\frac{\partial\phi_2}{\partial t} &= -\frac{1}{S(t)}\int_{\partial I(t)} (x - x_c)u_g^n ds,\\
\frac{\partial\phi_3}{\partial t} &= -\frac{1}{S(t)}\int_{\partial I(t)} (y - y_c)u_g^n ds,
\end{aligned} \tag{2.12}$$

where we omit the superscript '$ij$'.

In addition, the Gauss-Lobatto quadrature rule is used in this paper. We first define coordinate transformation that converts the quadrilateral cell $I_{ij}$ in the $x$-$y$ coordinates to the square $[-1/2, 1/2] \times [-1/2, 1/2]$ in the $\widetilde{x}$-$\widetilde{y}$ coordinates. The mapping relationship is as follows.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_1\widetilde{x}\widetilde{y} + a_2\widetilde{x} + a_3\widetilde{y} + a_4 \\ b_1\widetilde{x}\widetilde{y} + b_2\widetilde{x} + b_3\widetilde{y} + b_4 \end{pmatrix}, \tag{2.13}$$

where

$$\begin{aligned}
(a_1, b_1) &= \mathbf{x}_{i-\frac{1}{2},j-\frac{1}{2}} - \mathbf{x}_{i+\frac{1}{2},j-\frac{1}{2}} + \mathbf{x}_{i+\frac{1}{2},j+\frac{1}{2}} - \mathbf{x}_{i-\frac{1}{2},j+\frac{1}{2}},\\
(a_2, b_2) &= \frac{1}{2}(-\mathbf{x}_{i-\frac{1}{2},j-\frac{1}{2}} + \mathbf{x}_{i+\frac{1}{2},j-\frac{1}{2}} + \mathbf{x}_{i+\frac{1}{2},j+\frac{1}{2}} - \mathbf{x}_{i-\frac{1}{2},j+\frac{1}{2}}),\\
(a_3, b_3) &= \frac{1}{2}(-\mathbf{x}_{i-\frac{1}{2},j-\frac{1}{2}} - \mathbf{x}_{i+\frac{1}{2},j-\frac{1}{2}} + \mathbf{x}_{i+\frac{1}{2},j+\frac{1}{2}} + \mathbf{x}_{i-\frac{1}{2},j+\frac{1}{2}}),\\
(a_4, b_4) &= \frac{1}{4}(\mathbf{x}_{i-\frac{1}{2},j-\frac{1}{2}} + \mathbf{x}_{i+\frac{1}{2},j-\frac{1}{2}} + \mathbf{x}_{i+\frac{1}{2},j+\frac{1}{2}} + \mathbf{x}_{i-\frac{1}{2},j+\frac{1}{2}}).
\end{aligned}$$

And the determinant of Jacobian matrix is

$$|J(\widetilde{x}, \widetilde{y})| = (a_2b_1 - a_1b_2)\widetilde{x} + (a_1b_3 - a_3b_1)\widetilde{y} + a_2b_3 - a_3b_2. \tag{2.14}$$

We point out that the mapping (2.13) is only used to compute the integrals in the DG scheme (2.6) and is not used as a reference configuration of the flow. From the mapping relationship (2.13), we can find that the $a \times a$-point Gauss-Lobatto quadrature rule is exact for the integral of polynomials of degree $(a-2)$ in the $x$-$y$ coordinates. This is because a polynomial of degree $(a-2)$ in the $x$-$y$ coordinates becomes a polynomial of degree $(2a-4)$ in the $\widetilde{x}$-$\widetilde{y}$ coordinates, the Jacobian $|J(\widetilde{x}, \widetilde{y})|$ is a linear function, and the $a \times a$-point Gauss-Lobatto quadrature rule is exact for the integral of polynomials of degree up to $(2a-3)$ in the $\widetilde{x}$-$\widetilde{y}$ coordinates. We take

$a = 4$ for the second order scheme. Define the set of quadrature points for the square $[-1/2, 1/2] \times [-1/2, 1/2]$ to be $\{(\widetilde{x}_\alpha, \widetilde{y}_\beta),\ \alpha, \beta = 1, 2, 3, 4\}$, and its weights are $\omega_\alpha$ or $\omega_\beta$. Using (2.13), we can get the quadrature points $\{(x_{\alpha,\beta}, y_{\alpha,\beta}), \alpha, \beta = 1, 2, 3, 4\}$ for the cell $I_{ij}$. With the above definition, we can compute all the integrals in the scheme (2.6) numerically. For example, the integral for the mass matrix $\mathbf{A}$ is discretized by the following formula,

$$A_{pq} = \iint_{I_{ij}} \phi_p(x, y, t)\phi_q(x, y, t)dxdy = \sum_{\alpha=1}^{4}\sum_{\beta=1}^{4} [\omega_\alpha\omega_\beta\phi_p(x_{\alpha,\beta}, y_{\alpha,\beta}, t)\phi_q(x_{\alpha,\beta}, y_{\alpha,\beta}, t)|J(\widetilde{x}_\alpha, \widetilde{y}_\beta)|],$$

where $p, q = 1, 2, 3$. The line integral for the numerical fluxes is calculated by

$$\int_{\partial I_{ij}} \left(\widehat{\mathbf{W}} \cdot \mathbf{n}\phi(x, y, t)\right) ds = \sum_{m=1}^{4}\sum_{\alpha=1}^{4} \left[\omega_\alpha \left(\widehat{\mathbf{W}}(\mathbf{U}^{int}_{m,\alpha}, \mathbf{U}^{out}_{m,\alpha}) \cdot \mathbf{n}^m_{ij}\phi(x_\alpha, y_\alpha, t)\right) \Delta l^m_{ij}\right],$$

where $\widehat{\mathbf{W}} \cdot \mathbf{n} = \widehat{\mathbf{G}} \cdot \mathbf{n} + \widehat{\mathbf{U}}(\widehat{\mathbf{u}} \cdot \mathbf{n} - u^g_n)$, $\mathbf{U}^{int}_{m,\alpha}$ and $\mathbf{U}^{out}_{m,\alpha}$ are the vectors of the conserved variables inside and outside the cell $I_{ij}$ at the quadrature point $(x_\alpha, y_\alpha)$ along the the $m$-th edge, $\mathbf{n}^m_{ij}$ is the unit outward normal to the $m$-th edge, and $\Delta l^m_{ij}$ is the length of the $m$-th edge. Note that there are two main reasons why we use Gauss-Lobatto quadrature rule instead of the Gaussian quadrature rule. The first reason is that it is easier to implement the positivity-preserving technique when necessary, which can be seen in [12]. The second reason is to avoid the double calculation of the normal part of the vertex velocity, which has been calculated in numerical quadrature of the fluxes.

## 2.3 The determination of the vertex velocity

Now we introduce how to determine the vertex velocity, which is used in the formula (2.10) and the movement of the grid. In fact, the determination of the vertex velocity is similar to that in [8]. Briefly, we consider a vertex $(x_{i-\frac{1}{2}}, y_{j-\frac{1}{2}})$ shared by four edges which are given a serial number $m = 1, 2, 3, 4$. The tangential velocity at the vertex $(x_{i-\frac{1}{2}}, y_{j-\frac{1}{2}})$ along each edge $m$ is defined as the average of that on both sides, i.e.,

$$u^m_t = \frac{1}{2}(u^{m-}_t + u^{m+}_t),\ m = 1, 2, 3, 4,$$

where $u^{m\pm}_t = \mathbf{u}^{m\pm} \cdot \mathbf{t}^m$, $\mathbf{u}^{m\pm}$ is the left/right value of velocity at this vertex along the edge $m$, and $\mathbf{t}^m = (t^m_x, t^m_y)$ is the unit tangential of the edge $m$. For the normal velocity, we set $u^m_n = \widehat{\mathbf{u}} \cdot \mathbf{n}^m = S_*$, where $\mathbf{n}^m = (n^m_x, n^m_y)$ is the unit normal of the edge $m$. And then we convert $u^m_t$ and $u^m_n$ into the $x$-$y$ velocity components, which have the following form,

$$u^m_x = \frac{u^m_n t^m_y - u^m_t n^m_y}{n^m_x t^m_y - n^m_y t^m_x}, \quad u^m_y = \frac{-u^m_n t^m_x + u^m_t n^m_x}{n^m_x t^m_y - n^m_y t^m_x},\ m = 1, 2, 3, 4.$$

Finally, we can get the vertex velocity as follows,

$$u_{i-\frac{1}{2},j-\frac{1}{2}} = \frac{1}{4}\sum_{m=1}^{4} u^m_x, \quad v_{i-\frac{1}{2},j-\frac{1}{2}} = \frac{1}{4}\sum_{m=1}^{4} u^m_y. \tag{2.15}$$

## 2.4 Time discretization

In this paper, we use the second-order strong stability preserving Runge-Kutta time discretization [41]. In the Lagrangian simulation, the vertex velocity, the grid point position and the polynomials of the conserved variables need to be updated at each Runge-Kutta stage, that is, the SSP-RK method is applied to the time marching

for the semi-discrete scheme (2.6) and the grid point update formulation (2.4). Specifically, the second-order SSP-RK method has the following form.

Stage 1:

$$\begin{aligned} \mathbf{x}^{(1)}_{i-\frac{1}{2},j-\frac{1}{2}} &= \mathbf{x}^{n}_{i-\frac{1}{2},j-\frac{1}{2}} + \Delta t^n \mathbf{u}^{n}_{i-\frac{1}{2},j-\frac{1}{2}}, \\ \mathbf{A}^{(1)}\mathbf{U}^{(1)}_{c} &= \mathbf{A}^{n}\mathbf{U}^{n}_{c} + \Delta t^n \mathbf{L}(\mathbf{U}^n); \end{aligned} \tag{2.16}$$

Stage 2:

$$\begin{aligned} \mathbf{x}^{n+1}_{i-\frac{1}{2},j-\frac{1}{2}} &= \frac{1}{2}\mathbf{x}^{n}_{i-\frac{1}{2},j-\frac{1}{2}} + \frac{1}{2}\mathbf{x}^{(1)}_{i-\frac{1}{2},j-\frac{1}{2}} + \frac{1}{2}\Delta t^n \mathbf{u}^{(1)}_{i-\frac{1}{2},j-\frac{1}{2}}, \\ \mathbf{A}^{n+1}\mathbf{U}^{n+1}_{c} &= \frac{1}{2}\mathbf{A}^{n}\mathbf{U}^{n}_{c} + \frac{1}{2}\mathbf{A}^{(1)}\mathbf{U}^{(1)}_{c} + \frac{1}{2}\Delta t^n \mathbf{L}(\mathbf{U}^{(1)}), \end{aligned} \tag{2.17}$$

where $\mathbf{L}$ is the numerical spatial operator at the right hand of the scheme (2.6), $\mathbf{A}$ is the mass matrix, $\mathbf{U}_c$ represents the coefficients of the polynomials for any conserved variables on the cell $I_{ij}$. 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. And we omit the subscript '$ij$' in $\mathbf{A}$ and $\mathbf{U}_c$. The time step $\Delta t^n$ is calculated as follows,

$$\Delta t^n = \lambda \min_{i,j} \left( \Delta l_{ij} / \left( c^n_{ij} + |\mathbf{u}|^n_{ij} \right) \right), \tag{2.18}$$

where $\Delta l_{ij}$ is the shortest edge length of the cell $I_{ij}(t^n)$ and $\lambda$ is the Courant number.

Up to now, we have a fully discrete Lagrangian scheme with second-order accuracy in both space and time.

# 3 Second-order Lagrangian discontinuous Galerkin scheme for the Euler equations in the cylindrical coordinates

We have introduced the full algorithm of the Lagrangian DG scheme in Section 2. In this section, we develop the above algorithm to the two-dimensional cylindrical coordinates. We first construct two Lagrangian discontinuous Galerkin schemes for the axisymmetric compressible Euler equations: a true volume DG scheme and an area-weighted DG scheme. The aim is to design a Lagrangian DG scheme that preserves both the spherical symmetry and conservation properties by combining these two schemes.

## 3.1 The compressible Euler equations in two-dimensional cylindrical coordinates

The compressible Euler equations in two-dimensional cylindrical coordinates can be described as follows

$$\frac{\partial}{\partial t}\begin{pmatrix} \rho \\ \rho u_z \\ \rho u_r \\ E \end{pmatrix} + \frac{\partial}{\partial z}\begin{pmatrix} \rho u_z \\ \rho u_z u_z + p \\ \rho u_r u_z \\ (E+p)u_z \end{pmatrix} + \frac{1}{r}\frac{\partial}{\partial r}\begin{pmatrix} \rho u_r r \\ \rho u_z u_r r \\ (\rho u_r u_r + p)r \\ (E+p)u_r r \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ p/r \\ 0 \end{pmatrix}, \ (z,r) \in \Omega(t), \tag{3.1}$$

where $z$ and $r$ are the axial and radial directions respectively. We use the same notations as in the Cartesian coordinates. $\mathbf{u} = (u_z, u_r)^T$, $\mathbf{M} = \rho\mathbf{u} = (M_z, M_r)^T$, where $M_z, M_r, u_z, u_r$ are the momentum and velocity components in the $z$ and $r$ directions respectively. $\rho$ is density, $E$ is total energy, and $p$ is pressure given by the equation of state, such as the equation (2.3). Similarly, denote the vector of the conserved variables $\mathbf{U}$ and pressure flux function $\mathbf{G}$ as

$$\mathbf{U} = \begin{pmatrix} \rho \\ \mathbf{M} \\ E \end{pmatrix}, \quad \mathbf{G}(\mathbf{U}) = \begin{pmatrix} 0 \\ p\mathbf{I} \\ p\mathbf{u} \end{pmatrix}.$$

## 3.2 Basic notations

We introduce the notations used in the cylindrical coordinates. Similar to the notations in the Cartesian coordinates, the two-dimensional domain $\Omega(t)$ is discretized into $Q \times L$ computational cells. $I_{ql}$ is a quadrilateral cell with the four vertices $\{(z_{q-\frac{1}{2},l-\frac{1}{2}}, r_{q-\frac{1}{2},l-\frac{1}{2}}),\ (z_{q+\frac{1}{2},l-\frac{1}{2}}, r_{q+\frac{1}{2},l-\frac{1}{2}}), (z_{q+\frac{1}{2},l+\frac{1}{2}}, r_{q+\frac{1}{2},l+\frac{1}{2}}), (z_{q-\frac{1}{2},l+\frac{1}{2}}, r_{q-\frac{1}{2},l+\frac{1}{2}})\}$, $q = 1, \cdots, Q,\ l = 1. \cdots, L$. $S_{ql}$ denotes the area of the cell $I_{ql}$, and $V_{ql}$ denotes the volume of the cell obtained by rotating this cell around the $z$-axis, which can be formulated as $V_{ql} = \iint_{I_{ql}} r dr dz$ (without the $2\pi$ factor).

If we simulate a spherical symmetric problem, we hope to preserve the spherical symmetry property of the numerical solutions. Thus, we use the equal-angle-zoned grid at the initial time, see Fig. 3.1. $(\xi, \eta)$ is the cell's local polar coordinates used for the analysis of the symmetry-preserving property in the next section, where $\xi$ is the radial direction through the cell center and the origin, $\eta$ is the angular direction which is orthogonal to $\xi$ counterclockwise. Besides, we define $\Delta\theta$ as the angle between any two neighboring $l$ lines which is a constant for the equal-angle-zoned grid. Since the equal-angle-zoned cell is an equal-sided trapezoid (or isosceles triangle connecting the origin), the angles between $\xi$ and the two equal sides of the cell are $\frac{\Delta\theta}{2}$.

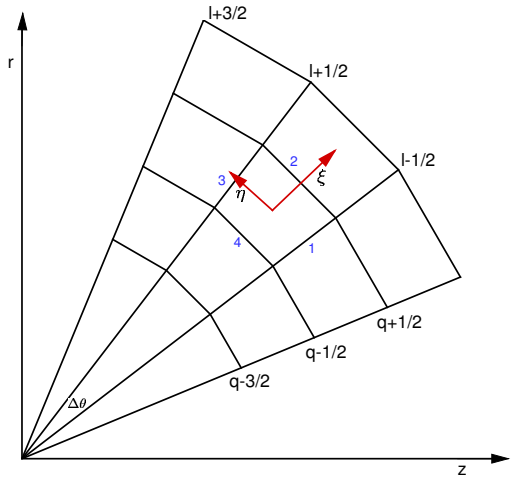



Figure 3.1: Equal-angle-zoned polar grid in the cylindrical geometry.

## 3.3 The true volume DG scheme

In this subsection, we design a Lagrangian DG scheme based on the true volume discretization in the cylindrical coordinates. It preserves the conservation of mass, momentum and total energy. Similar to the scheme (2.6), we can get the semi-discrete true volume Lagrangian DG scheme for solving (3.1): find $\mathbf{U} \in \mathbf{V}_h^k(t)$ such that for all the test functions $\varphi(z, r, t) \in V_h^k(t)$ and all $1 \le q \le Q,\ 1 \le l \le L$, we have

$$\begin{aligned}\frac{d}{dt}\iint_{I_{ql}(t)} (\mathbf{U}\varphi(z,r,t)r)\, dzdr = &\iint_{I_{ql}(t)} \left(\mathbf{U}\left(\mathbf{u}\cdot\nabla\varphi + \frac{\partial\varphi}{\partial t}\right) r\right) dzdr - \int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi r\right) ds \\ &+ \iint_{I_{ql}(t)} (\mathbf{G}\cdot\nabla\varphi r)\, dzdr - \int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{G}}\cdot\mathbf{n}\varphi r\right) ds + \iint_{I_{ql}(t)} \begin{pmatrix} 0 \\ 0 \\ p \\ 0 \end{pmatrix} \varphi dzdr,\end{aligned} \tag{3.2}$$

where $\mathbf{n} = (n_z, n_r)^T$ denotes the unit outward normal to $\partial I_{ql}$ in the $z$-$r$ coordinates, $\widehat{\mathbf{U}}$ is the vector of the numerical fluxes for the conserved variables $\mathbf{U}$, $\widehat{\mathbf{G}}\cdot\mathbf{n}$ is the vector of the numerical fluxes for pressure and

velocity functions $\mathbf{G}$, $u_n^g = \mathbf{u}_g \cdot \mathbf{n}$ and $\mathbf{u}_g = (u_g^z, u_g^r)^T$ is the grid velocity. There is the '$r$' factor in the integrals of the above scheme, because $2\pi \iint_\Omega r dz dr = \iiint_{\mathcal{W}} dx dy dz$ for the axisymmetric problems, where $\mathcal{W}$ is the three-dimensional domain obtained by rotating the two-dimensional area $\Omega$ about the $z$ axis. This means that the scheme is constructed on the genuine three-dimensional control volume. Moreover, the source term $\iint_{I_{ql}(t)} p\varphi dz dr$ exists in the formula of the momentum in the $r$ direction, which differs from that in the Cartesian coordinates.

The treatment of the flux term $\int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}} \cdot \mathbf{n} - u_n^g)\varphi r\right) ds$ is similar to that in the subsection 2.2. We only compute the flux term $\int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}} \cdot \mathbf{n} - u_n^g)\varphi r\right) ds$ in the equations using the second and third basis functions (the basis functions orthogonal to 1), i.e., if $\varphi = 1$, we assume that $\int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}} \cdot \mathbf{n} - u_n^g) r\right) ds = 0$. This treatment ensures that there is no mass exchange passing through the moving cell edges in (3.2), which is crucial for pure Lagrangian methods.

We use the same method as in the Cartesian coordinates to compute the vertex velocity, the numerical fluxes, and the time derivatives of basis functions. We also use the Gauss-Lobatto quadrature rule. Time discretization is chosen as the second-order SSP-RK method, see the subsection 2.4 for more details.

### 3.4 The area-weighted DG scheme

In the numerical simulation of the spherical symmetry problems in two-dimensional cylindrical coordinates, there is another Lagrangian scheme called the area-weighted scheme. In this subsection, we will construct an area-weighted DG scheme. We first rewrite the divergence term of the Euler system (3.1) in the following form

$$\frac{\partial \mathcal{H} u_z}{\partial z} + \frac{1}{r}\frac{\partial \mathcal{H} u_r r}{\partial r} = \frac{\partial \mathcal{H} u_z}{\partial z} + \frac{\partial \mathcal{H} u_r}{\partial r} + \mathcal{H}\frac{u_r}{r}, \tag{3.3}$$

where $\mathcal{H}$ represents any physical variables such as $\mathcal{H} = \rho,\ \mathbf{M},\ E$ or $p$. Using the equality (3.3), we can rewrite the Euler system (3.1) as follows

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho u_z\\ \rho u_r\\ E\end{pmatrix} + \frac{\partial}{\partial z}\begin{pmatrix}\rho u_z\\ \rho u_z u_z + p\\ \rho u_r u_z\\ (E+p)u_z\end{pmatrix} + \frac{\partial}{\partial r}\begin{pmatrix}\rho u_r\\ \rho u_z u_r\\ \rho u_r u_r + p\\ (E+p)u_r\end{pmatrix} + \frac{u_r}{r}\begin{pmatrix}\rho\\ \rho u_z\\ \rho u_r\\ E+p\end{pmatrix} = 0,\ (z,r) \in \Omega(t). \tag{3.4}$$

The semi-discrete area-weighted Lagrangian DG scheme for the system (3.4) is to find $\mathbf{U} \in \mathbf{V}_h^k(t)$ such that for all the test functions $\varphi(z,r,t) \in V_h^k(t)$ and all $1 \le q \le Q,\ 1 \le l \le L$,

$$\begin{aligned}\frac{d}{dt}\iint_{I_{ql}(t)} (\mathbf{U}\varphi(z,r,t))\, dz dr = &\iint_{I_{ql}(t)} \left(\mathbf{U}\left(\mathbf{u}\cdot\nabla\varphi + \frac{\partial\varphi}{\partial t}\right)\right) dz dr - \int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds\\ &+ \iint_{I_{ql}(t)} (\mathbf{G}\cdot\nabla\varphi)\, dz dr - \int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{G}}\cdot\mathbf{n}\varphi\right) ds - \iint_{I_{ql}(t)} \begin{pmatrix}\rho\\ M_z\\ M_r\\ E+p\end{pmatrix}\frac{u_r}{r}\varphi dz dr,\end{aligned} \tag{3.5}$$

where the symbols $\mathbf{n}$, $\widehat{\mathbf{U}}$, $\widehat{\mathbf{G}}\cdot\mathbf{n}$, $\widehat{\mathbf{u}}\cdot\mathbf{n}$ and $u_n^g$ are similar to those in the scheme (3.2). The vertex velocity, the numerical fluxes, the term $\int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\varphi\right) ds$ and the time derivatives of the basis functions are calculated in a similar way to the scheme (2.6). We will not repeat these details.

The true volume DG scheme (3.2) and the area-weighted DG scheme (3.5) are cell-centered Lagrangian schemes, where density, momentum and total energy are solved directly. Furthermore, the two schemes have their own features. On the one hand, the true volume DG scheme is based on true control volume discretization and preserves the conservation of mass, momentum and total energy, while the area-weighted DG scheme is

based on planar area discretization and is not conserved. On the other hand, the area-weighted DG scheme is easier to preserve one-dimensional spherical symmetry than the true volume DG scheme in the two dimensional cylindrical geometry, where we only need to carefully discretize the source terms in the scheme (3.5).

Next, we hope to combine the advantages of the above two schemes to construct a Lagrangian DG scheme that preserves both spherical symmetry and conservation properties, and maintains second-order accuracy.

### 3.5 The symmetry-preserving and conservative Lagrangian DG scheme

In the following, we will design a symmetry-preserving and conservative Lagrangian DG scheme in two-dimensional cylindrical coordinates. In fact, it is very difficult to design an area-weighted DG scheme or a true volume DG scheme that maintains both the symmetry and conservation properties. Here our basic idea is to calculate the cell averages using the true volume scheme (3.2), calculate the slopes of the numerical solution polynomials using the area-weighted scheme (3.5), and deal with the mass matrix and source terms carefully. The details are as follows.

Find $\mathbf{U} = (\rho, M_z, M_r, E)^T \in \mathbf{V}_h^1$ to satisfy the following semi-discrete scheme,

$$\frac{d}{dt}\iint_{I_{ql}(t)} \mathbf{U} r dz dr = -\int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{G}}\cdot\mathbf{n} r\right) ds + \iint_{I_{ql}(t)} \begin{pmatrix} 0 \\ 0 \\ p_a \\ 0 \end{pmatrix} dz dr, \tag{3.6}$$

$$\begin{aligned}\frac{d}{dt}\iint_{I_{ql}(t)} (\mathbf{U}\phi_i)\, dz dr =& \iint_{I_{ql}(t)} \left(\mathbf{U}\left(\mathbf{u}\cdot\nabla\phi_i + \frac{\partial \phi_i}{\partial t}\right)\right) dz dr - \int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\phi_i\right) ds \\ &+ \iint_{I_{ql}(t)} (\mathbf{G}\cdot\nabla\phi_i)\, dz dr - \int_{\partial I_{ql}(t)} \left(\widehat{\mathbf{G}}\cdot\mathbf{n}\phi_i\right) ds - \iint_{I_{ql}(t)} \begin{pmatrix} \rho \\ M_z \\ M_r \\ E+p \end{pmatrix} \frac{u_r^a}{r_a}\phi_i dz dr,\ i = 2, 3,\end{aligned} \tag{3.7}$$

where $\phi_2$ and $\phi_3$ are two basis functions orthogonal to $\phi_1 = 1$ in the DG space $V_h^1$, which have multiple expressions, such as

$$\begin{aligned}&\phi_1 = 1,\ \phi_2(z, r, t) = z - z_c^{ql}(t),\ \phi_3(z, t, t) = r - r_c^{ql}(t), \\ &z_c^{ql} = \frac{\iint_{I_{ql}} z dz dr}{S_{ql}},\ r_c^{ql} = \frac{\iint_{I_{ql}} r dz dr}{S_{ql}}.\end{aligned} \tag{3.8}$$

Note that (3.6) is from the true volume scheme, and (3.7) is from the area-weighted scheme. Thus, the details of the calculation in the scheme (3.6)-(3.7) are similar to the previous two schemes. And the time marching of this semi-discrete DG scheme is implemented by the second-order SSP-RK method.

In order to keep spherical symmetry, the mass matrix and source terms are determined as follows.

1. In (3.6), we set $\iint_{I_{ql}} \mathbf{U} r dz dr \approx r_c^{ql} \iint_{I_{ql}} \mathbf{U} dz dr$, i.e., for the mass matrix, we have

$$\iint_{I_{ql}} \begin{pmatrix} \phi_1^2 r & \phi_1\phi_2 r & \phi_1\phi_3 r \\ \phi_1\phi_2 & \phi_2^2 & \phi_2\phi_3 \\ \phi_1\phi_3 & \phi_2\phi_3 & \phi_3^2 \end{pmatrix} dz dr \approx \iint_{I_{ql}} \begin{pmatrix} \phi_1^2 r_c^{ql} & \phi_1\phi_2 r_c^{ql} & \phi_1\phi_3 r_c^{ql} \\ \phi_1\phi_2 & \phi_2^2 & \phi_2\phi_3 \\ \phi_1\phi_3 & \phi_2\phi_3 & \phi_3^2 \end{pmatrix} dz dr. \tag{3.9}$$

   Since the basis functions $\phi_2,\ \phi_3$ are orthogonal to $\phi_1 = 1$, we can get $\iint_{I_{ql}} \phi_1\phi_i dz dr = 0$ $(i = 2, 3)$, which means that the mass matrix is a symmetric matrix.

2. In (3.6), we set $\iint_{I_{ql}} p dz dr \approx p_a S_{ql}$, where $p_a$ is calculated as follows,

$$p_a = \frac{\sum_\alpha (\widehat{p}_1^\alpha \omega_\alpha \mu_1^\alpha + \widehat{p}_3^\alpha \omega_\alpha \mu_3^\alpha)}{\sum_\alpha (\omega_\alpha \mu_1^\alpha + \omega_\alpha \mu_3^\alpha)}, \tag{3.10}$$

where $\widehat{p}_1^{\alpha}$ and $\widehat{p}_3^{\alpha}$ are pressure numerical fluxes at the $\alpha$-th quadrature points $(z_1^{\alpha}, r_1^{\alpha})$ and $(z_3^{\alpha}, r_3^{\alpha})$ on the first and third edges of the cell $I_{ql}$ shown in Fig. 3.1, respectively. $\omega_{\alpha}$ is the quadrature weight. $\mu_m^{\alpha} = \sqrt{(z_m^{\alpha})^2 + (r_m^{\alpha})^2}$ represents the length of the line between the origin and the $\alpha$-th quadrature point $(z_m^{\alpha}, r_m^{\alpha})$ on the $m$-th edge of the cell $I_{ql}$.

3. In (3.7), we set $\iint_{I_{ql}} \widetilde{\mathbf{U}} \frac{u_r}{r} \phi_i dz dr \approx \frac{u_r^a}{r_a} \iint_{I_{ql}} \widetilde{\mathbf{U}} \phi_i dz dr$, where $\widetilde{\mathbf{U}} = (\rho, \mathbf{M}, E + p)^T$ and

$$\frac{u_r^a}{r_a} = \frac{u_r(z_c^{ql}, r_c^{ql})}{r_c^{ql}}. \tag{3.11}$$

The above three approximations do not affect conservation and second-order accuracy. In fact, for the first approximation, we can prove that when $r \neq 0$, $\frac{\iint_{I_{ql}} \mathbf{U} r dz dr}{\iint_{I_{ql}} r dz dr} = \frac{\iint_{I_{ql}} \mathbf{U} dz dr}{\iint_{I_{ql}} dz dr} + \mathcal{O}(h^2)$, where $h$ is the cell size. The discretization for the source term of pressure is similar to that in [13]. And we refer to [29] for the discretization of the term $\iint \widetilde{\mathbf{U}} \frac{u_r}{r} \phi_i dz dr$. Numerical experiments demonstrate the good performance of our designed Lagrangian DG scheme (3.6)-(3.7), such as symmetry, conservation and second-order accuracy. In the next section, we will give a theorem to show that this scheme can keep the spherical symmetry property on an equal-angle-zoned initial grid.

# 4 Analysis of the spherical symmetry property

## 4.1 Spherical symmetry condition in the cylindrical coordinates

In this subsection, we declare the definition of one-dimensional spherical symmetry in two-dimensional cylindrical coordinates for the linear polynomials of the numerical solutions such as density, momentum and total energy.

We use $\mathbf{U}^{ex} = (\rho^{ex}, \mathbf{M}^{ex}, E^{ex})^T$ and $\mathbf{U} = (\rho, \mathbf{M}, E)^T$ to represent exact and numerical solutions, respectively. Define $(\mu, \theta)$ as the local polar coordinates, where $\mu$ is the radial direction through any point $(z, r)$ and the origin, and $\theta$ is the angle between the $\mu$ direction and the $z$ axis. Thus we have

$$z = \mu \cos\theta, \ r = \mu \sin\theta. \tag{4.1}$$

Here the definition of $(\mu, \theta)$ is different from that of $(\xi, \eta)$ shown in Fig. 3.1, and we have the following transformation relationship

$$z = \xi \cos\theta_c^{ql} - \eta \sin\theta_c^{ql}, \ r = \xi \sin\theta_c^{ql} + \eta \cos\theta_c^{ql}, \tag{4.2}$$

and

$$\xi = \mu \cos(\theta - \theta_c^{ql}), \ \eta = \mu \sin(\theta - \theta_c^{ql}), \tag{4.3}$$

where $\theta_c^{ql}$ is the angle between the local $\xi$ direction and $z$ axis. The exact solutions $\mathbf{U}^{ex}$ are called spherically symmetric if they satisfy

$$\begin{aligned} &\rho^{ex}(\mu, \theta_1, t) = \rho^{ex}(\mu, \theta_2, t), \\ &M_{\mu}^{ex}(\mu, \theta_1, t) = M_{\mu}^{ex}(\mu, \theta_2, t), \\ &M_{\theta}^{ex}(\mu, \theta, t) = 0, \\ &E^{ex}(\mu, \theta_1, t) = E^{ex}(\mu, \theta_2, t), \end{aligned} \tag{4.4}$$

where $(\mu, \theta_1)$ and $(\mu, \theta_2)$ are any two points in the $\mu$-$\theta$ coordinates which have the same radial radius, $M_{\mu}^{ex} = M_z^{ex} \cos\theta + M_r^{ex} \sin\theta$ is the momentum component in the $\mu$ direction, $M_{\theta}^{ex} = -M_z^{ex} \sin\theta + M_r^{ex} \cos\theta$ is the momentum component in the direction orthogonal to $\mu$. If the conditions (4.4) is satisfied at the initial time, we can show that the conditions (4.4) is still valid at any time when there is no asymmetric external action,

which can be obtained by analyzing the Euler equations (3.1). We note that the symmetry conditions (4.4) for the exact solutions hold point by point, which implies derivative information, i.e.,

$$\frac{\partial}{\partial\mu}\begin{pmatrix}\rho^{ex}\\ M_\mu^{ex}\\ M_\theta^{ex}\\ E^{ex}\end{pmatrix}_{(\mu,\theta_1)} = \frac{\partial}{\partial\mu}\begin{pmatrix}\rho^{ex}\\ M_\mu^{ex}\\ 0\\ E^{ex}\end{pmatrix}_{(\mu,\theta_2)}, \quad \frac{\partial}{\partial\theta}\begin{pmatrix}\rho^{ex}\\ M_\mu^{ex}\\ M_\theta^{ex}\\ E^{ex}\end{pmatrix} = \mathbf{0}. \tag{4.5}$$

Next, we state the spherical symmetry conditions for the linear polynomials of the numerical solutions, which are based on the cell's local $\xi$-$\eta$ coordinates. For the sake of discussion, we choose the basis functions as

$$\phi_1 = 1,\ \phi_2(\xi,\eta,t) = \xi - \xi_c^{ql}(t),\ \phi_3(\xi,\eta,t) = \eta - \eta_c^{ql}(t), \tag{4.6}$$

where $\xi_c^{ql} = \frac{\iint_{I_{ql}}\xi dzdr}{S_{ql}}$ and $\eta_c^{ql} = \frac{\iint_{I_{ql}}\eta dzdr}{S_{ql}}$, so $\iint_I \phi_1\phi_2 dzdr = \iint_I \phi_1\phi_3 dzdr = 0$. In the equal-angle-zoned grid, $\xi_c^{ql}$ and $S_{ql}$ are independent of the index $l$ and $\eta_c^{ql} = 0$. Thus, we take $\xi_c^{ql} = \xi_c^q$, $S_{ql} = S_q$ and $\eta_c^{ql} = \eta_c^q = 0$. With the above notations, we can represent the numerical solutions in the cell $I_{ql}$ as follows,

$$\begin{pmatrix}\rho^{ql}\\ M_\xi^{ql}\\ M_\eta^{ql}\\ E^{ql}\end{pmatrix}(\xi,\eta) = \begin{pmatrix}\rho_1^{ql}\\ M_{\xi,1}^{ql}\\ M_{\eta,1}^{ql}\\ E_1^{ql}\end{pmatrix} + \begin{pmatrix}\rho_2^{ql}\\ M_{\xi,2}^{ql}\\ M_{\eta,2}^{ql}\\ E_2^{ql}\end{pmatrix}\phi_2(\xi,\eta) + \begin{pmatrix}\rho_3^{ql}\\ M_{\xi,3}^{ql}\\ M_{\eta,3}^{ql}\\ E_3^{ql}\end{pmatrix}\phi_3(\xi,\eta), \tag{4.7}$$

where $M_\xi^{ql}$ and $M_\eta^{ql}$ are momentum components in the $\xi$ and $\eta$ directions that satisfy $M_\xi^{ql} = M_z^{ql}\cos\theta_c^{ql} + M_r^{ql}\sin\theta_c^{ql}$ and $M_\eta^{ql} = -M_z^{ql}\sin\theta_c^{ql} + M_r^{ql}\cos\theta_c^{ql}$, $\mathbf{U}_j^{ql}(j = 1,2,3)$ are the coefficients of the corresponding polynomials. Here for simplicity and without any confusion, we omit the time variable $t$. To analyze the symmetry-preserving conditions, we need to re-decompose momentum into the $\mu$-$\theta$ coordinates, that is

$$\begin{aligned} M_\mu^{ql}(\xi,\eta) &= M_\xi^{ql}\cos(\theta-\theta_c^{ql}) + M_\eta^{ql}\sin(\theta-\theta_c^{ql}),\\ M_\theta^{ql}(\xi,\eta) &= -M_\xi^{ql}\sin(\theta-\theta_c^{ql}) + M_\eta^{ql}\cos(\theta-\theta_c^{ql}). \end{aligned} \tag{4.8}$$

It should be noted that we cannot use a formula to describe the spherical symmetry of the numerical solutions point by point, because the numerical solutions are linear polynomials. Therefore, we only consider the cell center $(\xi_c^{ql},\eta_c^{ql})$, and then we apply the conditions (4.4) and (4.5) to the numerical solutions $(\rho^{ql},\ M_\mu^{ql},\ M_\theta^{ql},\ E^{ql})$ at the cell center. We can get the following definition.

**Definition 4.1** *The numerical solutions defined in the formula* (4.7) *are called the one-dimensional spherically symmetric if they satisfy the following conditions,*

$$\begin{pmatrix}\rho_1^{ql}\\ M_{\xi,1}^{ql}\\ M_{\eta,1}^{ql}\\ E_1^{ql}\end{pmatrix} = \begin{pmatrix}\rho_1^{ql'}\\ M_{\xi,1}^{ql'}\\ 0\\ E_1^{ql'}\end{pmatrix}, \quad \begin{pmatrix}\rho_2^{ql}\\ M_{\xi,2}^{ql}\\ M_{\eta,2}^{ql}\\ E_2^{ql}\end{pmatrix} = \begin{pmatrix}\rho_2^{ql'}\\ M_{\xi,2}^{ql'}\\ 0\\ E_2^{ql'}\end{pmatrix}, \quad \begin{pmatrix}\rho_3^{ql}\\ M_{\xi,3}^{ql}\\ M_{\eta,3}^{ql}\\ E_3^{ql}\end{pmatrix} = \begin{pmatrix}0\\ 0\\ M_{\eta,3}^{ql'}\\ 0\end{pmatrix}, \tag{4.9}$$

*where* $1 \le q \le Q$ *and* $1 \le l,\ l' \le L$.

## 4.2 Proof of the spherical symmetry-preserving property

Next, we will prove our scheme (3.6)-(3.7) can preserve the spherical symmetry property if an equal-angle-zoned grid is used at the initial time. Notice that symmetry preservation for the Lagrangian solution includes two parts: the evolution of the conserved variables and the grid. The theorem is as follows.

**Theorem 4.1** *The scheme* (3.6)-(3.7) *on an equal-angle-zoned initial grid can preserve the one-dimensional spherical symmetry property. That is, if the initial solution has one-dimensional spherical symmetry, the numerical solution will keep this symmetry during the time evolution.*

**Proof** Since the SSP-RK method [41] is a convex combination of the Euler forward time discretization, we only need to prove the scheme (3.6)-(3.7) with Euler forward time stepping is symmetry-preserving. Without loss of the generality, we assume that the solutions of the scheme (3.6) -(3.7) have spherical symmetry property and the grid is a polar grid with equal angles at the $n$-th time step, and then we only need to prove that the numerical solution preserves the spherical symmetry and the grid is equal-angle-zoned at the $(n+1)$-th time step. For the convenience of notation, we omit the superscript '$n$' of the variables at the $n$-th time step. The proof is divided into two parts.

**1. The proof of symmetry preservation for the grid at the $(n+1)$-th step.**

We consider a vertex $(z_{q-\frac{1}{2},l-\frac{1}{2}}, r_{q-\frac{1}{2},l-\frac{1}{2}})$ shared by four edges '$L$', '$R$', '$B$', '$T$', which connects four cells $I_{q-1,l-1}$, $I_{q-1,l}$, $I_{q,l-1}$ and $I_{q,l}$. We define this vertex's local polar coordinates $(\widetilde{\xi},\widetilde{\eta})$ , where $\widetilde{\xi}$ is the radial direction through the vertex $(z_{q-\frac{1}{2},l-\frac{1}{2}}, r_{q-\frac{1}{2},l-\frac{1}{2}})$ and the origin, and $\widetilde{\eta}$ is orthogonal to $\widetilde{\xi}$ counterclockwise, see Fig. 4.1. Since the grid and the numerical solutions are symmetrical at the $n$-th step, we have

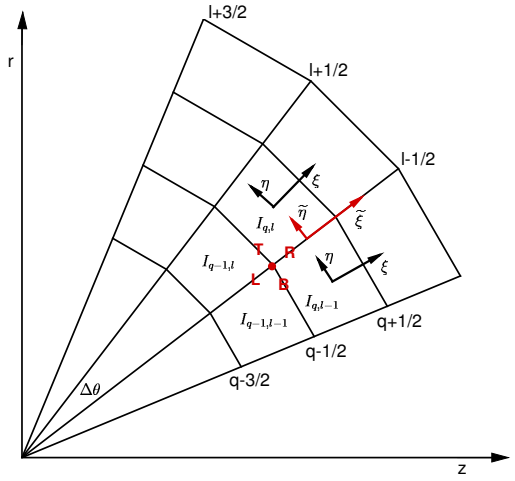



Figure 4.1: The local $\widetilde{\xi}$-$\widetilde{\eta}$ coordinates at the vertex $(z_{q-\frac{1}{2},l-\frac{1}{2}}, r_{q-\frac{1}{2},l-\frac{1}{2}})$.

$$\begin{pmatrix} \rho^{\kappa,l-1} \\ M_\xi^{\kappa,l-1} \\ M_\eta^{\kappa,l-1} \\ E^{\kappa,l-1} \end{pmatrix} (z_{q-\frac{1}{2},l-\frac{1}{2}}, r_{q-\frac{1}{2},l-\frac{1}{2}}) = \begin{pmatrix} \rho^{\kappa,l} \\ M_\xi^{\kappa,l} \\ -M_\eta^{\kappa,l} \\ E^{\kappa,l} \end{pmatrix} (z_{q-\frac{1}{2},l-\frac{1}{2}}, r_{q-\frac{1}{2},l-\frac{1}{2}}) \equiv \begin{pmatrix} \rho^{\kappa} \\ M_\xi^{\kappa} \\ M_\eta^{\kappa} \\ E^{\kappa} \end{pmatrix}, \ \kappa = q-1, \ q,$$

where we omit the superscript '$l-1$' and '$l$' for the variables which are independent of the $l$ index, $M_\xi^{\kappa,l-1}$ and $M_\eta^{\kappa,l-1}$ are the momentum components in the $I_{\kappa,l-1}$ cell's local $\xi$-$\eta$ coordinates, see Fig. 4.1. Here the numerical solutions represent the values of the corresponding variables at this vertex $(z_{q-\frac{1}{2},l-\frac{1}{2}}, r_{q-\frac{1}{2},l-\frac{1}{2}})$. We set the direction of each edge to be the increasing direction of the index $q$ or $l$, and we get

$$\begin{aligned}
&\rho^{L-} = \rho^{L+} = \rho^{B-} = \rho^{T-} = \rho^{q-1}, \quad \rho^{R-} = \rho^{R+} = \rho^{B+} = \rho^{T+} = \rho^{q}, \\
&E^{L-} = E^{L+} = E^{B-} = E^{T-} = E^{q-1}, \quad E^{R-} = E^{R+} = E^{B+} = E^{T+} = E^{q}, \\
&M_{\widetilde{\xi}}^{L+} = M_{\widetilde{\xi}}^{L-} = M_{\widetilde{\xi}}^{T-} = M_{\widetilde{\xi}}^{B-} = \cos(\Delta\theta/2) M_\xi^{q-1} + \sin(\Delta\theta/2) M_\eta^{q-1}, \\
&M_{\widetilde{\eta}}^{L+} = M_{\widetilde{\eta}}^{T-} = -M_{\widetilde{\eta}}^{L-} = -M_{\widetilde{\eta}}^{B-} = \sin(\Delta\theta/2) M_\xi^{q-1} - \cos(\Delta\theta/2) M_\eta^{q-1}, \\
&M_{\widetilde{\xi}}^{R+} = M_{\widetilde{\xi}}^{R-} = M_{\widetilde{\xi}}^{T+} = M_{\widetilde{\xi}}^{B+} = \cos(\Delta\theta/2) M_\xi^{q} + \sin(\Delta\theta/2) M_\eta^{q}, \\
&M_{\widetilde{\eta}}^{R+} = M_{\widetilde{\eta}}^{T+} = -M_{\widetilde{\eta}}^{R-} = -M_{\widetilde{\eta}}^{B+} = \sin(\Delta\theta/2) M_\xi^{q} - \cos(\Delta\theta/2) M_\eta^{q},
\end{aligned} \tag{4.10}$$

where $\mathbf{U}^{L+}$ and $\mathbf{U}^{L-}$ are the values of the corresponding variables at the vertex $(z_{q-\frac{1}{2},l-\frac{1}{2}}, r_{q-\frac{1}{2},l-\frac{1}{2}})$ along the two sides of the edge '$L$', $M_{\widetilde{\xi}}$ and $M_{\widetilde{\eta}}$ are the momentum components in this vertex's local $\widetilde{\xi}$-$\widetilde{\eta}$ coordinates. The unit normal and tangential vectors of the four edges in the local $\widetilde{\xi}$-$\widetilde{\eta}$ coordinates are represented as follows,

$$\begin{aligned}
&\mathbf{n}^L = \mathbf{n}^R = (0,1), \quad \mathbf{t}^L = \mathbf{t}^R = (1,0),\\
&\mathbf{n}^B = \left(\cos\frac{\Delta\theta}{2}, -\sin\frac{\Delta\theta}{2}\right), \quad \mathbf{t}^B = \left(\sin\frac{\Delta\theta}{2}, \cos\frac{\Delta\theta}{2}\right),\\
&\mathbf{n}^T = \left(\cos\frac{\Delta\theta}{2}, \sin\frac{\Delta\theta}{2}\right), \quad \mathbf{t}^T = \left(-\sin\frac{\Delta\theta}{2}, \cos\frac{\Delta\theta}{2}\right).
\end{aligned}$$

Let $M_n^{L\pm} = \mathbf{M}^{L\pm}\cdot\mathbf{n}^L$, $M_t^{L\pm} = \mathbf{M}^{L\pm}\cdot\mathbf{t}^L$ represent the normal and tangential momenta at the two sides of the edge '$L$', respectively. $M_n^{B\pm}$, $M_t^{B\pm}$, $M_n^{T\pm}$, $M_t^{T\pm}$, $M_n^{R\pm}$ and $M_t^{R\pm}$ are defined in the same way. Through the above relationship, we obtain

$$\begin{aligned}
&M_n^{L+} = -M_n^{L-}, \quad M_t^{L+} = M_t^{L-},\\
&M_n^{R+} = -M_n^{R-}, \quad M_t^{R+} = M_t^{R-},\\
&M_n^{B-} = M_n^{T-}, \quad M_t^{B-} = -M_t^{T-},\\
&M_n^{B+} = M_n^{T+}, \quad M_t^{B+} = -M_t^{T-}.
\end{aligned} \tag{4.11}$$

Using (4.10) and (4.11), we have

$$u_n^L = 0,\ u_n^R = 0,\ u_n^B = u_n^T, \tag{4.12}$$

where $u_n^m$ is the normal velocity calculated by the HLLC approximate Riemann solver at this vertex along the edge '$m$', $m = L, R, B, T$. The tangential velocities at each edge are as follows,

$$\begin{aligned}
u_t^L &= \frac{1}{2}(M_t^{L-}/\rho^{L-} + M_t^{L+}/\rho^{L+}),\\
u_t^R &= \frac{1}{2}(M_t^{R-}/\rho^{R-} + M_t^{R+}/\rho^{R+}),\\
u_t^B &= \frac{1}{2}(M_t^{B-}/\rho^{B-} + M_t^{B+}/\rho^{B+}),\\
u_t^T &= \frac{1}{2}(M_t^{T-}/\rho^{T-} + M_t^{T+}/\rho^{T+}) = -u_t^B.
\end{aligned} \tag{4.13}$$

Here $u_n^m$ and $u_t^m$ $(m = L, R, B, T)$ are independent of the $l$ index because all the variables in (4.10) and (4.11) are independent of the $l$ index. And then we convert $u_n^m$ and $u_t^m$ into the $\widetilde{\xi}$-$\widetilde{\eta}$ components, that is

$$\begin{aligned}
&(u_{\widetilde{\xi}}^L, u_{\widetilde{\eta}}^L) = (u_t^L,\ 0), \quad (u_{\widetilde{\xi}}^R, u_{\widetilde{\eta}}^R) = (u_t^R,\ 0),\\
&(u_{\widetilde{\xi}}^B, u_{\widetilde{\eta}}^B) = \left(\cos(\Delta\theta/2)u_n^B + \sin(\Delta\theta/2)u_t^B,\ -\sin(\Delta\theta/2)u_n^B + \cos(\Delta\theta/2)u_t^B\right),\\
&(u_{\widetilde{\xi}}^T, u_{\widetilde{\eta}}^T) = \left(\cos(\Delta\theta/2)u_n^T - \sin(\Delta\theta/2)u_t^T,\ \sin(\Delta\theta/2)u_n^T + \cos(\Delta\theta/2)u_t^T\right).
\end{aligned}$$

Therefore, we can get the the following vertex velocity,

$$\begin{aligned}
(u_{\widetilde{\xi}})_{q-\frac{1}{2},l-\frac{1}{2}} &= \frac{1}{4}(u_{\widetilde{\xi}}^L + u_{\widetilde{\xi}}^R + u_{\widetilde{\xi}}^B + u_{\widetilde{\xi}}^T) = \frac{1}{4}\left(u_t^L + u_t^R + 2\cos(\Delta\theta/2)u_n^B + 2\sin(\Delta\theta/2)u_t^B\right),\\
(u_{\widetilde{\eta}})_{q-\frac{1}{2},l-\frac{1}{2}} &= \frac{1}{4}(u_{\widetilde{\eta}}^L + u_{\widetilde{\eta}}^R + u_{\widetilde{\eta}}^B + u_{\widetilde{\eta}}^T) = 0.
\end{aligned} \tag{4.14}$$

Since $(u_{\widetilde{\eta}})_{q-\frac{1}{2},l-\frac{1}{2}} = 0$ and $(u_{\widetilde{\xi}})_{q-\frac{1}{2},l-\frac{1}{2}}$ depends only on the $q$ index, we can conclude that the vertex velocity is symmetric. Thus we have finished the proof of the symmetry for the grid at the $(n+1)$-th step.

**2. The proof of symmetry preservation for the conserved variables at the $(n+1)$-th step.**

In order to save space, we only give the proof of symmetry preservation for the momentum. The symmetry proof for density and total energy is similar. We consider the cell $I_{ql}$ with four edges $m = 1, 2, 3, 4$. We use the

$4 \times 4$-point Gauss-Lobatto quadrature rule. Since the proof is based on the $I_{ql}$ cell's local $\xi$-$\eta$ coordinates, we rewrite the momentum schemes in (3.6)-(3.7) as follows,

$$\frac{d}{dt}\iint_{I_{ql}(t)} \begin{pmatrix} M_\xi \\ M_\eta \end{pmatrix} r dz dr = -\int_{\partial I_{ql}(t)} \widehat{p} \begin{pmatrix} n_\xi \\ n_\eta \end{pmatrix} r ds + p_a S_{ql} \begin{pmatrix} \sin\theta_c^{ql} \\ \cos\theta_c^{ql} \end{pmatrix}, \tag{4.15}$$

$$\begin{aligned} \frac{d}{dt}\iint_{I_{ql}(t)} \begin{pmatrix} M_\xi \\ M_\eta \end{pmatrix} \phi_i dz dr = & \iint_{I_{ql}(t)} \begin{pmatrix} M_\xi \\ M_\eta \end{pmatrix} \left( \mathbf{u} \cdot \begin{pmatrix} \frac{\partial \phi_i}{\partial \xi} \\ \frac{\partial \phi_i}{\partial \eta} \end{pmatrix} + \frac{\partial \phi_i}{\partial t} \right) dz dr - \int_{\partial I_{ql}(t)} \begin{pmatrix} \widehat{M_\xi} \\ \widehat{M_\eta} \end{pmatrix} (\widehat{\mathbf{u}} \cdot \mathbf{n} - u_n^g) \phi_i ds \\ & + \iint_{I_{ql}(t)} p \begin{pmatrix} \frac{\partial \phi_i}{\partial \xi} \\ \frac{\partial \phi_i}{\partial \eta} \end{pmatrix} dz dr - \int_{\partial I_{ql}(t)} \widehat{p} \begin{pmatrix} n_\xi \\ n_\eta \end{pmatrix} \phi_i ds - \iint_{I_{ql}(t)} \begin{pmatrix} M_\xi \\ M_\eta \end{pmatrix} \phi_i dz dr \frac{u_r^a}{r_a}, \ i = 2, 3, \end{aligned} \tag{4.16}$$

where $(n_\xi, n_\eta)^T$ denotes the unit outward normal to $\partial I_{ql}$ in the $\xi$-$\eta$ coordinates, $\mathbf{u} = (u_\xi, u_\eta)^T$, and $\phi_i$ is chosen as that in (4.6). Next, we first analyze the symmetry of the cell averages in (4.15), and then prove the symmetry of the slopes of the momentum polynomials in (4.16).

**(1) The proof of symmetry preservation for the cell averages in** (4.15)**.**

Let $(z_m^\alpha, r_m^\alpha)$ and $(\xi_m^\alpha, \eta_m^\alpha)$ $(m = 1, \cdots, 4,\ \alpha = 1, \cdots, 4)$ be the $\alpha$-th Gauss-Lobatto quadrature point on the $m$-th edge of the $I_{ql}$ cell in the $z$-$r$ and the local $\xi$-$\eta$ coordinates, respectively, see Fig. 4.2. $\mu_m^\alpha = \sqrt{(z_m^\alpha)^2 + (r_m^\alpha)^2}$

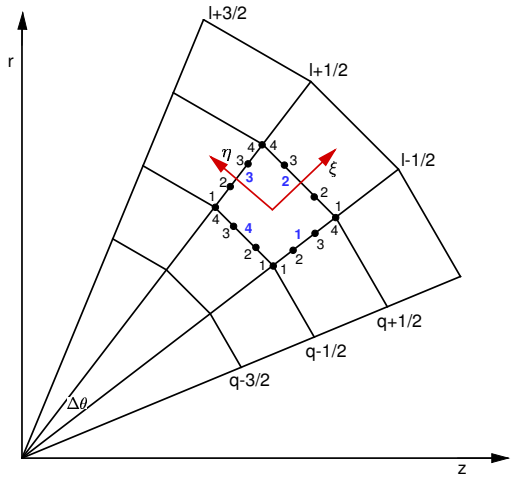



Figure 4.2: The Gauss-Lobatto quadrature points on the four edges of the $I_{ql}$ cell.

represents radial length of the corresponding quadrature point. The length of each cell edge is defined as $\Delta l^m,\ m = 1, \cdots, 4$. The radial length at the middle point of each edge is denoted as $\mu_e^m,\ m = 1, \cdots, 4$. Since the grid is symmetrical, $\Delta l^m$ and $\mu_e^m$ are independent of the $l$ index. Thus we denote them as

$$\begin{aligned} &\mu_e^1 = \mu_e^3 \equiv \mu_q,\ \mu_e^2 \equiv \mu_{q+\frac{1}{2}},\ \mu_e^4 \equiv \mu_{q-\frac{1}{2}}, \\ &\Delta l^1 = \Delta l^3 \equiv \Delta l_q,\ \Delta l^2 \equiv \Delta l_{q+\frac{1}{2}},\ \Delta l^4 \equiv \Delta l_{q-\frac{1}{2}}. \end{aligned}$$

The unit tangential vector $\mathbf{t}^m = (t_\xi^m, t_\eta^m)$ and unit outward normal vector $\mathbf{n}^m = (n_\xi^m, n_\eta^m)$ of the $m$-th edge in the local $\xi$-$\eta$ coordinates are represented as follows,

$$\begin{aligned} &\mathbf{n}^1 = \left( -\sin\frac{\Delta\theta}{2}, -\cos\frac{\Delta\theta}{2} \right), \quad \mathbf{t}^1 = \left( \cos\frac{\Delta\theta}{2}, -\sin\frac{\Delta\theta}{2} \right), \\ &\mathbf{n}^3 = \left( -\sin\frac{\Delta\theta}{2}, \cos\frac{\Delta\theta}{2} \right), \quad \mathbf{t}^3 = \left( \cos\frac{\Delta\theta}{2}, \sin\frac{\Delta\theta}{2} \right), \\ &\mathbf{n}^2 = (1, 0), \quad \mathbf{t}^2 = (0, 1), \quad \mathbf{n}^4 = (-1, 0), \quad \mathbf{t}^4 = (0, 1). \end{aligned}$$

We define $(\rho_m^{\alpha,int},(M_n)_m^{\alpha,int},(M_t)_m^{\alpha,int},E_m^{\alpha,int})$ and $(\rho_m^{\alpha,out},(M_n)_m^{\alpha,out},(M_t)_m^{\alpha,out},E_m^{\alpha,out})$ to represent the variables inside and outside the cell at the $\alpha$-th quadrature point on the $m$-th edge, where $M_n$ and $M_t$ are the normal and tangential momenta, respectively. According to the definition of one-dimensional spherical symmetry, we have

$$
\begin{aligned}
&(\mathbf{U}_t)_1^{\alpha,int}=(\mathbf{U}_t)_1^{\alpha,out}=(\mathbf{U}_t)_3^{\alpha,int}=(\mathbf{U}_t)_3^{\alpha,out},\ \alpha=1,\cdots,4,\\
&(M_n)_1^{\alpha,int}=-(M_n)_1^{\alpha,out}=(M_n)_3^{\alpha,int}=-(M_n)_3^{\alpha,out},\ \alpha=1,\cdots,4,\\
&(\mathbf{U}_n)_\delta^{1,\nu}=(\mathbf{U}_n)_\delta^{2,\nu}=(\mathbf{U}_n)_\delta^{3,\nu}=(\mathbf{U}_n)_\delta^{4,\nu},\ \delta=2,4,\ \nu=\text{'int' or 'out'},\\
&(M_t)_\delta^{1,\nu}=-(M_t)_\delta^{4,\nu},\ (M_t)_\delta^{2,\nu}=-(M_t)_\delta^{3,\nu},\ \delta=2,4,\ \nu=\text{'int' or 'out'},
\end{aligned}
$$

where $\mathbf{U}_t=(\rho,M_t,E)$, $\mathbf{U}_n=(\rho,M_n,E)$. By the above relationship, we get

$$
\widehat{p}_1^\alpha=\widehat{p}_3^\alpha,\quad \widehat{p}_2^1=\widehat{p}_2^4,\quad \widehat{p}_2^2=\widehat{p}_2^3,\quad \widehat{p}_4^1=\widehat{p}_4^4,\quad \widehat{p}_4^2=\widehat{p}_4^3,\ \alpha=1,\cdots,4, \tag{4.17}
$$

where $\widehat{p}_m^\alpha$ is the numerical flux of pressure computed by (2.7) at the $\alpha$-th quadrature point on the $m$-th edge, which is independent of the $l$ index. Moreover, using the symmetry property of grid at the $(n+1)$-th step, the area and volume of the cell $I_{ql}$ at the $(n+1)$-th step can be denoted as follows

$$
S_{ql}^{n+1}=S_q^{n+1}=\mu_q^{n+1}\Delta l_q^{n+1}\sin(\Delta\theta),\ V_{ql}^{n+1}=\frac{2}{3}\sin(\theta_c^{ql})\tan(\Delta\theta/2)\left((\mu_{q+\frac{1}{2}}^{n+1})^3-(\mu_{q-\frac{1}{2}}^{n+1})^3\right). \tag{4.18}
$$

Thus, the full discretization of the scheme (4.15) is as follows,

$$
\begin{pmatrix}(\overline{M}_\xi^{ql})^{n+1}\\(\overline{M}_\eta^{ql})^{n+1}\end{pmatrix}=\begin{pmatrix}\overline{M}_\xi^{ql}\\\overline{M}_\eta^{ql}\end{pmatrix}\frac{V_{ql}}{V_{ql}^{n+1}}+\frac{\Delta t^n}{V_{ql}^{n+1}}\begin{pmatrix}-\sum_{m=1}^4\sum_{\alpha=1}^4(\omega_\alpha\widehat{p}_m^\alpha r_m^\alpha n_\xi^m\Delta l^m)+p_a\sin(\theta_c^{ql})S_q\\-\sum_{m=1}^4\sum_{\alpha=1}^4(\omega_\alpha\widehat{p}_m^\alpha r_m^\alpha n_\eta^m\Delta l^m)+p_a\cos(\theta_c^{ql})S_q\end{pmatrix}, \tag{4.19}
$$

where the Euler forward time discretization is considered, $m$ represents the cell edge, $\alpha$ represents the quadrature point, and $\omega_\alpha$ is the quadrature weight. $\overline{M}_\xi^{ql}$ and $\overline{M}_\eta^{ql}$ are the cell averages of $M_\xi$ and $M_\eta$, respectively, which are also the first coefficients of the corresponding polynomials in (4.7). Using (4.17), (3.10), $\mu_1^\alpha=\mu_3^\alpha$ and

$$
\begin{aligned}
&r_1^\alpha=\mu_1^\alpha\sin\left(\theta_c^{ql}-\frac{\Delta\theta}{2}\right),\quad r_3^\alpha=\mu_3^\alpha\sin\left(\theta_c^{ql}+\frac{\Delta\theta}{2}\right),\\
&r_2^1+r_2^4=r_2^2+r_2^3=2\mu_{q+\frac{1}{2}}\sin(\theta_c^{ql}),\quad r_4^1+r_4^4=r_4^2+r_4^3=2\mu_{q-\frac{1}{2}}\sin(\theta_c^{ql}),
\end{aligned}
$$

we have

$$
\begin{aligned}
&\sum_{m=1}^4\sum_{\alpha=1}^4(\omega_\alpha\widehat{p}_m^\alpha r_m^\alpha n_\xi^m\Delta l^m)\\
=&-\sin(\Delta\theta/2)\Delta l_q\sum_{\alpha=1}^4(\widehat{p}_1^\alpha(r_1^\alpha+r_3^\alpha)\omega_\alpha)+\Delta l_{q+\frac{1}{2}}\left(\widehat{p}_2^1\omega_1(r_2^1+r_2^4)+\widehat{p}_2^2\omega_2(r_2^2+r_2^3)\right)\\
&-\Delta l_{q-\frac{1}{2}}\left(\widehat{p}_4^1\omega_1(r_4^1+r_4^4)+\widehat{p}_4^2\omega_2(r_4^2+r_4^3)\right)\\
=&\left[-\Delta l_q\sin(\Delta\theta)\sum_{\alpha=1}^4(\widehat{p}_1^\alpha\mu_1^\alpha\omega_\alpha)+2\Delta l_{q+\frac{1}{2}}\mu_{q+\frac{1}{2}}(\widehat{p}_2^1\omega_1+\widehat{p}_2^2\omega_2)-2\Delta l_{q-\frac{1}{2}}\mu_{q-\frac{1}{2}}(\widehat{p}_4^1\omega_1+\widehat{p}_4^2\omega_2)\right]\sin(\theta_c^{ql}),
\end{aligned} \tag{4.20}
$$

and

$$
\begin{aligned}
&-\sum_{m=1}^4\sum_{\alpha=1}^4(\omega_\alpha\widehat{p}_m^\alpha r_m^\alpha n_\eta^m\Delta l^m)+p_a\cos\theta_c^{ql}S_q\\
=&-\cos(\Delta\theta/2)\Delta l_q\sum_{\alpha=1}^4(\widehat{p}_1^\alpha(r_3^\alpha-r_1^\alpha)\omega_\alpha)+\frac{\sum_\alpha(\widehat{p}_1^\alpha\omega_\alpha\mu_1^\alpha+\widehat{p}_3^\alpha\omega_\alpha\mu_3^\alpha)}{\sum_\alpha(\omega_\alpha\mu_1^\alpha+\omega_\alpha\mu_3^\alpha)}\cos(\theta_c^{ql})\mu_q\Delta l_q\sin(\Delta\theta)\\
=&-\sin(\Delta\theta)\cos(\theta_c^{ql})\Delta l_q\sum_{\alpha=1}^4(\widehat{p}_1^\alpha\mu_1^\alpha\omega_\alpha)+\sum_{\alpha=1}^4(\widehat{p}_1^\alpha\omega_\alpha\mu_1^\alpha)\cos(\theta_c^{ql})\Delta l_q\sin(\Delta\theta)\\
=&0.
\end{aligned} \tag{4.21}
$$

By (4.18), (4.20) and (3.10), we know that $\frac{1}{V_{ql}^{n+1}}\left(-\sum_{m=1}^{4}\sum_{\alpha=1}^{4}(\omega_\alpha \widehat{p}_m^\alpha r_m^\alpha n_\xi^m \Delta l^m) + p_a \sin(\theta_c^{ql}) S_q\right)$ is independent of the $l$ index, so we simplify it to

$$\frac{1}{V_{ql}^{n+1}}\left(-\sum_{m=1}^{4}\sum_{\alpha=1}^{4}(\omega_\alpha \widehat{p}_m^\alpha r_m^\alpha n_\xi^m \Delta l^m) + p_a \sin(\theta_c^{ql}) S_q\right) \equiv \overline{\mathrm{RHS}}_\xi^q. \tag{4.22}$$

According to Definition 4.1, we get $\overline{M}_\xi^{ql} = \overline{M}_\xi^q,\ \overline{M}_\eta^{ql} = 0$. Substituting (4.18), (4.21) and (4.22) into (4.19), we obtain

$$\begin{pmatrix}(\overline{M}_\xi^{ql})^{n+1}\\(\overline{M}_\eta^{ql})^{n+1}\end{pmatrix} = \begin{pmatrix}\overline{M}_\xi^q\\0\end{pmatrix}\frac{\mu_{q+\frac{1}{2}}^3 - \mu_{q-\frac{1}{2}}^3}{\left(\mu_{q+\frac{1}{2}}^{n+1}\right)^3 - \left(\mu_{q-\frac{1}{2}}^{n+1}\right)^3} + \Delta t^n \begin{pmatrix}\overline{\mathrm{RHS}}_\xi^q\\0\end{pmatrix}. \tag{4.23}$$

By now, we can conclude that the cell averages of momentum at the $(n+1)$-th step are symmetric, i.e., $(\overline{M}_\xi^{ql})^{n+1}$ is independent of the $l$ index and $(\overline{M}_\eta^{ql})^{n+1} = 0$.

**(2) The proof of symmetry preservation for the scheme** (4.16)**.**

We first analyze the momentum component $M_\eta^{ql}$. Define the mass matrix of the scheme (4.16) as follows,

$$\mathcal{M}^{ql}(t) = \iint_{I_{ql}(t)} \begin{pmatrix}(\xi-\xi_c^q)^2 & (\xi-\xi_c^q)(\eta-\eta_c^q)\\(\xi-\xi_c^q)(\eta-\eta_c^q) & (\eta-\eta_c^q)^2\end{pmatrix} dzdr = \begin{pmatrix}\mathcal{M}_{11}^{ql}(t) & 0\\0 & \mathcal{M}_{22}^{ql}(t)\end{pmatrix},$$

where $\mathcal{M}_{11}^{ql}$ and $\mathcal{M}_{22}^{ql}$ are independent of the $l$ index. So we denote $\mathcal{M}^{ql}$ as

$$\mathcal{M}^{ql}(t) \equiv \mathcal{M}^q(t) \equiv \begin{pmatrix}\mathcal{M}_{11}^{q}(t) & 0\\0 & \mathcal{M}_{22}^{q}(t)\end{pmatrix}.$$

The fully discrete scheme of $M_\eta^{ql}$ in (4.16) is as follows,

$$\begin{pmatrix}(M_{\eta,2}^{ql})^{n+1}\\(M_{\eta,3}^{ql})^{n+1}\end{pmatrix} = ((\mathcal{M}^q)^{n+1})^{-1}\mathcal{M}^q \begin{pmatrix}M_{\eta,2}^{ql}\\M_{\eta,3}^{ql}\end{pmatrix} + ((\mathcal{M}^q)^{n+1})^{-1}\Delta t^n \begin{pmatrix}b_{\eta,2}^{ql}\\b_{\eta,3}^{ql}\end{pmatrix}, \tag{4.24}$$

where $M_{\eta,2}^{ql}$ and $M_{\eta,3}^{ql}$ are the last two coefficients of the $M_\eta^{ql}$ polynomial in (4.7). According to Definition 4.1, we get $M_{\eta,2}^{ql} = 0,\ M_{\eta,3}^{ql} = M_{\eta,3}^q$. Here $b_{\eta,2}^{ql}$ and $b_{\eta,3}^{ql}$ are the right hand terms of $M_\eta$ in (4.16), which are defined in detail as follows,

$$\begin{aligned}
b_{\eta,2}^{ql} =& \iint_{I_{ql}} M_\eta u_\xi dzdr - \frac{d\xi_c^q}{dt}\iint_{I_{ql}} M_\eta dzdr - \int_{\partial I_{ql}} \widehat{M_\eta}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)(\xi-\xi_c^q)ds\\
&- \int_{\partial I_{ql}} \widehat{p} n_\eta (\xi-\xi_c^q)ds - \frac{u_r^a}{r_a}\iint_{I_{ql}} M_\eta(\xi-\xi_c^q)dzdr,\\
b_{\eta,3}^{ql} =& \iint_{I_{ql}} M_\eta u_\eta dzdr - \int_{\partial I_{ql}} \widehat{M_\eta}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)(\eta-\eta_c^q)ds\\
&+ \iint_{I_{ql}} p dzdr - \int_{\partial I_{ql}} \widehat{p} n_\eta (\eta-\eta_c^q)ds - \frac{u_r^a}{r_a}\iint_{I_{ql}} M_\eta(\eta-\eta_c^q)dzdr.
\end{aligned} \tag{4.25}$$

Next, we analyze each term in (4.25). Let $\mathcal{H}_{\alpha,\beta},\ (\alpha,\beta = 1,\cdots,4)$ be the value of variable $\mathcal{H}$ at the $(\alpha,\beta)$ Gauss-Lobatto quadrature point in the $I_{ql}$ cell, see Fig. 4.3. Since the grid and the numerical solution are symmetrical at the $n$-th step, we have

$$\mathcal{F}_{\alpha,\beta} = \mathcal{F}_{\alpha',\beta},\ \mathcal{G}_{1,\beta} = -\mathcal{G}_{4,\beta},\ \mathcal{G}_{2,\beta} = -\mathcal{G}_{3,\beta},\ \alpha,\alpha',\beta = 1,\cdots,4, \tag{4.26}$$

where $\mathcal{F} = \rho, E, M_\xi, u_\xi, p, \xi, J$ and $\mathcal{G} = M_\eta, u_\eta, \eta$. Here $J$ is Jacobian of coordinate transformation defined in (2.14). Since the cell is an equal-sided trapezoid, we have $J_{\alpha,\beta} = J_{\alpha',\beta}$. And these variables are independent of

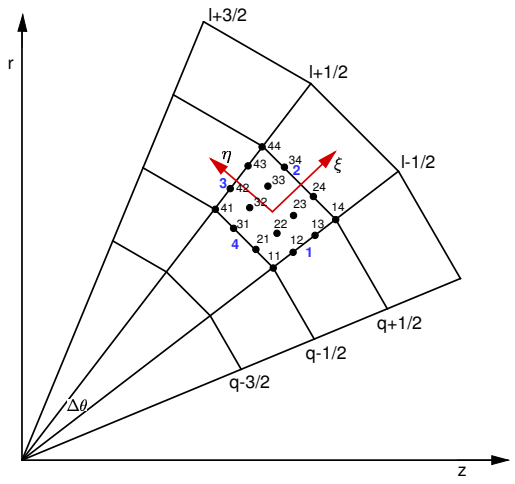



Figure 4.3: The Gauss-Lobatto quadrature points in the cell $I_{ql}$.

the $l$ index. By the above relationship, we get

$$\iint_{I_{ql}} \mathcal{F}\mathcal{G}dzdr = \sum_{\alpha,\beta=1}^{4} (\omega_{\alpha,\beta}\mathcal{F}_{\alpha,\beta}\mathcal{G}_{\alpha,\beta}J_{\alpha,\beta}) = 0,$$

$$\iint_{I_{ql}} \mathcal{F}\mathcal{F}dzdr = \sum_{\alpha,\beta=1}^{4} \left(\omega_{\alpha,\beta}\mathcal{F}^2_{\alpha,\beta}J_{\alpha,\beta}\right)$$

$$\iint_{I_{ql}} \mathcal{G}\mathcal{G}dzdr = \sum_{\alpha,\beta=1}^{4} \left(\omega_{\alpha,\beta}\mathcal{G}^2_{\alpha,\beta}J_{\alpha,\beta}\right) = 2\sum_{\beta=1}^{4} \left(\omega_{1,\beta}\mathcal{G}^2_{1,\beta}J_{1,\beta} + \omega_{2,\beta}\mathcal{G}^2_{2,\beta}J_{2,\beta}\right).$$

Specifically, we have

$$\iint_{I_{ql}} M_\eta u_\xi dzdr = 0, \iint_{I_{ql}} M_\eta dzdr = 0, \iint_{I_{ql}} M_\eta(\xi - \xi^q_c)dzdr = 0. \tag{4.27}$$

$\iint_{I_{ql}} M_\eta u_\eta dzdr$, $\iint_{I_{ql}} p dzdr$ and $\iint_{I_{ql}} M_\eta(\eta - \eta^q_c)dzdr$ are independent of the $l$ index. Using (3.11), we obtain

$$\frac{u^a_r}{r_a} = \frac{\overline{u}^{ql}_\xi \sin(\theta^{ql}_c)}{\xi^{ql}_c \sin(\theta^{ql}_c)} = \frac{\overline{u}^q_\xi}{\xi^q_c}. \tag{4.28}$$

Thus, $u^a_r/r_a$ depends only on the $q$ index. For the normal grid velocity $u^g_n$, by formulas (4.14) and (2.10), we have

$$u^g_n|_{1,3} = 0, \quad u^g_n|_2 = C^{q+\frac{1}{2}}, \quad u^g_n|_4 = C^{q-\frac{1}{2}},$$

where $u^g_n|_m$ represents the normal component of the grid velocity on the $m$-th edge of the cell, $C^{q+\frac{1}{2}}$ and $C^{q-\frac{1}{2}}$ are two constants that depend only on the $q$ index. Moreover, similar to the analysis of (4.12) and (4.17), it is easy to obtain

$$(\widehat{\mathbf{u}}\cdot\mathbf{n})^\alpha_1 = (\widehat{\mathbf{u}}\cdot\mathbf{n})^\alpha_3 = 0, \quad (\widehat{\mathbf{u}}\cdot\mathbf{n})^1_\delta = (\widehat{\mathbf{u}}\cdot\mathbf{n})^4_\delta, \quad (\widehat{\mathbf{u}}\cdot\mathbf{n})^2_\delta = (\widehat{\mathbf{u}}\cdot\mathbf{n})^3_\delta, \ \alpha = 1,\cdots,4, \ \delta = 2,4,$$

$$(\widehat{M}_\eta)^1_\delta = -(\widehat{M}_\eta)^4_\delta, \quad (\widehat{M}_\eta)^2_\delta = -(\widehat{M}_\eta)^3_\delta, \ \delta = 2,4,$$

where $\widehat{U}^\alpha_m$ is the corresponding numerical flux at the $\alpha$-th quadrature point on the $m$-th edge of the cell $I_{ql}$.

Thus, the integrals along the cell edges are expressed as follows,

$$\begin{aligned}
&\int_{\partial I_{ql}} \widehat{M_\eta}(\widehat{\mathbf{u}}\cdot\mathbf{n}-u_n^g)(\xi-\xi_c^q)ds\\
=&\Delta l_{q+\frac{1}{2}}(\mu_{q+\frac{1}{2}}-\xi_c^q)\left[\omega_1\left((\widehat{M_\eta})_2^1+(\widehat{M_\eta})_2^4\right)\left((\widehat{\mathbf{u}}\cdot\mathbf{n})_2^1-u_n^g|_2\right)+\omega_2\left((\widehat{M_\eta})_2^2+(\widehat{M_\eta})_2^3\right)\left((\widehat{\mathbf{u}}\cdot\mathbf{n})_2^2-u_n^g|_2\right)\right]\\
&+\Delta l_{q-\frac{1}{2}}(\mu_{q-\frac{1}{2}}-\xi_c^q)\left[\omega_1\left((\widehat{M_\eta})_4^1+(\widehat{M_\eta})_4^4\right)\left((\widehat{\mathbf{u}}\cdot\mathbf{n})_4^1-u_n^g|_4\right)+\omega_2\left((\widehat{M_\eta})_4^2+(\widehat{M_\eta})_4^3\right)\left((\widehat{\mathbf{u}}\cdot\mathbf{n})_4^2-u_n^g|_4\right)\right]\\
=&0,
\end{aligned} \tag{4.29}$$

$$\int_{\partial I_{ql}}(\widehat{p}n_\eta(\xi-\xi_c^q))ds=-\Delta l_q\cos\left(\frac{\Delta\theta}{2}\right)\sum_{\alpha=1}^{4}(\omega_\alpha\widehat{p}_1^\alpha(\xi_1^\alpha-\xi_c^q))+\Delta l_q\cos\left(\frac{\Delta\theta}{2}\right)\sum_{\alpha=1}^{4}(\omega_\alpha\widehat{p}_3^\alpha(\xi_3^\alpha-\xi_c^q))=0. \tag{4.30}$$

Analogously, $\int_{\partial I_{ql}}\widehat{M_\eta}(\widehat{\mathbf{u}}\cdot\mathbf{n}-u_n^g)(\eta-\eta_c^q)ds$ and $\int_{\partial I_{ql}}(\widehat{p}n_\eta(\eta-\eta_c^q))ds$ are independent of the $l$ index. By the above analyses, we obtain that $b_{\eta,2}^{ql}=0$ and $b_{\eta,3}^{ql}=b_{\eta,3}^q$ is independent of the $l$ index in (4.25). Thus, the scheme (4.24) is further written as

$$\begin{pmatrix}(M_{\eta,2}^{ql})^{n+1}\\(M_{\eta,3}^{ql})^{n+1}\end{pmatrix}=\begin{pmatrix}\mathcal{M}_{11}^q/(\mathcal{M}_{11}^q)^{n+1} & 0\\ 0 & \mathcal{M}_{22}^q/(\mathcal{M}_{22}^q)^{n+1}\end{pmatrix}\begin{pmatrix}0\\M_{\eta,3}^q\end{pmatrix}+\Delta t^n\begin{pmatrix}0\\b_{\eta,3}^q/(\mathcal{M}_{22}^q)^{n+1}\end{pmatrix}. \tag{4.31}$$

From the scheme (4.31), we can find that the last two coefficients of the $M_\eta^{ql}$ polynomial satisfy the symmetry condition at the $(n+1)$-th step, i.e., $(M_{\eta,2}^{ql})^{n+1}=0$, and $(M_{\eta,3}^{ql})^{n+1}$ is independent of the $l$ index. Similarly, we can analyze that the last two coefficients of the $M_\xi^{ql}$ polynomial satisfy the definition of the spherical symmetry. The proof of the theorem is completed. □

**Remark 4.1** *The Lagrangian DG scheme in the z-r coordinates is equivalent to that in the local $\xi$-$\eta$ coordinates. In the analysis of spherical symmetry preservation, we use the basis functions* (4.6) *and the expressions* (4.7) *of the conserved variables $(\rho, M_\xi, M_\eta, E)$ in the cell's local $\xi$-$\eta$ coordinates. In practice, we use the basis functions* (3.8) *and solve the conserved variables $(\rho, M_z, M_r, E)$ in the DG scheme* (3.6)-(3.7) *in the z-r coordinates.*

**Remark 4.2** *In this paper, we use the HLLC numerical flux and the upwind numerical flux. The other numerical fluxes such as Dukowicz or Godunov fluxes can also be selected, and the proof of symmetry preservation will also work well. Here we do not intend to compare the features and numerical performance between different numerical fluxes. Please refer to [8] for a more in-depth discussion of these numerical fluxes.*

# 5 Numerical tests

## 5.1 Accuracy tests

In this subsection, we test the accuracy of the Lagrangian DG schemes for the two-dimensional compressible Euler equations. The first three examples are computed in the Cartesian coordinates using the DG scheme (2.6), focusing on verifying the importance of the term $\int_{\partial I(t)}\left(\widehat{\mathbf{U}}(\mathbf{u}-\mathbf{u}_g)\cdot\mathbf{n}\phi_j\right)ds$ $(j=2,3)$ for maintaining second-order accuracy. The last example is computed in the cylindrical coordinates using the symmetry-preserving and conservative DG scheme (3.6)-(3.7) to verify that the design of this DG scheme in two-dimensional cylindrical geometry does not change the original second-order accuracy. The Courant number $\lambda=0.4$ is applied in these examples.

**Example 5.1**

In the Cartesian coordinates, let us first consider an example where the velocity is constant, which can be found in [21]. We set the initial condition is as follows:

$$\rho_0(x,y) = 1 + 0.5\sin(x+y),\ (u_0,v_0) = (1,1),\ p_0 = 1,\ (x,y) \in [0,2\pi]^2.$$

The exact solutions are

$$\rho(x,y,t) = 1 + 0.5\sin(x+y-2t),\ \mathbf{u}(x,y,t) = (1,1),\ p(x,y,t) = 1,$$

with $\gamma = 1.4$. The periodic boundary conditions are used.

The computational domain is divided into uniform squares with mesh size $h = \frac{2\pi}{N}$ at $t = 0$. We take final time $t = 0.1$. The numerical errors of the Lagrangian DG scheme (2.6) containing the term $\int_{\partial I(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\phi_j\right) ds$ $(j = 2, 3)$ are shown in Table 5.1, where we can see the second-order accuracy of density. Here we only show the error table of density. In fact, the error tables of momentum and total energy are the same as that of density, because the exact solutions for momentum and total energy are similar to the density, and the errors of pressure and velocity are close to machine zero.

If we test the scheme (2.6) without the term $\int_{\partial I(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\phi_j\right) ds$ $(j = 2, 3)$, the numerical convergence rates do not change. It shows that in the simple example where the velocity is constant, the DG scheme (2.6) without this term may maintain second-order accuracy. However, this conclusion cannot be generalized to the more complicated examples.

Table 5.1: Errors of the scheme (2.6) for density $\rho$ in Example 5.1.

| N(=M) | $L_2$ error | Order($L_2$) | $L_1$ error | Order($L_1$) | $L_\infty$ error | Order($L_\infty$) |
|---|---|---|---|---|---|---|
| 20 | 3.4271E-3 | - | 2.4601E-3 | - | 1.4087E-2 | - |
| 40 | 8.5931E-4 | 2.00 | 6.1788E-4 | 1.99 | 3.5401E-3 | 1.99 |
| 80 | 2.1499E-4 | 2.00 | 1.5465E-4 | 2.00 | 8.8617E-4 | 2.00 |
| 160 | 5.3756E-5 | 2.00 | 3.8673E-5 | 2.00 | 2.2161E-4 | 2.00 |

**Example 5.2**

The example is an isentropic problem with smooth solutions [25]. The initial condition is:

$$\rho_0(x,y) = \frac{1 + 0.2\sin(\frac{x+y}{2})}{\sqrt{6}},\ u_0(x,y) = v_0(x,y) = \sqrt{\frac{\gamma}{2}}\rho_0,\ p_0(x,y) = \rho_0^\gamma,\ (x,y) \in [0,4\pi]^2,$$

with the periodic boundary condition. If we take $\gamma = 3$, we can verify that $\sqrt{6}\rho(x,y,t)$ is the exact solution of the two-dimensional Burgers' equation:

$$\mu_t + \left(\frac{\mu^2}{2}\right)_x + \left(\frac{\mu^2}{2}\right)_y = 0, \quad \mu_0(x,y) = 1 + 0.2\sin\left(\frac{x+y}{2}\right),$$

and

$$u(x,y,t) = v(x,y,t) = \sqrt{\frac{\gamma}{2}}\rho(x,y,t),\ p(x,y,t) = \rho(x,y,t)^\gamma.$$

The errors and numerical rates of convergence at $t = 0.3$ are summarized in Table 5.2, where the Lagrangian DG scheme (2.6) containing $\int_{\partial I(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\phi_j\right) ds$ $(j = 2, 3)$ is used. One can observe that the numerical rates for all the variables are second-order.

For comparison, we test the DG scheme without the term $\int_{\partial I(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\phi_j\right) ds$ $(j = 2, 3)$, which is used in some references. The numerical results are shown in Table 5.3, where we can see that although the

accuracy for pressure and velocity is second-order, the DG scheme without this term cannot achieve the second-order accuracy of the conserved variables such as density, momentum and total energy. Therefore, this term $\int_{\partial I(t)} \left( \widehat{\mathbf{U}}(\widehat{\mathbf{u}} \cdot \mathbf{n} - u_n^g)\phi_j \right) ds$ $(j = 2, 3)$ is essential for maintaining the second-order accuracy of the conserved variables.

Table 5.2: Errors of the scheme (2.6) with the term $\int_{\partial I(t)} \left( \widehat{\mathbf{U}}(\widehat{\mathbf{u}} \cdot \mathbf{n} - u_n^g)\phi_j \right) ds$ $(j = 2, 3)$ for Example 5.2.

| | N(=M) | $L_2$ error | Order($L_2$) | $L_1$ error | Order($L_1$) | $L_\infty$ error | Order($L_\infty$) |
|---|---|---|---|---|---|---|---|
| density | 20 | 6.3343E-4 | - | 4.5095E-4 | - | 3.0721E-3 | - |
| | 40 | 1.6286E-4 | 1.96 | 1.1422E-4 | 1.98 | 8.0958E-4 | 1.92 |
| | 80 | 4.1050E-5 | 1.99 | 2.8664E-5 | 1.99 | 2.0611E-4 | 1.97 |
| | 160 | 1.0289E-5 | 2.00 | 7.1734E-6 | 2.00 | 5.1881E-5 | 1.99 |
| energy | 20 | 6.7266E-4 | - | 4.6420E-4 | - | 4.2210E-3 | - |
| | 40 | 1.7167E-4 | 1.97 | 1.1616E-4 | 2.00 | 1.1053E-3 | 1.93 |
| | 80 | 4.2988E-5 | 2.00 | 2.8843E-5 | 2.01 | 2.8064E-4 | 1.98 |
| | 160 | 1.0733E-5 | 2.00 | 7.1822E-6 | 2.01 | 7.0365E-5 | 2.00 |
| momentum | 20 | 9.2059E-4 | - | 6.7070E-4 | - | 5.0251E-3 | - |
| | 40 | 2.3772E-4 | 1.95 | 1.7340E-4 | 1.95 | 1.3125E-3 | 1.94 |
| | 80 | 5.9815E-5 | 1.99 | 4.3714E-5 | 1.99 | 3.3311E-4 | 1.98 |
| | 160 | 1.4961E-5 | 2.00 | 1.0938E-5 | 2.00 | 8.3708E-5 | 1.99 |
| pressure | 20 | 3.6204E-4 | - | 2.6384E-4 | - | 2.0501E-3 | - |
| | 40 | 9.1724E-5 | 1.98 | 6.5348E-5 | 2.01 | 5.4701E-4 | 1.91 |
| | 80 | 2.2957E-5 | 2.00 | 1.6165E-5 | 2.02 | 1.3935E-4 | 1.97 |
| | 160 | 5.7382E-6 | 2.00 | 4.0202E-6 | 2.01 | 3.5081E-5 | 1.99 |
| velocity | 20 | 1.2683E-3 | - | 9.6882E-4 | - | 5.3977E-3 | - |
| | 40 | 3.3400E-4 | 1.92 | 2.5803E-4 | 1.91 | 1.3724E-3 | 1.98 |
| | 80 | 8.4498E-5 | 1.98 | 6.5417E-5 | 1.98 | 3.5264E-4 | 1.96 |
| | 160 | 2.1157E-5 | 2.00 | 1.6387E-5 | 2.00 | 8.9245E-5 | 1.98 |

Table 5.3: Errors of the scheme (2.6) without the term $\int_{\partial I(t)} \left( \widehat{\mathbf{U}}(\widehat{\mathbf{u}} \cdot \mathbf{n} - u_n^g)\phi_j \right) ds$ $(j = 2, 3)$ for Example 5.2.

| | N(=M) | $L_2$ error | Order($L_2$) | $L_1$ error | Order($L_1$) | $L_\infty$ error | Order($L_\infty$) |
|---|---|---|---|---|---|---|---|
| density | 20 | 7.2387E-4 | - | 5.2452E-4 | - | 3.5514E-3 | - |
| | 40 | 2.4817E-4 | 1.54 | 1.8622E-4 | 1.49 | 1.1039E-3 | 1.69 |
| | 80 | 9.7553E-5 | 1.35 | 7.4449E-5 | 1.32 | 3.7777E-4 | 1.55 |
| | 160 | 4.3194E-5 | 1.18 | 3.2767E-5 | 1.18 | 1.4939E-4 | 1.34 |
| energy | 20 | 6.8450E-4 | - | 4.6609E-4 | - | 4.3997E-3 | - |
| | 40 | 1.8829E-4 | 1.86 | 1.2743E-4 | 1.87 | 1.2299E-3 | 1.84 |
| | 80 | 5.3852E-5 | 1.81 | 3.7582E-5 | 1.76 | 3.4847E-4 | 1.82 |
| | 160 | 1.7216E-5 | 1.65 | 1.2502E-5 | 1.59 | 1.0491E-4 | 1.73 |
| momentum | 20 | 9.6597E-4 | - | 7.1278E-4 | - | 5.3497E-3 | - |
| | 40 | 2.8686E-4 | 1.75 | 2.1535E-4 | 1.73 | 1.5837E-3 | 1.76 |
| | 80 | 9.3255E-5 | 1.62 | 7.1284E-5 | 1.60 | 4.8409E-4 | 1.71 |
| | 160 | 3.5299E-5 | 1.40 | 2.7241E-5 | 1.39 | 1.6269E-4 | 1.57 |
| pressure | 20 | 3.6305E-4 | - | 2.5892E-4 | - | 2.1125E-3 | - |
| | 40 | 9.4237E-5 | 1.95 | 6.5639E-5 | 1.98 | 5.7127E-4 | 1.89 |
| | 80 | 2.3813E-5 | 1.98 | 1.6391E-5 | 2.00 | 1.4632E-4 | 1.97 |
| | 160 | 5.9640E-6 | 2.00 | 4.0891E-6 | 2.00 | 3.6827E-5 | 1.99 |
| velocity | 20 | 1.2703E-3 | - | 9.7402E-4 | - | 5.3031E-3 | - |
| | 40 | 3.3391E-4 | 1.93 | 2.5815E-4 | 1.92 | 1.3777E-3 | 1.94 |
| | 80 | 8.4382E-5 | 1.98 | 6.5299E-5 | 1.98 | 3.5432E-4 | 1.96 |
| | 160 | 2.1120E-5 | 2.00 | 1.6339E-5 | 2.00 | 8.9342E-5 | 1.99 |

**Example 5.3 Taylor-Green vortex problem**

The two-dimensional Taylor-Green vortex problem [4] is a benchmark test with analytical solutions. The initial condition is prescribed by

$$\rho_0 = 1,\ u_0(x,y) = \sin(\pi x)\cos(\pi y),\ v_0(x,y) = -\cos(\pi x)\sin(\pi y),\ p_0(x,y) = \frac{1}{4}(\cos(2\pi x) + \cos(2\pi y)) + 1.$$

The working fluid is ideal gas with $\gamma = \frac{5}{3}$. In the compressible inviscid case, a source term is required in the energy equation to make the flow steady state:

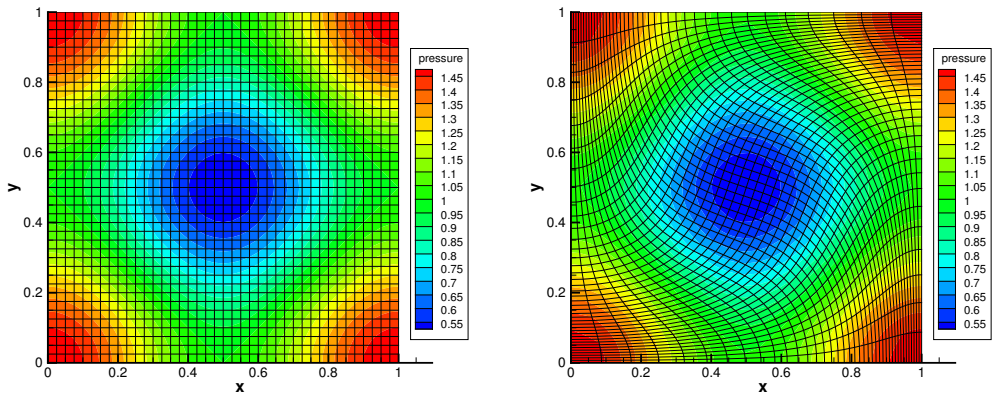
$$S = \frac{\pi}{4(\gamma-1)}(\cos(3\pi x)\cos(\pi y) - \cos(\pi x)\cos(3\pi y)).$$

The computational domain is a square $[0,1]\times[0,1]$. We set the final time to $t = 0.4$. The final mesh and the pressure results with $N = M = 40$ are shown in Fig. 5.1. The numerical errors at $t = 0.4$ are listed in Table 5.4. From the table, we can find the second-order accuracy for all the variables.

Moreover, if we test the DG scheme (2.6) without $\int_{\partial I(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\phi_j\right) ds$ $(j = 2, 3)$, the numerical errors are shown in Table 5.5, where we can find that the numerical rates of convergence for the conserved variables are only first-order. Therefore, we can not simply throw the term $\int_{\partial I(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\phi_j\right) ds$ away.

Table 5.4: Errors of the scheme (2.6) with the term $\int_{\partial I(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}}\cdot\mathbf{n} - u_n^g)\phi_j\right) ds$ $(j = 2, 3)$ for Example 5.3.

| | N(=M) | $L_2$ error | Order($L_2$) | $L_1$ error | Order($L_1$) | $L_\infty$ error | Order($L_\infty$) |
|---|---|---|---|---|---|---|---|
| density | 20 | 2.1267E-3 | - | 1.6721E-3 | - | 6.5795E-3 | - |
| | 40 | 5.0453E-4 | 2.08 | 3.8701E-4 | 2.11 | 1.7409E-3 | 1.92 |
| | 80 | 1.2411E-4 | 2.02 | 9.4499E-5 | 2.03 | 4.6227E-4 | 1.91 |
| | 160 | 3.1072E-5 | 2.00 | 2.3649E-5 | 2.00 | 1.1993E-4 | 1.95 |
| energy | 20 | 4.3517E-3 | - | 3.2839E-3 | - | 2.0350E-2 | - |
| | 40 | 1.1348E-3 | 1.94 | 8.4958E-4 | 1.95 | 5.4646E-3 | 1.90 |
| | 80 | 2.9833E-4 | 1.93 | 2.2105E-4 | 1.94 | 1.4044E-3 | 1.96 |
| | 160 | 7.7635E-5 | 1.94 | 5.7020E-5 | 1.95 | 3.5544E-4 | 1.98 |
| momentum | 20 | 7.2207E-3 | - | 5.8144E-3 | - | 2.5404E-2 | - |
| | 40 | 1.9873E-3 | 1.86 | 1.5857E-3 | 1.87 | 7.1634E-3 | 1.83 |
| | 80 | 5.2970E-4 | 1.91 | 4.1931E-4 | 1.92 | 1.9207E-3 | 1.90 |
| | 160 | 1.3780E-4 | 1.94 | 1.0839E-4 | 1.95 | 5.0126E-4 | 1.94 |
| pressure | 20 | 2.9431E-3 | - | 2.1581E-3 | - | 1.2579E-2 | - |
| | 40 | 7.8672E-4 | 1.90 | 5.4266E-4 | 1.99 | 3.6839E-3 | 1.77 |
| | 80 | 2.0911E-4 | 1.91 | 1.3823E-4 | 1.97 | 1.0339E-3 | 1.83 |
| | 160 | 5.4608E-5 | 1.94 | 3.5325E-5 | 1.97 | 2.8015E-4 | 1.88 |
| velocity | 20 | 6.8693E-3 | - | 5.5689E-3 | - | 2.3222E-2 | - |
| | 40 | 1.9169E-3 | 1.84 | 1.5328E-3 | 1.86 | 6.9587E-3 | 1.74 |
| | 80 | 5.1440E-4 | 1.90 | 4.0665E-4 | 1.91 | 1.9592E-3 | 1.83 |
| | 160 | 1.3425E-4 | 1.94 | 1.0526E-4 | 1.95 | 5.2496E-4 | 1.90 |




(a) initial grid

(b) grid at $t = 0.4$

Figure 5.1: The mesh and pressure results of the Taylor-Green vortex problem.

Table 5.5: Errors of the scheme (2.6) without the term $\int_{\partial I(t)} \left(\widehat{\mathbf{U}}(\widehat{\mathbf{u}} \cdot \mathbf{n} - u_n^g)\phi_j\right) ds$ $(j = 2, 3)$ for Example 5.3.

| | N(=M) | $L_2$ error | Order($L_2$) | $L_1$ error | Order($L_1$) | $L_\infty$ error | Order($L_\infty$) |
|---|---|---|---|---|---|---|---|
| density | 20 | 2.1017E-2 | - | 1.2833E-2 | - | 1.1689E-1 | - |
| | 40 | 8.0879E-3 | 1.38 | 5.4191E-3 | 1.24 | 4.2067E-2 | 1.47 |
| | 80 | 3.5251E-3 | 1.20 | 2.5289E-3 | 1.10 | 1.7455E-2 | 1.27 |
| | 160 | 1.7617E-3 | 1.00 | 1.2797E-3 | 0.98 | 8.6082E-3 | 1.02 |
| energy | 20 | 8.4084E-3 | - | 5.8782E-3 | - | 4.0676E-2 | - |
| | 40 | 3.1346E-3 | 1.42 | 1.8599E-3 | 1.66 | 1.8472E-2 | 1.14 |
| | 80 | 1.3451E-3 | 1.22 | 7.9885E-4 | 1.22 | 7.8895E-3 | 1.23 |
| | 160 | 6.4433E-4 | 1.06 | 4.0886E-4 | 0.97 | 3.5949E-3 | 1.13 |
| momentum | 20 | 1.2845E-2 | - | 9.4996E-3 | - | 5.7429E-2 | - |
| | 40 | 5.2018E-3 | 1.30 | 3.5022E-3 | 1.44 | 2.9045E-2 | 0.98 |
| | 80 | 2.5820E-3 | 1.01 | 1.7468E-3 | 1.00 | 1.4451E-2 | 1.01 |
| | 160 | 1.3707E-3 | 0.91 | 9.4824E-4 | 0.88 | 7.1802E-3 | 1.01 |
| pressure | 20 | 4.1164E-3 | - | 2.6122E-3 | - | 2.4156E-2 | - |
| | 40 | 1.2405E-3 | 1.73 | 7.2898E-4 | 1.84 | 7.7175E-3 | 1.65 |
| | 80 | 3.4340E-4 | 1.85 | 1.9837E-4 | 1.88 | 2.1386E-3 | 1.85 |
| | 160 | 9.0444E-5 | 1.92 | 5.2362E-5 | 1.92 | 5.5093E-4 | 1.96 |
| velocity | 20 | 8.3221E-3 | - | 6.1654E-3 | - | 4.0584E-2 | - |
| | 40 | 2.3862E-3 | 1.80 | 1.6399E-3 | 1.91 | 1.4311E-2 | 1.50 |
| | 80 | 6.8717E-4 | 1.80 | 4.8047E-4 | 1.77 | 4.4504E-3 | 1.69 |
| | 160 | 2.4293E-4 | 1.50 | 1.8721E-4 | 1.36 | 1.3144E-3 | 1.76 |

**Example 5.4 The free expansion problem**

In the cylindrical coordinates, we test the accuracy of the Lagrangian DG scheme (3.6)-(3.7) on a free expansion problem [40]. The initial computational domain is $[0, 1] \times [0, \pi/2]$ defined in the polar coordinates. Its initial condition is:

$$\rho_0 = 1,\ u_{z,0} = u_{r,0} = 0,\ p_0(z, r) = 1 - (z^2 + r^2).$$

The problem has the following analytical solutions,

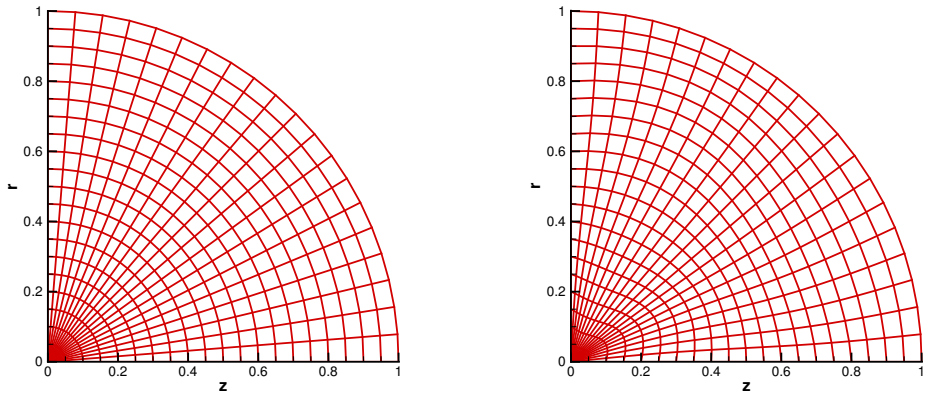
$$
\begin{aligned}
&R(t) = \sqrt{(1+2t^2)},\\
&u_z(z,r,t) = \frac{2t}{1+2t^2}z,\ u_r(z,r,t) = \frac{2t}{1+2t^2}r,\\
&\rho(z,r,t) = \frac{1}{R^3},\\
&p(z,r,t) = \frac{1}{R^5}(1-\frac{z^2+r^2}{R^2}),
\end{aligned}
$$

where $R$ is radius of the free outer boundary.

We perform the test on two different types of grids as shown in Fig. 5.2. The first one is an initially equal-angled polar grid. The second one is an initially non-uniform smooth polar grid, for which each internal grid vertex is obtained by a smooth perturbation from an equal-angled polar grid as follows

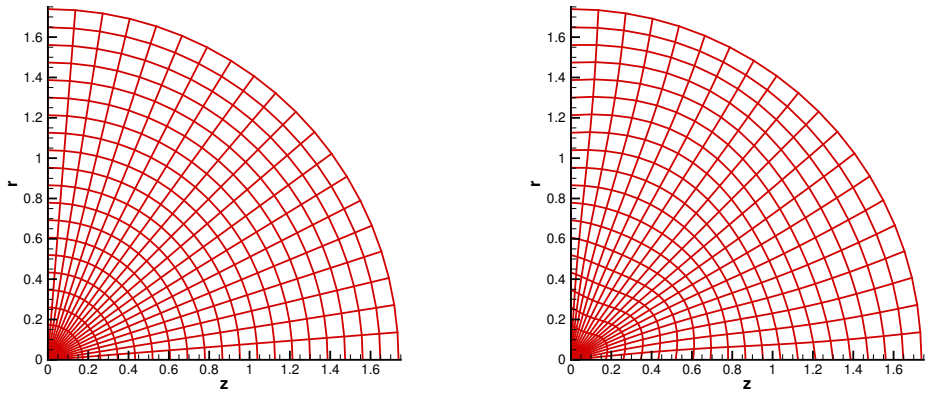
$$
\begin{aligned}
z_{q-\frac{1}{2},l-\frac{1}{2}} &= \xi_{q-\frac{1}{2}}\cos(\frac{1}{2}\pi\theta_{l-\frac{1}{2}}) + \epsilon\sin(2\pi\xi_{q-\frac{1}{2}})\sin(2\pi\theta_{l-\frac{1}{2}}),\\
r_{q-\frac{1}{2},l-\frac{1}{2}} &= \xi_{q-\frac{1}{2}}\sin(\frac{1}{2}\pi\theta_{l-\frac{1}{2}}) + \epsilon\sin(2\pi\xi_{q-\frac{1}{2}})\sin(2\pi\theta_{l-\frac{1}{2}}),\ q=1,\cdots,Q,\ l=1,\cdots,L,
\end{aligned}
$$

where $\xi_{q-\frac{1}{2}} = \frac{q-1}{Q}$, $\theta_{l-\frac{1}{2}} = \frac{l-1}{L}$, $(z_{q-\frac{1}{2},l-\frac{1}{2}}, r_{q-\frac{1}{2},l-\frac{1}{2}})$ is the grid point in the $z$-$r$ coordinates with the sequential indices $(q,l)$ in the radial and angular directions respectively. $Q, L$ represent the number of cells in the above mentioned two directions. $\epsilon$ is a parameter which is chosen as 0.015 in this test. Free boundary condition is applied on the outer boundary.




(a) equal-angled polar grid (b) non-uniform smooth polar grid

Figure 5.2: The initial grid of the free expansion problem with $20 \times 20$ cells.




(a) equal-angled polar grid

(b) non-uniform smooth polar grid

Figure 5.3: The final grid of the free expansion problem with $20 \times 20$ cells at $t = 1$.

Fig. 5.3 shows the final grids at $t = 1$. The errors of the spherical symmetry-preserving DG scheme (3.6)-(3.7) on these two kinds of grids are listed in Tables 5.6-5.7, which are measured on the interval $[\frac{1}{5}Q, \frac{4}{5}Q] \times [1, L]$ to remove the influence from the boundary. In Tables 5.6 and 5.7, we can observe the expected second-order accuracy, which means that the design of the symmetry-preserving DG scheme (3.6)-(3.7) in two-dimensional cylindrical geometry does not change the accuracy.

Table 5.6: Errors of the scheme (3.6)-(3.7) for Example 5.4 using $Q \times L$ initially equal-angled polar grid cells.

| | Q(=L) | $L_2$ error | Order($L_2$) | $L_1$ error | Order($L_1$) | $L_\infty$ error | Order($L_\infty$) |
|---|---|---|---|---|---|---|---|
| density | 10 | 2.8103E-3 | - | 2.3963E-3 | - | 5.0009E-3 | - |
| | 20 | 8.2328E-4 | 1.77 | 6.9510E-4 | 1.79 | 1.4304E-3 | 1.81 |
| | 40 | 2.2858E-4 | 1.85 | 1.9084E-4 | 1.86 | 3.5966E-4 | 1.99 |
| | 80 | 6.0328E-5 | 1.92 | 4.9539E-5 | 1.95 | 9.8956E-5 | 1.86 |
| energy | 10 | 7.0455E-4 | - | 6.1108E-4 | - | 1.3147E-3 | - |
| | 20 | 2.1672E-4 | 1.70 | 1.9592E-4 | 1.64 | 3.8867E-4 | 1.76 |
| | 40 | 6.1611E-5 | 1.81 | 5.7048E-5 | 1.78 | 9.7387E-5 | 2.00 |
| | 80 | 1.6580E-5 | 1.89 | 1.5453E-5 | 1.88 | 2.6085E-5 | 1.90 |
| momentum | 10 | 1.3741E-3 | - | 9.8534E-4 | - | 3.1078E-3 | - |
| | 20 | 3.8569E-4 | 1.83 | 2.7143E-4 | 1.86 | 9.1867E-4 | 1.76 |
| | 40 | 1.0225E-4 | 1.92 | 7.2552E-5 | 1.90 | 2.3750E-4 | 1.95 |
| | 80 | 2.5644E-5 | 2.00 | 1.8597E-5 | 1.96 | 5.5974E-5 | 2.09 |

Table 5.7: Errors of the scheme (3.6)-(3.7) for Example 5.4 using $Q \times L$ initially non-uniform smooth polar grid cells.

| | Q(=L) | $L_2$ error | Order($L_2$) | $L_1$ error | Order($L_1$) | $L_\infty$ error | Order($L_\infty$) |
|---|---|---|---|---|---|---|---|
| density | 10 | 2.9124E-3 | - | 2.4682E-3 | - | 5.8851E-3 | - |
| | 20 | 8.4991E-4 | 1.78 | 7.1383E-4 | 1.79 | 1.6567E-3 | 1.83 |
| | 40 | 2.3443E-4 | 1.86 | 1.9492E-4 | 1.87 | 4.2777E-4 | 1.95 |
| | 80 | 6.1537E-5 | 1.93 | 5.0254E-5 | 1.96 | 1.0970E-4 | 1.96 |
| energy | 10 | 7.2766E-4 | - | 6.2697E-4 | - | 1.6082E-3 | - |
| | 20 | 2.2168E-4 | 1.71 | 1.9951E-4 | 1.65 | 4.6636E-4 | 1.79 |
| | 40 | 6.2534E-5 | 1.83 | 5.7703E-5 | 1.79 | 1.2011E-4 | 1.96 |
| | 80 | 1.6744E-5 | 1.90 | 1.5503E-5 | 1.90 | 2.9233E-5 | 2.04 |
| momentum | 10 | 1.4726E-3 | - | 1.0473E-3 | - | 3.8238E-3 | - |
| | 20 | 4.1065E-4 | 1.84 | 2.8895E-4 | 1.86 | 1.1034E-3 | 1.79 |
| | 40 | 1.0767E-4 | 1.93 | 7.6503E-5 | 1.92 | 2.9088E-4 | 1.92 |
| | 80 | 2.6733E-5 | 2.01 | 1.9395E-5 | 1.98 | 7.0604E-5 | 2.04 |

Next, we focus on verifying the properties of spherical symmetry and non-oscillation of the second-order DG scheme (3.6)-(3.7).

## 5.2 Spherical symmetry-preserving tests in two-dimensional cylindrical coordinates

In this subsection, we test six well-known examples in two-dimensional cylindrical coordinates. The second-order DG scheme (3.6)-(3.7) on initially equal-angled polar grid is used in the following tests unless otherwise stated. The material is ideal gas with the ratio of specific heat capacities $\gamma = \frac{5}{3}$. Reflective boundary conditions are applied to the $z$ and $r$ axes in all the tests. $\mu = \sqrt{z^2 + r^2}$ is the radial coordinate. $u_\mu$ and $u_\theta$ represent the values of velocity in the radial and angular directions in local polar coordinates. The multi-resolution WENO limiter [48] is used to avoid spurious oscillations in most examples except the Sod and Kidder problems, and we use the local characteristic decomposition along the normal direction of the cell edges when applying the WENO limiter. According to [13], in order not to affect the symmetry-preserving property, the WENO limiter is accomplished in the cell's local $\xi$-$\eta$ coordinates.

**Example 5.5 The one-dimensional spherical Sod Riemann problem**

The spherical Sod shock tube [42] is a classical test problem. For this problem, the initial computational domain is $[0, 20] \times [0, \pi/2]$ defined in the polar coordinates with the following initial condition:

$$(\rho,\ u_\mu,\ u_\theta,\ p) = \begin{cases} (1, 0, 0, 1), & 0 \le \mu \le 10, \\ (0.125, 0, 0, 0.1), & 10 \le \mu \le 20. \end{cases}$$

The reflective boundary condition is applied on the outer boundary. The initial grid consists of $400 \times 10$ equal-angled polar cells. The final time is set to $t = 1.4$. Fig. 5.4 shows the numerical results of the mesh and density contour of the second-order DG scheme (3.6)-(3.7), and density as a function of the radial radius at all grid points performed by both the second-order scheme and the first-order scheme. Here the reference solution is the converged result obtained by using a one-dimensional second-order Eulerian code in the spherical coordinate with 10000 grid points. We observe the good symmetry behavior of the scheme. And the numerical results of the second-order DG scheme are less dissipative than the first-order solutions.

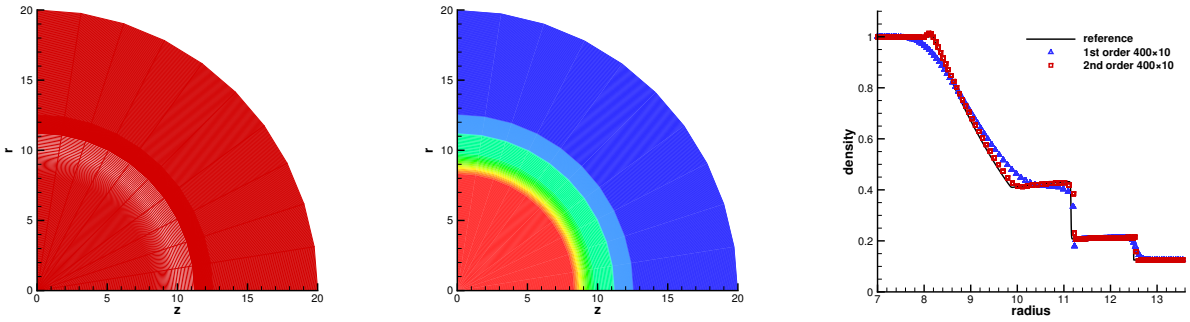



Figure 5.4: The results of the Sod problem with $400 \times 10$ cells at $t = 1.4$. Left: final grid; Middle: density contour of the second-order scheme; Right: density $\rho$ versus radial radius $\mu$ at all grid points.

**Example 5.6 The Noh problem in the cylindrical coordinate system**

The Noh problem [38] is a typical test problem which is widely used to validate the performance of Lagrangian schemes on strong shocks. In this problem, the initial condition is as follows:

$$\rho = 1, \quad u_\mu = -1, \quad u_\theta = 0, \quad p = 10^{-5}.$$

The equal-angled polar grid is applied in the $\frac{1}{4}$-circle computational domain defined in the polar coordinates by $[0, 1] \times [0, \frac{\pi}{2}]$. Free boundary condition is used on the outer boundary. The shock is generated by bringing the cold gas to rest at the origin. The analytical post shock density is 64 and the shock speed is 1/3. Fig. 5.5 shows the numerical results with $100 \times 20$ initially equal-angled polar cells at $t = 0.6$. We can find that the final mesh and density contour are symmetric. The scatter plots of density obtained from the first-order scheme and the second-order DG scheme are non-oscillatory. And the density from the second-order DG scheme is closer to the analytical solution than that from the first-order scheme.

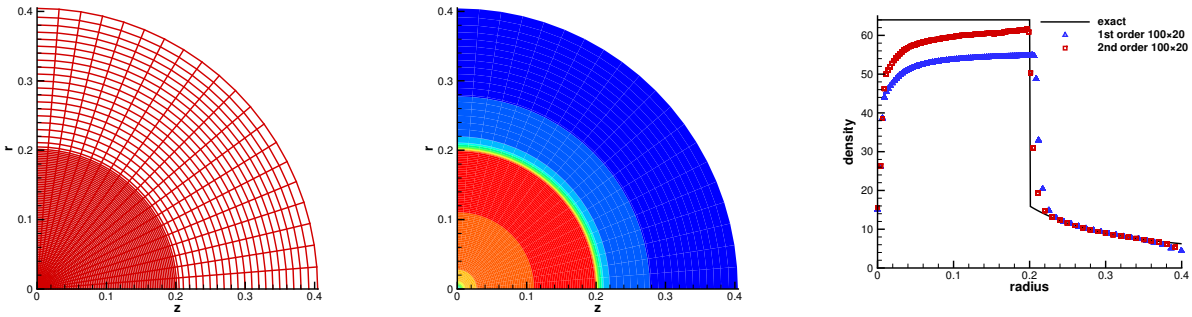



Figure 5.5: The results of the Noh problem with $100 \times 20$ cells at $t = 0.6$. Left: final grid; Middle: density contour of the second-order scheme; Right: density $\rho$ versus radial radius $\mu$ at all grid points.

**Example 5.7 Implosion problem of Lazarus**

Implosion problem of Lazarus [24] has self-similar solutions. In this problem, the initial condition is described as follows:

$$\rho = 1, \quad u_\mu = u_\theta = 0, \quad e = 10^{-5}.$$

A sphere of unit initial radius is driven by an inward radial velocity given by

$$u_\mu(t) = -\frac{\alpha f}{(1-ft)^{1-\alpha}},$$

where $\alpha = 0.6883545,\ f = 1-\epsilon t-\delta t^3,\ \epsilon = 0.185,\ \delta = 0.28$. The initial computational domain is $[0,1]\times[0,\pi/2]$ defined in the polar coordinates. Free boundary condition is applied on the outer boundary. We test the problem on an initially equal-angled polar grid of $200\times 30$ cells. Fig. 5.6 shows the results of the DG scheme (3.6)-(3.7) including the final grid and density solutions at $t = 0.74,\ 0.8$. Here the reference solutions are computed using a one-dimensional second-order Lagrangian code in the spherical coordinate with 10000 cells. In the plots of grid and density contour, we observe the expected symmetry. The non-oscillatory property is well reflected in the scatter plot of density. Compared with the first-order scheme, we find that the numerical solutions from the second-order DG scheme have higher resolution.

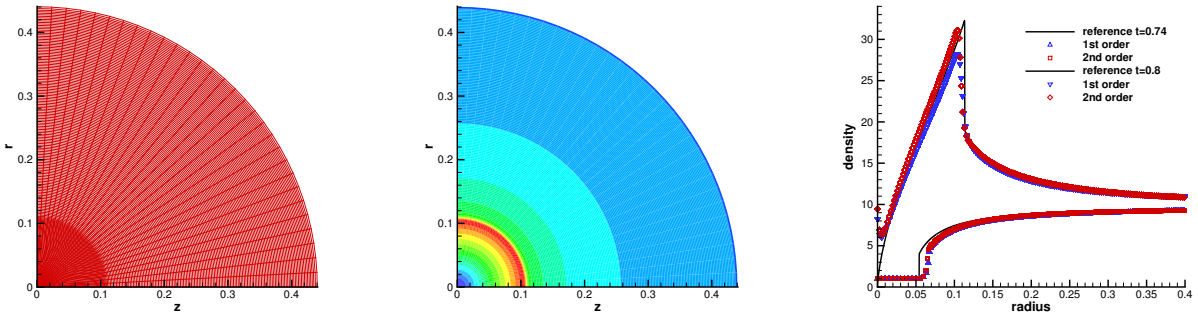



Figure 5.6: The results of the Lazarus problem with $200\times 30$ cells at $t = 0.74,\ 0.8$. Left: final grid; Middle: density contour of the second-order scheme; Right: density $\rho$ versus radial radius $\mu$ at all grid points.

**Example 5.8 Kidder's isentropic compression problem [23, 34]**

In [23], Kidder has computed the analytical solution of the self-similar isentropic compression for a shell filled with perfect gas. At the initial time, we set the computational region as $[\mu_1,\mu_2]\times[0,\pi/2]$ in the polar coordinates. The initial density, velocity and pressure have the following expression,

$$\rho_0(\mu) = \left(\frac{\mu_2^2-\mu^2}{\mu_2^2-\mu_1^2}\rho_1^{\gamma-1} + \frac{\mu^2-\mu_1^2}{\mu_2^2-\mu_1^2}\rho_2^{\gamma-1}\right)^{\frac{1}{\gamma-1}}, \quad \mathbf{u}_0 = \mathbf{0}, \quad p_0(\mu) = s\left(\rho_0(\mu)\right)^\gamma,$$

where $\mu_1 = 0.9,\ \mu_2 = 1.0,\ \rho_1 = 6.31\times 10^{-4},\ \rho_2 = 10^{-2},\ s = 2.15\times 10^4$. With the time marching, the pressure $p_1(t)$ and $p_2(t)$ are imposed continuously at the internal and external boundaries respectively which have the following representation

$$p_1(t) = p_1^0 a(t)^{-\frac{2\gamma}{\gamma-1}}, \quad p_2(t) = p_2^0 a(t)^{-\frac{2\gamma}{\gamma-1}},$$

where $p_1^0 = 0.1,\ p_2^0 = 10$ and $a(t) = \sqrt{1-\left(\frac{t}{\tau}\right)^2}$. $\tau = 6.72\times 10^{-3}$ is the focusing time of the shell and $t\in[0,\tau)$ is the evolving time. The analytical solutions of the three fundamental variables for this problem in spherical geometry are as follows:

$$\begin{aligned}
\rho(\zeta(\mu,t),t) &= \rho_0(\mu)a(t)^{-\frac{2}{\gamma-1}},\\
u_\xi(\zeta(\mu,t),t) &= \mu\frac{\mathrm{d}}{\mathrm{d}t}a(t),\\
p(\zeta(\mu,t),t) &= p_0(\mu)a(t)^{-\frac{2\gamma}{\gamma-1}},
\end{aligned}$$

where $\zeta(\mu,t)$ represents the radius at time $t$ of a point initially located at radius $\mu$. Its analytical solution is $\zeta(\mu,t) = a(t)\mu$.

We test the problem on an initially $40 \times 80$ equal-angular grid. The final time is set to be $t = 0.99\tau$. We display the numerical results of the DG scheme (3.6)-(3.7) in Fig. 5.7 and Fig. 5.8. From the figures, we observe the non-oscillatory and symmetry-preserving properties of the DG scheme. Compared with the first-order scheme, the numerical solutions from the second-order DG scheme are much closer to the analytical solutions.

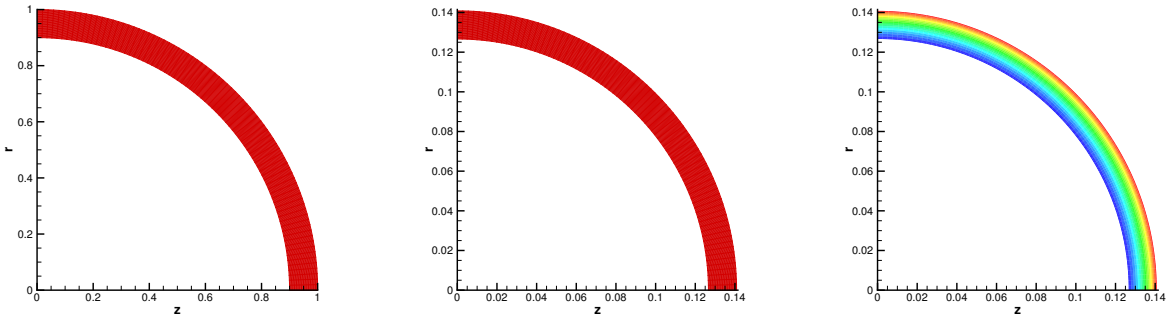



Figure 5.7: The results of the Kidder problem with $40 \times 80$ cells at $t = 0.99\tau$. Left: initial grid; Middle: final grid of the second-order scheme; Right: density contour of the second-order scheme.

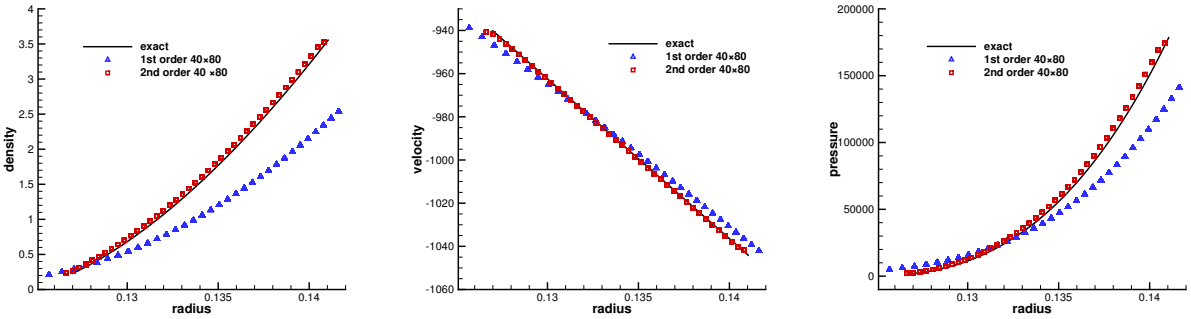



Figure 5.8: Scatter plot of three fundamental variables as a function of the radial position at all grid points in the Kidder problem. Left: density $\rho$ versus radial radius $\mu$; Middle: velocity $u_\mu$ versus radial radius $\mu$; Right: pressure $p$ versus radial radius $\mu$.

**Example 5.9 The spherical Sedov problem in cylindrical coordinate system**

The spherical Sedov blast wave problem [40] in the cylindrical coordinates is a commonly used example of a diverging shock wave. Consider this problem with the initial condition as:

$$\rho = 1, \ u_\mu = u_\theta = 0,$$

and the initial internal energy $e = 10^{-6}$ almost everywhere except in the cells connected to the origin where they share a total value of 0.2468. Reflective boundary conditions are used on all boundaries. The analytical solution is a shock with a peak density of 4 at radius unity at $t = 1$.

In this work, we consider two different types of grids. The first type of grid is the initially equal-angled polar grid with $30 \times 30$ cells, where the initial computational domain is $[0, 1.125] \times [0, \pi/2]$ defined in the polar coordinates. The second type of grid is the initially rectangular grid consisting of $30 \times 30$ uniform cells, where the initial computational domain is a $1.125 \times 1.125$ square. Fig. 5.9 shows the numerical results on initially

equal-angled polar grid at $t = 1$. From the figure, we observe the expected symmetry of grid and density. The shock position and peak density obtained by the second-order DG scheme are more consistent with the analytical solution than those obtained by the first-order scheme, which demonstrates the advantages of the second-order DG scheme (3.6) -(3.7).

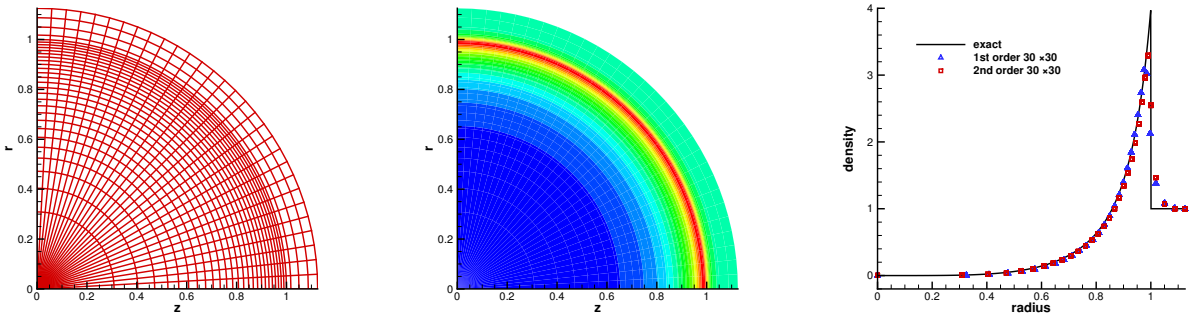



Figure 5.9: The results of the Sedov blast wave problem on $30 \times 30$ initially equal-angled polar grid at $t = 1$. Left: final grid; Middle: density contour of the second-order scheme; Right: density $\rho$ versus radial radius $\mu$ at all grid points.

As a contrast, we test this problem on the initially rectangular grid. In this case, the shock direction is not aligned with the grid lines. Fig. 5.10 shows the second-order results on $30 \times 30$ initially rectangular grid at $t = 1$. Although the symmetry-preserving property of the second-order DG scheme is only proved on the initially equal-angled polar grid, we observe that the numerical results on this non-polar grid are roughly symmetric in the region near the shock wave.

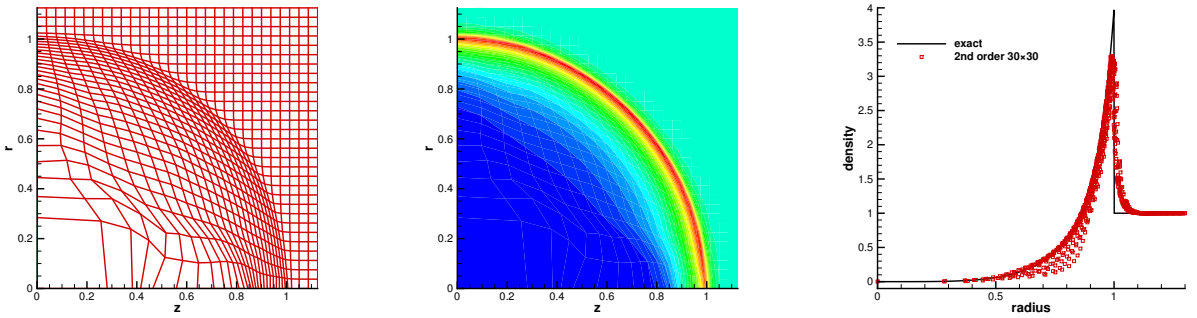



Figure 5.10: The second-order results of the Sedov blast wave problem on $30 \times 30$ initially rectangular grid at $t = 1$. Left: final grid; Middle: density contour; Right: density $\rho$ versus radial radius $\mu$ at all grid points.

**Example 5.10 Coggeshall expansion problem [19]**

In this example, we try to apply Coggeshall's adiabatic compression problem to test the performance of the second-order DG scheme (3.6)-(3.7) on a two-dimensional asymmetric problem. The computational domain is $[0, 1] \times [0, \frac{\pi}{2}]$ defined in the polar coordinates. The initial condition is

$$\rho = 1, \ (u_z, u_r) = (-z/4, -r), \ e = (3z_c/8)^2,$$

where $z_c$ is the $z$ coordinates of the cell center. We set $100 \times 10$ initially equal-angular cells. The final time is set to be $t = 0.8$, and the analytical solution for density is a constant with a value of 37.4. Fig. 5.11 shows

the results of the grids, density contour and density plotted as a function of the radial radius along each radial line, where we can observe the numerical results of the second-order DG scheme converge well to the analytical solutions except in the small region near the origin. This example demonstrates that our proposed DG scheme in two-dimensional cylindrical coordinates can not only effectively simulate the spherical symmetric problems, but also simulate the asymmetric problems well.

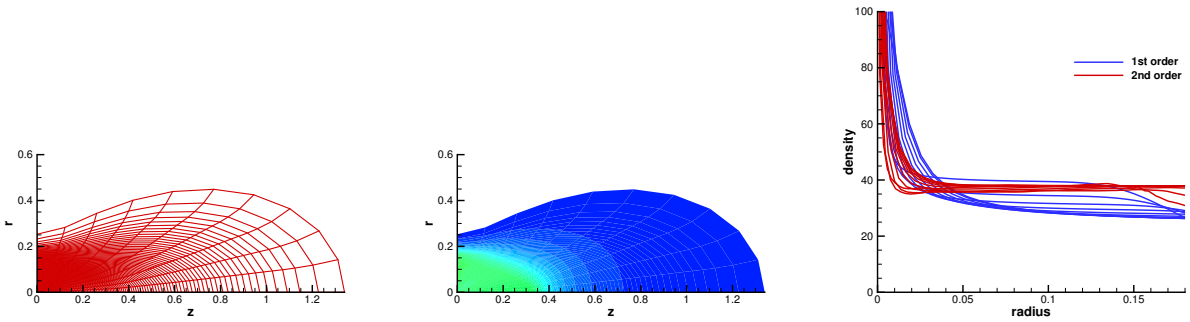



Figure 5.11: The results of the Coggeshall problem with $100 \times 10$ cells at $t = 0.8$. Left: final grid; Middle: density contour; Right: density $\rho$ versus radial radius $\mu$ along each radial line.

# 6 Concluding remarks

In this paper, we have constructed a new second-order cell-centered Lagrangian DG scheme for solving the compressible Euler equations on quadrilateral meshes. This DG scheme is based on the moving physical configuration, and the conserved variables are solved directly. It is conservative for the density, momentum and total energy. Compared to previous research, our proposed Lagrangian DG scheme is truly second-order accurate for all the variables such as density, momentum, total energy, pressure and velocity. The key point is to properly deal with the flux term of the difference between the fluid velocity and the grid velocity. On the one hand, we need to ensure that there is no mass exchange between neighboring cells, which is crucial for the pure Lagrangian methods. On the other hand, the design of this flux term should not destroy second-order accuracy of any variables. To balance these two issues, we compute this flux term only to equations where the basis functions are not equal to 1. Our strategy can obtain a truly second-order Lagrangian DG scheme without mass exchange between neighboring cells. Based on this framework, we have further designed a second-order Lagrangian DG scheme in the two-dimensional cylindrical geometry that maintains both the spherical symmetry and conservation properties. The basic idea is to combine the true volume scheme and the area-weighted scheme, and carefully discretize the mass matrix and source terms. We have tested many well-known numerical examples in the two-dimensional Cartesian and cylindrical coordinates to demonstrate the good performance of the Lagrangian DG schemes in terms of accuracy, symmetry and non-oscillation.

Finally, we point out that our proposed pure Lagrangian DG schemes can achieve at most second-order accuracy for multi-dimensional problems on the straight-line grid, and the discussion of higher order Lagrangian type DG schemes on the straight-line grid is a part of our future work.

# References

[1] D.J. Benson, *Momentum advection on a staggered mesh*, J. Comput. Phys. 100 (1992) 143-162.

[2] D.J. Benson, *Computational methods in Lagrangian and Eulerian hydrocodes*, Comput. Methods Appl. Mech. Eng. 99 (1992) 235-394.

[3] P.L. Browne, *Integrated gradients: a derivation of some difference forms for the equation of motion for compressible flow in two-dimensional Lagrangian hydrodynamics, using integration of pressures over surfaces*, Los Alamos National Laboratory Report LA-2105872-MS, 1986.

[4] D.E. Burton, N.R. Morgan, T.C. Carney, M.A. Kenamond, *Reduction of dissipation in Lagrange cell-centered hydrodynamics (CCH) through corner gradient reconstruction (CGR)*, J. Comput. Phys. 227 (2007) 1567-1596.

[5] J.C. Campbell, M.J. Shashkov, *A tensor artificial viscosity using a mimetic finite difference algorithm*, J. Comput. Phys. 172 (2001) 739-765.

[6] E.J. Caramana, D.E. Burton, M.J. Shashkov, P.P. Whalen, *The construction of compatible hydrodynamics algorithms utilizing conservation of total energy*, J. Comput. Phys. 146 (1998) 227-262.

[7] E.J. Caramana, M.J. Shashkov, P.P. Whalen, *Formulations of artificial viscosity for multidimensional shock wave computations*, J. Comput. Phys. 144 (1998) 70-97.

[8] J. Cheng, C.-W. Shu, *A high order ENO conservative Lagrangian type scheme for the compressible Euler equations*, J. Comput. Phys. 227 (2007) 1567-1596.

[9] J. Cheng, C.-W. Shu, *A third order conservative Lagrangian type scheme on curvilinear meshes for the compressible Euler equation*, Commun. Comput. Phys. 4 (2008) 1008-1024.

[10] J. Cheng, C.-W. Shu, *A cell-centered Lagrangian scheme with the preservation of symmetry and conservation properties for compressible fluid flows in two-dimensional cylindrical geometry*, J. Comput. Phys. 229 (2010) 7191-7206.

[11] J. Cheng, C.-W. Shu, *Improvement on spherical symmetry in two-dimensional cylindrical coordinates for a class of control volume Lagrangian schemes*, Commun. Comput. Phys. 11 (2012) 1144-1168.

[12] J. Cheng, C.-W. Shu, *Positivity-preserving Lagrangian scheme for multi-material compressible flow*, J. Comput. Phys. 257 (2014) 143-168.

[13] J. Cheng, C.-W. Shu, *Second order symmetry-preserving conservative Lagrangian scheme for compressible Euler equations in two-dimensional cylindrical coordinates*, J. Comput. Phys. 272 (2014) 245-265.

[14] B. Cockburn, S. Hou, C.-W. Shu, *The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case*, Math. Comput. 54 (1990) 545-581.

[15] 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.

[16] 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.

[17] B. Cockburn, C.-W. Shu, *The Runge-Kutta local projection-discontinuous-Galerkin finite element method for scalar conservation laws*, ESAIM: Math. Model. Numer. Anal. 25 (1991) 337-361.

[18] B. Cockburn, C.-W. Shu, *The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems*, J. Comput. Phys. 141 (1998) 199-224.

[19] S.V. Coggeshall, J. Meyer-ter-Vehn, *Group invariant solutions and optimal systems for multidimensional hydrodynamics*, J. Math. Phys. 33 (1992) 3585-3601.

[20] B. Després, C. Mazeran, *Lagrangian gas dynamics in two dimensions and lagrangian systems*, Arch. Rational Mech. Anal. 178 (2005) 327-372.

[21] P. Fu, Y. Xia, *The positivity preserving property on the high order arbitrary Lagrangian-Eulerian discontinuous Galerkin method for Euler equations*, J. Comput. Phys. 470 (2022) 111600.

[22] Z. Jia, S. Zhang, *A new high-order discontinuous Galerkin spectral finite element method for Lagrangian gas dynamics in two-dimensions*, J. Comput. Phys. 230 (2011) 2496-2522.

[23] R.E. Kidder, *Laser-driven compression of hollow shells: power requirements and stability limitations*, Nucl. Fusion 1 (1976) 3-14.

[24] R. Lazarus, *Self-similar solutions for converging shocks and collapsing cavities*, SIAM J. Numer. Anal. 18 (1981) 316-371.

[25] N. Lei, J. Cheng, C.-W. Shu, *A high order positivity-preserving polynomial projection remapping method*, J. Comput. Phys. 474 (2023) 111826.

[26] D. Ling, J. Cheng, C.-W. Shu, *Positivity-preserving and symmetry-preserving Lagrangian schemes for compressible Euler equations in cylindrical coordinates*, Comput. Fluids. 157 (2017) 112-130.

[27] K. Lipnikov, M. Shashkov, *A framework for developing a mimetic tensor artificial viscosity for Lagrangian hydrocodes on arbitrary polygonal meshes*, J. Comput. Phys. 229 (2010) 7911-7941.

[28] X. Liu, N. Morgan, D. Burton, *A Lagrangian discontinuous Galerkin hydrodynamic method*, Comput. Fluids. 163 (2018) 68-85.

[29] X. Liu, N. Morgan, D. Burton, *Lagrangian discontinuous Galerkin hydrodynamic methods in axisymmetric coordinates*, J. Comput. Phys. 373 (2018) 253-283.

[30] X. Liu, N. Morgan, D. Burton, *A high-order Lagrangian discontinuous Galerkin hydrodynamic method for quadratic cells using a subcell mesh stabilization scheme*, J. Comput. Phys. 386 (2019) 110-157.

[31] X. Liu, N. Morgan, E. Lieberman, D. Burton, *A fourth-order Lagrangian discontinuous Galerkin method using a hierarchical orthogonal basis on curvilinear grids*, J. Comput. Appl. Math. 404 (2022) 113890.

[32] R. Loubère, J. Ovadia, R. Abgrall, *A Lagrangian discontinuous Galerkin-type method on unstructured meshes to solve hydrodynamics problems*, Int J Numer Methods Fluids. 44 (2004) 645-663.

[33] P.-H. Maire, *A high-order cell-centered Lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes*, J. Comput. Phys. 228 (2009) 2391-2425.

[34] P.-H. Maire, *A high-order cell-centered Lagrangian scheme for compressible fluid flows in two-dimensional cylindrical geometry*, J. Comput. Phys. 228 (2009) 6882-6915.

[35] P.-H. Maire, R. Abgrall, J. Breil, J. Ovadia, *A cell-centered Lagrangian scheme for compressible flow problems*, SIAM J. Sci. Comput. 29 (2007) 1781-1824.

[36] C.D. Munz, *On Godunov-type schemes for Lagrangian gas dynamics*, SIAM J. Numer. Anal. 31 (1994) 17-42.

[37] J. von Neumann, R.D. Richtmyer, *A method for the calculation of hydrodynamics shocks*, J. Appl. Phys. 21 (1950) 232-237.

[38] W.F. Noh, *Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux*, J. Comput. Phys. 72 (1987) 78-120.

[39] W.H. Reed, T.R. Hill, *Triangular mesh method for the neutron transport equation*, Los Alamos Scientific Laboratory Report LA-UR-73-479, Los Alamos, NM, 1973.

[40] L.I. Sedov, *Similarity and Dimensional Methods in Mechanics*, Academic Press, New York, 1959.

[41] C.-W. Shu, S. Osher, *Efficient implementation of essentially non-oscillatory shock-capturing schemes*, J. Comput. Phys. 77 (1988) 439-471.

[42] G. Sod, *A survey of several finite difference methods for systems of non-linear hyperbolic conservation laws*, J. Comput. Phys. 27 (1978) 1-31.

[43] A. Solov'ev, M. Shashkov, *Difference scheme for the Dirichlet particle method in cylindrical coordinates, conserving symmetry of gas-dynamical flow*, Differ. Equ. 24 (1988) 817-823.

[44] F. Vilar, *Cell-centered discontinuous Galerkin discretization for two-dimensional Lagrangian hydrodynamics*, Comput. Fluids. 64 (2012) 64-73.

[45] F. Vilar, P.-H. Maire, R. Abgrall, *A discontinuous Galerkin discretization for solving the two-dimensional gas dynamics equations written under total Lagrangian formulation on general unstructured grids*, J. Comput. Phys. 276 (2014) 188-234.

[46] C. Wang, H. Luo, M. Shashkov, *A reconstructed discontinuous Galerkin method for compressible flows in Lagrangian formulation*, Comput. Fluids. 202 (2020) 104522.

[47] T. Wu, M. Shashkov, N. Morgan, D. Kuzmin, H. Luo, *An updated Lagrangian discontinuous Galerkin hydrodynamic method for gas dynamics*, Comput. Math. Appl. 78 (2019) 258-273.

[48] J. Zhu, Chi-Wang Shu, J. Qiu, *High-order Runge-Kutta discontinuous Galerkin methods with a new type of multi-resolution WENO limiters on triangular meshes*, Appl. Numer. Math. 153 (2020) 519-539.