Journal of Scientific Computing (2021) 88:8
https://doi.org/10.1007/s10915-021-01525-9

An Arbitrary-Lagrangian-Eulerian High-Order Gas-Kinetic Scheme for Three-Dimensional Computations

Liang Pan[1] · **Kun Xu**[2,3]

Received: 5 October 2020 / Revised: 24 March 2021 / Accepted: 14 May 2021

Abstract

In the previous study [J. Comput. Phys. 417 (2020) 109558], under arbitrary-Lagrangian-Eulerian (ALE) formulation a high-order gas-kinetic scheme has been developed for the computation of two-dimensional flows. For the three-dimensional flows, due to the distorted mesh, it becomes more difficult to develop robust high-order ALE methods with the precise preservation of geometric conservation law. In this paper, the high-order gas-kinetic ALE scheme will be constructed for three-dimensional flows. The key ingredients of the scheme are the use of weighted essentially non-oscillatory (WENO) scheme for spatial reconstruction and the two-stage fourth-order discretization for temporal evolution. In the ALE formulation, in order to release the problems associated with mesh distortion and non-coplanar vertexes of a control volume, in the spatial reconstruction the selection of candidate stencils and the topologically independent linear weights have to be carefully designed. In the surface integrals for the flux transport, a bilinear interpolation is used to parameterize both grid coordinates and grid moving velocity with the preservation of the geometric conservation law. In the computation, the grid velocity is determined by the variational formulation based Lagrangian nodal solver. Numerical examples are presented to evaluate the accuracy, robustness, and the preservation of geometric conservation law of the current scheme.

Keywords Weighted essentially non-oscillatory (WENO)scheme · Arbitrary Lagrangian-Eulerian (ALE)method · High-order gas-kinetic scheme

---

✉ Liang Pan
panliang@bnu.edu.cn

Kun Xu
makxu@ust.hk

[1] Laboratory of Mathematics and Complex Systems, School of Mathematical Sciences, Beijing Normal University, Beijing, China

[2] Department of Mathematics, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

[3] Shenzhen Research Institute, Hong Kong University of Science and Technology, Shenzhen, China

1 Introduction

In the computational fluid dynamics, there exist two coordinate systems to describe flow motions, i.e. Eulerian and Lagrangian methods. Considerable progress has been made in the numerical simulations from both methods. The Eulerian methods describe the flow motions with time-independent mesh, while the mesh moves with fluid velocity in the Lagrangian formulation. Generally, the Lagrangian methods can be classified into the staggered methods and cell-centered methods. The staggered methods [10,11,13] define physical variables of density, internal energy and pressure at zone centers, but velocities are defined at grid points. The artificial viscosity is needed to resolve the shock. The cell-centered methods [29–31] define the physical variables and velocities at zone centers, which need to solve the Riemann problem on the cell boundary. The Lagrangian methods can track the material interfaces clearly, but have intrinsic difficulties to tackle the multidimensional flow problems with large-deformation, such as the flow associated with shear and vorticity. In Eulerian methods [42], the mesh is fixed in space and the fluid moves through it. The Eulerian methods are relatively simple, but smear the contact discontinuities and slip lines badly. Integrating the advantages of Eulerian and Lagrangian methods, the arbitrary Lagrangian-Eulerian (ALE) method was originally developed by Hirt [19]. The essence of ALE methods is that the mesh motion can be chosen arbitrarily, providing additional flexibility and accuracy. Generally, there are two types of ALE methods, i.e. the direct and indirect ALE methods. It is common to separate the indirect ALE methods into three distinct stages, i.e., Lagrangian stage, rezoning stage [9], and remapping stage [24]. In the Lagrangian stage, the solution and computational mesh are updated simultaneously. To release the error due to mesh deformation, the computational mesh is adjusted to the optimal position in the rezoning stage. In the remapping stage, the Lagrangian solution is redistributed into the rezoned mesh. Further developments and great achievements have been made for the indirect ALE method [2,3,28]. As the unsteady flow calculations with moving boundaries and interfaces become important, such as the flutter simulation of wings and turbo-machinery blades, the methods with dynamically deforming computational domain are required. In order to incorporate with the boundary deformation dynamically, the rezoning and remapping stages are required every step, which reduce the efficiency of computation. Based on the unified coordinate transformation, the moving mesh methods were developed for Euler and Navier-Stokes equations [21,22]. With the integral form of the fluid dynamic equations, the remapping-free ALE gas-kinetic schemes were developed [32,47]. They can be considered as a direct ALE method, in which the remapping stage is avoided and the grid velocity can be chosen arbitrarily in the flux calculation. In the framework of one-step ADER-WENO schemes [43,44], significant progresses of direct ALE method have also been made for three-dimensional flows [5–7]. The high order direct ALE schemes with topology changes were also developed [17,18]. With the curvilinear finite elements methods, a general framework of high-order Lagrangian discretization was proposed for compressible shock hydrodynamics equations [1,15].

In the past decades, the gas-kinetic schemes (GKS) based on Bhatnagar-Gross-Krook (BGK) model [4,14] have been developed systematically for computations from the low speed flow to supersonic ones [45,46]. Different from the classical numerical methods based on Riemann fluxes [42], the gas-kinetic scheme presents a gas evolution process from kinetic scale to hydrodynamic scale. Both inviscid and viscous fluxes can be recovered from a time-dependent and multi-dimensional gas distribution function at a cell interface. Recently, based on the time-dependent flux function, a two-stage fourth-order method was developed for the Lax-Wendroff type flow solvers [27], particularly applied for the hyperbolic conservation laws [36,38]. With the two-stage temporal discretization, a reliable framework was provided for constructing fourth-order and even higher-order temporal accuracy. More importantly, this scheme is as robust as the second-order one and works perfectly from the subsonic to hypersonic flows. A high-order WENO-IMEX scheme for the BGK model is also developed on general polygonal meshes [8].

Recently, with the third-order WENO scheme on unstructured meshes [20,48], a high-order ALE gas-kinetic scheme was developed for two-dimensional computations [38], in which the expected order of accuracy is achieved and geometric conservation law is preserved. In this paper, the high-order ALE gas-kinetic scheme will be extended to three-dimensional flows. In the classical WENO scheme, the linear weights are obtained by solving the linear system, which is dependent on the geometrical topology. However, in the ALE computations, the efficiency will be reduced dramatically for solving the linear weight every time step. In order to achieve the order of accuracy and improve the efficiency, a newly developed WENO scheme [37] is used, in which the ideas of simple WENO reconstruction [49–51] and central WENO reconstruction [25,26] are adopted. A simple strategy of selecting stencils for reconstruction is adopted and the topology independent linear weights are chosen, which improve the efficiency and robustness of the classical WENO reconstruction. For the three-dimensional ALE computation, it is also common to generate the meshes with non-coplanar vertexes, which introduce extra difficulties for three-dimensional methods. The trilinear interpolation is used to parameterize hexahedron cells for the evolution of cell averaged variables. For the surface integral of numerical fluxes, the bilinear interpolation is used to parameterize both grid coordinates and grid velocities. Taken the variation of grid velocity into account, the geometric conservation law can be preserved. Numerical examples are presented to validate the accuracy, robustness, and geometry conservation law of the current scheme. In the computations, the mesh velocity can be given by the variational approach [40] for local mesh adaptation, and the Lagrangian nodal solver [29] for tracking material interface.

This paper is organized as follows. In Sect. 2, the gas-kinetic scheme in ALE formulation is introduced. The implementation of high-order scheme is presented in Sect. 3, including numerical fluxes, spatial reconstruction and temporal discretization. Section 4 includes the numerical examples. The last section is the conclusion.

2 Gas-Kinetic Scheme in ALE Formulation

The three-dimensional BGK equation [4,14] can be written as

$$f_t + uf_x + vf_y + wf_z = \frac{g-f}{\tau}, \tag{1}$$

where $\boldsymbol{u} = (u, v, w)$ is the particle velocity, f is the gas distribution function, g is the three-dimensional Maxwellian distribution and τ is the collision time. The collision term satisfies the compatibility condition

$$\int \frac{g-f}{\tau}\psi \mathrm{d}\Xi = 0, \tag{2}$$

where $\psi = (\psi_1, ..., \psi_5)^T = \left(1, u, v, w, \frac{1}{2}\left(u^2 + v^2 + w^2 + \varsigma^2\right)\right)^T$, the internal variable $\varsigma^2 = \varsigma_1^2 + ... + \varsigma_K^2$, $\mathrm{d}\Xi = \mathrm{d}u\mathrm{d}v\mathrm{d}w\mathrm{d}\varsigma^1...\mathrm{d}\varsigma^K$, γ is the specific heat ratio and $K = (5-3\gamma)/(\gamma-1)$ is the degree of freedom for three-dimensional flows. According to the Chapman-Enskog expansion for BGK equation, the macroscopic governing equations can be derived [45,46]. In the continuum region, the BGK equation can be rearranged and the gas distribution function can be expanded as

$$f = g - \tau D_{\boldsymbol{u}} g + \tau D_{\boldsymbol{u}}(\tau D_{\boldsymbol{u}})g - \tau D_{\boldsymbol{u}}[\tau D_{\boldsymbol{u}}(\tau D_{\boldsymbol{u}})g] + ...,$$

where $D_{\boldsymbol{u}} = \dfrac{\partial}{\partial t} + \boldsymbol{u} \cdot \nabla$. With the zeroth-order truncation $f = g$, the Euler equations can be obtained. For the first-order truncation

$$f = g - \tau(ug_x + vg_y + wg_z + g_t),$$

the Navier-Stokes equations can be obtained. With higher-order truncations, the Burnett and super-Burnett equations can be obtained as well.

In this paper, a high-order gas-kinetic scheme (HGKS) will be constructed in the arbitrary-Lagrangian-Eulerian (ALE) framework for three-dimensional flows. The BGK equation on moving reference can be written as

$$f_t + \left(u - U^g\right) f_x + \left(v - V^g\right) f_y + \left(w - W^g\right) f_z = \frac{g - f}{\tau}, \tag{3}$$

where $\boldsymbol{U}^g = (U^g, V^g, W^g)$ is the grid velocity given arbitrarily. Taking moments of Eq. (3) and integrating with respect to space, the semi-discretized finite volume scheme can be expressed as

$$\frac{\mathrm{d}(|\Omega_i|Q_i)}{\mathrm{d}t} = -\mathcal{L}(\Omega_i, Q_i), \tag{4}$$

where Q_i is the cell averaged conservative value of Ω_i and $|\Omega_i|$ is the volume of Ω_i. The operator $\mathcal{L}$ is defined as

$$\mathcal{L}(\Omega_i, Q_i) = -\sum_{p=1}^{6} \mathbb{F}_p(t) = -\sum_{p=1}^{6} \iint_{\Sigma_p} \boldsymbol{F}(Q, t)\mathrm{d}\sigma, \tag{5}$$

where Σ_p is the cell interface of Ω_i, $\boldsymbol{n}$ is the outer normal direction and the numerical flux is given as

$$\boldsymbol{F}(Q, t) = \int \psi f\left(\boldsymbol{x}, t, \boldsymbol{u}, \varsigma\right)\left(\boldsymbol{u} - \boldsymbol{U}^g\right) \cdot \boldsymbol{n}\mathrm{d}\Xi.$$

The structured mesh is used in this paper, and the extension to unstructured mesh will be considered in the future.

For simplicity, the grid velocity is assumed to be constant during a time interval. For two-dimensional computations, the segments of each cell keep straight with constant grid velocity, and no special treatment is needed in the computation. However, for three-dimensional computations, the vertexes might become non-coplanar even with constant grid velocity, which introduces more difficulties to preserve the order of accuracy and geometric conservation law. Thus, a trilinear interpolation is introduced for hexahedron cell with non-coplanar vertexes

$$\boldsymbol{X}(\xi, \eta, \zeta) = \sum_{i=1}^{8} \boldsymbol{x}_i \psi_i(\xi, \eta, \zeta),$$

where $(\xi, \eta, \zeta) \in [-1/2, 1/2]^3$, $\boldsymbol{x}_i$ is the vertex of a hexahedron and ψ_i is the base function given as follows

$$\psi_1 = \frac{1}{8}(1 - 2\xi)(1 - 2\eta)(1 - 2\zeta),\ \psi_2 = \frac{1}{8}(1 - 2\xi)(1 - 2\eta)(1 + 2\zeta),$$

$$\psi_3 = \frac{1}{8}(1-2\xi)(1+2\eta)(1-2\zeta),\ \psi_4 = \frac{1}{8}(1-2\xi)(1+2\eta)(1+2\zeta),$$
$$\psi_5 = \frac{1}{8}(1+2\xi)(1-2\eta)(1-2\zeta),\ \psi_6 = \frac{1}{8}(1+2\xi)(1-2\eta)(1+2\zeta),$$
$$\psi_7 = \frac{1}{8}(1+2\xi)(1+2\eta)(1-2\zeta),\ \psi_8 = \frac{1}{8}(1+2\xi)(1+2\eta)(1+2\zeta).$$

With the trilinear interpolation, the triple integral over a control volume can be simply given by Gaussian quadrature as follows

$$\begin{aligned}\iiint_{\Omega} x^a y^b z^c \mathrm{d}x\mathrm{d}y\mathrm{d}z &= \iiint_{\Omega} x^a y^b z^c(\xi,\eta,\zeta)\left|\frac{\partial(x,y,z)}{\partial(\xi,\eta,\zeta)}\right|\mathrm{d}\xi\mathrm{d}\eta\mathrm{d}\zeta \\ &= \sum_{l,m,n=1}^{2} \omega_{lmn} x^a y^b z^c(\xi_l,\eta_m,\zeta_n)\left|\frac{\partial(x,y,z)}{\partial(\xi,\eta,\zeta)}\right|_{(\xi_l,\eta_m,\zeta_n)}\Delta\xi\Delta\eta\Delta\zeta,\end{aligned}$$

where ω_{lmn} is the quadrature weight and (ξ_l,η_m,ζ_n) is the quadrature point. For each cell interface, the trilinear interpolation reduces to a bilinear interpolation. For example, the cell interface with $\xi = -1/2$ is defined as

$$\boldsymbol{X}(\eta,\zeta) = \sum_{i=1}^{4} \boldsymbol{x}_i\phi_i(\eta,\zeta), \tag{6}$$

where $(\eta,\zeta)\in[-1/2,1/2]^2$, $\boldsymbol{x}_i$ is the vertex of the interface and ϕ_i is the base function

$$\phi_1 = \frac{1}{4}(1-2\eta)(1-2\zeta),\ \phi_2 = \frac{1}{4}(1-2\eta)(1+2\zeta),$$
$$\phi_3 = \frac{1}{4}(1+2\eta)(1-2\zeta),\ \phi_4 = \frac{1}{4}(1+2\eta)(1+2\zeta).$$

With the parameterized cell interface, the local coordinate $(\boldsymbol{n}_x,\boldsymbol{n}_y,\boldsymbol{n}_z)$ can be given by the procedure as follows

$$\begin{aligned}\boldsymbol{n}_x &= \left(\boldsymbol{X}_\eta\times\boldsymbol{X}_\zeta\right)/\|\boldsymbol{X}_\eta\times\boldsymbol{X}_\zeta\|,\\ \boldsymbol{n}_z &= \boldsymbol{X}_\zeta/\|\boldsymbol{X}_\zeta\|,\\ \boldsymbol{n}_y &= \boldsymbol{n}_z\times\boldsymbol{n}_x.\end{aligned} \tag{7}$$

With the constant grid velocity during a time interval, the velocity of curved interface can be given by the bilinear interpolation as well

$$\boldsymbol{U}^g(\eta,\zeta) = \sum_{i=1}^{4}\boldsymbol{U}_i^g(t)\phi_i(\eta,\zeta), \tag{8}$$

where $\boldsymbol{U}_i^g$ is the constant grid velocity. The procedure above is the key to achieve the geometric conservation law.

With the parameterized cell interface, the numerical flux is provided by Gaussian quadrature of surface integral over the cell interface

$$\begin{aligned}\mathbb{F}_p(t) &= \iint_{\Sigma_p}\boldsymbol{F}(Q,t)\mathrm{d}\sigma = \int_{-1/2}^{1/2}\int_{-1/2}^{1/2}\boldsymbol{F}(Q(\boldsymbol{X}(\eta,\zeta)),t)\|\boldsymbol{X}_\eta\times\boldsymbol{X}_\zeta\|\mathrm{d}\eta\mathrm{d}\zeta\\ &= \sum_{m_1,m_2=1}^{2}\omega_{m_1,m_2}\boldsymbol{F}_{m_1,m_2}(Q,t)\|\boldsymbol{X}_\eta\times\boldsymbol{X}_\zeta\|_{m_1,m_2}\Delta\eta\Delta\zeta,\end{aligned} \tag{9}$$

where ω_{m_1,m_2} is the quadrature weight. The numerical flux $\boldsymbol{F}_{m_1,m_2}(Q,t)$ can be obtained by taking moments of the gas distribution function

$$\boldsymbol{F}_{m_1,m_2}(t)=\begin{pmatrix} F^{\rho}_{m_1,m_2}\\ F^{\rho U}_{m_1,m_2}\\ F^{\rho V}_{m_1,m_2}\\ F^{\rho W}_{m_1,m_2}\\ F^{\rho E}_{m_1,m_2}\end{pmatrix}=\int \psi f(\boldsymbol{x}_{m_1,m_2},t,\boldsymbol{u},\varsigma)(\boldsymbol{u}-\boldsymbol{U}^g_{m_1,m_2})\cdot\boldsymbol{n}_{m_1,m_2}\mathrm{d}\Xi,$$

where $\boldsymbol{n}_{m_1,m_2}$ is the outer normal direction, $\boldsymbol{x}_{m_1,m_2}$ is the coordinate of quadrature point and $\boldsymbol{U}^g_{m_1,m_2}$ is the velocity of quadrature point given by the bilinear interpolation Eq.(6) and Eq.(8). In the actual computation, the numerical flux can be obtained by taking moments of gas distribution function in local coordinate, and the component-wise form can be written as

$$\widetilde{\boldsymbol{F}}_{m_1,m_2}(t)=\begin{pmatrix} F^{\widetilde{\rho}}_{m_1,m_2}\\ F^{\widetilde{\rho U}}_{m_1,m_2}\\ F^{\widetilde{\rho V}}_{m_1,m_2}\\ F^{\widetilde{\rho W}}_{m_1,m_2}\\ F^{\widetilde{\rho E}}_{m_1,m_2}\end{pmatrix}=\int \widetilde{u}\begin{pmatrix} 1\\ \widetilde{u}\\ \widetilde{v}\\ \widetilde{w}\\ \frac{1}{2}(\widetilde{u}^2+\widetilde{v}^2+\widetilde{w}^2+\varsigma^2)\end{pmatrix} f(\boldsymbol{x}_{m_1,m_2},t,\widetilde{\boldsymbol{u}},\varsigma)\mathrm{d}\widetilde{\Xi}, \quad (10)$$

where the relative particle velocity in the local coordinate Eq.(7) is given by

$$\widetilde{\boldsymbol{u}}=(\boldsymbol{u}-\boldsymbol{U}^g_{m_1,m_2})\cdot(\boldsymbol{n}_x,\boldsymbol{n}_y,\boldsymbol{n}_z)_{m_1,m_2}.$$

Each component of $\boldsymbol{F}_{m_1,m_2}(t)$ can be given by the combination of $\widetilde{\boldsymbol{F}}_{m_1,m_2}(t)$ as follows

$$\begin{cases} F^{\rho}_{m_1,m_2} =F^{\widetilde{\rho}}_{m_1,m_2},\\ F^{\rho U}_{m_1,m_2} =U^g_{m_1,m_2}F^{\widetilde{\rho}}_{m_1,m_2}+a_{11}F^{\widetilde{\rho U}}_{m_1,m_2}+a_{12}F^{\widetilde{\rho V}}_{m_1,m_2}+a_{13}F^{\widetilde{\rho W}}_{m_1,m_2},\\ F^{\rho V}_{m_1,m_2} =V^g_{m_1,m_2}F^{\widetilde{\rho}}_{m_1,m_2}+a_{21}F^{\widetilde{\rho U}}_{m_1,m_2}+a_{22}F^{\widetilde{\rho V}}_{m_1,m_2}+a_{23}F^{\widetilde{\rho W}}_{m_1,m_2},\\ F^{\rho W}_{m_1,m_2} =W^g_{m_1,m_2}F^{\widetilde{\rho}}_{m_1,m_2}+a_{31}F^{\widetilde{\rho U}}_{m_1,m_2}+a_{32}F^{\widetilde{\rho V}}_{m_1,m_2}+a_{33}F^{\widetilde{\rho W}}_{m_1,m_2},\\ F^{\rho E}_{m_1,m_2} =F^{\widetilde{\rho E}}_{m_1,m_2}+\dfrac{1}{2}(U^g_{m_1,m_2})^2+(V^g_{m_1,m_2})^2+(W^g_{m_1,m_2})^2)F^{\widetilde{\rho}}_{m_1,m_2}\\ \qquad +(a_{11}U^g_{m_1,m_2}+a_{21}V^g_{m_1,m_2}+a_{31}W^g_{m_1,m_2})F^{\widetilde{\rho U}}_{m_1,m_2}\\ \qquad +(a_{12}U^g_{m_1,m_2}+a_{22}V^g_{m_1,m_2}+a_{32}W^g_{m_1,m_2})F^{\widetilde{\rho V}}_{m_1,m_2}\\ \qquad +(a_{13}U^g_{m_1,m_2}+a_{23}V^g_{m_1,m_2}+a_{33}W^g_{m_1,m_2})F^{\widetilde{\rho W}}_{m_1,m_2}. \end{cases}$$

where (a_{ij}) is denoted as the inverse matrix of $(\boldsymbol{n}_x,\boldsymbol{n}_y,\boldsymbol{n}_z)_{m_1,m_2}$. Therefore, the time-dependent numerical fluxes with mesh velocity $\mathbb{F}_p(t)$ in Eq.(9) over cell interface Σ_p can be constructed.

3 Implementation of High-Order Scheme

3.1 Numerical Flux

To obtain the numerical flux, the gas distribution function needs to be constructed at quadrature point, which can be provided by the integral solution of BGK equation Eq.(1) as follows

$$f(\boldsymbol{x}_{m_1,m_2}, t, \boldsymbol{u}, \varsigma) = \frac{1}{\tau}\int_0^t g\left(\boldsymbol{x}', t', \boldsymbol{u}, \varsigma\right) e^{-(t-t')/\tau} \mathrm{d}t' + e^{-t/\tau} f_0(-\boldsymbol{u}t, \varsigma),$$

where $\widetilde{\boldsymbol{u}}$ in Eq. (10) is denoted as $\boldsymbol{u}$ for simplicity, $\boldsymbol{u} = (u, v, w)$ is the particle velocity in the local coordinate, $\boldsymbol{x}_{m_1,m_2} = \boldsymbol{x}' + \boldsymbol{u}(t - t')$ is the trajectory of particles, f_0 is the initial gas distribution function and g is the corresponding equilibrium state. With the reconstruction of macroscopic variables, the second-order gas distribution function can be expressed as

$$\begin{aligned} f(\boldsymbol{x}_{m_1,m_2}, t, \boldsymbol{u}, \varsigma) = &\left(1 - e^{-t/\tau}\right) g_0 + \left((t+\tau)e^{-t/\tau} - \tau\right)(\overline{a}_1 u + \overline{a}_2 v + \overline{a}_3 w) g_0 \\ &+ \left(t - \tau + \tau e^{-t/\tau}\right)\bar{A} g_0 \\ &+ e^{-t/\tau} g_r[1 - (\tau + t)\left(a_1^r u + a_2^r v + a_3^r w\right) - \tau A^r)]H(u) \\ &+ e^{-t/\tau} g_l[1 - (\tau + t)\left(a_1^l u + a_2^l v + a_3^l w\right) - \tau A^l)](1 - H(u)), \end{aligned} \tag{11}$$

where $H(u)$ is the Heaviside function, g_l, g_r are the equilibrium states corresponding to the reconstructed variables Q_l, Q_r at both sides of cell interface. The equilibrium state g_0 and corresponding conservative variables Q_0 are given by the compatibility condition Eq. (2)

$$\int \psi g_0 \mathrm{d}\Xi = Q_0 = \int_{u>0} \psi g_l \mathrm{d}\Xi + \int_{u<0} \psi g_r \mathrm{d}\Xi.$$

The coefficients in Eq. (11) can be obtained by the reconstructed directional derivatives and compatibility condition

$$\begin{aligned} &\left\langle a_1^k\right\rangle = \frac{\partial Q_k}{\partial \boldsymbol{n}_x}, \left\langle a_2^k\right\rangle = \frac{\partial Q_k}{\partial \boldsymbol{n}_y}, \left\langle a_3^k\right\rangle = \frac{\partial Q_k}{\partial \boldsymbol{n}_z}, \left\langle a_1^k u + a_2^k v + a_3^k w + A^k\right\rangle = 0, \\ &\langle \overline{a}_1\rangle = \frac{\partial Q_0}{\partial \boldsymbol{n}_x}, \langle \overline{a}_2\rangle = \frac{\partial Q_0}{\partial \boldsymbol{n}_y}, \langle \overline{a}_3\rangle = \frac{\partial Q_0}{\partial \boldsymbol{n}_z}, \left\langle \overline{a}_1 u + \overline{a}_2 v + \overline{a}_3 w + \overline{A}\right\rangle = 0, \end{aligned}$$

where $k = l, r$ and $\langle ...\rangle$ is the moment of the equilibrium g defined by

$$\langle ...\rangle = \int g(...)\psi \mathrm{d}\Xi.$$

The spatial derivatives $\frac{\partial Q_{l,r}}{\partial \boldsymbol{n}_x}, \frac{\partial Q_{l,r}}{\partial \boldsymbol{n}_y}, \frac{\partial Q_{l,r}}{\partial \boldsymbol{n}_z}$ can be determined by spatial reconstruction. In the classical gas-kinetic scheme, an extra reconstruction is needed for the equilibrium part, which introduces extra difficulty for the genuine multidimensional scheme [38]. In this paper, the spatial derivatives of the equilibrium part can be given by

$$\begin{aligned} \frac{\partial Q_0}{\partial \boldsymbol{n}_x} &= \int_{u>0} \psi a_1^l g_l \mathrm{d}\Xi + \int_{u<0} \psi a_1^r g_r \mathrm{d}\Xi, \\ \frac{\partial Q_0}{\partial \boldsymbol{n}_y} &= \int_{u>0} \psi a_2^l g_l \mathrm{d}\Xi + \int_{u<0} \psi a_2^r g_r \mathrm{d}\Xi, \end{aligned}$$

$$\frac{\partial Q_0}{\partial \boldsymbol{n}_z}=\int_{u>0}\psi a_3^l g_l \mathrm{d}\Xi+\int_{u<0}\psi a_3^r g_r \mathrm{d}\Xi .$$

More details of gas-kinetic scheme can be found in [45].

3.2 Temporal Discretization and Geometrical Conservation Law

Recently, a two-stage fourth-order temporal discretization was developed for Lax-Wendroff flow solvers, such as the generalized Riemann problem (GRP) solver [27] and gas-kinetic scheme (GKS) [36]. Consider the following time-dependent equation

$$\frac{\partial Q}{\partial t}=\mathcal{L}(Q),$$

where $\mathcal{L}$ is an operator for spatial derivative of flux, the flow variable Q^{n+1} at $t_{n+1}=t_n+\Delta t$ can be updated with the following formula

$$\begin{aligned}
Q^{*}&=Q^{n}+\frac{1}{2}\Delta t\mathcal{L}\left(Q^{n}\right)+\frac{1}{8}\Delta t^{2}\frac{\partial}{\partial t}\mathcal{L}(Q^{n}),\\
Q^{n+1}&=Q^{n}+\Delta t\mathcal{L}\left(Q^{n}\right)+\frac{1}{6}\Delta t^{2}\left(\frac{\partial}{\partial t}\mathcal{L}\left(Q^{n}\right)+2\frac{\partial}{\partial t}\mathcal{L}\left(Q^{*}\right)\right).
\end{aligned}$$

It can be proved that for hyperbolic equations the above temporal discretization provides a fourth-order time accurate solution [27,36]. To implement the two-stage fourth-order method for Eq.(4), $\mathcal{L}$ and $\partial_t\mathcal{L}$ at the initial and intermediate stages need to be given. For the stationary mesh, they can be simply given as follows

$$\mathcal{L}\left(\Omega_i^n, Q_i^n\right)=-\sum_{p=1}^{6}\mathbb{F}_p^n,\ \ \partial_t\mathcal{L}\left(\Omega_i^n, Q_i^n\right)=-\sum_{p=1}^{6}\partial_t\mathbb{F}_p^n. \tag{12}$$

To determine these coefficients, the time dependent numerical flux $\mathbb{F}_p(t)$ in Eq. (9) can be approximate as a linear function

$$\mathbb{F}_p(t)=\mathbb{F}_p^n+\partial_t\mathbb{F}_p^n(t-t_n). \tag{13}$$

For the gas-kinetic scheme, the gas evolution is a relaxation process from kinetic to hydrodynamic scale through the exponential function, and the corresponding flux is a complicated function of time. In order to obtain the time derivatives of the flux function, the flux function should be approximated as a linear function of time within a time interval. Integrating Eq. (13) over $[t_n, t_n+\Delta t/2]$ and $[t_n, t_n+\Delta t]$, we have the following two equations

$$\begin{cases}
\mathbb{F}_p^n\Delta t+\dfrac{1}{2}\partial_t\mathbb{F}_p^n\Delta t^2=\displaystyle\int_{t_n}^{t+_n\Delta t}\mathbb{F}_p(t)\mathrm{d}t,\\
\dfrac{1}{2}\mathbb{F}_p^n\Delta t+\dfrac{1}{8}\partial_t\mathbb{F}_p^n\Delta t^2=\displaystyle\int_{t_n}^{t_n+\Delta t/2}\mathbb{F}_p(t)\mathrm{d}t.
\end{cases} \tag{14}$$

The coefficients $\mathbb{F}_p^n$ and $\partial_t\mathbb{F}_p^n$ can be obtained by solving the linear system, and $\mathcal{L}\left(\Omega_i^n, Q_i^n\right)$ and $\partial_t\mathcal{L}\left(\Omega_i^n, Q_i^n\right)$ can be given by Eq. (12). More details for two-stage fourth-order method can be found in [36].

For the ALE computation, the variation of geometrical variable and grid velocity need to be considered for the numerical flux. Otherwise, the geometrical conservation law (GCL) can not be preserved. Generally, the geometric conservation law is often referred as the preservation of free stream, which means that a uniform flow on a grid that moves arbitrarily in space as a function of time must be preserved for any numerical discretization scheme [35,41]. In this section, a brief analysis for GCL is given. For simplicity, a stationary uniform flow with $\rho = 1$ is considered. For a cell interface of Ω_i, which is denoted as $\boldsymbol{X}_1\boldsymbol{X}_2\boldsymbol{X}_3\boldsymbol{X}_4$, the grid velocity of $\boldsymbol{X}_i$ is $\boldsymbol{U}_i^g$, and $\boldsymbol{U}_i^g$ is constant in a time step. The hexahedron swept by such interface during a time interval $[t_n, t_n + t]$ is parameterized by

$$\mathbb{X}(\xi,\eta,\zeta) = \sum_{i=1}^{4} \Big(\boldsymbol{X}_i \psi_i(\xi,\eta,\zeta) + \Big(\boldsymbol{X}_i + \boldsymbol{U}_i^g \Delta t\Big)\psi_{i+4}(\xi,\eta,\zeta)\Big) = \boldsymbol{X} + \frac{1}{2}(t-t_n)(1+2\xi)\boldsymbol{U}^g,$$

where $\boldsymbol{X}$ and $\boldsymbol{U}^g$ are given by Eq. (6) and (8). The volume of such hexahedron can be calculated as

$$V(t) = \int_{-1/2}^{1/2}\int_{-1/2}^{1/2}\int_{-1/2}^{1/2} |\mathbb{X}_\xi, \mathbb{X}_\eta, \mathbb{X}_\zeta| \mathrm{d}\xi \mathrm{d}\eta \mathrm{d}\zeta.$$

According to the Gaussian quadrature, the volume can be expressed as follows

$$V(t) = \sum_{m_1,m_2=1}^{2} \omega_{m_1,m_2}\left(\left|\boldsymbol{U}^g, \boldsymbol{X}_\eta, \boldsymbol{X}_\zeta\right|(t-t_n) + V_{m1,m2}(t)\right)\Delta\eta\Delta\zeta,$$

where

$$V_{m_1,m_2}(t) = \left(\frac{1}{2}\left(\left|\boldsymbol{U}^g, \boldsymbol{X}_\eta, \boldsymbol{U}_\zeta^g\right| + \left|\boldsymbol{U}^g, \boldsymbol{U}_\eta^g, \boldsymbol{X}_\zeta\right|\right)(t-t_n)^2 + \frac{1}{3}\left|\boldsymbol{U}^g, \boldsymbol{U}_\eta^g, \boldsymbol{U}_\zeta^g\right|(t-t_n)^3\right)_{m_1,m_2}.$$

For such steady uniform flow, we have the density flux $\boldsymbol{F}_{m_1,m_2}^\rho = -\boldsymbol{U}_{m_1,m_2}^g$ and the first term can be rewritten as

$$\begin{aligned}\left|\boldsymbol{U}^g, \boldsymbol{X}_\eta, \boldsymbol{X}_\zeta\right|_{m_1,m_2} &= U_{m_1,m_2}^g \cdot \boldsymbol{n}_{m1,m_2}\left\|\boldsymbol{X}_\eta \times \boldsymbol{X}_\zeta\right\|_{m_1,m_2}\\ &= -\boldsymbol{F}_{m_1,m_2}^\rho\left\|\boldsymbol{X}_\eta \times \boldsymbol{X}_\zeta\right\|_{m_1,m_2},\end{aligned}$$

where $\boldsymbol{n}_{m_1,m_2}$ is the unit outer normal direction of quadrature point, and this term is identical with Eq. (12). The later three terms represent the variation of geometry, and can be also written as the flux form

$$\begin{aligned}\left|\boldsymbol{U}^g, \boldsymbol{U}_\eta^g, \boldsymbol{X}_\zeta\right|_{m_1,m_2} &= -(\boldsymbol{F}_1)_{m_1,m_2}^\rho\left\|\boldsymbol{U}_\eta^g \times \boldsymbol{X}_\zeta\right\|_{m_1,m_2},\\ \left|\boldsymbol{U}^g, \boldsymbol{X}_\eta, \boldsymbol{U}_\zeta^g\right|_{m_1,m_2} &= -(\boldsymbol{F}_2)_{m_1,m_2}^\rho\left\|\boldsymbol{X}_\eta \times \boldsymbol{U}_\zeta^g\right\|_{m_1,m_2},\\ \left|\boldsymbol{U}^g, \boldsymbol{U}_\eta^g, \boldsymbol{U}_\zeta^g\right|_{m_1,m_2} &= -(\boldsymbol{F}_3)_{m_1,m_2}^\rho\left\|\boldsymbol{U}_\eta^g \times \boldsymbol{U}_\zeta^g\right\|_{m_1,m_2},\end{aligned}$$

where $(\boldsymbol{F}_i)_{m_1,m_2}^\rho$ is the density flux in normal direction $(\boldsymbol{n}_i)_{m_1,m_2}$

$$(\boldsymbol{F}_i)_{m_1,m_2}^\rho = -U_{m_1,m_2}^g \cdot (\boldsymbol{n}_i)_{m_1,m_2}$$

and the normal directions are given by

$$\begin{aligned}(\boldsymbol{n}_1)_{m_1,m_2} &= \left(\boldsymbol{U}_\eta^g \times \boldsymbol{X}_\zeta\right)/\left\|\boldsymbol{U}_\eta^g \times \boldsymbol{X}_\zeta\right\|_{m_1,m_2},\\ (\boldsymbol{n}_2)_{m_1,m_2} &= \left(\boldsymbol{X}_\eta \times \boldsymbol{U}_\zeta^g\right)/\left\|\boldsymbol{X}_\eta \times \boldsymbol{U}_\zeta^g\right\|_{m_1,m_2},\\ (\boldsymbol{n}_3)_{m_1,m_2} &= \left(\boldsymbol{U}_\eta^g \times \boldsymbol{U}_\zeta^g\right)/\left\|\boldsymbol{U}_\eta^g \times \boldsymbol{U}_\zeta^g\right\|_{m_1,m_2}.\end{aligned}$$

The local coordinate can be obtained by the similar procedure of orthogonalization Eq. (7). Thus, the integrated extra flux at the quadrature point can be written as

$$\begin{aligned}\mathbb{V}_{m_1,m_2}(t) =\Big(\frac{1}{2}\Big(&\boldsymbol{F}_1\big\|\boldsymbol{U}^g_\eta \times \boldsymbol{X}_\zeta\big\| \\ &+ \boldsymbol{F}_2\big\|\boldsymbol{X}_\eta \times \boldsymbol{U}^g_\zeta\big\|\Big)(t-t_n)^2 + \frac{1}{3}\boldsymbol{F}_3\big\|\boldsymbol{U}^g_\eta \times \boldsymbol{U}^g_\zeta\big\|(t-t_n)^3\Big)_{m_1,m_2},\end{aligned}$$

where $(\boldsymbol{F}_1)_{m_1,m_2}$, $(\boldsymbol{F}_2)_{m_1,m_2}$, $(\boldsymbol{F}_3)_{m_1,m_2}$ are the fluxes in the local coordinate related to $(\boldsymbol{n}_i)_{m_1,m_2}$. The total numerical flux $\mathbb{F}_p(t)$ in Eq. (9) can be modified as

$$\mathbb{F}_p(t) = \sum_{m_1,m_2=1}^{2} \omega_{m_1,m_2}\Big(\boldsymbol{F}_{m_1,m_2}(Q,t)\,\big\|\boldsymbol{X}_\eta \times \boldsymbol{X}_\zeta\big\|_{m_1,m_2} + \partial_t \mathbb{V}_{m1,m2}(t)\Big)\Delta\eta\Delta\zeta.$$

To implement the two-stage fourth-order method, the similar procedures given by Eq. (13) and Eq. (14) are used. Both the order of accuracy and geometric conservation law can be preserved.

3.3 Spatial Reconstruction

In the classical WENO schemes [20,39], the high-order accuracy is achieved by the non-linear combination of lower order polynomials. For example, to achieve a third-order accuracy, a quadratic polynomial $P^2(\boldsymbol{x})$ and some linear polynomials $P^1_j(\boldsymbol{x}), j = 1, ..., M$ are constructed based on the candidate stencils. The linear weights γ_j are determined according to the following relation

$$P^2(\boldsymbol{x}_G) = \sum_{j=1}^{M} \gamma_j P^1_j(\boldsymbol{x}_G),$$

where M is the number of candidate linear polynomials and $\boldsymbol{x}_G$ is the Gaussian quadrature point. Linear weights can be obtained by solving a linear system, and the non-linear weights are introduced to deal with the discontinuities. However, the very large linear weights and negative weights appear with the lower mesh quality. In order to improve the robustness of WENO schemes, an optimization approach is given to deal with the very large linear weights [48]. With the two-stage temporal discretization and improved WENO scheme, the high-order ALE gas-kinetic scheme works well for two-dimensional computations, especially for the flows with strong discontinuities [38].

However, the linear weights need to be updated every time step when the mesh is moving, and the efficiency will be reduced greatly for three-dimensional ALE computations. Due to the distorted meshes, the robustness of WENO reconstruction is also reduced. To overcome the drawbacks above, the newly developed third-order WENO scheme [37] is used for spatial reconstruction during the moving-mesh procedure, in which the ideas of simple WENO reconstruction [49–51] and central WENO reconstruction [25,26] are adopted. With the reconstructed quadratic polynomial from large stencil and linear polynomials from sub-stencils, the linear weights are chosen as positive numbers with the requirement that their sum equals one and be independent of local mesh topology. With the construction of non-linear weights, the accuracy can be achieved. More importantly, the robustness is as well as the classical one-dimensional WENO scheme.

For the third-order WENO scheme, a quadratic polynomial and several polynomials are needed to achieve the expected accuracy. In this section, we consider the structured mesh for

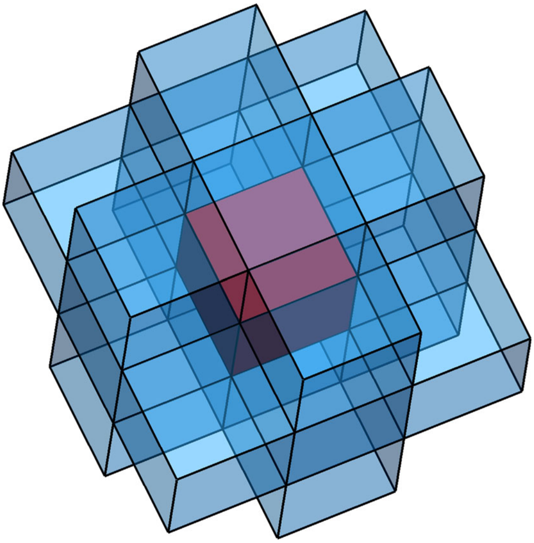


Fig. 1 Stencils of cell Ω_{ijk} (red cubic) for three-dimensional reconstruction

simplicity. To construct the quadratic polynomial, a large stencil shown in Fig. 1 for the call Ω_{ijk} is selected and the index of elements is rearranged as follows

$$\begin{aligned} S &= \{\Omega_{i+i_0, j+j_0, k+k_0},\ i_0, j_0, k_0 = -1, 0, 1,\ i_0 \cdot j_0 \cdot k_0 \neq \pm 1\} \\ &= \{\Omega_n, n = 0, ..., 18\}. \end{aligned}$$

where $\Omega_0 = \Omega_{ijk}$. With the large stencil, a quadrature polynomial is constructed

$$P_0(\boldsymbol{x}) = Q_{ijk} + \sum_{|\boldsymbol{n}|=1}^{2} a_n p_n(\boldsymbol{x}),$$

where Q_{ijk} is the cell averaged conservative variables over Ω_{ijk}, $\boldsymbol{n} = (n_1, n_2, n_3)$, $|\boldsymbol{n}| = n_1 + n_2 + n_3$ and

$$p_n(\boldsymbol{x}) = x^{n_1} y^{n_2} z^{n_3} - \frac{1}{\left|\Omega_{ijk}\right|} \iiint_{\Omega_{ijk}} x^{n_1} y^{n_2} z^{n_3} \mathrm{d}V.$$

For the linear polynomials, too many sub-stencils are used in the previous paper [37], and another type of sub-stencils are given as follows

$$\begin{aligned} S_1' &= \{\Omega_{ijk}, \Omega_{i+1,j,k}, \Omega_{i,j+1,k}, \Omega_{i,j,k+1}\}, & S_5' &= \{\Omega_{ijk}, \Omega_{i+1,j,k}, \Omega_{i,j+1,k}, \Omega_{i,j,k-1}\}, \\ S_2' &= \{\Omega_{ijk}, \Omega_{i-1,j,k}, \Omega_{i,j+1,k}, \Omega_{i,j,k+1}\}, & S_6' &= \{\Omega_{ijk}, \Omega_{i-1,j,k}, \Omega_{i,j+1,k}, \Omega_{i,j,k-1}\}, \\ S_3' &= \{\Omega_{ijk}, \Omega_{i+1,j,k}, \Omega_{i,j-1,k}, \Omega_{i,j,k+1}\}, & S_7' &= \{\Omega_{ijk}, \Omega_{i+1,j,k}, \Omega_{i,j-1,k}, \Omega_{i,j,k-1}\}, \\ S_4' &= \{\Omega_{ijk}, \Omega_{i-1,j,k}, \Omega_{i,j+1,k}, \Omega_{i,j,k+1}\}, & S_8' &= \{\Omega_{ijk}, \Omega_{i-1,j,k}, \Omega_{i,j+1,k}, \Omega_{i,j,k-1}\}. \end{aligned}$$

For simplicity, the index of elements of the sub-stencils is also rearranged as follows

$$S_m' = \{\Omega_{n'}^m, n' = 0, ..., 4\}.$$

The linear polynomials are constructed on candidate sub-stencils

$$P_m(\boldsymbol{x}) = Q_{ijk} + \sum_{|n|=1} b_n^m p_n(\boldsymbol{x}),$$

where $m = 1, ...8$. With the following constrains

$$Q_n = \frac{1}{|\Omega_n|} \iiint_{\Omega_n} P_0(\boldsymbol{x}) \mathrm{d}V,$$
$$Q_{n'}^m = \frac{1}{|\Omega_n|} \iiint_{\Omega_{n'}^m} P_m(\boldsymbol{x}) \mathrm{d}V,$$

the coefficients of $P_0(\boldsymbol{x})$ and $P_m(\boldsymbol{x}), m = 1, ..., 8$ can be fully determined. Therefore, the reconstructed point-value of the conservative variables $Q(\boldsymbol{x}_G)$ at Gaussian quadrature point can be given by the following non-linear combination

$$Q(\boldsymbol{x}_G) = \overline{\omega}_0 \left(\frac{1}{\gamma_0} P_0(\boldsymbol{x}_G) - \sum_{m=1}^{8} \frac{\gamma_m}{\gamma_0} P_m(\boldsymbol{x}_G) \right) + \sum_{m=1}^{8} \overline{\omega}_m P_m(\boldsymbol{x}_G). \tag{15}$$

The non-linear weights ω_m and normalized non-linear weights $\overline{\omega}_m$ are defined as

$$\overline{\omega}_m = \frac{\omega_m}{\sum_{m=0}^{8} \omega_m}, \ \omega_m = \gamma_m \Big[1 + \Big(\frac{\tau}{\beta_m + \epsilon} \Big) \Big],$$

where ϵ is a small positive number and the parameter τ is chosen as

$$\tau = \sum_{m=1}^{8} \left(\frac{|\beta_0 - \beta_m|}{8} \right).$$

The smooth indicator β_m is defined as

$$\beta_m = \sum_{|l|=1}^{r_m} \big| \Omega_{ijk} \big|^{\frac{2|l|}{3} - 1} \int_{\Omega_{ijk}} \left(\frac{\partial^l P_m}{\partial_x^{l_1} \partial_y^{l_2} \partial_z^{l_3}} (x, y, z) \right)^2 \mathrm{d}V,$$

where $r_0 = 2$ and $r_m = 1, m \neq 0$. It can be proved that Eq. (15) ensures the third order of accuracy for the current scheme. With the reconstructed polynomial, the spatial derivatives at Gaussian quadrature point, which will be used in the gas-kinetic solver, can be given as follows

$$\partial_x Q(\boldsymbol{x}_G) = \overline{\omega}_0 \Big(\frac{1}{\gamma_0} \partial_x P_0(\boldsymbol{x}_G) - \sum_{m=1}^{8} \frac{\gamma_m}{\gamma_0} \partial_x P_m(\boldsymbol{x}_G) \Big) + \sum_{m=1}^{8} \overline{\omega}_m \partial_x P_m(\boldsymbol{x}_G),$$
$$\partial_y Q(\boldsymbol{x}_G) = \overline{\omega}_0 \Big(\frac{1}{\gamma_0} \partial_y P_0(\boldsymbol{x}_G) - \sum_{m=1}^{8} \frac{\gamma_m}{\gamma_0} \partial_y P_m(\boldsymbol{x}_G) \Big) + \sum_{m=1}^{8} \overline{\omega}_m \partial_y P_m(\boldsymbol{x}_G),$$
$$\partial_z Q(\boldsymbol{x}_G) = \overline{\omega}_0 \Big(\frac{1}{\gamma_0} \partial_z P_0(\boldsymbol{x}_G) - \sum_{m=1}^{8} \frac{\gamma_m}{\gamma_0} \partial_z P_m(\boldsymbol{x}_G) \Big) + \sum_{m=1}^{8} \overline{\omega}_m \partial_z P_m(\boldsymbol{x}_G).$$

Thus, the spatial reconstruction is fully given. With the reconstructed variables, the gas distribution function can be determined completely.

In order to eliminate the spurious oscillation and improve the stability, the reconstruction can be performed for the characteristic variables in local moving coordinate for each Gaussian quadrature point. The characteristic variable is defined as $\omega = R^{-1} Q$, where Q is variable in the local moving coordinate with the grid velocity taken into account, and where R is the right eigenmatrix of Jacobian matrix $(\partial F / \partial Q)_G$ at Gaussian quadrature point. With the reconstructed variable, the conservative variables can be obtained by the inverse projection. To

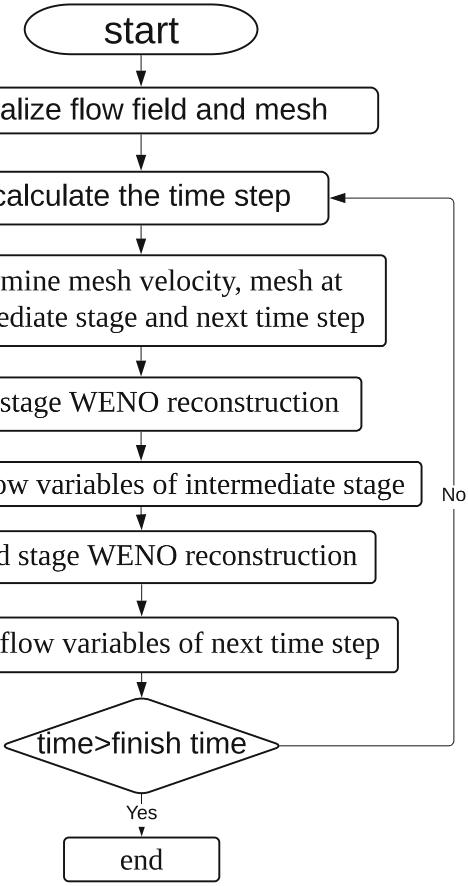


Fig. 2 Flow chart of high-order ALE gas-kinetic scheme

improve the robustness, a simple limiting procedure is used. For the reconstructed variables $P_m(\boldsymbol{x}_G), m = 0, ..., 8$ from quadrature and linear polynomials, if any one value of the densities $\rho_m(\boldsymbol{x}_G)$ and pressures $p_m(\boldsymbol{x}_G), m = 0, ..., 8$ become negative, the derivatives are set as zero and first-order reconstruction is adopted (Fig. 2).

3.4 Mesh Velocity

In the scheme above, the grid velocity $\boldsymbol{U}^g$ can be arbitrary for the control volume. The strategies of mesh velocity are given and the different methods are chosen according to the requirement of numerical cases.

1. The mesh velocity can be specified directly. This is the simplest type of mesh velocity and is mainly adopted in the accuracy tests in this paper.
2. The mesh velocity can be determined by the cell centered Lagrangian nodal solver [31], which is a three-dimensional extension of [29,30]. Denote $C(p)$ is the set of control volumes that share the common vertex p, $F_P(cp)$ is the set of faces of cell c that share the common vertex p, S_f and $\boldsymbol{N}_f^c$ are the area and unit outward normal direction of face

f, the grid velocity $\boldsymbol{U}_p$ can be given by the geometrical variables and cell averaged flow variables

$$\boldsymbol{U}_p^g = \mathbb{M}_p^{-1} \sum_{c \in C(p), f \in F_P(cp)} \Big[S_f p_c \boldsymbol{N}_f^c + \mathbb{M}_{pcf} \boldsymbol{U}_c \Big],$$

where $\rho_c, \boldsymbol{U}_c, p_c, a_c$ is density, fluid velocity, pressure, sound speed and

$$\mathbb{M}_p = \sum_{c \in C(p), f \in F_P(cp)} \mathbb{M}_{pcf} = \sum_{c \in C(p), f \in F_P(cp)} S_f \rho_c a_c (\boldsymbol{N}_f^c \otimes \boldsymbol{N}_f^c).$$

The matrix $\mathbb{M}_p$ is symmetric positive definite. Therefore, the system will always admit a unique solution, and the coordinate of each point can be given by

$$\boldsymbol{x}_{ijk}^{n+1} = \boldsymbol{x}_{ijk}^n + \boldsymbol{U}_{ijk}^g \Delta t.$$

3. The mesh velocity can be determined by the variational approach [40], and the corresponding Euler-Lagrange equations can be obtained from

$$\begin{aligned} (\omega x_\xi)_\xi + (\omega x_\eta)_\eta + (\omega x_\zeta)_\zeta &= 0, \\ (\omega y_\xi)_\xi + (\omega y_\eta)_\eta + (\omega y_\zeta)_\zeta &= 0, \\ (\omega z_\xi)_\xi + (\omega z_\eta)_\eta + (\omega z_\zeta)_\zeta &= 0, \end{aligned}$$

where (x, y, z) and (ξ, η, ζ) denote the physical and computational coordinates, ω is the monitor function and can be chosen as a function of the flow variables, such as the density, velocity, pressure, or their gradients. In the numerical tests, without no special statement, the monitor function takes the form

$$\omega = \sqrt{1 + \alpha |\nabla \rho|^2}.$$

A second-order central difference scheme is used to discretize the Euler-Lagrange equations. The Jacobian iterative method is used to solve the discretized equations with boundary condition, and the coordinate of $\boldsymbol{x}_{ijk}^{n+1}$ can be obtained. Actually, we do not need the converged solutions, only four or five iterative precess is enough to provide the local refined mesh. Thus, the mesh distribution can be directly generated and the grid velocity is obtained by

$$\boldsymbol{U}_{ijk}^g = \frac{\boldsymbol{x}_{ijk}^{n+1} - \boldsymbol{x}_{ijk}^n}{\Delta t}.$$

More details can be found in [40].

With the Lagrangian velocity and adaption velocity, the meshes become distorted and computation might break down as shown in Fig. 11. Therefore, the meshes need to be smoothed by the following procedure

$$\begin{aligned} \widetilde{\boldsymbol{x}}_{ijk}^{n+1} = (4\boldsymbol{x}_{ijk}^{n+1} &+ \boldsymbol{x}_{i-1,j,k}^{n+1} + \boldsymbol{x}_{i+1,j,k}^{n+1} + \boldsymbol{x}_{i,j-1,k}^{n+1} \\ &+ \boldsymbol{x}_{i,j+1,k}^{n+1} + \boldsymbol{x}_{i,j,k-1}^{n+1} + \boldsymbol{x}_{i,j,k+1}^{n+1})/10, \end{aligned}$$

where $\widetilde{\boldsymbol{x}}_{ijk}^{n+1}$ is the coordinate of smoothed mesh. With the new mesh point, the new grid velocity is replaced by

$$\boldsymbol{U}_{ijk}^g = \frac{\widetilde{\boldsymbol{x}}_{ijk}^{n+1} - \boldsymbol{x}_{ijk}^n}{\Delta t}.$$

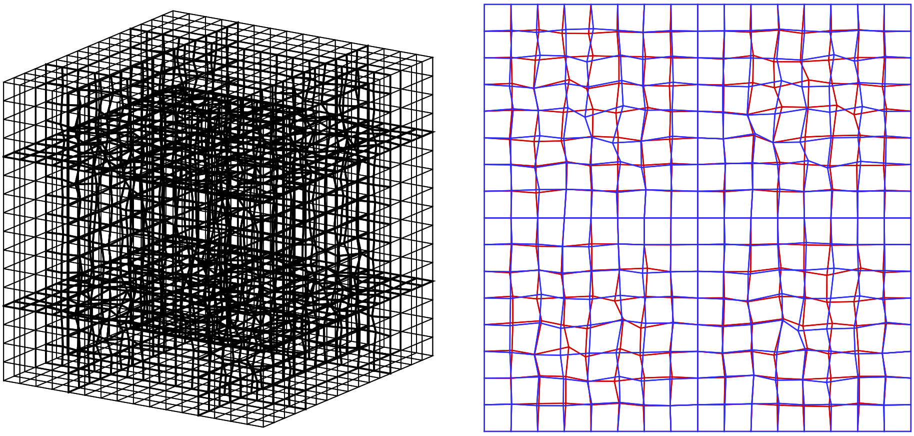


Fig. 3 Accuracy test: moving-mesh Type 2 at $t = 0.5$ with $16 \times 16 \times 16$ cells

Table 1 Accuracy test: the advection of density perturbation with uniform stationary meshes

mesh	L^1 error	Order	L^2 error	Order
$16 \times 16 \times 16$	1.4612E-01		5.7820E-02	
$32 \times 32 \times 32$	2.0241E-02	2.8517	7.9355E-03	2.8651
$64 \times 64 \times 64$	2.5712E-03	2.9768	1.0083E-03	2.9763
$128 \times 128 \times 128$	3.2240E-04	2.9955	1.2633E-04	2.9966

This procedure is conducted for some certain steps, which can be considered as a simple rezoning stage in the classical ALE methods. In all the following tests, the mesh is smoothed every 20 steps to avoid the tangling of the mesh for the Lagrangian velocity.

The mesh velocities at Gaussian quadrature points can be obtained by the bilinear interpolation of mesh velocity at the grid points. For each Gaussian quadrature point, the constant mesh velocity is also used in the time interval $[t_n, t_n + \Delta t]$. Thus, we have fully given the numerical scheme.

4 Numerical Tests

In this section, numerical tests for the inviscid flows will be presented to validate the performance of current scheme. The collision time τ for the flows with discontinuities takes

$$\tau = \epsilon \Delta t + C \left| \frac{p_l - p_r}{p_l + p_r} \right| \Delta t,$$

where $\epsilon = 0.01$ and $C = 1$, p_l and p_r denote the pressure on the left and right sides of the cell interface. For the inviscid smooth flows and the gas distribution function reduces to

$$f\left(\boldsymbol{x}_{m_1,m_2}, t, \boldsymbol{u}, \varsigma\right) = g_0\left(1 + \bar{A}t\right).$$

The computation can be simplified greatly. Without special statement, the specific heat ratio $\gamma = 1.4$ and the CFL number $CFL = 0.35$ are used in the computation.

Table 2 Accuracy test: the advection of density perturbation with mesh velocities Type 1- Type 4

	L^1 error	Order	L^2 error	Order
Type 1				
16 × 16 × 16	1.6475E-01		6.4872E-02	
32 × 32 × 32	2.3757E-02	2.7938	9.4385E-03	2.7809
64 × 64 × 64	3.0657E-03	2.9540	1.2195E-03	2.9521
128 × 128 × 128	3.8585E-04	2.9901	1.5341E-04	2.9908
Type 2				
16 × 16 × 16	1.4708E-02		5.8174E-02	
32 × 32 × 32	2.0355E-02	2.8531	7.9957E-03	2.8630
64 × 64 × 64	2.5906E-03	2.9740	1.0163E-03	2.9758
128 × 128 × 128	3.2667E-04	2.9873	1.2811E-04	2.9878
Type 3				
16 × 16 × 16	1.5108E-01		5.9875E-02	
32 × 32 × 32	2.0965E-02	2.8493	8.3161E-03	2.8479
64 × 64 × 64	2.6675E-03	2.9744	1.0577E-03	2.9748
128 × 128 × 128	3.3443E-04	2.9957	1.3264E-04	2.9953
Type 4				
16 × 16 × 16	1.5521E-01		6.1490E-02	
32 × 32 × 32	2.1794E-02	2.8322	8.6659E-03	2.8269
64 × 64 × 64	2.7846E-03	2.9684	1.1074E-03	2.9680
128 × 128 × 128	3.4951E-04	2.9940	1.3901E-04	2.9939

4.1 Accuracy Test

The advection of density perturbation for three-dimensional flows is presented to test the order of accuracy. For this case, the physical domain is [0, 2] × [0, 2] × [0, 2] and the initial condition is set as follows

$$\begin{aligned}&\rho_0(x,y,z)=1+0.2\sin(\pi(x+y+z)),\ \ p_0(x,y,z)=1,\\&U_0(x,y,z)=1,\ \ V_0(x,y,z)=1,\ \ W_0(x,y,z)=1.\end{aligned}$$

The periodic boundary conditions are applied at boundaries, and the exact solution is

$$\begin{aligned}&\rho(x,y,z,t)=1+0.2\sin(\pi(x+y+z-t)),\ \ p(x,y,z,t)=1,\\&U(x,y,z,t)=1,\ \ V(x,y,z,t)=1,\ \ W(x,y,z,t)=1.\end{aligned}$$

As reference, the uniform stationary meshes with $\Delta x=\Delta y=\Delta z=2/N$ are tested. The L^1 and L^2 errors and orders of accuracy at $t=2$ are presented in Table 1. To validate the order of accuracy with moving-meshes, the following four time dependent moving meshes are considered

$$\text{Type 1}:\begin{cases}x_i=\xi_i+0.1\sin(\pi\xi_i)\sin(\pi\eta_j)\sin(\pi\zeta_k)\sin\pi t,\\y_j=\eta_j+0.1\sin(\pi\xi_i)\sin(\pi\eta_j)\sin(\pi\zeta_k)\sin\pi t,\\z_k=\zeta_k+0.1\sin(\pi\xi_i)\sin(\pi\eta_j)\sin(\pi\zeta_k)\sin\pi t,\end{cases}\tag{16}$$

Table 3 Isotropic vortex propagation: L^2 errors and orders for stationary meshes and moving-meshes

3D mesh	Stationary meshes L^2 error	 Order	Moving-meshes L^2 error	 Order
$16 \times 16 \times 3$	1.0602E-01		1.0656E-01	
$32 \times 32 \times 6$	1.8726E-02	2.5012	1.9073E-02	2.4820
$64 \times 64 \times 12$	2.7360E-03	2.7749	2.8362E-03	2.7495
$128 \times 128 \times 24$	3.5119E-04	2.9617	4.5136E-04	2.6516

$$\text{Type 2}: \begin{cases} x_i = \xi_i + \mathrm{R}_{ijk}\sin(\pi\xi_i)\sin(\pi\eta_j)\sin(\pi\zeta_k)\sin\pi t, \\ y_j = \eta_j + \mathrm{R}_{ijk}\sin(\pi\xi_i)\sin(\pi\eta_j)\sin(\pi\zeta_k)\sin\pi t, \\ z_k = \zeta_k + \mathrm{R}_{ijk}\sin(\pi\xi_i)\sin(\pi\eta_j)\sin(\pi\zeta_k)\sin\pi t, \end{cases} \tag{17}$$

$$\text{Type 3}: \begin{cases} x_i = \xi_i + 0.05\sin(\pi\xi_i)\sin\pi t, \\ y_j = \eta_j + 0.05\sin(\pi\eta_j)\sin\pi t, \\ z_k = \zeta_k + 0.05\sin(\pi\zeta_k)\sin\pi t, \end{cases} \tag{18}$$

$$\text{Type 4}: \begin{cases} x_i = \xi_i + 0.05(\sin(\pi\xi_i) + \sin(\pi\xi_i)\sin(\pi\eta_j)\sin(\pi\zeta_k))\sin\pi t, \\ y_j = \eta_j + 0.05(\sin(\pi\eta_j) + \sin(\pi\xi_i)\sin(\pi\eta_j)\sin(\pi\zeta_k))\sin\pi t, \\ z_k = \zeta_k + 0.05(\sin(\pi\zeta_k) + \sin(\pi\xi_i)\sin(\pi\eta_j)\sin(\pi\zeta_k))\sin\pi t, \end{cases} \tag{19}$$

where $(\xi, \eta, \zeta) \in [0, 2] \times [0, 2] \times [0, 2]$, (ξ_i, η_j, ζ_k) are given uniformly with $\Delta\xi = \Delta\eta = \Delta\zeta = 2/N$ and $R_{ijk} \in [-0.25\Delta\xi, 0.25\Delta\xi]$ is the randomly given number. The mesh with Type 2 velocity at $t = 0.5$ are given in Fig. 3 as example. The periodic boundary condition is imposed for the mesh. The L^1 and L^2 errors and orders of accuracy at $t = 2$ are presented in Table 2 from Type 1 to Type 4, where the expected orders of accuracy are achieved by the current scheme.

The second one is the isotropic vortex propagation problem. It is a non-linear case for accuracy test, and the three-dimensional code is run for the original two-dimensional problem. The mean flow is $(\rho, U, V, W, p) = (1, 1, 1, 0, 1)$, and an isotropic vortex is added to the mean flow, i.e., with perturbation in U, V and temperature $T = p/\rho$, and no perturbation in entropy $S = p/\rho^\gamma$. The perturbation is given by

$$(\delta U, \delta V, \delta W) = \frac{\epsilon}{2\pi} e^{\frac{(1-r^2)}{2}}(-y, x, 0), \quad \delta T = -\frac{(\gamma-1)\epsilon^2}{8\gamma\pi^2} e^{1-r^2}, \qquad \delta S = 0,$$

where $r^2 = x^2 + y^2$ and the vortex strength $\epsilon = 5$. The computational domain is $[0, 10] \times [0, 10] \times [0, 1]$, the periodic boundary conditions are imposed on the boundaries in all directions. The exact solution is the perturbation which propagates with the velocity $(1, 1, 0)$. For the accuracy with moving-mesh velocity, the following time dependent meshes are considered as follows

$$\begin{cases} x_i = \xi_i + 0.05\sin(\pi\xi_i)\sin(\pi\eta_j)\sin 0.2\pi t, \\ y_j = \eta_j + 0.05\sin(\pi\xi_i)\sin(\pi\eta_j)\sin 0.2\pi t, \\ z_k = \zeta_k, \end{cases}$$

where (ξ_i, η_j, ζ_k) are given uniformly with $\Delta\xi = \Delta\eta = \Delta\zeta = 10/N$. The mesh velocity can be directly given during a time step. The L^2 errors and orders at $t = 10$ are presented in Table 3. As reference, the errors and orders of accuracy with stationary meshes are also given

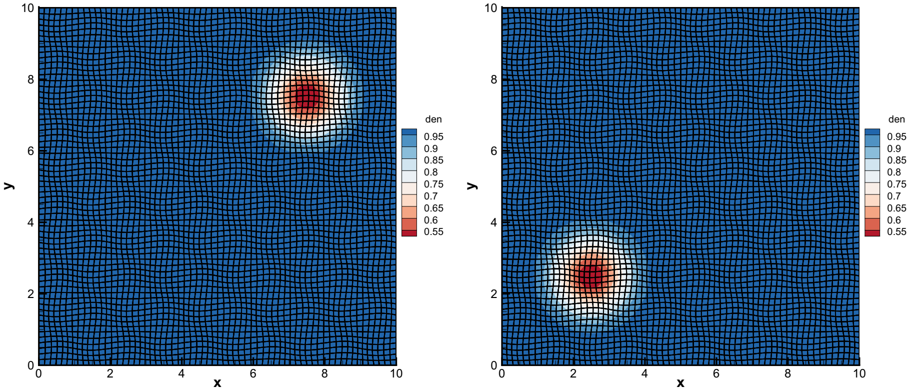


Fig. 4 Isotropic vortex propagation: the density and mesh distributions with 64 × 64 cells at $t = 2.5$ and 7.5

Table 4 Steady isotropic vortex: L^2 errors and orders at $t = 1$ and 2

	$t = 1$		$t = 2$	
3D mesh	L^2 error	Order	L^2 error	Order
16 × 16 × 3	5.7987E-02		8.8941E-02	
32 × 32 × 6	1.3800E-02	2.0709	2.0232E-02	2.1361
64 × 64 × 12	2.0308E-03	2.7645	3.4747E-03	2.5416
128 × 128 × 24	2.5678E-04	2.9834	5.2166E-04	2.7357

in Table 3. The expected accuracy is obtained with the moving-mesh procedure as well. The density and mesh distributions with 64 × 64 cells at $t = 2.5$ and 7.5 are given in Fig. 4, where the vortex is well preserved with the moving-mesh procedure.

To test the accuracy with Lagrangian velocity, the steady isotropic vortex with mean flow $(\rho, U, V, W, p) = (1, 0, 0, 0, 1)$ is tested. The identical isotropic vortex is added to the mean flow, and the same computational domain and initial meshes are used. With the periodic boundary condition, the exact solution is the steady vortex and grid pints move with the flow velocity, which is given by the Lagrangian nodal solver [31]. The L^2 errors and orders at $t = 1$ and 2 are presented in Table 4. The expected accuracy is also obtained even with the highly distorted meshes. The density and mesh distributions with 64 × 64 cells at $t = 1$, 2 and 3.5 are given in Fig. 5, where the vortex is preserved very well at $t = 3.5$ with the highly distorted mesh. With a few more steps, the computation will brake down.

4.2 Geometric Conservation Law

In this case, the geometric conservation law is tested, which is mainly about the maintenance of uniform flow passing through the moving-mesh. The initial condition is given as follows

$$\rho_0(x, y, z) = 1, \ \ p_0(x, y, z) = 1, \ \ U_0(x, y, z) = 1, \ \ V_0(x, y, z) = 1, \ \ W_0(x, y, z) = 1.$$

The above moment of computational mesh is used, and the periodic boundary conditions are adopted as well. The L^1 and L^2 errors at $t = 0.5$ are given in Table 5 from Type 1 to Type 4. The results show that the errors reduce to the machine zero, which implies the satisfaction of geometric conservation law. For the mesh velocity Type 1 and Type 2, $\mathbb{V}_{m1,m2}(t)$ is always

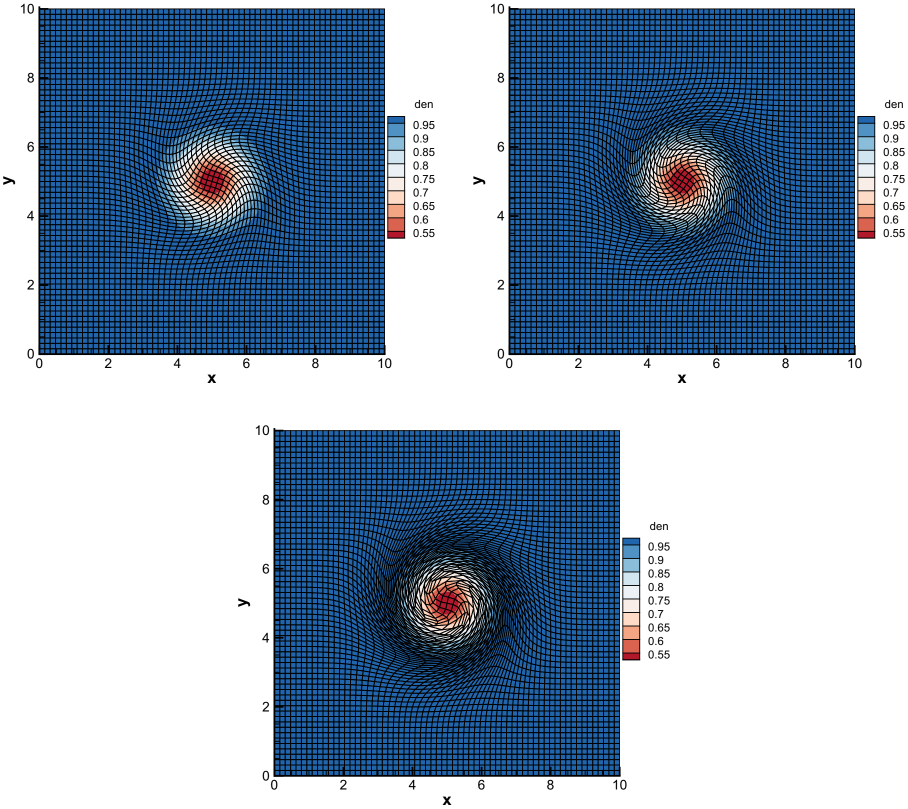


Fig. 5 Steady isotropic vortex: the density and mesh distributions with 64×64 cells at $t = 1$, 2 and 3.5

0, the geometric conservation law can be simply preserved. For the mesh velocity Type 3 and Type 4, the geometrical variation needs to be included in the numerical fluxes in order to preserve the geometric conservation law.

4.3 Riemann Problem

As an extension of the Sod problem, the spherically symmetric Sod problem is considered, and the initial conditions are given by

$$(\rho, U, V, W, p) = \begin{cases} (1, 0, 0, 0, 1), & 0 \leq \sqrt{x^2 + y^2 + z^2} \leq 0.5, \\ (0.125, 0, 0, 0, 0.1), & 0.5 < \sqrt{x^2 + y^2 + z^2} \leq 1. \end{cases}$$

The computational domain is $(x, y, z) \in [0, 1] \times [0, 1] \times [0, 1]$, and the symmetric boundary condition is imposed on the plane with $x = 0$, $y = 0$ and $z = 0$, and the non-reflection boundary condition is imposed on the plane with $x = 1$, $y = 1$ and $z = 1$. The exact solution of spherically symmetric problem can be given by the following one-dimensional system with geometric source terms

$$\frac{\partial Q}{\partial t} + \frac{\partial F(Q)}{\partial r} = S(Q),$$

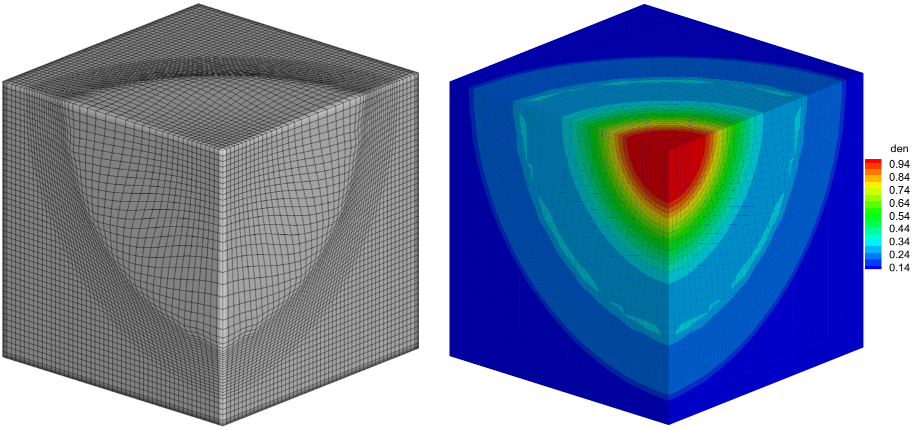


Fig. 6 Riemann problem: the 3D mesh and density distributions with Lagrangian velocity

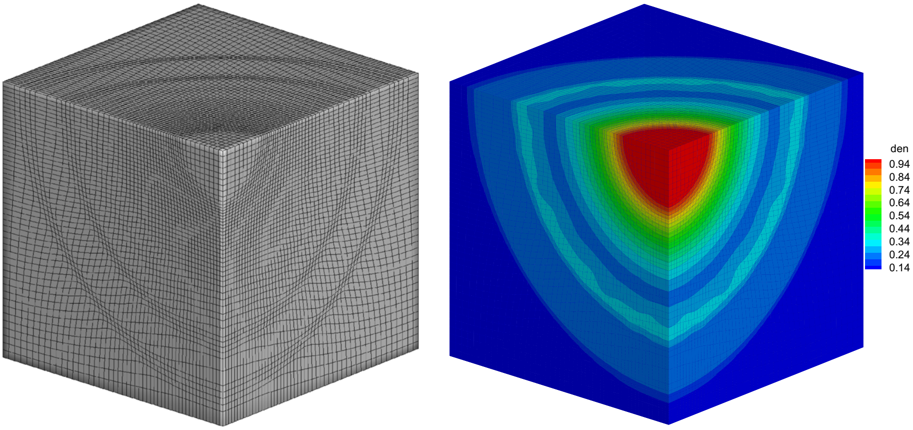


Fig. 7 Riemann problem: the 3D mesh and density distribution with adaptive velocity

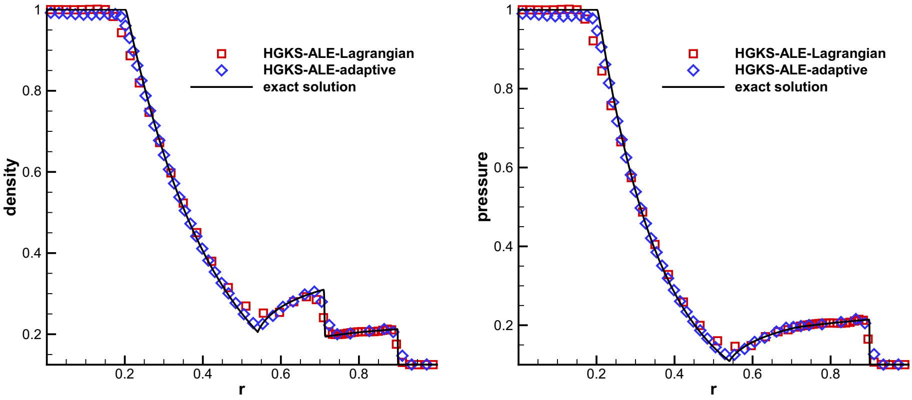


Fig. 8 Riemann problem: the density and pressure profiles with respect to radius r with Lagrangian and adaptive velocities

Table 5 Accuracy test: geometric conservation law for moving-mesh Type 1–Type 4

	L^1 error	L^2 error
Type 1		
$16 \times 16 \times 16$	1.1925E-14	5.4002E-15
$32 \times 32 \times 32$	3.1108E-14	1.4144E-14
$64 \times 64 \times 64$	7.7908E-14	3.6674E-14
$128 \times 128 \times 128$	1.8682E-13	8.9173E-14
Type 2		
$16 \times 16 \times 16$	1.1949E-14	5.4535E-15
$32 \times 32 \times 32$	2.7951E-14	1.2703E-14
$64 \times 64 \times 64$	6.2853E-14	2.8663E-14
$128 \times 128 \times 128$	1.3732E-13	6.2618E-14
Type 3		
$16 \times 16 \times 16$	2.9113E-14	1.2682E-14
$32 \times 32 \times 32$	8.2226E-14	3.6947E-14
$64 \times 64 \times 64$	1.9279E-13	8.7113E-14
$128 \times 128 \times 128$	5.9072E-13	2.6789E-13
Type 4		
$16 \times 16 \times 16$	2.0825E-14	9.4588E-15
$32 \times 32 \times 32$	7.7791E-14	7.2702E-14
$64 \times 64 \times 64$	7.2621E-13	5.4365E-13
$128 \times 128 \times 128$	2.0065E-12	1.1122E-12

where

$$Q = \begin{pmatrix} \rho \\ \rho U \\ \rho E \end{pmatrix}, F(Q) = \begin{pmatrix} \rho U \\ \rho U^2 + p \\ U(\rho E + p) \end{pmatrix}, S(Q) = -\frac{d-1}{r} \begin{pmatrix} \rho U \\ \rho U^2 \\ U(\rho E + p) \end{pmatrix}.$$

The radial direction is denoted by r, U is the radial velocity, d is the number of space dimensions. Initially, $50 \times 50 \times 50$ cells are equally distributed. The adaptive velocity and Lagrangian velocity are chosen as mesh velocities. For the adaptive procedure, the parameter α in the monitor function takes 1. The smooth procedure is used every 20 steps for the adaptive velocity and Lagrangian velocity. The three-dimensional mesh and density distributions at $t = 0.25$ are presented in Figs. 6 and 7, and the density and pressure profiles along $y = z = 0$ at $t = 0.25$ with respect to radius r are presented in Fig. 8. The numerical solutions agree well with the exact solutions. Due the local mesh adaptation, the discontinuities are well resolved by the current scheme.

4.4 Sedov Problem

This is a three-dimensional explosion problem to model blast wave from an energy deposited singular point, which is a standard benchmark problem for the Lagrangian method and ALE method. Due to the increased distortion undergone by the elements, the three-dimensional Sedov problem is more challenging than its two-dimensional counterpart. Similar with the two-dimensional case, the fluid is modeled by the ideal gas EOS with $\gamma = 1.4$. The initial density has a uniform unit distribution, and the pressure is 10^{-6} everywhere, except in the

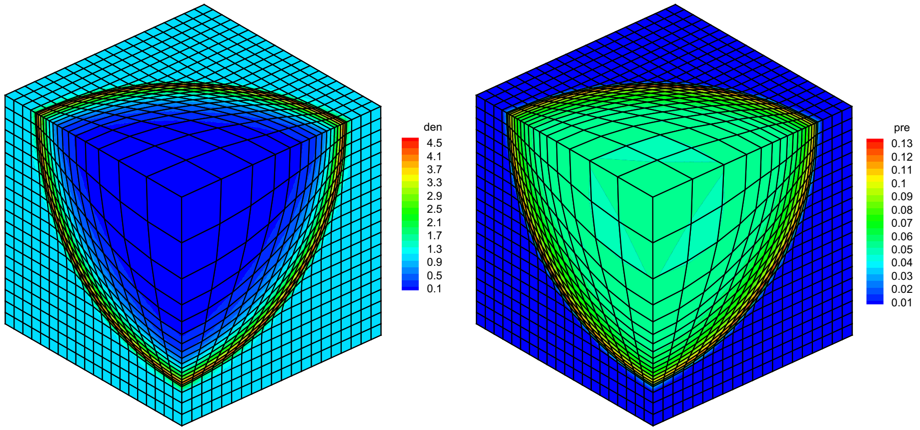


Fig. 9 Sedov problem: the three-dimensional density and pressure distributions at $t = 1$ with $20 \times 20 \times 20$ cells

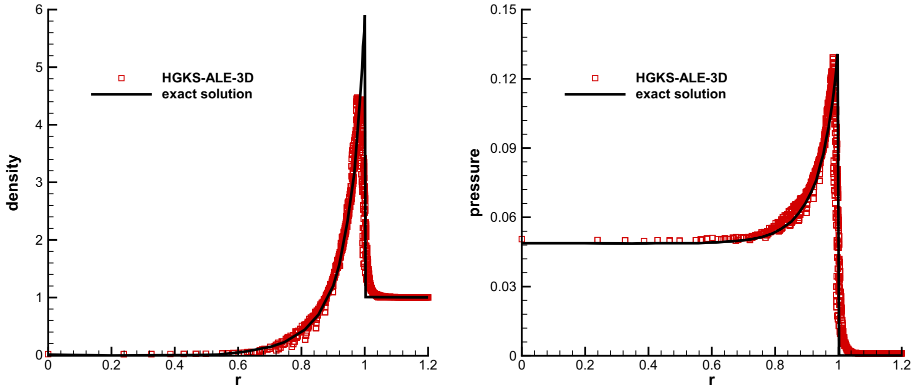
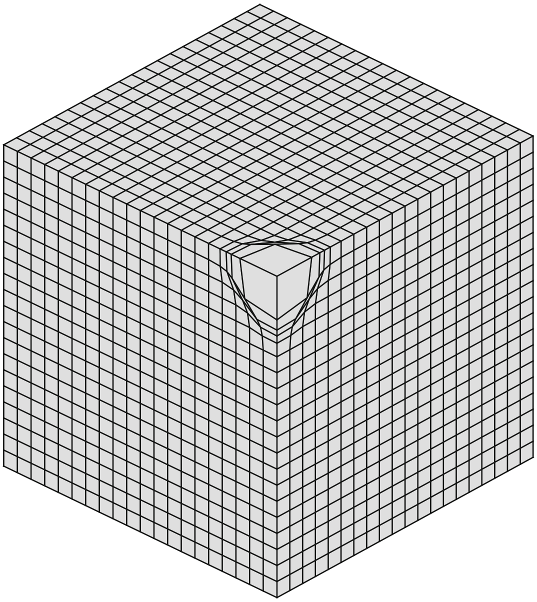


Fig. 10 Sedov problem: the density and pressure in all cells with respect to the radius r at $t = 1$ with $20 \times 20 \times 20$ cells

cell containing the origin. For this cell containing the origin, the pressure is defined as $p = (\gamma - 1)\varepsilon_0/V$, where $\varepsilon_0 = 0.106384$ is the total amount of released energy and V is the cell volume. The computation domain is $[0, 1.2] \times [0, 1.2] \times [0, 1.2]$ and the symmetric boundary condition is imposed on the plane with $x = 0$, $y = 0$ and $z = 0$, and the non-reflection boundary condition is imposed on the plane with $x = 1.2$, $y = 1.2$ and $z = 1.2$. Initially, $20 \times 20 \times 20$ cells are equally distributed. The solution consists of a diverging infinite strength shock wave whose front is located at radius $r = 1$ at $t = 1$ [23]. Due to the singularity at the origin, a small CFL number $CFL = 0.01$ is used. After 10 steps, a normal CFL number $CFL = 0.35$ is used. The Lagrangian velocity is chosen as mesh velocities and the smooth procedure is used every 20 time steps. The mesh distribution with pure Lagrangian velocity is given in Fig. 11 with $20 \times 20 \times 20$ cells at $t = 0.028$, where the mesh is distorted severely and the computation will soon break down. It shows the impact of mesh smooth procedure, i.e. ALE procedure. The three-dimensional density and pressure distributions and the density and pressure in all cells with respect to the radius r at $t = 1$ are given in Fig. 9 and Fig. 10. As reference, the density and pressure profiles long $y = z = 0$ at

Fig. 11 Sedov problem: mesh distribution with pure Lagrangian velocity with $20 \times 20 \times 20$ cells at $t = 0.028$



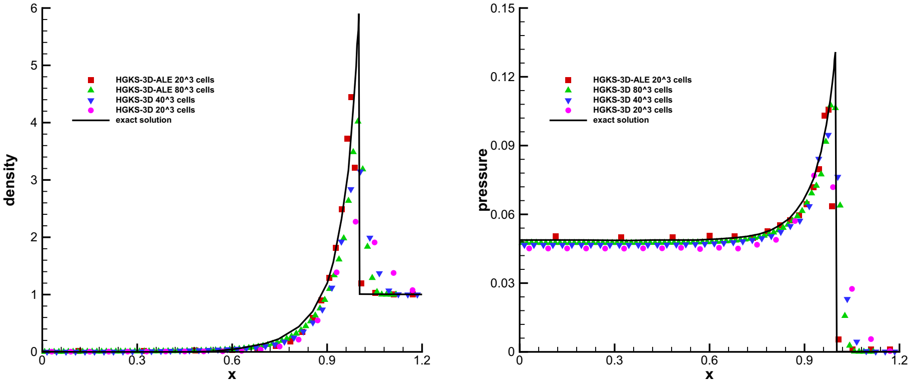


Fig. 12 Sedov problem: the density and pressure profiles long $y = z = 0$ at $t = 1$ for ALE computation with $20 \times 20 \times 20$ cells, and for stationary mesh computation with $20 \times 20 \times 20$, $40 \times 40 \times 40$ and $80 \times 80 \times 80$ cells

$t = 1$ with $20 \times 20 \times 20$, $40 \times 40 \times 40$ and $80 \times 80 \times 80$ stationary cells as well as the ALE results with $20 \times 20 \times 20$ cells are given in Fig. 12. Compared with the numerical results with stationary meshes, the ALE scheme resolves the peak of shock pretty well even with much less number of cells.

4.5 Noh Problem

This case is a three-dimensional extension of Noh problem [33], which is also a benchmark case for Lagrangian and ALE codes. The initial domain is $(x, y, z) \in [0, 1.2] \times [0, 1.2] \times [0, 1.2]$. The gas is initiated with $\rho = 1, e = 10^{-4}, \gamma = 5/3$, and the initial velocity

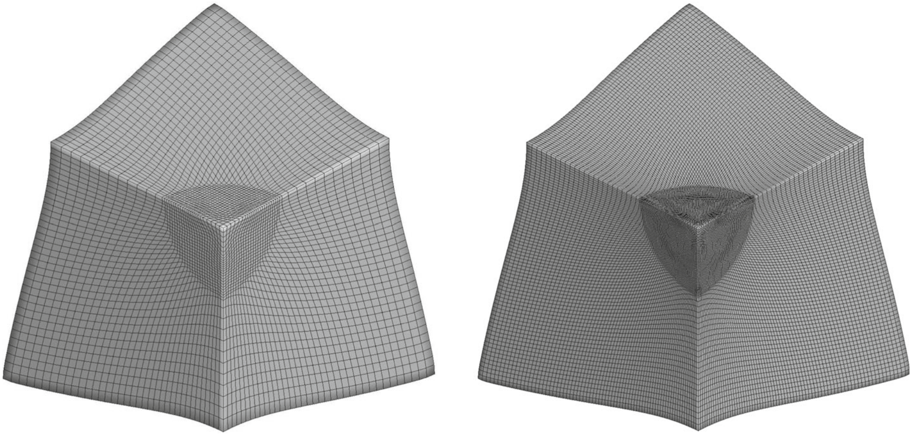


Fig. 13 Noh problem: the mesh distributions with $36 \times 36 \times 36$ and $72 \times 72 \times 72$ cells at $t = 0.6$

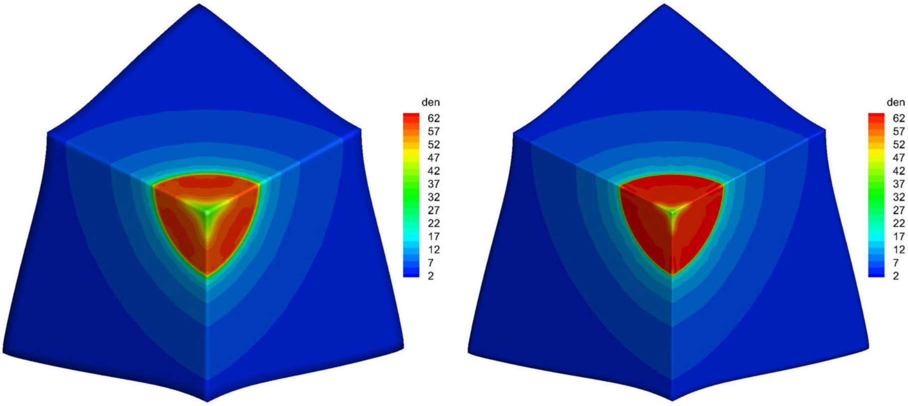


Fig. 14 Noh problem: the density distributions with $36 \times 36 \times 36$ and $72 \times 72 \times 72$ cells at $t = 0.6$

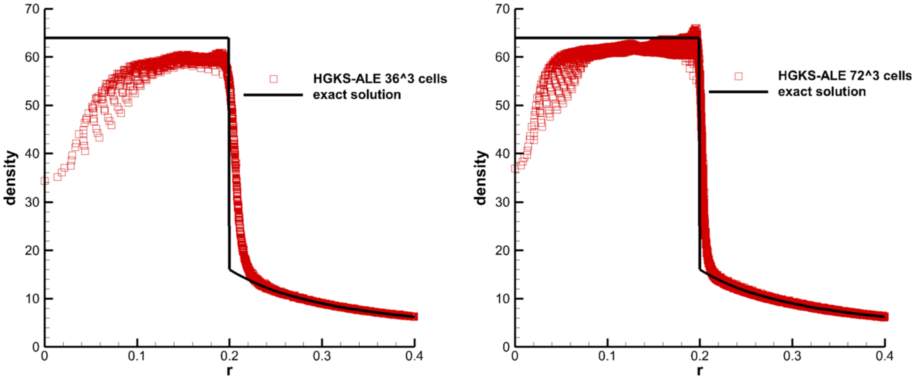


Fig. 15 Noh problem: the density distributions in all cells with $36 \times 36 \times 36$ and $72 \times 72 \times 72$ cells at $t = 0.6$

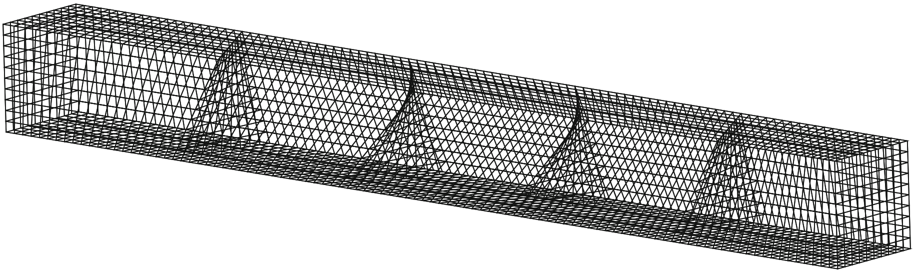


Fig. 16 Saltzman problem: the initial mesh

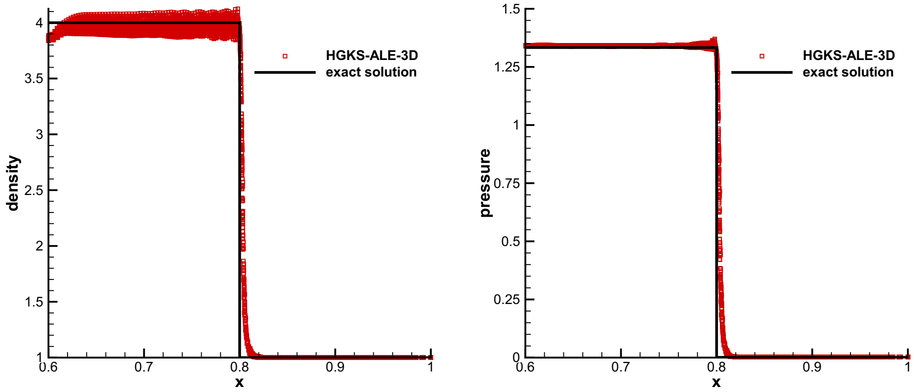


Fig. 17 Saltzman problem: the density and pressure distributions in all cells at $t = 0.6$

$\mathbf{U} = (U, V, W)$ is given by

$$U = \frac{-x}{\sqrt{x^2 + y^2 + z^2}}, V = \frac{-y}{\sqrt{x^2 + y^2 + z^2}}, W = \frac{-z}{\sqrt{x^2 + y^2 + z^2}}.$$

A spherical shock wave is generated at the origin and further propagates. The final time is chosen at $t = 0.6$. At this time, the exact solution is given by the position of shock at $r_s = 0.2$, and the post-shock density is $\rho = 64$ and $p = 64/3$, whereas the preshock density is given as a function of radius $r = \sqrt{x^2 + y^2 + z^2}$, i.e., $\rho = (1 + t/r)^2$ [34]. Initially, the uniform hexahedral cells are used to represent the initial domain. The symmetric boundary condition is imposed on the plane with $x = 0$, $y = 0$ and $z = 0$. For the other three boundaries, the non-reflective boundary condition is used. The Lagrangian velocity is chosen for mesh movement and the smooth procedure is used every 20 time steps. The three-dimensional mesh and density distributions at $t = 0.6$ are presented in Figs. 13 and 14 with $36 \times 36 \times 36$ and $72 \times 72 \times 72$ cells. The density distributions with respect to r are given in Fig. 15, which agree well with the exact solution.

4.6 Saltzman Problem

In this case, we consider the Saltzman problem, which is about the motion of a planar shock wave on a skewed Cartesian grid. This case is also a well known case to validate the robustness of Lagrangian and ALE codes. It was initially defined for two-dimensional flows, and we

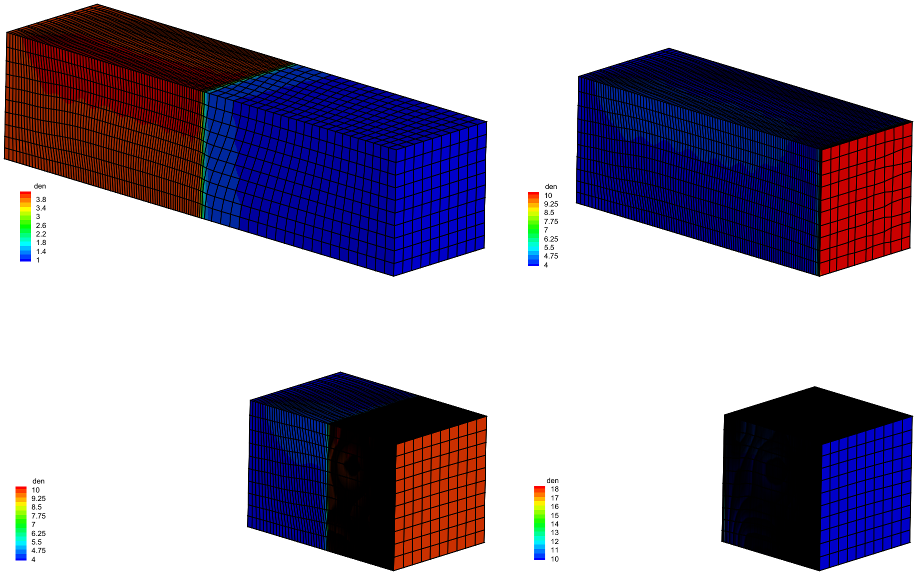


Fig. 18 Saltzman problem: the meshes at $t = 0.6, 0.75\ 0.85$ and 0.9

consider the three-dimensional extension of this case [12,16]. The computational domain is $[0, 1] \times [0, 0.1] \times [0, 0.1]$, and the initial mesh shown in Fig. 16 is given by the following transformation

$$\begin{aligned} x_{ijk} &= \xi_i + \left(0.5\eta_j + \zeta_k - 15\eta_j\zeta_k\right)\sin\left(\pi\xi_i\right), \\ y_{ijk} &= \eta_j, \\ z_{ijk} &= \zeta_k, \end{aligned}$$

where $(\xi, \eta, \zeta) \in [0, 1] \times [0, 0.1] \times [0, 0.1]$, and $\left(\xi_i, \eta_j, \zeta_k\right)$ are given uniformly with $\Delta\xi = \Delta\eta = \Delta\zeta = 0.01$. An ideal monatomic gas with $\rho = 1, e = 10^{-4}, \gamma = 5/3$ is filled in the box. The left-hand side wall acts as a piston with a constant velocity $U_{pis} = 1$, and other boundaries are reflective walls. As a consequence, a strong shock wave is generated from the left end. For this case, the purely Lagrangian method will cause mesh deformation, and the ALE method works well because of the properly modified mesh velocity. In the computation, the Lagrangian velocity is chosen as mesh velocities and the smooth procedure is used every 20 time steps. At time $t = 0.6$, the shock is expected to be located at $x = 0.8$, and the post shock solutions are $\rho = 4$ and $p = 1.333$. The density and pressure distributions with respect to x are given in Fig. 17. The numerical results agree well with the exact solutions. The shock wave will hit the face $x = 1$ at time $t = 0.75$ and be reflected by the solid wall. A second shock wave is produced which is traveling back towards to the piston. Finally, the computation stops $t = 0.9$ when the reflecting shock wave hits the piston. The mesh distributions at $t = 0.6, 0.75, 0.85$ and 0.9 are given in Fig. 18. These results demonstrate that our 3D scheme behaves similarly with the 2D calculation.

5 Conclusion

In the paper, a high-order gas-kinetic ALE scheme has been developed for three-dimensional flows. With the introduction of grid velocity in the BGK equation, the semi-discretized finite volume scheme is constructed in the ALE formulation. The newly developed third-order WENO reconstruction and two-stage fourth-order temporal discretization are used in the scheme. The main requirement for the scheme is to keep the order of accuracy and preserve the geometric conservation law. For the surface integral of numerical fluxes, a bilinear interpolation is used to parameterize the grid coordinate and its velocity. With the account of the variation of grid velocity within the interface surface, the geometric conservation law has been preserved. In the WENO reconstruction, a simple strategy of selecting stencils is adopted and the topologically independent linear weights are chosen for the improvement of the efficiency and robustness of the classical WENO scheme. Numerical examples from the smooth flows to the flows with strong discontinuities are presented to validate the accuracy, robustness, and geometrical conservation law. In these numerical examples, the grid velocity is determined by the variational approach for local mesh adaptation and by the Lagrangian nodal solver for tracking material interface. The extension of the current scheme to unstructured mesh will be considered in the future.

Acknowledgements The current research of Liang Pan is supported by National Natural Science Foundation of China (11701038, 11702030) and the Fundamental Research Funds for the Central Universities (2018NTST19). The work of Kun Xu is supported by National Natural Science Foundation of China (11772281, 91852114), Hong Kong research grant council (16206617), and the National Numerical Windtunnel project.

References

1. Anderson, R.W., Dobrev, V.A., Kolev, T.V., Rieben, R.N., Tomov, V.Z.: High-order multi-material ale hydrodynamics. SIAM J. Sci. Comput. **40**, B32–B58 (2018)
2. Barlow, A.J., Maire, P.H., Rider, W.J., Rieben, R.N., Shashkov, M.J.: Arbitrary Lagrangian Eulerian methods for modeling high-speed compressible multimaterial flows. J. Comput. Phys. **322**, 603–665 (2016)
3. Benson, D.J.: Computational methods in Lagrangian and Eulerian hydrocodes. Comput. Methods Appl. Mech. Eng. **99**, 235–394 (1992)
4. Bhatnagar, P.L., Gross, E.P., Krook, M.: A model for collision processes in gases I: small amplitude processes in charged and neutral one-component systems. Phys. Rev. **94**, 511–525 (1954)
5. Boscheri, W., Dumbser, M.: Arbitrary-Lagrangian-Eulerian one-step WENO finite volume schemes on unstructured triangular meshes. Commun. Comput. Phys. **14**, 1174–1206 (2013)
6. Boscheri, W., Dumbser, M.: A direct Arbitrary-Lagrangian-Eulerian ADER-WENO finite volume scheme on unstructured tetrahedral meshes for conservative and non-conservative hyperbolic systems in 3D. J. Comput. Phys. **275**, 484–523 (2014)
7. Boscheri, W., Balsara, D.S.: High order direct Arbitrary-Lagrangian-Eulerian (ALE) $P_N P_M$ schemes with WENO Adaptive-Order reconstruction on unstructured meshes. J. Comput. Phys. **398**, 108899 (2019)
8. Boscheri, W.: Dimarco, high order central WENO-Implicit-Explicit Runge Kutta schemes for the BGK model on general polygonal meshes. J. Comput. Phys. **422**, 109766 (2020)
9. Brackbill, J.U., Saltzman, J.S.: Adaptive zoning for singular problems in two dimensions. J. Comput. Phys. **46**, 342–368 (1982)
10. Burton, D.E.: Exact Conservation of Energy and Momentum in Staggered-grid Hydrodynamics with Arbitrary Connectivity. Advances in the Free Lagrange Method. Springer, New York (1990)
11. Burton, D.E.: Multidimensional discretization of conservation laws for unstructured polyhedral grids. Technical Report UCRL-JC-118306, Lawrence Livermore National Laboratory (1990)
12. Caramana, E.J., Rousculp, C.L., Burton, D.E.: A compatible, energy and symmetry preserving Lagrangian hydrodynamics algorithm in three-dimensional Cartesian geometry. J. Comput. Phys. **157**, 89–119 (2000)
13. Caramana, E.J., Shashkov, M.J., Whalen, P.P.: Formulations of artificial viscosity for multi-dimensional shock wave computations. J. Comput. Phys. **144**, 70–97 (1998)

14. Chapman, S., Cowling, T.G.: The Mathematical Theory of Non-Uniform Gases, 3rd edn. Cambridge University Press, Cambridge (1990)
15. Dobrev, V., Kolev, T., Rieben, R.: High-order curvilinear finite element methods for Lagrangian hydrodynamics. SIAM J. Sci. Comput. **34**, B606–B641 (2012)
16. Dukowicz, J.K., Meltz, B.: Vorticity errors in multidimensional lagrangian codes. J. Comput. Phys. **99**, 115–134 (1992)
17. Gaburro, E., Boscheri, W., Chiocchetti, S., Klingenberg, C., Springel, V., Dumbser, M.: High order direct Arbitrary-Lagrangian-Eulerian schemes on moving Voronoi meshes with topology changes. J. Comput. Phys. **407**, 109167 (2020)
18. Gaburro, E.: A unified framework for the solution of hyperbolic pde systems using high order direct Arbitrary-Lagrangian-Eulerian schemes on moving unstructured meshes with topology change. Arch. Comput. Methods Eng. (2020). https://doi.org/10.1007/s11831-020-09411-7
19. Hirt, C.W., Armsden, A.A., Cook, J.L.: An arbitrary Lagrangian Eulerian computing method for all flow speed. J. Comput. Phys. **135**, 203–216 (1997)
20. Hu, C., Shu, C.W.: Weighted essentially non-oscillatory schemes on triangular meshes. J. Comput. Phys. **150**, 97–127 (1999)
21. Hui, W.H., Li, P.Y., Li, Z.W.: A unified coordinated system for solving the two-dimensional Euler equations. J. Comput. Phys. **153**, 596–637 (1999)
22. Jin, C.Q., Xu, K.: A unified moving grid gas-kinetic method in Eulerian space for viscous flow computation. J. Comput. Phys. **222**, 155–175 (2007)
23. Kamm, J.R., Timmes, F.X.: On Efficient Generation of Numerically Robust Sedov Solutions, Technical Report LA-UR-07-2849, Los Alamos National Laboratory (2007)
24. Kucharik, M., Shashkov, M.: Conservative multi-material remap for staggered multi-material Arbitrary Lagrangian-Eulerian methods. J. Comput. Phys. **258**, 268–304 (2014)
25. Levy, D., Puppo, G., Russo, G.: Central WENO schemes for hyperbolic conservation laws. ESAIM Math. Model. Numer. Anal. **33**, 547–571 (1999)
26. Levy, D., Puppo, G., Russo, G.: Compact central WENO schemes for multidimensional conservation laws. SIAM J. Sci. Comput. **22**, 656–672 (2000)
27. Li, J.Q., Du, Z.F.: A two-stage fourth order time-accurate discretization for Lax-Wendroff type flow solvers I. hyperbolic conservation laws. SIAM J. Sci. Comput. **38**, 3046–3069 (2016)
28. Liu, X.D., Morgan, N.R., Burton, D.E.: A Lagrangian discontinuous Galerkin hydrodynamic method. Comput. Fluids **163**, 68–85 (2018)
29. Maire, P.H., Abgrall, R., Breil, J., Ovadia, J.: A cell-centered Lagrangian scheme for two-dimensional compressible flow problems. SIAM J. Sci. Comput. **29**, 1781–1824 (2007)
30. Maire, P.H.: A high-order cell-centered Lagrangian scheme for compressible fluid flows in two-dimensional cylindrical geometry. J. Comput. Phys. **228**, 6882–6915 (2009)
31. Maire, P.H., Nkonga, B.: Multi-scale Godunov-type method for cell-centered discrete Lagrangian hydrodynamics. J. Comput. Phys. **228**, 799–821 (2009)
32. Ni, G.X., Jiang, S., Xu, K.: Remapping-free ALE-type kinetic method for flow computations. J. Comput. Phys. **228**, 3154–3171 (2009)
33. Noh, W.F.: Errors for calculations of strong shocks using artificial viscosity and an artificial heat flux. J. Comput. Phys. **72**, 78–120 (1987)
34. Omang, M., Bøve, S., Trulsen, J.: SPH in spherical and cylindrical coordinates. J. Comput. Phys. **213**, 391–412 (2006)
35. Persson, P.O., Bonet, J., Peraire, J.: Discontinuous Galerkin solution of the Navier-Stokes equations on deformable domains. Comput. Methods Appl. Mech. Eng. **198**, 1585–1595 (2009)
36. Pan, L., Xu, K., Li, Q.B., Li, J.Q.: An efficient and accurate two-stage fourth-order gas-kinetic scheme for the Navier-Stokes equations. J. Comput. Phys. **326**, 197–221 (2016)
37. Pan, L., Xu, K.: High-order gas-kinetic scheme with three-dimensional WENO reconstruction for the Euler and Navier-Stokes solutions. Comput. Fluids **198**, 104401 (2020)
38. Pan, L., Zhao, F.X., Xu, K.: High-order ALE gas-kinetic scheme with WENO reconstruction. J. Comput. Phys. **417**, 109558 (2020)
39. Shi, J., Hu, C., Shu, C.W.: A Technique of Treating Negative Weights in WENO Schemes. J. Comput. Phys. **175**, 108–127 (2002)
40. Tang, H.Z., Tang, T.: Adaptive mesh methods for one-and two-dimensional hyperbolic conservation laws. SIAM J. Numer. Anal. **41**, 487–515 (2003)
41. Thomas, P., Lombard, C.: Geometric conservation law and its application to flow computations on moving grids. AIAA J. **17**, 1030–1037 (1979)
42. Toro, E.F.: Riemann Solvers and Numerical Methods for Fluid Dynamics, 3rd edn. Springer, New York (2009)

43. Titarev, V.A., Toro, E.F.: ADER: arbitrary high order Godunov approach. J. Sci. Comput. **17**, 609–618 (2002)
44. Titarev, V.A., Toro, E.F.: ADER schemes for three-dimensional nonlinear hyperbolic systems. J. Comput. Phys. **204**, 715–736 (2005)
45. Xu, K.: Direct Modeling for Computational Fluid Dynamics: Construction and Application of Unfied Gas Kinetic Schemes. World Scientific (2015)
46. Xu, K.: A gas-kinetic BGK scheme for the Navier-Stokes equations and its connection with artificial dissipation and Godunov method. J. Comput. Phys. **171**, 289–335 (2001)
47. Xu, X.H., Ni, G.X., Jiang, S.: A high-order moving mesh kinetic scheme based on WENO reconstruction for compressible flows on unstructured meshes. J. Sci. Comput. **57**, 278–299 (2013)
48. Zhao, F.X., Pan, L., Wang, S.H.: Weighted essentially non-oscillatory scheme on unstructured quadrilateral and triangular meshes for hyperbolic conservation laws. J. Comput. Phys. **374**, 605–624 (2018)
49. Zhu, J., Qiu, J.X.: A new fifth order finite difference weno scheme for solving hyperbolic conservation laws. J. Comput. Phys. **318**, 110–121 (2016)
50. Zhu, J., Qiu, J.X.: New finite volume weighted essentially non-oscillatory scheme on triangular meshes. SIAM J. Sci. Comput. **40**, 903–928 (2018)
51. Zhu, J., Shu, C.W.: A new type of third-order finite volume multi-resolution WENO schemes on tetrahedral meshes. J. Comput. Phys. **406**, 109212 (2020)