


Linear discrete velocity model-based lattice Boltzmann flux solver for simulating acoustic propagation in fluids

Ningyu Zhan , Rongqian Chen *, Qiaochu Song, and Yancheng You
School of Aerospace Engineering, Xiamen University, Xiamen 361005, China

 (Received 11 March 2022; accepted 26 May 2022; published 21 June 2022)

A linear lattice velocity model is developed, and a corresponding lattice Boltzmann flux solver (LBFS) is constructed based on it. This solver calculates the fluxes of the linearized Euler equations (LEEs), which are discretized by the finite volume method (FVM), and can simulate acoustic propagation in fluids. First, the expressions for the distribution function and the lattice velocity of the linear discrete velocity model are constructed using the moment relations in linear form. Second, based on Chapman-Enskog analysis and moment relations, the mesoscopic flux expression can be constructed by comparing the linear lattice Boltzmann equation and LEEs. Finally, the developed scheme is used to calculate the LEE flux term. The developed linear lattice velocity model-based LBFS has the following advantages: (i) It extends the lattice Boltzmann model-based flux solver from solving the macroscopic equations of fluid dynamics to solving linear equations; (ii) it also inherits the advantages of the Boltzmann model-type flux solver. The variables at the interface are calculated from the local solution of the lattice Boltzmann equation, making it more physical. In addition, least-squares-based finite differences and Gaussian integration are used for the FVM to discretize the LEE, making it a high-precision algorithm. Thus, the developed algorithm can accurately capture acoustic propagation in fluids and acoustic scattering in complex geometries. Several numerical cases for propagating acoustics in fluids are simulated to validate the accuracy and robustness of the present algorithm, and an accuracy test shows it approaches fourth-order accuracy.

DOI: [10.1103/PhysRevE.105.065303](https://doi.org/10.1103/PhysRevE.105.065303)

I. INTRODUCTION

Many practical engineering problems [1–3] involve acoustic waves passing through complex flow structures such as vortices and shear layers. For example, in an acoustic wind tunnel experiment, waves from the sound source must pass through the shear layer formed by the exit jet of the wind tunnel and the static surrounding air to reach the microphone array in the far field. Research has shown that refraction, reflection, and scattering occur when acoustic waves pass through shear layers, changing the amplitude, frequency, and phase of the signal received by the microphone and influencing the accuracy of aerodynamic noise measurement and sound source localization [4]. Therefore, the study of acoustic wave propagation characteristics in fluids is important.

Numerical simulation is essential for studying acoustic waves propagating in fluids, and researchers have applied computational fluid dynamics (CFD) technology to study aeroacoustics problems. The essence of the acoustic waves is the disturbance. Therefore, to simulate acoustic propagation is to simulate the disturbance propagating and evolving in fluids. At present, there are two main methods to simulate acoustic propagation in fluids using CFD. The first of these, direct numerical simulation (DNS) [5], is used to simulate the full flow field, including the disturbance. However, the disturbance is generally orders of magnitude smaller than

the base flow, so it is necessary to use a very fine grid and a small time step to accurately capture the acoustic waves, resulting in huge computational demands. The second option is to use the acoustic propagation equations. The variables are decomposed into the mean flow and disturbance parts and substituted into the governing fluid dynamics equations. The linear acoustic propagation equations [6–8] can then be derived by ignoring the nonlinear terms. The computational requirements of this approach are much smaller than for direct numerical simulation. In addition, the viscous effects of the fluids on the acoustics can generally be ignored, so the linearized Euler equations (LEEs) are used to simulate the disturbance propagation.

The generally used method for solving LEEs is the finite difference method (FDM). However, it is often performed in structured grids, which makes it suitable for simulation of simple geometries; otherwise, it needs to use techniques such as the coordinate transformation, multiblock grid, etc., and increase the complexity of the algorithm. The finite volume method [9] (FVM) is a common discrete method for solving the governing equations of fluid dynamics and can be used for unstructured grids. Thus, it is suitable for the simulation of flows in complex geometries. The calculation of fluxes at the interfaces between cells is a key requirement of the FVM. The traditional flux scheme often regards the interface as the Riemann discontinuity when calculating inviscid fluxes and obtains the fluxes at the interface by solving the Riemann problem [10,11]. However, the solution to the multidimensional Riemann problem is difficult to obtain [12], so most

*rqchen@xmu.edu.cn

schemes are based on one-dimensional Riemann problems to calculate the fluxes. As a result, it is necessary to approximate the components tangent to the interface. In addition, the viscous and inviscid fluxes should be calculated with different schemes in the traditional method, which increases the algorithmic complexity.

In recent years, lattice and continuous Boltzmann model-based flux solvers have been gradually developed, such as the lattice Boltzmann flux solver (LBFS) and simplified distribution function-based gas kinetic schemes proposed by Shu and co-workers [13,14], the unified gas kinetic scheme (UGKS) proposed by Xu and co-workers [15], and the discrete unified gas kinetic scheme proposed by Guo and co-workers [16]. These types of schemes have been applied widely [17–22]. From a mesoscopic perspective, the Boltzmann equation describes the flow field using the density distribution function of a particle. These Boltzmann model-based flux solvers can be conveniently extended to multiple dimensions so that no approximations are needed to calculate the variables tangential to the interface. Furthermore, the inviscid and viscous fluxes are coupled and calculated simultaneously, resulting in a more consistent algorithm. Schemes of this type, having advantages compared to traditional schemes, have been applied to various flow problems [23–27]. However, they start from the Boltzmann equation or lattice Boltzmann equation and use the Maxwell function [28] or simplified distribution function [29] for the equilibrium distribution function to derive the flux expressions of the Navier-Stokes or Euler equations, which is not suitable for LEE simulating acoustic propagation problems. Although Vergnault *et al.* [30] have derived the linear form of the lattice Boltzmann equation and corresponding linear Maxwell function, it is essentially a lattice Boltzmann method and does not construct the flux scheme. In addition, the truncated Maxwell function is used, making it suitable for incompressible flow without solving the energy equation. Therefore, there is a need to develop a compressible Boltzmann model-based flux solver for LEE.

In terms of the compressible Boltzmann equation, Qu *et al.* [31] have inversely derived the equilibrium distribution function based on moment relations and developed the D1Q4L2 and D2Q13L2 models, and Yang *et al.* have improved the lattice model and provided the corresponding D1Q4 and D1Q5 models [32,33]. They can be applied to compressible flows, and the form of the equilibrium distribution function is simpler than the Maxwell function. This model provides a good reference for deriving a flux solver for the LEE based on the compressible Boltzmann model. It is foreseeable that the linear lattice Boltzmann model and corresponding distribution function in linear form can be derived based on the moment relations from the linear lattice Boltzmann equation (LLBE), and the flux scheme for the LEE can be constructed. In the present work, a linear D1Q4 model is developed, and the corresponding flux scheme for the LEE is constructed. Specifically, from the LLBE, the moment relations to recover the LEE are derived based on Chapman-Enskog (CE) analysis, and these relations are solved by treating the equilibrium distribution function and lattice velocity as the unknowns. As a consequence, the linear D1Q4 model can be defined. When calculating the fluxes at the interfaces between cells, the LLBE is solved locally. The LLBE is multiscale expanded

with respect to the Knudsen number using CE analysis, and the flux expression can be constructed based on the moment relations and the developed linear D1Q4 model.

Finally, the linear D1Q4 model-based LBFS is formed and applied to simulate acoustic propagation in fluids. Because the disturbance is much smaller than the base flow, a high-precision algorithm is needed to provide sufficient resolution to capture the disturbance. In recent years, Liu *et al.* proposed a high-precision least-squares-based finite-difference–finite-volume (LSFD-FV) method [25,34,35] for unstructured grids, which uses high-order Taylor series to approximate the variable distributions with the flux terms integrated using high-order Gaussian integration [36]. The unknown derivatives are calculated from the values of the control cell and neighboring cells using the mesh-free least-squares method [37]. Research has shown that the method’s accuracy and efficiency are better than the k -exact method [38], and that it is suitable for flow in complex geometries. Therefore, in the present work, the high-precision FVM is used to discretize the LEE, the Taylor series are used to approximate the distribution of variables within and around the control cell, the LSFD is used to calculate the unknown derivatives, and Gaussian integration is used to calculate the total fluxes of a cell.

This article is organized as follows. In Sec. II, the relations between the LLBE and LEE, the derivation and application of the linear D1Q4, and the computational sequence are discussed to introduce the principle of the algorithm. In Sec. III, several numerical cases are simulated to validate the proposed algorithm. Finally, conclusions are drawn in Sec. IV.

II. METHODOLOGY

A. Discretization of governing equations

The governing equations for simulating acoustic propagation in fluids are the LEE in terms of disturbed density, velocity, and pressure, and the integrated form in the x - y coordinate system can be written as

$$\int_V \frac{\partial \mathbf{W}'}{\partial t} dV + \int_S \mathbf{F}' dS = \int_V \mathbf{S}' dV, \quad (1)$$

where t is the time, V is the volume of the control cell, and S represents the surfaces of the control cell. The conservative variables \mathbf{W}' and inviscid fluxes \mathbf{F}' can be written as

$$\mathbf{W}' = \begin{Bmatrix} \rho' \\ \rho' \bar{u} + \bar{\rho} u' \\ \rho' \bar{v} + \bar{\rho} v' \\ \varepsilon' \end{Bmatrix}, \quad (2)$$

$$\mathbf{F}' = \begin{Bmatrix} \rho' \bar{u}_n \bar{u} + \bar{\rho} u'_n \bar{u} + \bar{\rho} \bar{u}_n u' + p' n_x \\ \rho' \bar{u}_n \bar{v} + \bar{\rho} u'_n \bar{v} + \bar{\rho} \bar{u}_n v' + p' n_y \\ u'_n (\bar{\varepsilon} + \bar{p}) + \bar{u}_n (\varepsilon' + p') \end{Bmatrix}, \quad (3)$$

where ρ , u , v , p , and ε are, respectively, the density, velocity in the x direction, velocity in the y direction, pressure, and the total energy per unit volume. n_x and n_y are the x and y components of the boundary unit normal vector. The bar represents the base flow variables, and the prime represents the perturbed flow variables. The subscript n denotes the direction normal to the control cell boundary. There are four unknown variables,

ρ' , u' , v' , and p' , because p' can be obtained from

$$\varepsilon' = \frac{p'}{\gamma - 1} + \bar{\rho}(u'\bar{u} + v'\bar{v}) + \frac{1}{2}\rho'(\bar{u}^2 + \bar{v}^2). \quad (4)$$

Consequently, the four unknown variables can be solved from the four equations.

The source terms in Eq. (1) have not been defined yet. They are related to the perturbed sources of the density and pressure and in the present work are defined as follows:

$$\mathbf{S}' = \left\{ \begin{array}{c} \rho'_s \\ \rho'_s\bar{u} + \bar{\rho}u'_s \\ \rho'_s\bar{v} + \bar{\rho}v'_s \\ \frac{1}{2}\rho'_s(\bar{u}^2 + \bar{v}^2) + \frac{p'_s}{\gamma-1} + \bar{\rho}(\bar{u}u'_s + \bar{v}v'_s) \end{array} \right\}, \quad (5)$$

where γ is the specific heat ratio. The specific forms of the perturbed density source ρ'_s , perturbed velocity sources u'_s ,

and v'_s , and perturbed pressure source p'_s are defined later. The relation $p'_s = \rho'_s c_0^2$ is also available [1], where the base flow sound velocity is $c_0 = \sqrt{\gamma \bar{p} / \bar{\rho}}$.

A two-point Gauss integration is performed in the surface integral of Eq. (1) to discretize the governing equations,

$$\frac{\partial}{\partial t} \int_V \mathbf{W}' dV + \sum_{n=1}^{N_s} \sum_{m=1}^2 (w_m \mathbf{F}'_m S_m)_{ns} = \int_V \mathbf{S}' dV, \quad (6)$$

where m represents the m th Gauss point, and ns represents the ns th surface of a cell where N_s is the total number of cell surfaces, and w is the Gauss integration weight. The conservative variables \mathbf{W}' within a control cell can be expressed using a third-order Taylor series expansion,

$$\mathbf{W}'(x, y) = \mathbf{W}'(x_i, y_i) + (\Delta \mathbf{x})^T \nabla \mathbf{W}'(x_i, y_i) + \frac{1}{2} (\Delta \mathbf{x})^T \mathbf{H} (\Delta \mathbf{x}) + \frac{1}{6} (\Delta^2 \mathbf{x})^T \mathbf{G} (\Delta \mathbf{x}) + O(\Delta x^4, \Delta y^4), \quad (7)$$

where (x_i, y_i) is the centroid position of cell i , the distance between two points is $\Delta \mathbf{x} = (x - x_i, y - y_i)^T$, and $\Delta^2 \mathbf{x} = [(x - x_i)^2, (y - y_i)^2]^T$. The matrix \mathbf{G} and Hessian matrix \mathbf{H} are

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 \mathbf{W}'}{\partial x^2} \Big|_{(x_i, y_i)} & \frac{\partial^2 \mathbf{W}'}{\partial x \partial y} \Big|_{(x_i, y_i)} \\ \frac{\partial^2 \mathbf{W}'}{\partial x \partial y} \Big|_{(x_i, y_i)} & \frac{\partial^2 \mathbf{W}'}{\partial y^2} \Big|_{(x_i, y_i)} \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \frac{\partial^3 \mathbf{W}'}{\partial x^3} \Big|_{(x_i, y_i)} & 3 \frac{\partial^3 \mathbf{W}'}{\partial x^2 \partial y} \Big|_{(x_i, y_i)} \\ 3 \frac{\partial^3 \mathbf{W}'}{\partial x \partial y^2} \Big|_{(x_i, y_i)} & \frac{\partial^3 \mathbf{W}'}{\partial y^3} \Big|_{(x_i, y_i)} \end{bmatrix}. \quad (8)$$

Therefore, the volume integral $\int_V \mathbf{W}' dV$ can be calculated based on Eq. (7) and written as $\mathbf{W}'(x_i, y_i)V_i + \mathbf{D}_i^T \mathbf{C}_i$, where vectors \mathbf{D}_i and \mathbf{C}_i are

$$\mathbf{D}_i = \begin{bmatrix} \frac{\partial \mathbf{W}'}{\partial x} \Big|_{(x_i, y_i)} & \frac{\partial \mathbf{W}'}{\partial y} \Big|_{(x_i, y_i)} & \frac{\partial^2 \mathbf{W}'}{\partial x^2} \Big|_{(x_i, y_i)} & \frac{\partial^2 \mathbf{W}'}{\partial y^2} \Big|_{(x_i, y_i)} & \frac{\partial^2 \mathbf{W}'}{\partial x \partial y} \Big|_{(x_i, y_i)} \\ \times \frac{\partial^3 \mathbf{W}'}{\partial x^3} \Big|_{(x_i, y_i)} & \frac{\partial^3 \mathbf{W}'}{\partial y^3} \Big|_{(x_i, y_i)} & \frac{\partial^3 \mathbf{W}'}{\partial x^2 \partial y} \Big|_{(x_i, y_i)} & \frac{\partial^3 \mathbf{W}'}{\partial x \partial y^2} \Big|_{(x_i, y_i)} \end{bmatrix}^T, \quad (9)$$

$$\mathbf{C}_i = \begin{bmatrix} \int_V (x - x_i) dV & \int_V (y - y_i) dV & \frac{1}{2} \int_V (x - x_i)^2 dV & \frac{1}{2} \int_V (y - y_i)^2 dV & \int_V (x - x_i)(y - y_i) dV \\ \times \frac{1}{6} \int_V (x - x_i)^3 dV & \frac{1}{6} \int_V (y - y_i)^3 dV & \frac{1}{2} \int_V (x - x_i)^2 (y - y_i) dV & \frac{1}{2} \int_V (x - x_i)(y - y_i)^2 dV \end{bmatrix}^T. \quad (10)$$

The vector \mathbf{C}_i is only related to the grids, while the vector \mathbf{D}_i contains unknown derivatives, which are to be determined by the LSFD method; i.e., it is calculated from the neighbor points to minimize its two-norm error. As a result, \mathbf{D}_i can be expressed by the conservative variables of cell i and its neighbor cells with the expression

$$\mathbf{D}_i = \mathbf{Q}_i^T \begin{bmatrix} \mathbf{W}'(x_1, y_1) - \mathbf{W}'(x_i, y_i) \\ \mathbf{W}'(x_2, y_2) - \mathbf{W}'(x_i, y_i) \\ \dots \\ \mathbf{W}'(x_N, y_N) - \mathbf{W}'(x_i, y_i) \end{bmatrix}. \quad (11)$$

The derivation details are not provided here; readers can refer to Ref. [25] for additional information. The resulting final form of the integral $\int_V \mathbf{W}' dV$ is $(V_i - \sum_{j=1}^N B_{ij}) \mathbf{W}'(x_i, y_i) + \sum_{j=1}^N B_{ij} \mathbf{W}'(x_j, y_j)$, where N is the number of the neighbor cells of cell i , and B_{ij} is the j th component of the vector \mathbf{B}_i , calculated as $\mathbf{B}_i = \mathbf{Q}_i^T \mathbf{C}_i$. The matrix \mathbf{Q}_i can be written as

$$\mathbf{Q}_i = \mathbf{A}_i [(\mathbf{P}_i \mathbf{A}_i)^T \mathbf{M}_i (\mathbf{P}_i \mathbf{A}_i)]^{-1} (\mathbf{P}_i \mathbf{A}_i)^T \mathbf{M}_i, \quad (12)$$

where matrixes \mathbf{P} , \mathbf{A} , and \mathbf{M}_i are

$$\mathbf{P}_i = \begin{bmatrix} \Delta x_1 & \Delta y_1 & \Delta x_1^2/2 & \Delta y_1^2/2 & \Delta x_1 \Delta y_1 & \Delta x_1^3/6 & \Delta y_1^3/6 & \Delta x_1^2 \Delta y_1/2 & \Delta x_1 \Delta y_1^2/2 \\ \Delta x_2 & \Delta y_2 & \Delta x_2^2/2 & \Delta y_2^2/2 & \Delta x_2 \Delta y_2 & \Delta x_2^3/6 & \Delta y_2^3/6 & \Delta x_2^2 \Delta y_2/2 & \Delta x_2 \Delta y_2^2/2 \\ \Delta x_3 & \Delta y_3 & \Delta x_3^2/2 & \Delta y_3^2/2 & \Delta x_3 \Delta y_3 & \Delta x_3^3/6 & \Delta y_3^3/6 & \Delta x_3^2 \Delta y_3/2 & \Delta x_3 \Delta y_3^2/2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \Delta x_N & \Delta y_N & \Delta x_N^2/2 & \Delta y_N^2/2 & \Delta x_N \Delta y_N & \Delta x_N^3/6 & \Delta y_N^3/6 & \Delta x_N^2 \Delta y_N/2 & \Delta x_N \Delta y_N^2/2 \end{bmatrix}, \quad (13)$$

$$\mathbf{A}_i = \text{diag}[d_i^{-1} \quad d_i^{-1} \quad d_i^{-2} \quad d_i^{-2} \quad d_i^{-2} \quad d_i^{-3} \quad d_i^{-3} \quad d_i^{-3} \quad d_i^{-3}], \quad (14)$$

$$\mathbf{M}_i = \text{diag}[w_{i1} \quad w_{i2} \quad \dots \quad w_{iN}], \quad (15)$$

and where $\Delta x_j = x_j - x_i$, $\Delta y_j = y_j - y_i$ for $j = 1, \dots, N$. d_i is chosen as $1.1[\max_{j=1}^N(\sqrt{\Delta x_j^2 + \Delta y_j^2})]$ and $w_{ij} = 1/\sqrt{\Delta x_j^2 + \Delta y_j^2}$. It is worth noting that the vector \mathbf{B}_i and matrix \mathbf{Q}_i are only dependent on the grid and can be computed before the iterations begin. As a consequence, the governing equations can be expressed as

$$\frac{\partial}{\partial t} \left[\left(V_i - \sum_{j=1}^N B_{ij} \right) \mathbf{W}'(x_i, y_i) + \sum_{j=1}^N B_{ij} \mathbf{W}'(x_j, y_j) \right] = -\mathbf{R}_i[\mathbf{W}'(x_i, y_i)], \quad (16)$$

where the residual term is $\mathbf{R}_i = \sum_{n=1}^{N_s} \sum_{m=1}^2 (w_m \mathbf{F}'_m \mathcal{S}_m)_n - \mathbf{S}' V_i$. The time derivative is also needed to complete the discretization. In the present work, all the test cases are unsteady problems; therefore, the dual time stepping method is adopted, and the pseudotime level is introduced as

$$\frac{\partial}{\partial t^*} \left[\left(V_i - \sum_{j=1}^N B_{ij} \right) \mathbf{W}'(x_i, y_i) + \sum_{j=1}^N B_{ij} \mathbf{W}'(x_j, y_j) \right] = -\mathbf{R}_i^*[\mathbf{W}'(x_i, y_i)], \quad (17)$$

where t^* is the pseudotime and R^* is defined as

$$\mathbf{R}_i^*[\mathbf{W}'(x_i, y_i)] = - \left\{ \mathbf{R}_i[\mathbf{W}'(x_i, y_i)] + \frac{\partial}{\partial t} \left[\left(V_i - \sum_{j=1}^N B_{ij} \right) \mathbf{W}'(x_i, y_i) + \sum_{j=1}^N B_{ij} \mathbf{W}'(x_j, y_j) \right] \right\}. \quad (18)$$

The time derivative of R^* is approximated by second-order finite difference and the three-stage Runge-Kutta scheme [39,40] is used for pseudotime derivatives. The details are not repeated here, because readers can refer to Refs. [25,41] for details.

The governing equations have now been discretized in both time and space. The next task is to determine the fluxes \mathbf{F}'_m in the residual term, which are discussed in Sec. II D.

B. Recovery of the LEE from the LLBE with CE analysis

In this work, the linear LBFS, which constructs the flux expression based on the local solution of the LLBE, is used to calculate the fluxes. Therefore, it is necessary to determine the relationships between the LLBE and LEE. In the x - y coordinate system, without external forces, the LLBE based on the Bhatnagar-Gross-Krook (BGK) [42] collision model can be written as

$$\begin{aligned} & f'_\alpha(\mathbf{x} + \boldsymbol{\xi}_\alpha \delta_t, t + \delta_t) - f'_\alpha(\mathbf{x}, t) \\ &= -\frac{1}{\tau} [f'_\alpha(\mathbf{x}, t) - g'_\alpha(\mathbf{x}, t)]. \end{aligned} \quad (19)$$

The subscript α represents the α th particle. f'_α is the density distribution function and g'_α is the equilibrium distri-

bution function of the α th particle. δ_t is the time step, τ is the dimensionless relaxation time, and $\boldsymbol{\xi}_\alpha$ is the lattice velocity of the α th particle which reduce to the velocity normal to the interface (a scalar) in one dimension. The equilibrium distribution function g'_α and lattice velocity $\boldsymbol{\xi}_\alpha$ are defined in Sec. II C.

Based on the CE expansion analysis [43], a relationship can be built between kinetic theory and the macroscopic equations. According to the conservation of mass, momentum, and energy, three moment relations can be derived:

$$\begin{aligned} \sum_\alpha g'_\alpha &= \rho', \\ \sum_\alpha g'_\alpha \boldsymbol{\xi}_\alpha &= \rho' \bar{\mathbf{u}} + \bar{\rho} \mathbf{u}', \\ \sum_\alpha g'_\alpha (\boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha + 2e'_p) &= 2e', \end{aligned} \quad (20)$$

where $e'_p = [1-D(\gamma-1)/2]e'$ is the particle potential energy; D is the dimension, chosen as 1 if the D1Q4 model is used; and $e' = p'/[\rho'(\gamma-1)]$. First, the LLBE (19) is expanded in a first-order Taylor series,

$$\frac{\partial f'_\alpha}{\partial t} \delta_t + \boldsymbol{\xi}_\alpha \delta_t \nabla f'_\alpha = -\frac{1}{\tau} (f'_\alpha - g'_\alpha) + O(\delta_t^2), \quad (21)$$

where $\nabla = \frac{\partial}{\partial \mathbf{x}}$ is the Hamiltonian, which reduces to $\frac{\partial}{\partial x}$ in one dimension. The distribution function, time derivative, and space derivative are multiscale expanded as

$$\begin{aligned} f'_\alpha &= f'_\alpha^{(0)} + Kn f'_\alpha^{(1)} + Kn^2 f'_\alpha^{(2)} + \dots, \\ \frac{\partial}{\partial t} &= Kn \frac{\partial}{\partial t_1} + Kn^2 \frac{\partial}{\partial t_2}, \\ \frac{\partial}{\partial x} &= Kn \frac{\partial}{\partial x_1}, \end{aligned} \quad (22)$$

where Kn is a small parameter identified with the Knudsen number (Kn). By substituting Eq. (22) into Eq. (21), the coefficient of each order of Kn results in the following equations

$$Kn^0 : f'_\alpha^{(0)} = g'_\alpha, \quad (23)$$

$$Kn^1 : \frac{\partial f'_\alpha^{(0)}}{\partial t_1} + \xi_\alpha \nabla_1 f'_\alpha^{(0)} = -\frac{1}{\tau \delta_t} f'_\alpha^{(1)}, \quad (24)$$

$$Kn^2 : \frac{\partial f'_\alpha^{(0)}}{\partial t_2} + \frac{\partial f'_\alpha^{(1)}}{\partial t_1} + \xi_\alpha \nabla_1 f'_\alpha^{(1)} = -\frac{1}{\tau \delta_t} f'_\alpha^{(2)}. \quad (25)$$

By multiplying Eqs. (24) and (25) with the collisional invariants $\boldsymbol{\varphi} = [1 \ \xi_\alpha \ \xi_\alpha \xi_\alpha / 2 + e'_p]^T$ and performing integration in the lattice velocity space, one can obtain

$$\begin{aligned} \frac{\partial}{\partial t_1} \begin{pmatrix} \rho' \\ \rho' \bar{u}_n + \bar{\rho} u'_n \\ \varepsilon' \end{pmatrix} + \frac{\partial}{\partial x_1} \begin{pmatrix} \rho' \bar{u}_n + \bar{\rho} u'_n \\ \rho' \bar{u}_n \bar{u}_n + \bar{\rho} u'_n \bar{u}_n + \bar{\rho} \bar{u}_n u'_n + p' \\ \sum_\alpha g'_\alpha (\xi_\alpha \xi_\alpha + 2e'_p) \xi_\alpha \end{pmatrix} \\ = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \end{aligned} \quad (26)$$

$$\begin{aligned} \frac{\partial}{\partial t_2} \begin{pmatrix} \rho' \\ \rho' \bar{u}_n + \bar{\rho} u'_n \\ \varepsilon' \end{pmatrix} + \frac{\partial}{\partial x_1} \begin{pmatrix} \sum_\alpha f'_\alpha^{(1)} \xi_\alpha \\ \sum_\alpha f'_\alpha^{(1)} \xi_\alpha \xi_\alpha \\ \sum_\alpha f'_\alpha^{(1)} \xi_\alpha (\xi_\alpha \xi_\alpha / 2 + e'_p) \end{pmatrix} \\ = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \end{aligned} \quad (27)$$

where Eq. (23) and the moment relations Eq. (20) have been used. Adding Eqs. (26) and (27) should result in the LEE, implying that $f'_\alpha^{(1)} = 0$ ($f'_\alpha = g'_\alpha$). In other words, the distribution function only needs to be expanded to the zeroth order of Kn to recover the Euler equations. After adding Eqs. (26) and (27), the result is exactly the LEE in one dimension:

$$\frac{\partial}{\partial t} \begin{pmatrix} \rho' \\ \rho' \bar{u}_n + \bar{\rho} u'_n \\ \varepsilon' \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} \rho' \bar{u}_n + \bar{\rho} u'_n \\ \rho' \bar{u}_n \bar{u}_n + \bar{\rho} u'_n \bar{u}_n + \bar{\rho} \bar{u}_n u'_n + p' \\ \bar{u}_n (\bar{\rho} \bar{u}_n u'_n + 0.5 \rho' \bar{u}_n \bar{u}_n + \frac{\gamma}{\gamma-1} p') + u'_n (0.5 \bar{\rho} \bar{u}_n \bar{u}_n + \frac{\gamma}{\gamma-1} \bar{p}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \quad (28)$$

The coordinate system is established in the direction normal to the interface, and the subscript n represents the interface normal direction. From Eq. (28), another moment relation can be found:

$$\sum_\alpha g'_\alpha (\xi_\alpha \xi_\alpha + 2e'_p) \xi_\alpha = \bar{u}_n \left(2\bar{\rho} \bar{u}_n u'_n + \rho' \bar{u}_n \bar{u}_n + \frac{2\gamma}{\gamma-1} p' \right) + u'_n \left(\bar{\rho} \bar{u}_n \bar{u}_n + \frac{2\gamma}{\gamma-1} \bar{p} \right). \quad (29)$$

Since the discrete velocity model is one dimensional, approximations for variables in other directions should be made. Equation (28) can be extended to two dimensions with the approximations

$$\sum_\alpha g'_\alpha \xi_\alpha \bar{u}_\tau = \bar{\rho} u'_n \bar{u}_\tau + \rho' \bar{u}_n \bar{u}_\tau, \quad (30)$$

$$\sum_\alpha g'_\alpha \xi_\alpha \bar{u}_\tau^2 = \bar{\rho} u'_n \bar{u}_\tau^2 + \rho' \bar{u}_n \bar{u}_\tau^2, \quad (31)$$

where the subscript τ represents the direction tangential to the interface. The perturbed tangential velocity u'_τ at the interface can be approximated by the Roe average.

C. Development of linear discrete velocity model from moment relations

The discrete velocity model is an important role for the lattice Boltzmann equation. In the present work, the LLBE is used. Therefore, a corresponding linear discrete velocity model is discussed in this section. The developed linear discrete velocity model, called L-D1Q4, is a one-dimensional model with four lattice velocities and works for linear gov-

erning equations. A schematic of this model is shown in Fig. 1, where d is the lattice velocity. If the equilibrium density distribution functions and lattice velocities are considered unknowns, there are six unknowns in this model (four

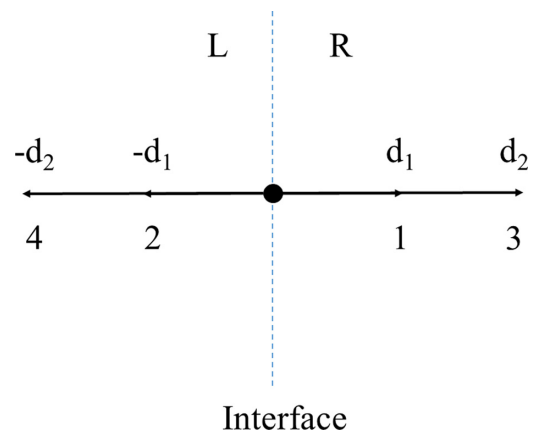


FIG. 1. Schematic of the L-D1Q4 model.

equilibrium distribution functions g'_1, g'_2, g'_3, g'_4 and two lattice velocities d_1, d_2 .

For clarity, the moment relations Eqs. (20) and (29) are reorganized and rewritten as

$$\begin{aligned}\sum_{\alpha} g'_{\alpha} &= \rho', \\ \sum_{\alpha} g'_{\alpha} \xi_{\alpha} &= \rho' \bar{u}_n + \bar{\rho} u'_n, \\ \sum_{\alpha} g'_{\alpha} \xi_{\alpha} \xi_{\alpha} &= \rho' \bar{u}_n \bar{u}_n + 2\bar{\rho} u'_n \bar{u}_n + p',\end{aligned}$$

$$\begin{aligned}\sum_{\alpha} g'_{\alpha} \xi_{\alpha} \xi_{\alpha} \xi_{\alpha} &= \bar{u}_n (2\bar{\rho} \bar{u}_n u'_n + \rho' \bar{u}_n \bar{u}_n + 3p') \\ &+ u'_n \left(\bar{\rho} \bar{u}_n \bar{u}_n + \frac{2\gamma}{\gamma-1} \bar{p} - 2\bar{\rho} e' + \bar{\rho} c'^2 \right),\end{aligned}\quad (32)$$

where $c' = \sqrt{(\gamma-1)e'}$. For the L-D1Q4 lattice velocity model, α varies as 1–4 and ξ_{α} is d_1 or d_2 . Finally, one can obtain four equations by substituting the model parameter into Eq. (32).

$$\begin{aligned}g'_1 + g'_2 + g'_3 + g'_4 &= \rho', \\ g'_1 d_1 - g'_2 d_1 + g'_3 d_2 - g'_4 d_2 &= \rho' \bar{u}_n + \bar{\rho} u'_n, \\ g'_1 d_1^2 + g'_2 d_1^2 + g'_3 d_2^2 + g'_4 d_2^2 &= \rho' \bar{u}_n \bar{u}_n + 2\bar{\rho} u'_n \bar{u}_n + p', \\ g'_1 d_1^3 - g'_2 d_1^3 + g'_3 d_2^3 - g'_4 d_2^3 &= \bar{u}_n (2\bar{\rho} \bar{u}_n u'_n + \rho' \bar{u}_n \bar{u}_n + 3p') + u'_n \left(\bar{\rho} \bar{u}_n \bar{u}_n + \frac{2\gamma}{\gamma-1} \bar{p} - 2\bar{\rho} e' + \bar{\rho} c'^2 \right).\end{aligned}\quad (33)$$

Among the six unknown variables, the lattice velocity can be determined by local total energy E [44], that is, $d_2 = \sqrt{E}$, $d_1 = d_2/4$. Therefore, there are four unknown variables (g'_1, g'_2, g'_3, g'_4), and they can be solved from the closed equation system given by Eq. (33). The algebraic operations are not repeated here because they can be performed using MATLAB or MAPLE software. The final expressions for the distribution functions are

$$\begin{aligned}g'_1 &= [u'_n (2\bar{\varepsilon} + 2\bar{p} + 2\bar{\rho} \bar{u}_n^2 - 2\bar{\rho} e' + \bar{\rho} c'^2 - \bar{\rho} d_2^2) + \bar{u}_n (3p' + \rho' \bar{u}_n^2 + \rho' d_1 \bar{u}_n - \rho' d_2^2 + 2\bar{\rho} d_1 u'_n) + p' d_1 - \rho' d_1 d_2^2] / \\ &[2d_1 (d_1^2 - d_2^2)], \\ g'_2 &= -[u'_n (2\bar{\varepsilon} + 2\bar{p} + 2\bar{\rho} \bar{u}_n^2 + \bar{\rho} c'^2 - 2\bar{\rho} e' - \bar{\rho} d_2^2) + \bar{u}_n (3p' + \rho' \bar{u}_n^2 - \rho' d_1 \bar{u}_n - \rho' d_2^2 - 2\bar{\rho} d_1 u'_n) - p' d_1 + \rho' d_1 d_2^2] / \\ &[2d_1 (d_1^2 - d_2^2)], \\ g'_3 &= -[u'_n (2\bar{\varepsilon} + 2\bar{p} + 2\bar{\rho} \bar{u}_n^2 - 2\bar{\rho} e' + \bar{\rho} c'^2 - \bar{\rho} d_1^2) + \bar{u}_n (3p' - \rho' d_1^2 + \rho' \bar{u}_n^2 + \rho' d_2 \bar{u}_n + 2\bar{\rho} d_2 u'_n) + p' d_2 - \rho' d_1^2 d_2] / \\ &[2d_2 (d_1^2 - d_2^2)], \\ g'_4 &= [u'_n (2\bar{\varepsilon} + 2\bar{p} + 2\bar{\rho} \bar{u}_n^2 - 2\bar{\rho} e' + \bar{\rho} c'^2 - \bar{\rho} d_1^2) + \bar{u}_n (3p' + \rho' \bar{u}_n^2 - \rho' d_1^2 - \rho' d_2 \bar{u}_n - 2\bar{\rho} d_2 u'_n) - p' d_2 + \rho' d_1^2 d_2] / \\ &[2d_2 (d_1^2 - d_2^2)],\end{aligned}\quad (34)$$

where $\bar{\varepsilon} = \bar{p}/(\gamma-1) + 0.5\bar{\rho} \bar{u}_n^2$ is the total energy per unit volume of the base flow. Every parameter of the L-D1Q4 model has now been defined. Another approach to constructing the discrete velocity model is based on a known distribution function, such as the Maxwellian function, while the construction method of this section is an ‘‘inverse design,’’ which strictly satisfies the moment relations. A similar D1Q4 model used for macroscopic governing equations can be found in Refs. [32,33].

D. Implementation of the L-D1Q4-based LBFS

From Secs. II B and II C, the L-D1Q4-based LBFS can be derived to construct the fluxes $(\mathbf{F}'_m)_{ns}$ in Eq. (6) explicitly. Again, $(\mathbf{F}'_m)_{ns}$ is the m th Gaussian point of the n th surface of a cell. Based on the discussion in Sec. II C, the fluxes \mathbf{F}'_n in one dimension can be written as

$$\mathbf{F}'_n = \begin{bmatrix} F'_{n1} \\ F'_{n2} \\ F'_{n3} \end{bmatrix} = \begin{bmatrix} \sum_{\alpha} g'_{\alpha} \xi_{\alpha} \\ \sum_{\alpha} g'_{\alpha} \xi_{\alpha} \xi_{\alpha} \\ \sum_{\alpha} g'_{\alpha} \xi_{\alpha} \xi_{\alpha} \xi_{\alpha} / 2 + e'_p \end{bmatrix}. \quad (35)$$

By combining Eqs. (30) and (31), the fluxes $(\mathbf{F}'_m)_{ns}$ can be expressed as

$$\mathbf{F}' = \begin{bmatrix} F'_{n1} \\ F'_{n2} n_x + (\bar{\rho} \bar{u}_n u'_{\tau} + F'_{n1} \bar{u}_{\tau}) n_y \\ F'_{n2} (-n_y) + (\bar{\rho} \bar{u}_n u'_{\tau} + F'_{n1} \bar{u}_{\tau}) n_x \\ F'_{n3} + 0.5 F'_{n1} \bar{u}_{\tau}^2 + \bar{\rho} \bar{u}_n \bar{u}_{\tau} u'_{\tau} \end{bmatrix}. \quad (36)$$

The subscripts m and ns have been omitted to make the formula concise. All the base flow variables in Eq. (36) are known, so the remaining problem is determining the distribution function g'_{α} and lattice velocity ξ_{α} at the Gauss points of the interface.

If a Gauss point coordinate is denoted as \mathbf{x}_m . The distribution function at the position \mathbf{x}_m can be calculated from the conservative variables at that position. If the lattice velocity ξ_{α} is greater than zero, the conservative variables are interpolated from the left cell, and if it is less than zero, the conservative variables are interpolated from the right cell. The distribution

function $g'_\alpha(\mathbf{x}_m, t)$ can be calculated as

$$g'_\alpha(\mathbf{x}_m, t) = \begin{cases} g'_\alpha(\mathbf{W}'_L) & \text{if } \xi_\alpha > 0 \\ g'_\alpha(\mathbf{W}'_R) & \text{if } \xi_\alpha < 0, \end{cases} \quad (37)$$

where $g'_\alpha(\mathbf{W}'_{L/R})$ means that the distribution function is calculated from the conservative variables $\mathbf{W}'_{L/R}$ based on Eq. (34), and the conservative variables $\mathbf{W}'_{L/R}$ are interpolated as

$$\begin{aligned} \mathbf{W}'_L &= \mathbf{W}'(x_L, y_L) + (\Delta \mathbf{x}_L)^T \nabla \mathbf{W}'(x_L, y_L) \\ &+ \frac{1}{2}(\Delta \mathbf{x}_L)^T \mathbf{H}_L(\Delta \mathbf{x}_L) + \frac{1}{6}(\Delta^2 \mathbf{x}_L)^T \mathbf{G}_L(\Delta \mathbf{x}_L) \\ &+ O(\Delta x^4, \Delta y^4), \\ \mathbf{W}'_R &= \mathbf{W}'(x_R, y_R) + (\Delta \mathbf{x}_R)^T \nabla \mathbf{W}'(x_R, y_R) \end{aligned}$$

$$\begin{aligned} &+ \frac{1}{2}(\Delta \mathbf{x}_R)^T \mathbf{H}_R(\Delta \mathbf{x}_R) + \frac{1}{6}(\Delta^2 \mathbf{x}_R)^T \mathbf{G}_R(\Delta \mathbf{x}_R) \\ &+ O(\Delta x^4, \Delta y^4), \end{aligned} \quad (38)$$

where $\mathbf{x}_{L/R} = (x_{L/R}, y_{L/R})$ is the coordinate of the left and right control cell of the interface and $\Delta \mathbf{x}_{L/R} = \mathbf{x}_m - \mathbf{x}_{L/R}$ (δt approaches zero) is the distance between the Gauss point m and the control cell. Once the distribution function $g'_\alpha(\mathbf{x}_m, t)$ has been calculated, the distribution function $g'_\alpha(\mathbf{x}_m, t)$ can be obtained according to Eq. . Therefore, the final expression for $g'_\alpha(\mathbf{x}_m, t)$ is

$$g'_\alpha(\mathbf{x}_m, t) = \begin{cases} g'_\alpha(\mathbf{W}'_L) & \text{for } \alpha = 1, 3 \\ g'_\alpha(\mathbf{W}'_R) & \text{for } \alpha = 2, 4. \end{cases} \quad (39)$$

Now that the distribution function $g'_\alpha(\mathbf{x}_m, t)$ has been obtained, it can be substituted into Eqs. (35) and (36) to calculate the interface fluxes $(\mathbf{F}'_m)_{ns}$. The final expression for the fluxes at the Gauss point is

$$\begin{aligned} \mathbf{F}'_{II} &= \begin{bmatrix} F'_{II1} \\ F'_{II2} \\ F'_{II3} \\ F'_{II4} \end{bmatrix} \\ &= \begin{bmatrix} \sum_{\alpha=1,3} g'_\alpha(\mathbf{W}'_L) \xi_\alpha + \sum_{\alpha=2,4} g'_\alpha(\mathbf{W}'_R) \xi_\alpha \\ [\sum_{\alpha=1,3} g'_\alpha(\mathbf{W}'_L) \xi_\alpha + \sum_{\alpha=2,4} g'_\alpha(\mathbf{W}'_R) \xi_\alpha] n_x + (\bar{\rho} \bar{u}_n u'_\tau + F'_{II1} \bar{u}_\tau) n_y \\ [\sum_{\alpha=1,3} g'_\alpha(\mathbf{W}'_L) \xi_\alpha + \sum_{\alpha=2,4} g'_\alpha(\mathbf{W}'_R) \xi_\alpha] (-n_y) + (\bar{\rho} \bar{u}_n u'_\tau + F'_{II1} \bar{u}_\tau) n_x \\ \sum_{\alpha=1,3} g'_\alpha(\mathbf{W}'_L) \xi_\alpha [\xi_\alpha \xi_\alpha / 2 + e'_p(\mathbf{W}'_L)] + \sum_{\alpha=2,4} g'_\alpha(\mathbf{W}'_R) \xi_\alpha [\xi_\alpha \xi_\alpha / 2 + e'_p(\mathbf{W}'_R)] + 0.5 F'_{II1} \bar{u}_\tau^2 + \bar{\rho} \bar{u}_n \bar{u}_\tau u'_\tau \end{bmatrix}. \end{aligned} \quad (40)$$

All of the algorithm steps have now been addressed and can be performed. The following provides a further discussion of the flux calculation.

If the distribution function $g'_\alpha(\mathbf{x}_m, t)$ is substituted into Eq. (20), the conservative variables at the Gauss point \mathbf{W}'_m for one dimension can be calculated:

$$\mathbf{W}'_m = \begin{bmatrix} \sum_{\alpha=1,3} g'_\alpha(\mathbf{W}'_L) + \sum_{\alpha=2,4} g'_\alpha(\mathbf{W}'_R) \\ \sum_{\alpha=1,3} g'_\alpha(\mathbf{W}'_L) \xi_\alpha + \sum_{\alpha=2,4} g'_\alpha(\mathbf{W}'_R) \xi_\alpha \\ \sum_{\alpha=1,3} g'_\alpha(\mathbf{W}'_L) [0.5 \xi_\alpha \xi_\alpha + e'_p(\mathbf{W}'_L)] + \sum_{\alpha=2,4} g'_\alpha(\mathbf{W}'_R) [0.5 \xi_\alpha \xi_\alpha + e'_p(\mathbf{W}'_R)] \end{bmatrix} = \begin{bmatrix} \rho' \\ \bar{\rho} u'_n + \rho' \bar{u}_n \\ \varepsilon' \end{bmatrix}_m. \quad (41)$$

Consequently, the primitive variables at the Gauss point $(\rho'_m, u'_m, v'_m, p'_m)$ can be obtained and substituted directly into Eq. (3) to calculate the flux:

$$\mathbf{F}'_I = \begin{bmatrix} \rho' \bar{u}_n + \bar{\rho} u'_n \\ \rho' \bar{u}_n \bar{u} + \bar{\rho} u'_n \bar{u} + \bar{\rho} \bar{u}_n u' + p' n_x \\ \rho' \bar{u}_n \bar{v} + \bar{\rho} u'_n \bar{v} + \bar{\rho} \bar{u}_n v' + p' n_y \\ u'_n (\bar{\varepsilon} + \bar{p}) + \bar{u}_n (\varepsilon' + p') \end{bmatrix}_m. \quad (42)$$

If one needs to recover the Navier-Stokes equations from the LBM, there is a nonequilibrium part of the distribution function:

$$f'_\alpha = g'_\alpha + f'^{neq}_\alpha. \quad (43)$$

According to Ref. [23], the nonequilibrium part can be written as

$$f'^{neq}_\alpha = -\tau_0 [g'_\alpha(\mathbf{x}_m, t) - g'_\alpha(\mathbf{x}_m - \xi_\alpha \delta_t, t - \delta_t)]. \quad (44)$$

In this situation, the fluxes \mathbf{F}'_{II} can be regarded as the contribution from $g'_\alpha(\mathbf{x}_m - \xi_\alpha \delta_t, t - \delta_t)$ by interpolating the

conservative variables to the corresponding position using Eq. (38) and \mathbf{F}'_I is the contribution from $g'_\alpha(\mathbf{x}_m, t)$. Therefore, the expression for the fluxes is $\mathbf{F}'_I + \tau_0(\mathbf{F}'_{II} - \mathbf{F}'_I)$. However, because the governing equations are inviscid, the flux contributed from the nonequilibrium part should be zero. If one calculates the flux this way, the extra part $\tau_0(\mathbf{F}'_{II} - \mathbf{F}'_I)$ plays a role similar to numerical viscosity and increases the algorithm's stability. Because $\tau_0(\mathbf{F}'_{II} - \mathbf{F}'_I)$ no longer has an actual physical meaning, the value τ_0 is not critical. Therefore, τ_0 can be used as an adjustment function to control the

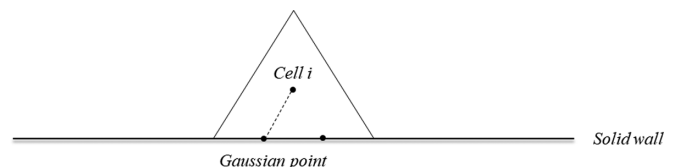


FIG. 2. Schematic of a Gauss point at the boundary.

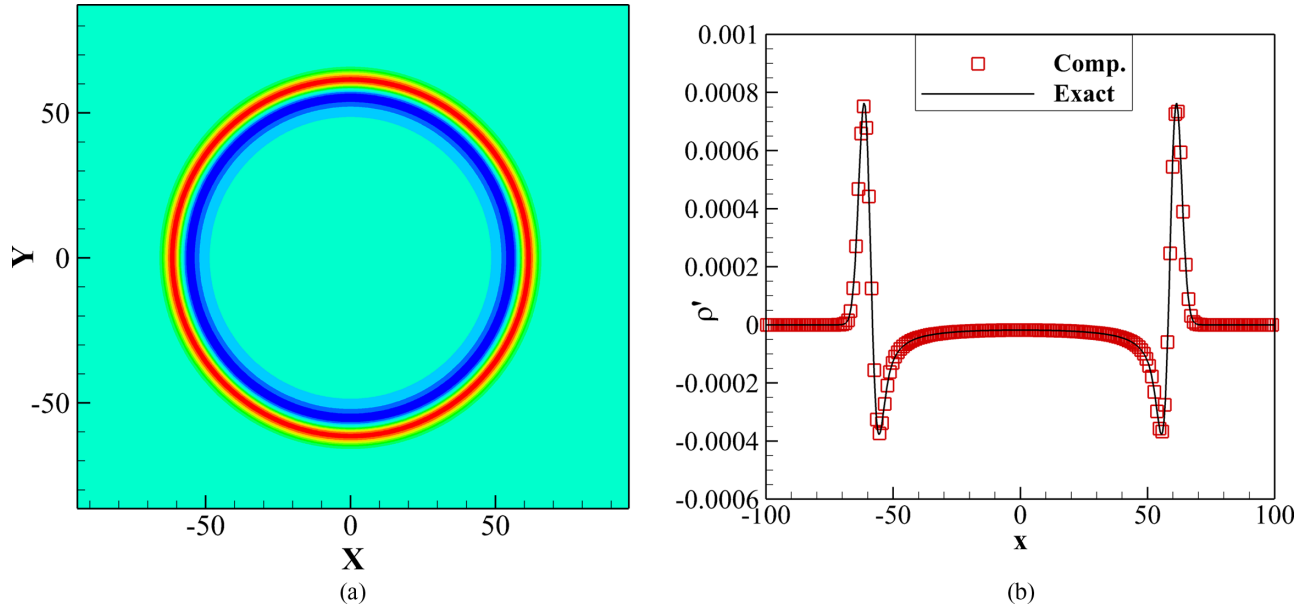


FIG. 3. Simulation results for a Gaussian pulse in a static flow with grid size $\Delta x = 0.5$ at $t = 60$: (a) density contours; (b) instantaneous perturbed pressure along the centerline $y = 0$.

numerical viscosity. In this work, it is defined as

$$\tau_0 = \tanh\left(C \frac{|p'_L - p'_R|}{p_L + p_R}\right), \quad (45)$$

where the subscripts L or R represent the value calculated from \mathbf{W}'_L or \mathbf{W}'_R , and C is a parameter with the value 1–100. Finally, the expression for the fluxes \mathbf{F}' is

$$\mathbf{F}' = (1 - \tau_0)\mathbf{F}'_I + \tau_0\mathbf{F}'_{II}. \quad (46)$$

It is noted that if the adjustment function τ_0 is zero, the flux expression is reduced to Eq. (40) or (42), considering that Eqs. (40) and (42) are equivalent when τ_0 is zero.

E. Boundary conditions

1. Impermeable condition

The solid walls for the small-disturbance problems are impermeable boundaries. The governing equations are the LEEs, so defining the perturbed variables (perturbed density, velocity, pressure) at the boundaries is necessary. Impermeability requires that the velocity normal to the boundary is zero, i.e., $\bar{u}_n = u'_n = 0$. According to Eq. (3), the fluxes at the boundary can be calculated as

$$\mathbf{F}'_w = \begin{Bmatrix} 0 \\ p'_w n_x \\ p'_w n_y \\ 0 \end{Bmatrix}, \quad (47)$$

where the subscript w represents the value on the solid wall. The perturbed pressure at the solid wall can be interpolated from the values of the neighbor cells using Eq. (7). As shown in Fig. 2, the perturbed pressure value at the solid wall can be interpolated from cell i .

2. Far-field condition

Similar to the impermeable boundaries, the variables at the far-field boundaries must also be defined. The difference is that these can either be interpolated from the inner cells or directly specified as inflow variables. The method used depends on whether a particular boundary is an inflow or outflow boundary and whether the inflow is subsonic or supersonic. Equation (7) can be used if the variables are interpolated.

The problems simulated in this work include both subsonic and supersonic flows. The perturbed density and pressure

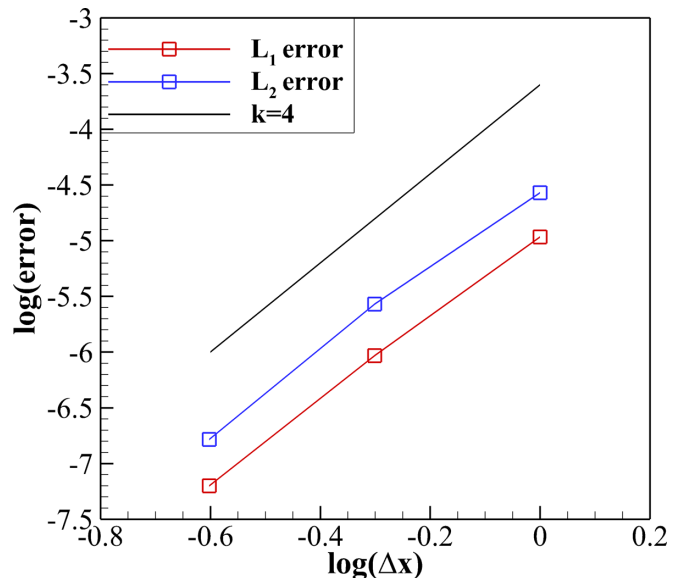


FIG. 4. L_1 error and L_2 error of the perturbed density for a Gaussian pulse in a static flow.

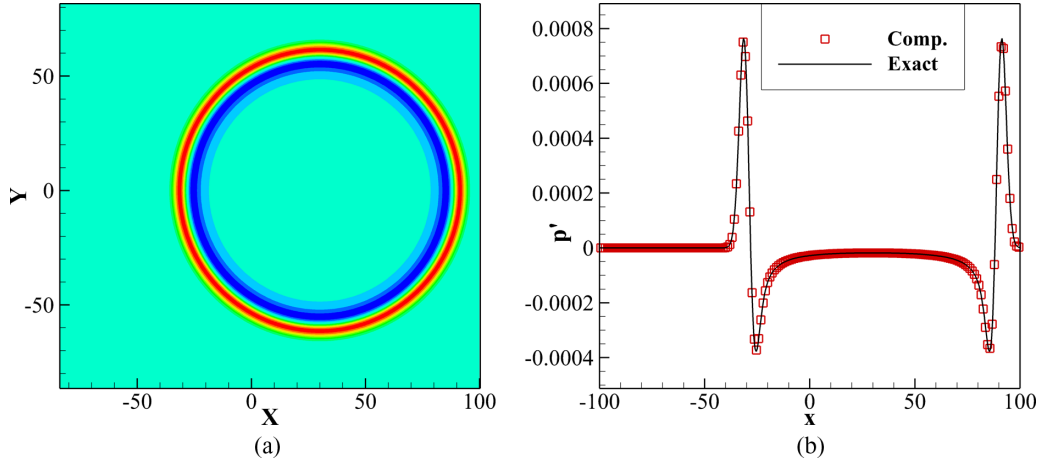


FIG. 5. Simulation results for a Gaussian pulse in a uniform flow at $Ma = 0.5$ at $t = 60$: (a) perturbed pressure contour; (b) instantaneous perturbed pressure along the centerline $y = 0$.

values are specified directly for inflow boundaries and interpolated from the inner cells for outflow boundaries for subsonic flows. For the perturbed velocity, it is just the opposite. For supersonic flows, all variables (perturbed density, pressure, velocity) are specified directly as inflow variables for the inflow boundary and interpolated from the inner cells for the outflow boundary.

In addition, a sponge grid layer is used outside the far-field boundary to prevent disturbances from being reflected from the boundaries.

F. Computational sequence

The computational sequence of the algorithm in the present work is as follows:

- (1) Use the Gaussian integral to discretize the governing equation in space as in Eq. (6).
 - (2) Use the Taylor series to expand the conservative variables within a cell and calculate the coefficient B_{ij} in Eq. (16), which only depends on the grid information.
 - (3) Initialize the flow field (assign initial values for the conservative variables to each cell).
 - (4) Interpolate the conservative variables $\mathbf{W}'_{L/R}$ to the left and right of the interfaces using Eq. (38).
 - (5) Calculate the distribution functions $g'_\alpha(\mathbf{x}_m, t)$ and lattice velocities ξ_α at the interfaces with Eqs. (39) and (34).
 - (6) Interpolate the base flow variables and perturbed variables tangential to the interface using the Roe average.
 - (7) Use Eq. (40) to calculate the fluxes \mathbf{F}'_{II} and Eq. (41) to calculate the conservative variables at the interfaces \mathbf{W}'_m .
 - (8) Solve for the primitive variables $(\rho'_m, u'_m, v'_m, p'_m)$ at the interfaces from \mathbf{W}'_m and calculate the fluxes \mathbf{F}'_I .
 - (9) Calculate the total fluxes with Eq. (46) and the relevant sources for specific problems. After that, the residual terms of Eq. (16) are obtained.
 - (10) Approach the conservative variables at the current time level and obtain them for the next time step.
- Repeat steps (4)–(10) until convergence is reached.

III. NUMERICAL EXAMPLES

In this section, several examples are simulated to validate the algorithm, including (i) a Gaussian pulse in a uniform flow; (ii) a Gaussian pulse scattered by a cylinder; (iii) the acoustic and vortical Gaussian pulse propagation problem; (iv) a pulsating source in a uniform flow; (v) a pulsating source and a solid wall; (vi) acoustic waves passing through a shear layer; (vii) acoustic waves scattered by multiple cylinders.

Case (i) is used to perform an accuracy test of the algorithm. Case (ii) tests the algorithm’s simulation of a single pulse interacting with a curved wall. The algorithm is tested for multiple waves (acoustic and vortical waves) propagating in a uniform flow by case (iii). Case (iv), with a periodic acoustic source, tests the algorithm in simulating waves generated by a pulsating source propagating in a uniform flow. Case (v) includes a pulsating source and a solid wall and tests the algorithm in simulating waves interacting with a wall in a uniform flow. Case (vi) simulates a pulsating source in a shear layer to test the algorithm’s robustness. Finally, case (vii) tests the algorithm’s simulation of a high-frequency pulsating source interacting with a solid wall.

The variables in these numerical cases are nondimensionalized as follows:

$$\rho^* = \frac{\rho}{\rho_\infty}, \quad u^* = \frac{u}{c_\infty}, \quad p^* = \frac{P}{\rho_\infty c_\infty^2}, \quad (48)$$

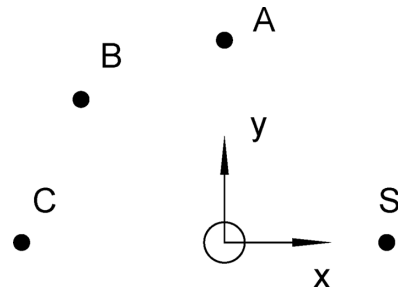


FIG. 6. Schematic showing the positions of the source (S), monitoring points (A, B, and C), and cylinder (located at the origin).

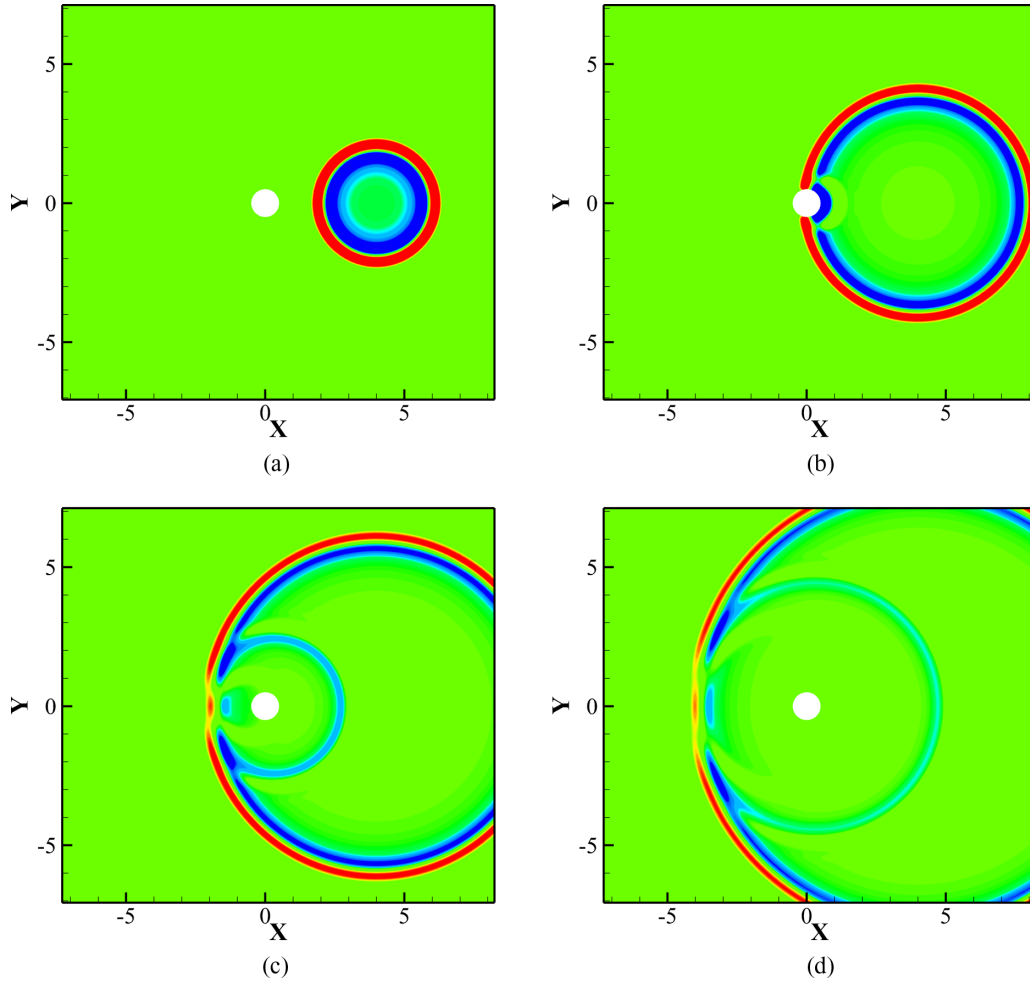


FIG. 7. Instantaneous perturbed density contours for a Gaussian source scattered by a cylinder at different times: (a) $t = 2$; (b) $t = 4$; (c) $t = 6$; (d) $t = 8$.

where the * superscript represents a dimensionless quantity, and the ∞ subscript represents the inflow variables. The sound speed is calculated as $c_\infty = \sqrt{\gamma p_\infty / \rho_\infty}$, where γ is the specific heat ratio. The * superscript is omitted in the following.

A. Propagation of a Gaussian pulse

This example simulates a pulse in a static flow field. The source terms of Eq. (1) are set to zero, while an initial disturbance is added with the following form:

$$\begin{aligned} \rho' = p' &= 0.01 \exp \left[-\ln(2) \frac{(x^2 + y^2)}{9} \right], \\ u' = v' &= 0. \end{aligned} \quad (49)$$

The dimensionless base flow is specified as $\bar{\rho} = 1$, $\bar{p} = 1/\gamma$, $\bar{u} = \bar{v} = 0$. The computational domain is $[-100, 100] \times [-100, 100]$ with $\Delta x = 1$, $\Delta x = 0.5$, and $\Delta x = 0.25$ grid sizes considered.

The instantaneous perturbed pressure at $t = 60$ for each grid size is compared with the analytical solution. For example, when the grid scale is $\Delta x = 0.5$, the pressure contours

and the instantaneous perturbed pressure distribution along the centerline $y = 0$ are shown in Fig. 3.

The L_1 error and L_2 error for each grid size are calculated and compared to perform an accuracy analysis, where the L_1 error and L_2 error are calculated as

$$\begin{aligned} L_1 \text{ error} &= \frac{1}{N_C} \sum |\rho' - \rho'_e|, \\ L_2 \text{ error} &= \sqrt{\frac{1}{N_C} \sum (\rho' - \rho'_e)^2}, \end{aligned} \quad (50)$$

where N_C is the number of the control cells, and subscript e represents the exact value. The variation of the logarithm of the L_1 error (solid red line) and L_2 error (dashed blue line) with the logarithm of grid sizes are compared with a line having slope $k = 4$ (solid black line) in Fig. 4. The algorithm approaches fourth-order accuracy because Gaussian integration and the fourth-order Taylor series expansion in Eq. (7) are used.

A base flow with a Mach number of 0.5 is also simulated. The initial condition and computational domain are the same as before. The four domain boundaries are specified with far-field conditions.

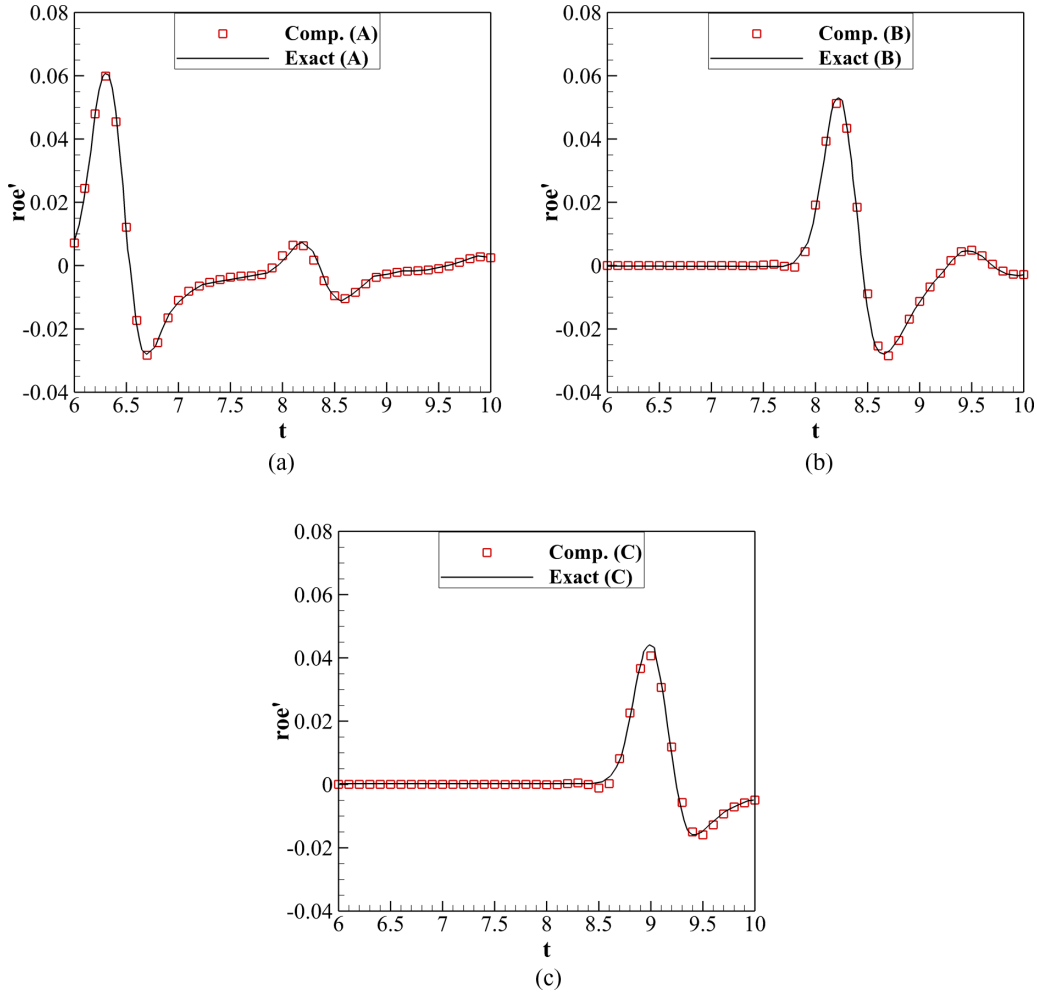


FIG. 8. Time histories of the perturbed density for a Gaussian source scattered by a cylinder at monitoring point (a) A; (b) B; (c) C.

In this situation, the pulse is transported with the base flow. In Fig. 5(a), the pressure contours at $t = 60$ show that the wave not only propagates due to the pressure gradient but also travels in the flow direction by convection. The instantaneous perturbed pressure along the centerline $y = 0$ at $t = 60$ is compared with the exact solution in Fig. 5(b), showing good consistency. This case indicates that the present method can also simulate waves propagating in a moving medium.

B. Gaussian pulse scattered by a cylinder

This case is a benchmark problem from the Second Computational Aeroacoustics Workshop [45] of a Gaussian pulse propagating and interacting with a solid curved boundary. As shown in Fig. 6, a cylinder with diameter $D = 1$ is located at the origin, an acoustic source is at the position $(4, 0)$, and three monitoring points A, B, C are located at $(0, 5)$, $[5 \cos(3\pi/4), 5 \sin(3\pi/4)]$, $(-5, 0)$, respectively. The grid size is 0.05, and a Gaussian pulse describes the source at the initial time:

$$\rho' = p' = \exp \left[-\ln 2 \frac{(x - 4)^2 + y^2}{0.04} \right]. \quad (51)$$

As shown in Fig. 7, the instantaneous density contours are displayed at dimensionless times $t = 2, t = 4, t = 6,$ and $t =$

8, from which it can be seen that the pulse propagates and interacts with the cylinder. The perturbed density at the three monitoring points at times from $t = 6$ to $t = 10$ are compared quantitatively with the exact solution in Fig. 8. The results agree well with the exact solution, showing that the algorithm has been validated for waves interacting with curved walls.

C. Long distance advection

In this case, an entropy wave propagates under a mean flow with Mach number 0.8; the entropy wave at the initial states

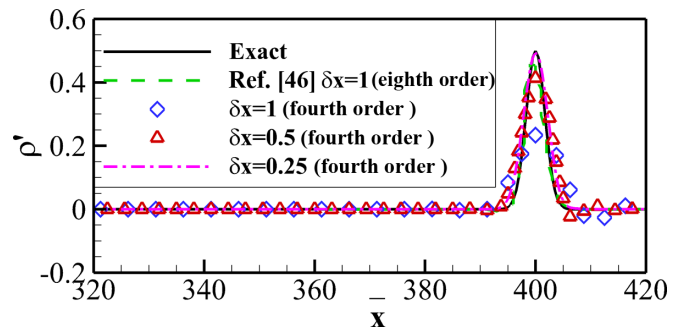


FIG. 9. The result of the long distance advection at $t = 400$ ($\bar{x} = x/Ma$).

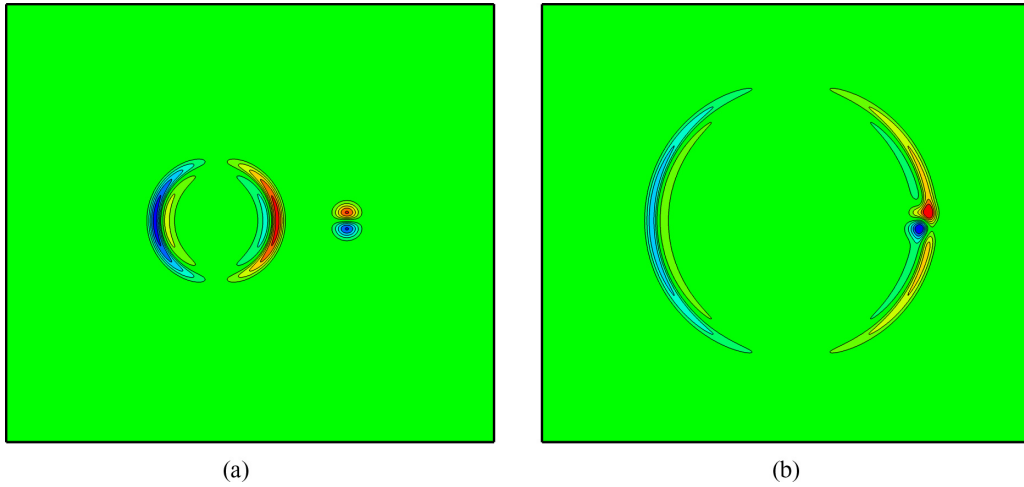


FIG. 10. Perturbed velocity contours u' for acoustic and vortical pulses in a uniform flow at times (a) $t = 30$; (b) $t = 70$.

is described as

$$\rho' = 0.5 \exp\left(-\ln 2 \frac{x^2 + y^2}{4}\right). \quad (52)$$

The base flow is a uniform flow with $\bar{\rho} = 1$, $\bar{p} = 1/\gamma$. Considering that the entropy wave will propagate in the base flow direction rather than all directions, the computational domain is a long and narrow rectangle $[-50, 450] \times [-50, 50]$ to save computational consumption. The grid scale is 1, 0.5, or 0.25 and the time step is 0.01. Figure 9 shows the result at $t = 400$ and the exact solution. A similar test is also presented in Ref. [46] which uses a compact eighth-order scheme and presents more accurate solutions.

D. Propagation of acoustic and vortical pulses in a uniform flow

In this case, a pressure pulse initiated at the origin and a vortical pulse initiated at position $(67, 0)$ are convected by a base flow with a Mach number of 0.5. The computational

domain is a square $[-200, 200] \times [-200, 200]$ with the grid size $\Delta x = \Delta y = 0.5$, and far-field boundary conditions are used. The pulses are described by

$$\begin{aligned} \rho' &= p' = \varepsilon_1 e^{-\alpha_1(x^2+y^2)}, \\ u' &= \varepsilon_2 y e^{-\alpha_2[(x-67)^2+y^2]}, \\ v' &= -\varepsilon_2(x-67) e^{-\alpha_2[(x-67)^2+y^2]}, \end{aligned} \quad (53)$$

with parameters $\varepsilon_1 = 0.01$, $\varepsilon_2 = 0.0004$, $\alpha_1 = \ln 2/9$, $\alpha_2 = \ln 2/25$.

The perturbed velocity contours u' at times $t = 30$ and $t = 70$ are shown in Fig. 10. The acoustic pulse and the vortical pulse propagate downstream due to the base flow at $Ma = 0.5$, and the acoustic pulse eventually catches up with the vortical pulse. The instantaneous perturbed velocities along the centerline $y = 0$ at times $t = 30$ and $t = 70$ are calculated and compared with the analytical solution in Figs. 11 and 12. The results agree well with the analytical solution, which

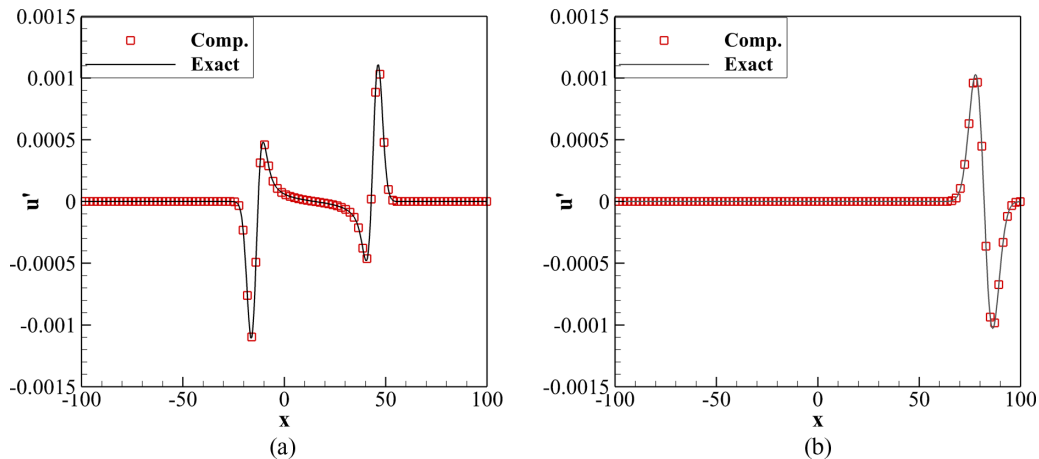


FIG. 11. Perturbed velocity along the centerline $y = 0$ for acoustic and vortical pulses in a uniform flow at time $t = 30$: (a) perturbed velocity u' ; (b) perturbed velocity v' .

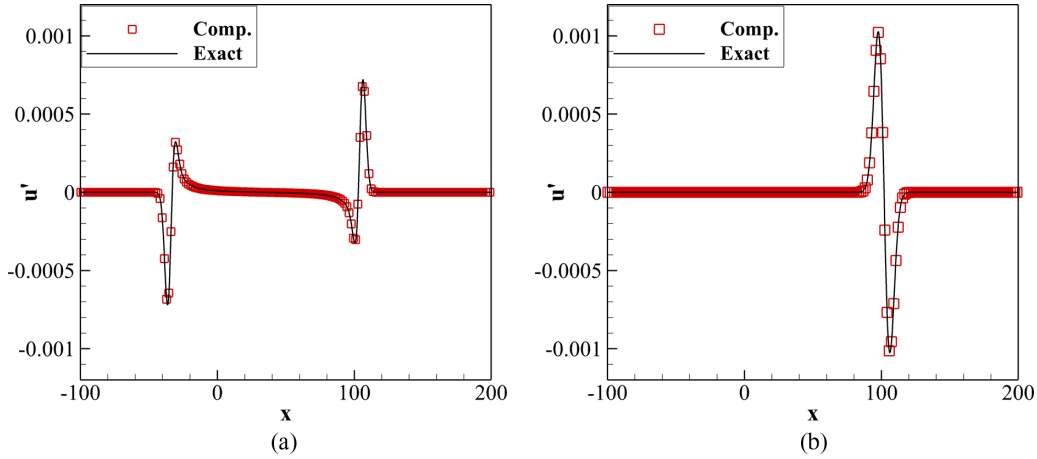


FIG. 12. Perturbed velocity along the centerline $y = 0$ for acoustic and vortical pulses in a uniform flow at time $t = 60$: (a) perturbed velocity u' ; (b) perturbed velocity v' .

validates the algorithm for simulating waves propagating and interacting with each other.

E. Propagation of a pulsating source in a uniform flow

The source terms in Eq. (5) must be defined to simulate periodic acoustic sources. In these cases, there are pulsating sources at the origin in a base flow with Mach numbers of 0.5 and 1.5. For a subsonic base flow ($Ma = 0.5$), the source terms are defined as

$$\begin{aligned} \rho'_s &= 0.5 \exp\left(-\ln 2 \frac{x^2 + y^2}{2}\right) \sin(\omega t), \\ p'_s &= \rho'_s c_\infty^2, \\ u'_s &= v'_s = 0, \end{aligned} \tag{54}$$

where the angular frequency ω equals $\pi/15$. The computational domain is a $[-100, 100] \times [-100, 100]$ square with grid size $\Delta x = \Delta y = 1$, and all boundaries are specified with far-field conditions.

The perturbed pressure contours at $t = 240$ are shown in Fig. 13. The waves are generated at the origin and propagate outward with a Doppler effect due to the base flow at Mach number 0.5. The instantaneous perturbed pressure distribution along the centerline $y = 0$ at that time is compared with the exact solution in Fig. 14, showing good consistency.

For a supersonic base flow, the periodic source is specified at position $(-50, 0)$, and is defined as

$$\begin{aligned} \rho'_s &= 0.5 \exp\left[-\ln 2 \frac{(x + 50)^2 + y^2}{2}\right] \sin(\omega t), \\ p'_s &= \rho'_s c_\infty^2, \\ u'_s &= v'_s = 0, \end{aligned} \tag{55}$$

where the angular frequency is $\pi/15$. A larger square domain $[-400, 400] \times [-400, 400]$ with the same grid size and a longer computational time is used to obtain a more comprehensive flow field range. The perturbed pressure contours at $t = 304$ are shown in Fig. 15. The waves form a Mach cone because of the supersonic effect (the disturbance does not propagate upstream in a supersonic flow).

The instantaneous perturbed pressure distribution along the centerline $y = 0$ at $t = 304$ is compared quantitatively with

the exact solution in Fig. 16. The simulation and exact results are both displayed using lines for clarity. It can be seen that the computational and exact results agree well except for the sound source point $(-50, 0)$, where a singularity exists. Reference [47] also shows this phenomenon.

These simulations validate the use of source terms in the algorithm and its ability to simulate a pulsating source in a subsonic or supersonic base flow.

F. Interaction of a pulsating source with a solid wall

The waves generated from a pulsating source interacting with a solid wall in a uniform flow at Mach number 0.5 are considered in this case. The source terms are the same as those described in Sec. II E. The computational domain is a $[-100, 200] \times [-100, 130]$ rectangle, and the grid size is $\Delta x = \Delta y = 1$. Other than the bottom boundary, which is

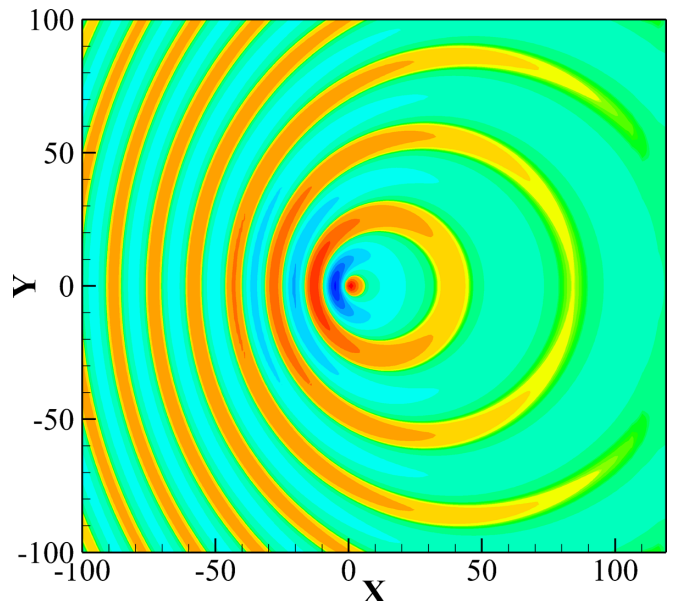


FIG. 13. Perturbed pressure contours for a pulsating source in a subsonic uniform flow ($Ma = 0.5$) at time $t = 240$.

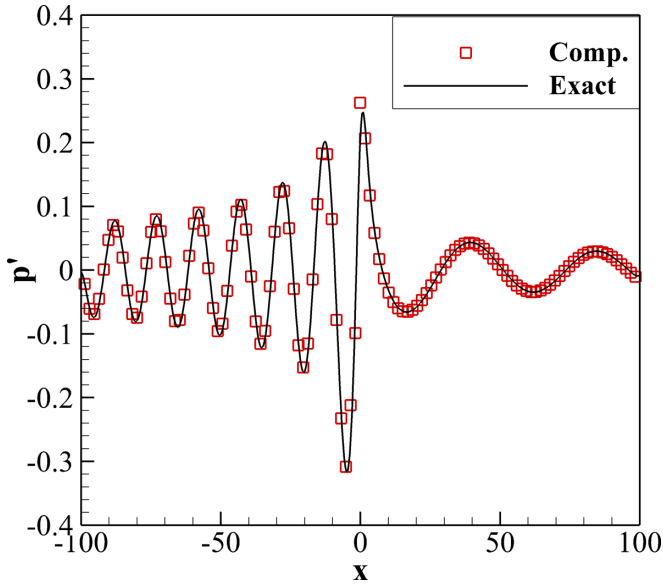


FIG. 14. Instantaneous perturbed pressure distribution along the centerline $y = 0$ for a pulsating source in a subsonic uniform flow ($Ma = 0.5$) at time $t = 240$.

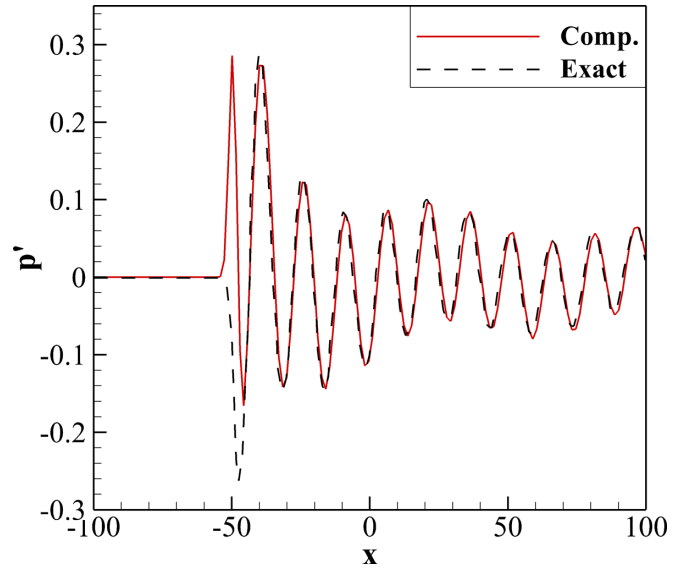


FIG. 16. Instantaneous perturbed pressure distribution along the centerline $y = 0$ for a pulsating source in a supersonic uniform flow ($Ma = 1.5$) at time $t = 304$.

specified as impermeable, all boundaries are specified with far-field conditions.

The instantaneous perturbed pressure contours at $t = 90$ are displayed in Fig. 17. The wave structures are captured clearly, convecting downstream in the uniform flow and reflecting from the solid wall. The perturbed pressure distribution along the line $x = 0$ at $t = 90$ is compared with the exact solution in Fig. 18. The simulation results are in good agreement with the exact solution except for the sound source position because the analytical solution (semi-infinite two-dimensional Green’s function) is singular at the origin

[1]. This case validates the algorithm for simulating waves interacting with a wall in a uniform flow.

G. Acoustic waves passing through a shear layer

This problem is the test case of Bogey *et al.* [48], which describes periodic acoustic waves passing through a shear layer. The background velocity field is fitted from the time-average results of a large eddy simulation:

$$\begin{aligned} \bar{u} &= \frac{\bar{u}_1 + \bar{u}_2}{2} + \frac{\bar{u}_2 - \bar{u}_1}{2} \tanh [2y/\delta_w(x)], \\ \bar{v} &= 0, \\ \bar{p} &= 1, \\ \bar{p} &= 1/\gamma, \end{aligned} \tag{56}$$

where $\bar{u}_1 = 0.118$, $\bar{u}_2 = 0.470$, and $\delta_w(x)$ is the vorticity thickness taken as

$$\delta_w(x) = \frac{3}{2} + \frac{1}{2} \tanh [(x - 70)/10]. \tag{57}$$

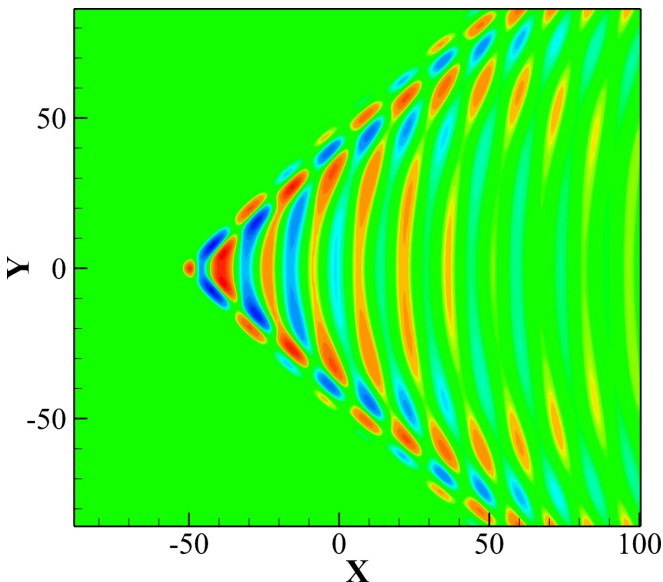


FIG. 15. Perturbed pressure contours for a pulsating source in a supersonic uniform flow ($Ma = 1.5$) at time $t = 304$.

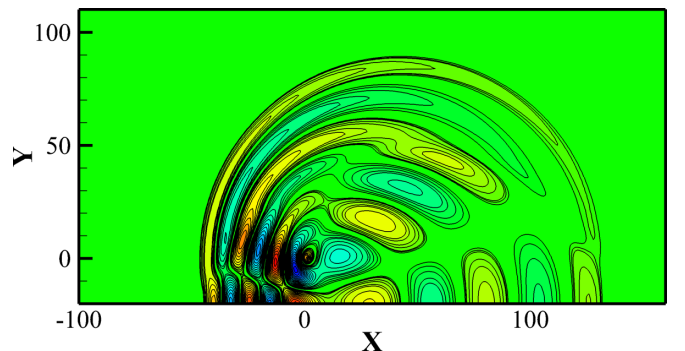


FIG. 17. Perturbed pressure contours at $t = 90$ for a pulsating source interacting with a solid wall.

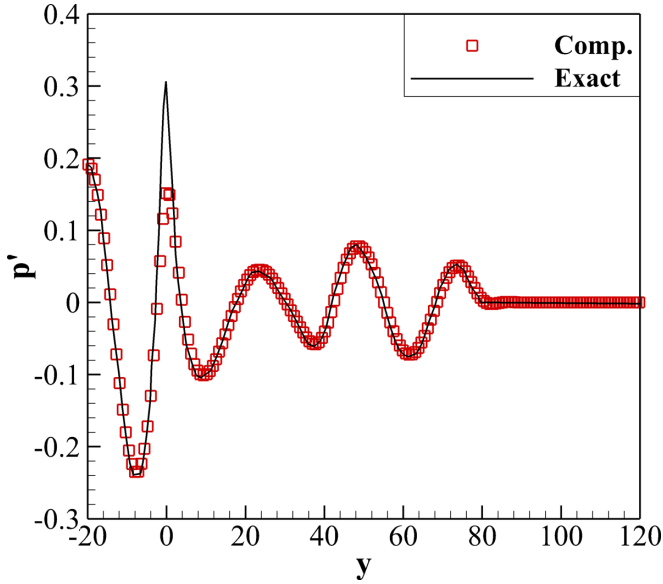


FIG. 18. Perturbed pressure distribution along the line $x = 0$ for a pulsating source interacting with a solid wall at $t = 90$.

The source terms are defined as

$$\begin{aligned} \rho'_s &= \varepsilon \exp \left[-\ln 2 \frac{(x - 70)^2 + y^2}{(3\Delta)^2} \right] \sin(\omega t), \\ \rho'_s &= p'_s / c_\infty^2, \\ u'_s &= v'_s = 0, \end{aligned} \quad (58)$$

where the amplitude is $\varepsilon = 10^{-4}$, the half width of the Gaussian profile is 3Δ , $\Delta = 0.24$, and the angular frequency is $\omega = 2\pi c_\infty / \lambda$. The wavelength λ is chosen as 51.5, and the sound source position is $(70, 0)$. The computational domain is a $[0, 300] \times [-200, 200]$ rectangle with a coarser outer sponge layer grid to absorb the waves passing through the boundaries.

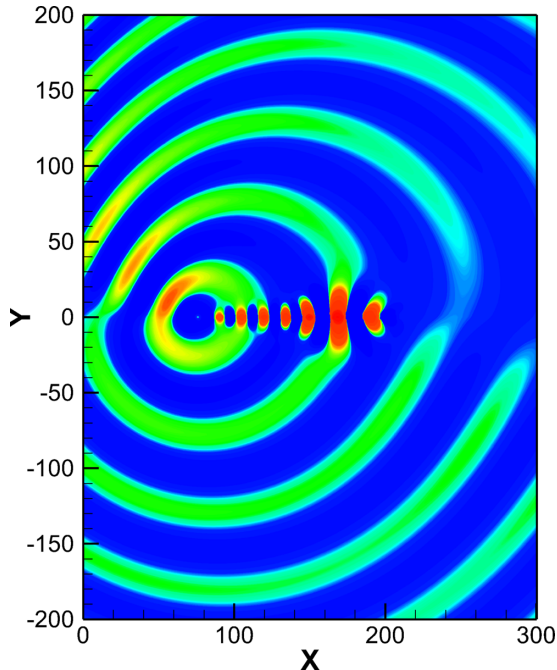


FIG. 19. Instantaneous pressure contours for acoustic waves passing through a shear layer at $t = 342$.

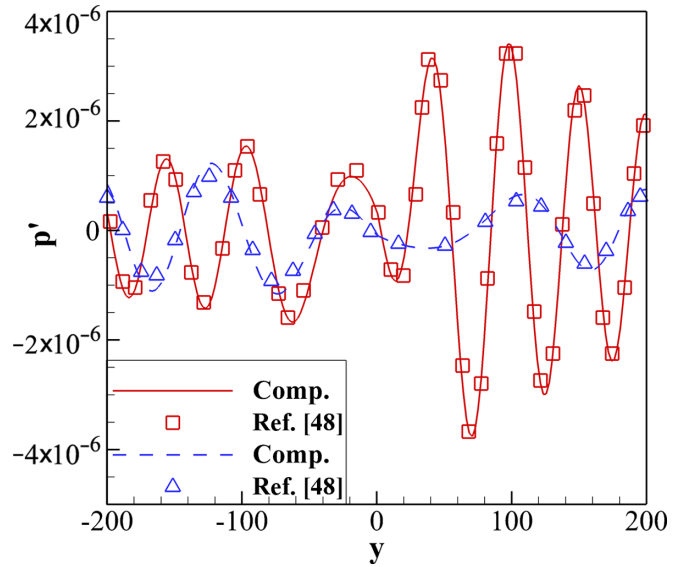


FIG. 20. Transverse perturbed pressure profiles for acoustic waves passing through a shear layer at $t = 342$ (solid line, squares, $x = 0$; dashed line, triangles $x = 300$). Results shown as symbols (squares and triangles) are from Bogey *et al.* [48].

A uniform 301×401 grid is used in the rectangular domain, and the dimensionless time step is $\Delta t = 0.1$.

The instantaneous perturbed pressure contours at $t = 342$ are shown in Fig. 19. Some instability waves are generated in the shear layer and convected downstream. This phenomenon can be explained by classical Kelvin-Helmholtz theory, which states that inviscid and nonthickness shear flow is always unstable [3]. In addition, the results of Bogey *et al.* [48] also demonstrate this phenomenon. The instantaneous perturbed pressure distributions along the lines $x = 0$ (red solid line and squares) and $x = 300$ (blue dashed line and triangles) are extracted and compared with the Bogey *et al.* [48] results in Fig. 20. The calculated and the reference results show good agreement, demonstrating the robustness of the present method in simulating waves propagating in a shear flow.

H. Acoustic waves scattered by multiple cylinders

As shown in Fig. 21, two cylinders are positioned at $(-4, 0)$ and $(4, 0)$ with diameters $D_1 = 1.0$ and $D_2 = 0.5$. An acoustic source, located at the origin, is defined as follows,

$$\begin{aligned} \rho'_s &= \exp \left[-\ln 2 \frac{x^2 + y^2}{0.04} \right] \sin(\omega t), \\ \rho'_s &= p'_s, \\ u'_s &= v'_s = 0, \end{aligned} \quad (59)$$

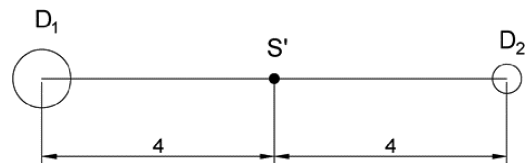


FIG. 21. Schematic of the model for acoustic wave scattering by multiple cylinders.

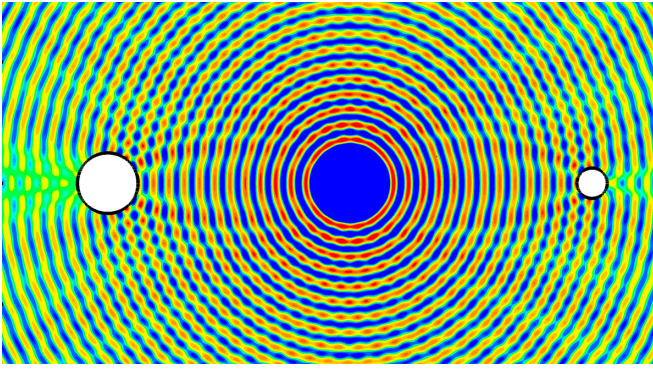


FIG. 22. Snapshot of the perturbed pressure contours for acoustic wave scattering by multiple cylinders.

where the angular frequency $\omega = 8\pi$. The computational domain is divided into a computational region and a sponge grid layer. A finer grid with a 0.01 grid size is chosen for the computational domain, and a grid growing exponentially coarser is used in the sponge layer to prevent wave reflection from the boundaries. The time step is chosen as 0.001.

A snapshot of the perturbed pressure contours is shown in Fig. 22. The high-frequency periodic waves are captured clearly. The root mean square of the perturbed pressure a distance 0.02 away from the cylinder surfaces is plotted and compared to the analytical solution in Fig. 23, which shows that the computed results agree well with the exact solution. This case validates the algorithm to simulate a high-frequency pulsating source that propagates and interacts with multiple cylinders.

IV. CONCLUSIONS

A linear discrete velocity model (L-D1Q4) is developed and implemented in a LBFS to calculate the fluxes of the LEE. First, the expressions for the equilibrium distribution function and the lattice velocity of the linear discrete velocity model are derived from the moment relations. Second, by comparing the LLBE and the LEE, the flux expression in terms of the mesoscopic variables can be derived based on CE analysis and the moment relations. Finally, the L-D1Q4-based LBFS is used to calculate the LEE fluxes.

In addition, the high-precision LSFD FV method is used to discretize the governing equations to make the algorithm fourth-order accurate, and the present method is used to simulate acoustic propagation in fluids. The algorithm inherits the advantage of the LBFS in that the flux is constructed from the local solution of the Boltzmann equation, and it can be

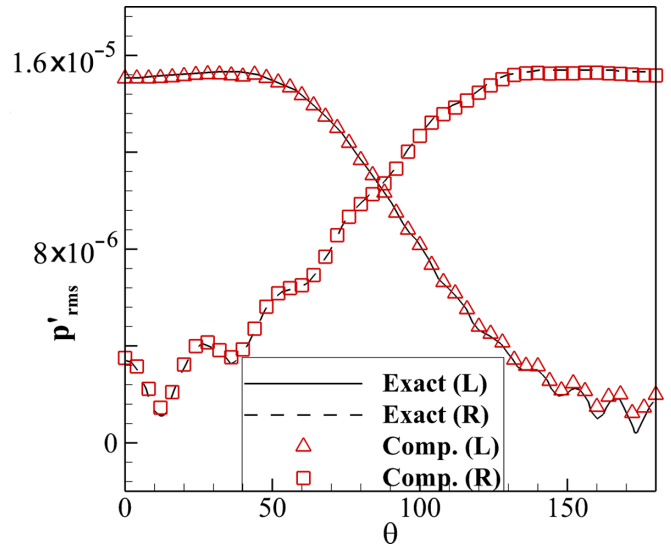


FIG. 23. Root mean square of the perturbed pressure of the scattered waves a distance 0.02 away from the cylinder surfaces (L: left cylinder; R: right cylinder).

conveniently extended to a multidimensional scheme. Furthermore, the algorithm provides an approach for the lattice Boltzmann model-based flux solver to calculate fluxes for linear equations.

Various cases, including a single pulse propagating and interacting with a vorticity pulse and solid wall in a uniform flow, and a pulsating source propagating in subsonic, supersonic, and shear flow, and interacting with multiple solid walls, are simulated to validate the accuracy and robustness of the present method. The calculated results demonstrate very good consistency with the reference results, and the convergence analysis shows that it can achieve fourth-order accuracy.

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ACKNOWLEDGMENTS

The work was supported by the National Natural Science Foundation of China under Grants No. 12072305 and No. 11602209, Aeronautical Science Foundation under Grant No. 20200057068001, National Key Laboratory of Science and Technology on Aerodynamic Design and Research under Grant No. 614220121030202, and Key Laboratory of Aerodynamic Noise Control under Grant No. ANCL20210201.

- [1] A. McAlpine, C. J. Powles, and B. J. Tester, A weak-scattering model for turbine-tone haystacking, *J. Sound Vib.* **332**, 3806 (2013).
- [2] A. Agarwal, P. J. Morris, and R. Mani, Calculation of sound propagation in nonuniform flows: Suppression of instability waves, *AIAA J.* **42**, 80 (2004).

- [3] R. J. Astley, Numerical methods for noise propagation in moving flows, with application to turbofan engines, *Acoust. Sci. Technol.* **30**, 227 (2009).
- [4] G. H. Xu, Y. J. Shi, Q. J. Zhao, Y. Wang, and F. Fan, New research progress in helicopter rotor aerodynamic noise, *Acta Aeronaut. Astronaut. Sin.* **38**, 16 (2017).

- [5] H. Ji, X. Xu, X. Guo, S. Ye, J. Chen, and X. Yang, Direct FVM simulation for sound propagation in an ideal wedge, *Shock Vib.* **2016**, 3703974 (2016).
- [6] S. Redonnet, E. Manoha, and P. Sagaut, Numerical simulation of propagation of small perturbations interacting with flows and solid bodies, in *7th AIAA/CEAS Aeroacoustics Conference and Exhibit* (AIAA, Reston, VA, 2001), p. 2223.
- [7] J. M. Pérez, A. Aguilar, and V. Theofilis, Lattice Boltzmann methods for global linear instability analysis, *Theor. Comput. Fluid Dyn.* **31**, 643 (2017).
- [8] B. Kaltenbacher and V. Nikoli, The inviscid limit of third-order linear and nonlinear acoustic equations, *SIAM J. Appl. Math.* **81**, 1461 (2021).
- [9] Q. Wang and Z. Zhang, A modified immersed finite volume element method for elliptic interface problems, *ANZIAM J.* **62**, 42 (2020).
- [10] S. S. Kim, C. Kim, O. H. Rho, and S. K. Hong, Cure for shock instability: Development of a shock-stable Roe scheme, *Comput. Phys. Commun.* **185**, 342 (2003).
- [11] G.-S. Yeom, K.-S. Chang, and S. W. Baek, Robust WAF-HLL scheme for compressible two-pressure two-velocity multiphase flow model, *Eng. Appl. Comput. Fluid Mech.* **6**, 144 (2012).
- [12] D. S. Balsara and M. Dumbser, Multidimensional Riemann problem with self-similar internal structure. Part II—Application to hyperbolic conservation laws on unstructured meshes, *J. Comput. Phys.* **287**, 269 (2015).
- [13] Y. Wang, C. Shu, L.-M. Yang, and H.-Z. Yuan, Development of axisymmetric lattice Boltzmann flux solver for complex multiphase flows, *Mod. Phys. Lett. B* **32**, 1840005 (2018).
- [14] L. M. Yang, C. Shu, W. M. Yang, and J. Wu, A simple gas kinetic scheme for simulation of 3D incompressible thermal flows, *Numer. Heat. Transfer, Part B* **72**, 450 (2017).
- [15] W. J. Sun, S. Jiang, and K. Xu, An asymptotic preserving implicit unified gas kinetic scheme for frequency-dependent radiative transfer equations, *Int. J. Numer. Anal. Model.* **15**, 134 (2018).
- [16] Y. Zhang, L. Zhu, P. Wang, and Z. Guo, Discrete unified gas kinetic scheme for flows of binary gas mixture based on the McCormack model, *Phys. Fluids* **31**, 017101 (2018).
- [17] L. M. Yang, C. Shu, J. Wu, and Y. Sun, Comparative study of 1D, 2D and 3D simplified gas kinetic schemes for simulation of inviscid compressible flows, *Appl. Math. Modell.* **43**, 85 (2017).
- [18] C. Zhang, Q. B. Li, S. Fu, and Z. J. Wang, A third-order gas-kinetic CPR method for the Euler and Navier–Stokes equations on triangular meshes, *J. Comput. Phys.* **363**, 329 (2018).
- [19] S. Z. Chen, Z. L. Guo, and K. Xu, A simplification of the unified gas kinetic scheme, *Phys. Rev. E* **94**, 023313 (2016).
- [20] M. Righi, A gas-kinetic scheme for the simulation of turbulent flows, in *Proceedings of the 29th International Symposium on Rarefied Gas Dynamics*, AIP Conf. Proc. No. 1628 (AIP, Melville, NY, 2014), pp. 1363–1370.
- [21] N. Y. Zhan, R. Q. Chen, and Y. C. You, Meshfree method based on discrete gas-kinetic scheme to simulate incompressible/compressible flows, *Phys. Fluids* **33**, 017112 (2021).
- [22] L. Pan L and K. Xu, A third-order compact gas-kinetic scheme on unstructured meshes for compressible Navier-Stokes solutions, *J. Comput. Phys.* **318**, 327 (2016).
- [23] L. M. Yang, C. Shu, and J. Wu, Extension of lattice Boltzmann flux solver for simulation of 3D viscous compressible flows, *Comput. Math. Appl.* **71**, 2069 (2016).
- [24] N. Y. Zhan, R. Q. Chen, J. Q. Liu, R. Qiu, and Y. You, Meshfree lattice Boltzmann flux solver for compressible inviscid flows, *Int. J. Numer. Methods Fluids* **93**, 1378 (2020).
- [25] Y. Y. Liu, C. Shu, H. W. Zhang, and L. M. Yang, A high order least square-based finite difference-finite volume method with lattice Boltzmann flux solver for simulation of incompressible flows on unstructured grids, *J. Comput. Phys.* **401**, 109019 (2019).
- [26] Y. Y. Liu, C. Shu, H. W. Zhang, and L. M. Yang, A high-order implicit least square-based finite difference-finite volume method for incompressible flows on unstructured grids, *Phys. Fluids* **33**, 053601 (2021).
- [27] N. Y. Zhan, R. Q. Chen, and Y. C. You, Discrete gas-kinetic scheme-based arbitrary Lagrangian-Eulerian method for moving boundary problems, *Phys. Fluids* **33**, 067101 (2021).
- [28] K. Xu, A gas-kinetic BGK scheme for the Navier–Stokes equations and its connection with artificial dissipation and Godunov method, *J. Comput. Phys.* **171**, 289 (2001).
- [29] L. M. Yang, C. Shu, and J. Wu, A simple distribution function-based gas-kinetic scheme for simulation of viscous incompressible and compressible flows, *J. Comput. Phys.* **274**, 611 (2014).
- [30] E. Vergnault, O. Malaspinas, and P. Sagaut, A lattice Boltzmann method for nonlinear disturbances around an arbitrary base flow, *J. Comput. Phys.* **231**, 8070 (2012).
- [31] K. Qu, C. Shu, and Y. T. Chew, Simulation of shock-wave propagation with finite volume lattice Boltzmann method, *Int. J. Mod. Phys. C* **18**, 447 (2007).
- [32] L. M. Yang, C. Shu, and J. Wu, Development and comparative studies of three non-free parameter lattice Boltzmann models for simulation of compressible flows, *Adv Appl. Math. Mech.* **4**, 454 (2012).
- [33] L. M. Yang, C. Shu, and J. Wu, A moment conservation-based non-free parameter compressible lattice Boltzmann model and its application for flux evaluation at cell interface, *Comput. Fluids* **79**, 190 (2013).
- [34] Y. Y. Liu, L. M. Yang, C. Shu C, and H. W. Zhang, Three-dimensional high-order least square-based finite difference-finite volume method on unstructured grids, *Phys. Fluids* **32**, 123604 (2020).
- [35] Y. Y. Liu, H. W. Zhang, L. M. Yang, and C. Shu, High-order least-square-based finite-difference-finite-volume method for simulation of incompressible thermal flows on arbitrary grids, *Phys. Rev. E* **100**, 063308 (2019).
- [36] Y. L. Liu, W. W. Zhang, and C. N. Li, A novel multi-dimensional limiter for high-order finite volume methods on unstructured grids, *Commun. Comput. Phys.* **22**, 1385 (2017).
- [37] W. X. Wu, C. Shu, and C. M. Wang, Mesh-free least-squares-based finite difference method for large-amplitude free vibration analysis of arbitrarily shaped thin plates, *J. Sound Vib.* **317**, 955 (2008).
- [38] T. J. Barth and P. Frederickson, Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction, in *Proceedings of the 28th Aerospace Sciences Meetings* (AIAA, 1990), p. 13.
- [39] A. J. Jameson, W. Schmidt, and E. Turkel, Numerical solution of the Euler equations by finite volume methods using

- Runge-Kutta time-stepping schemes, in *14th Fluid and Plasma Dynamics Conference* (AIAA, 1981).
- [40] S. Gottlieb and S. E. Tadmor, Strong stability-preserving high-order time discretization methods, *SIAM Rev.* **43**, 89 (2001).
- [41] A. Jameson, Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings, in *10th Computational Fluid Dynamics Conference* (AIAA, 1991).
- [42] P. L. Bhatnagar, E. P. Gross, and M. K. Krook, A model for collision processes in gases I: Small amplitude processes in charged and neutral one-component systems, *Phys. Rev.* **94**, 511 (1954).
- [43] S. Chapman and T. G. Cowling, The mathematical theory of non-uniform gases, *Am. J. Phys.* **30**, 389 (1962).
- [44] K. Qu, Development of lattice Boltzmann method for compressible flows, Doctoral dissertation, National University of Singapore, 2008.
- [45] Second Computational Aeroacoustics (CAA) Workshop on Benchmark Problems, edited by C. K. W. Tam and J. C. Hardin, NASA Conference Publication 3352, 1997.
- [46] F. Zhao, X. Ji, W. Shyy *et al.*, An acoustic and shock wave capturing compact high-order gas-kinetic scheme with spectral-like resolution, *Int. J. Comput. Fluid Dyn.* **34**, 731 (2020).
- [47] C. Bailly and D. Juvé, Numerical solution of acoustic propagation problems using linearized Euler equations, *AIAA J.* **38**, 22 (2000).
- [48] C. Bogey, C. Bailly, and D. Juvé, Computation of flow noise using source terms in linearized Euler's equations, *AIAA J.* **40**, 235 (2002).