Asymptotic method for entropic multiple relaxation time model in lattice Boltzmann method

Xiangshuo Tang [*]
Department of Mechanical Engineering and Mechanics, Lehigh University, Bethlehem, Pennsylvania 18015, USA

Yue Yu
Department of Mathematics, Lehigh University, Bethlehem, Pennsylvania 18015, USA

Alparslan Oztekin
Department of Mechanical Engineering and Mechanics, Lehigh University, Bethlehem, Pennsylvania 18015, USA

(Received 29 October 2021; accepted 9 June 2022; published 7 July 2022)

To improve the numerical stability of the lattice Boltzmann method, Karlin *et al.* [Phys. Rev. E **90**, 031302(R) (2014)] proposed the entropic multiple relaxation time (EMRT) collision model. The idea behind EMRT is to construct an optimal postcollision state by maximizing its local entropy value. The critical step of the EMRT model is to solve the entropy maximization problem under certain constraints, which is often computationally expensive and even not feasible. In this paper, we propose to employ perturbation theory and obtain an asymptotic solution to the maximum entropy state. With mathematical analysis of particular cases under relaxed constraints, we obtain the unperturbed form of the original problem and derive the asymptotic solution. We show that the asymptotic solution well approximates the optimal states; thus, our approach provides an efficient way to solve the constrained maximum entropy problem in the EMRT model. Also, we use the same idea of the EMRT model for the initial condition of the distribution function and propose to leave the entropy function to determine the missing information at the initial nodes. Finally, we numerically verify that the simulation results of the EMRT model obtained via the perturbation theory agree well with the exact solution to the Taylor-Green vortex problem. Furthermore, we also demonstrate that the EMRT model exhibits excellent stability performance for under-resolved simulations in the doubly periodic shear layer flow problem.

DOI: 10.1103/PhysRevE.106.015303

I. INTRODUCTION

The lattice Boltzmann method (LBM) is an effective computational tool to simulate fluid flows [1]. The LBM represents each computational node by distribution functions. Distribution functions could be considered as the density distribution of microscopic particles. The evolution of this kinetic system could recover the Navier-Stokes (N-S) equations in its hydrodynamic limit at a small Mach number (Ma). The advantage of this numerical scheme is its feature of locality, i.e., the computation is performed almost locally at each fluid node. It makes the LBM easy to be implemented for parallel computing.

The most commonly used collision model in LBM is the Bhatnagar-Gross-Krook (BGK) model [2,3]. However, this model would become numerically unstable as the Reynolds number (Re) increases. To improve the numerical stability of LBM, different collision models have been developed.

In the BGK model, there is only one relaxation parameter, and everything will relax to the equilibrium state at the same rate. The multiple relaxation time (MRT) collision model [4] proposes a generalization of the BGK model such that different moments of distribution functions could relax to the equilibrium state at different rates. In the MRT model, to recover the desired fluid viscosity, one only needs to fix the relaxation parameters up to the second-order moments. The relaxation parameters can be tuned without restrictions for the higher-order moments since they would not affect the recovery of N-S equations, which offers extra flexibility for the model. However, one drawback of the MRT model is the lack of a clear physical interpretation of these higher-order moments. Therefore, parameters tuning in the MRT model often rely on one's experiences rather than the physical interpretation.

The entropic collision model successfully derived the entropy function of the discrete system in the LBM [5,6]. The remarkable thing about this derivation is that it introduces the second law of thermodynamics into the LBM. Before introducing the entropy function, the constraints on collision are only about the conservation of mass and momentum. The entropy function expands our understanding of the collision process and helps us predict the direction of time in which the collision would most probably proceed. The entropic model is initially built on the BGK collision model (EBGK), where there is only a single relaxation parameter. The idea is to let the collision process always increase the value of local entropy [5,7]. To realize this idea, it first finds the boundary of the maximum step size, within which the entropy function is guaranteed to increase along the direction of the BGK collision model. Then it rescales the relaxation parameter by this maximum step size to let the postcollision state remain

[*] xit213@alum.lehigh.edu

within the boundary. The entropic BGK (EBGK) model exhibits excellent numerical stability. However, this model has a potential accuracy issue. Because the EBGK needs to dynamically rescale the only relaxation parameter, it would inevitably modify the local viscosity during the collision. Also, it is computationally expensive to find the boundary of the maximum step size. Several optimization strategies have been proposed to improve efficiency [8–10].

Karlin, Bösch, and Chikatamarla [11,12] proposed an entropic multiple relaxation time (EMRT) collision model. This model combines the idea of a multiple relaxation time model and entropy function. As mentioned above, the MRT model allows us to freely tune higher-order moments' relaxation parameters, while the issue is that the higher-order moments lack the physical interpretation. The EMRT model looks for the physical interpretation of the entropy function. It proposes to set the postcollision state to be such a state that the entropy function could reach the local maximum value. To recover the desired N-S equation, we need to fix the relaxation parameters up to the second-order moments during the collision. For the relaxation of higher-order moments, it leaves the entropy function to determine the optimal values. Thus, the postcollision states would be the local maximum entropy states under certain constraints. Compared to the EBGK model, the EMRT model addresses the potential accuracy issue because the tuning of higher-order moments would not affect fluid viscosity. It is worth mentioning that there are subtle differences between how EMRT and EBGK models treat the entropy functions. The EBGK model ensures that the postcollision state would always lie within the boundary, and the entropy value would guarantee to increase after the collision step. On the other hand, the EMRT model is like a "best-effort method": it fixes relaxation parameters up to the second-order moments and then does its best to make the postcollision state's entropy as large as possible. Wilson *et al.* [13] also provided another interpretation of the entropy function based on the "Principle of Minimum Discrimination Information". They derived a similarly constrained minimization of the Kullback-Leibler Divergence problem.

The key to realizing the EMRT model is to solve the constrained entropy maximization problem, i.e., seeking the maximum entropy state under constraints of the density, momentum, and second-order moments. However, the entropy maximization is a challenging problem to be solved. To the best of our knowledge, the exact solution to this problem is still not available. One option is to look for the numerical solutions to get the approximations. For example, we could use gradient descent methods or Newton's method [14] to solve the problem. However, the numerical methods are computationally expensive. They generally need to take several iterations for the solutions to be converged. To improve the efficiency, Wilson *et al.* solved the constrained optimization problem by only taking a single step of Newton's method [13]. This strategy is helpful, and we will show that a single Newton's step could give a good approximation based on a reasonable initial guess. However, it still requires us to construct and solve a 3×3 matrix. Moreover, to implement the EMRT model, we need to repeatedly solve the constrained entropy maximization problem at each node and time step. Therefore, developing a more efficient algorithm is necessary.

Karlin *et al.* [11,12] first assumed that all the higher-order moments relax to an equilibrium state at the same rate to seek the approximate solutions. By this assumption, there would be two relaxation parameters in the collision. A fixed one is for the relaxation up to the second-order moments, and an adaptive one is for the higher-order ones. A higher-order relaxation parameter function represents the entropy value of the postcollision state, and it would reach the maximum value at its critical point (i.e., the derivative is zero). Although it is difficult to find the exact solution to the critical point, Karlin *et al.* take the Taylor expansion on the derivative of entropy function around the local equilibrium state, simplify the expression by removing all the terms which are proven to be the value of zero, and get the approximate solution by setting the first nonzero term to zero. This approximation to the higher-order relaxation parameter is also referred to as the KBC stabilizer. The idea of a KBC stabilizer could be further applied to the general form, where the higher-order moments would have different relaxation parameters [15]. However, this generalization is inefficient because it requires the inverse of the "correlation matrix" to approximate the optimal values for the higher-order relaxation parameters. Recently, Krämer *et al.* [16] attempted to solve the entropy maximization problem in its general form. They proposed to take the Taylor expansion of the entropy function around the global equilibrium state (i.e., the equilibrium state under the unit fluid density and zero velocities) and make a quadratic approximation. This entropy function is slightly different from the original one and is referred to as the "quadratic pseudoentropy."

In this paper, we propose to apply perturbation theory [17,18] to solve the entropy maximization problem in its general form, where all the higher-order moments could relax to the equilibrium states at their rates. The advantage of asymptotic solutions is that it represents the solutions in the analytical forms; thus, it avoids the iterative scheme in the numerical methods and would be more efficient. At the same time, the asymptotic solutions could give good approximations of the maximum entropy states in practice. The asymptotic solutions are the main results of this paper, and we develop an efficient algorithm to implement the idea of the EMRT model. We will discuss discovering the "unperturbed" form of the original problem by analyzing particular cases under relaxed constraints. Furthermore, we will show the detailed derivation of the asymptotic solution and compare it to Newton's method, which is considered as a reference solution.

Another benefit of solving the constrained entropy maximization problem in its general form is that it could help us develop the initial and boundary conditions [19,20]. Of course, there could be a problem with imposing initial and boundary conditions since only the moments up to the second order are available initially and at the boundary nodes. However, we could still apply the idea of the EMRT model and let the entropy function determine the missing information about the higher-order moments. More specifically, we could assume the distribution functions for all nodes initially and boundary nodes at all times would be the states with maximum entropy under constraints up to the second-order moments. To implement this idea for the initial and boundary

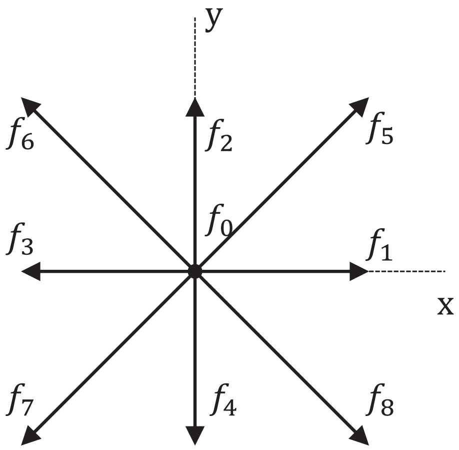


FIG. 1. Particle velocities in the D2Q9 lattice model.

conditions, we need to solve the entropy maximization problem in its general form.

II. LATTICE BOLTZMANN METHOD

This paper considers the popular two-dimensional nine-velocity lattice model (D2Q9), where all the microscopic particles are constrained to nine discrete velocities (see Fig. 1).

The distribution function $f_i(\boldsymbol{x}, t)$ represents the density distribution function for the microscopic particle with discrete velocity $\boldsymbol{v}_i$ at lattice nodes $\boldsymbol{x}$ for the time step t. At each computational node, it is convenient to consider all nine distribution functions as a vector $\boldsymbol{f}(\boldsymbol{x}, t) \in \mathbb{R}^9$. The equation of LBM with BGK collision model [2,3] writes

$$\boldsymbol{f}(\boldsymbol{x}+\boldsymbol{v}, t+1) = \boldsymbol{f}^{\text{post}}(\boldsymbol{x}, t) = \boldsymbol{f}(\boldsymbol{x}, t) + \omega[\boldsymbol{f}^{\text{equ}}(\boldsymbol{x}, t) - \boldsymbol{f}(\boldsymbol{x}, t)], \tag{1}$$

where $\boldsymbol{f}^{\text{post}}(\boldsymbol{x}, t)$ and $\boldsymbol{f}^{\text{equ}}(\boldsymbol{x}, t)$ are the postcollision states and local equilibrium states, respectively. The scalar ω is the relaxation parameter in the BGK model, which is associated with the fluid viscosity following the relation

$$\nu = c_s^2\left(\frac{1}{\omega} - \frac{1}{2}\right), \tag{2}$$

where $c_s = 1/\sqrt{3}$ is the speed of sound in this lattice model. The local equilibrium states $\boldsymbol{f}^{\text{equ}}(\boldsymbol{x}, t)$ has the same fluid density ρ and velocities $\boldsymbol{u}$ as distribution $\boldsymbol{f}(\boldsymbol{x}, t)$, and we take the Maxwell-Boltzmann distribution as the equilibrium states. Under the small Mach number assumption, the discrete equilibrium states $f_i^{\text{equ}}(\rho, \boldsymbol{u})$ are expanded to the polynomial form as

$$f_i^{\text{equ}}(\rho, \boldsymbol{u}) = \rho w_i\left(1 + \frac{u_\alpha v_{i\alpha}}{c_s^2} + \frac{u_\alpha u_\beta\left(v_{i\alpha}v_{i\beta} - c_s^2\delta_{\alpha\beta}\right)}{2c_s^4}\right) + O(\boldsymbol{u}^3), \tag{3}$$

where ρ and $\boldsymbol{u}$ are the local fluid density and velocity and v_i represents the microscopic particle velocity. The Einstein summation notation is used. The function of $\delta_{\alpha\beta}$ is the Kronecker delta function. The weight w_i is corresponding to the particles with different velocities, it has the values

$$w_i = \begin{cases} 4/9, & i = 0, \\ 1/9, & i = 1, 2, 3, 4, \\ 1/36, & i = 5, 6, 7, 8. \end{cases} \tag{4}$$

The evolution of this kinetic system (1) consists of two steps: streaming and collision. The microscopic particles move to the neighbor lattice nodes in the streaming step depending on their velocity. In the collision step, the particle distribution functions would relax toward the local equilibrium state $\boldsymbol{f}^{\text{equ}}$ and eventually be rearranged to reach the postcollision states $\boldsymbol{f}^{\text{post}}$. The streaming and collision steps occur alternatively, and this system could recover the N-S equations under the hydrodynamic limits. The recovery of N-S equations from LBM equations could be derived by Chapman-Enskog expansion [12,21,22]. We will not discuss the detail of the derivation in this paper.

A remarkable advancement in the LBM is the derivation of the entropy function for this system [5,6]. The entropy function introduces the second law of thermodynamics into the LBM and expands our understanding of the collision step. The H function has the form

$$H(\boldsymbol{f}(\boldsymbol{x}, t)) = \sum_{i=0}^{8} f_i(\boldsymbol{x}, t) \ln\left(\frac{f_i(\boldsymbol{x}, t)}{w_i}\right), \tag{5}$$

where the weight w_i has the same values in Eq. (4).

III. MOMENT REPRESENTATION

For the streaming step, the representation of $\boldsymbol{f}(\boldsymbol{x}, t)$ is a natural choice because each element f_i directly represents the particle distribution with velocity $\boldsymbol{v}_i$. However, it is more convenient for the collision step to use moment representation. From the perspective of linear algebra, we could represent the same vector $\boldsymbol{f} \in \mathbb{R}^9$ by choosing any other basis. The idea behind moment representation is to pick a basis so that all the elements in this coordinate system are the moments of the distribution function. The moment representation makes it mathematically convenient to discuss the collision step.

In LBM, the fluid variables (e.g., fluid density and velocities) are represented by the moments of distribution functions $\boldsymbol{f}$. The moments (up to the second order) of $\boldsymbol{f}$ are defined as

$$\rho = \sum_{i=0}^{8} f_i, \tag{6a}$$

$$\rho u_\alpha = \sum_{i=0}^{8} f_i v_{i\alpha}, \tag{6b}$$

$$\rho \Pi_{\alpha\beta} = \sum_{i=0}^{8} f_i v_{i\alpha} v_{i\beta}, \tag{6c}$$

where the fluid density ρ and velocity u_α are related to the zeroth- and first-order moment, respectively. In the LBM, the fluid density is also related to pressure. The second-order moment $\Pi_{\alpha\beta}$ is a tensor, and it represents the flux of momentum. In the Chapman-Enskog expansion, we could derive the relation between the nonequilibrium parts of the second-order moment and velocity gradients. In general, we define the moments as [12]

$$\rho m_{pq} = \sum_{i=0}^{8} f_i v_{ix}^p v_{iy}^q, \tag{7}$$

where m_{pq} is related to the order of the moments, and we normalize them by the fluid density ρ. For example, $m_{00} = 1$ is related to zeroth order (normalized by ρ itself); $m_{10} = u_x$ and $m_{01} = u_y$ are the fluid velocities, which are related to the first-order moments (6b); $m_{20} = \Pi_{xx}$, $m_{02} = \Pi_{yy}$, and $m_{11} = \Pi_{xy}$ are the elements related to the second-order moments tensor (6c). Finding the corresponding physical interpretation for the higher-order moments is difficult. But we still need them for computation.

We need to choose nine linearly independent vectors for moment representation as to its basis. For the moments up to the second order (6), it is natural to choose all of them (i.e., 1, u_x, u_y, Π_{xx}, Π_{yy}, and Π_{xy}) since they have clear physical interpretation and are directly related to the recovering of N-S equations. Then, we still need to pick three more vectors from the higher-order moments to construct the basis. We do not think one choice could be better because the higher-order moments lack the physical interpretation, and it is just a linear transformation. In this paper, we follow the choice in [12], i.e., $Q_{xyy} = m_{12}$, $Q_{yxx} = m_{21}$, and $Q_{xxyy} = m_{22}$. After choosing the basis, we write the moment representation $\boldsymbol{m} \in \mathbb{R}^9$ as

$$\boldsymbol{m} = (1, u_x, u_y, \Pi_{xx}, \Pi_{yy}, \Pi_{xy}, Q_{xyy}, Q_{yxx}, Q_{xxyy})^\intercal, \tag{8}$$

where $\intercal$ represents the transpose. With the transforming matrix M and its inverse M^{-1}, we easily transform distribution functions between its regular representation $\boldsymbol{f}$ and moment representation $\boldsymbol{m}$ as

$$\rho \boldsymbol{m} = M\boldsymbol{f}, \tag{9a}$$

$$\boldsymbol{f} = \rho M^{-1}\boldsymbol{m}, \tag{9b}$$

where ρ is the density. The transformation matrix M (9a) is

$$M = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}, \tag{10}$$

and its inverse matrix M^{-1} (9b) is

$$M^{-1} = \begin{bmatrix} 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0.5 & 0 & 0.5 & 0 & 0 & -0.5 & 0 & -0.5 \\ 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & -0.5 & -0.5 \\ 0 & -0.5 & 0 & 0.5 & 0 & 0 & 0.5 & 0 & -0.5 \\ 0 & 0 & -0.5 & 0 & 0.5 & 0 & 0 & 0.5 & -0.5 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0.25 & 0.25 \\ 0 & 0 & 0 & 0 & 0 & -0.25 & -0.25 & 0.25 & 0.25 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & -0.25 & -0.25 & 0.25 \\ 0 & 0 & 0 & 0 & 0 & -0.25 & 0.25 & -0.25 & 0.25 \end{bmatrix}. \tag{11}$$

IV. THE DIAGRAM OF COLLISIONS

In order to improve the numerical stability, different collision models have been developed for the LBM. We discuss some of these collision models and eventually introduce the EMRT collision model. Finally, we draw the diagram of different collision models and compare their postcollision states.

A. BGK collision model

Let us first revisit the BGK collision model [2,3], the most commonly used model in the LBM. The postcollision state is

$$\boldsymbol{f}^{\mathrm{BGK}} = \boldsymbol{f} + \omega(\boldsymbol{f}^{\mathrm{equ}} - \boldsymbol{f}), \tag{12}$$

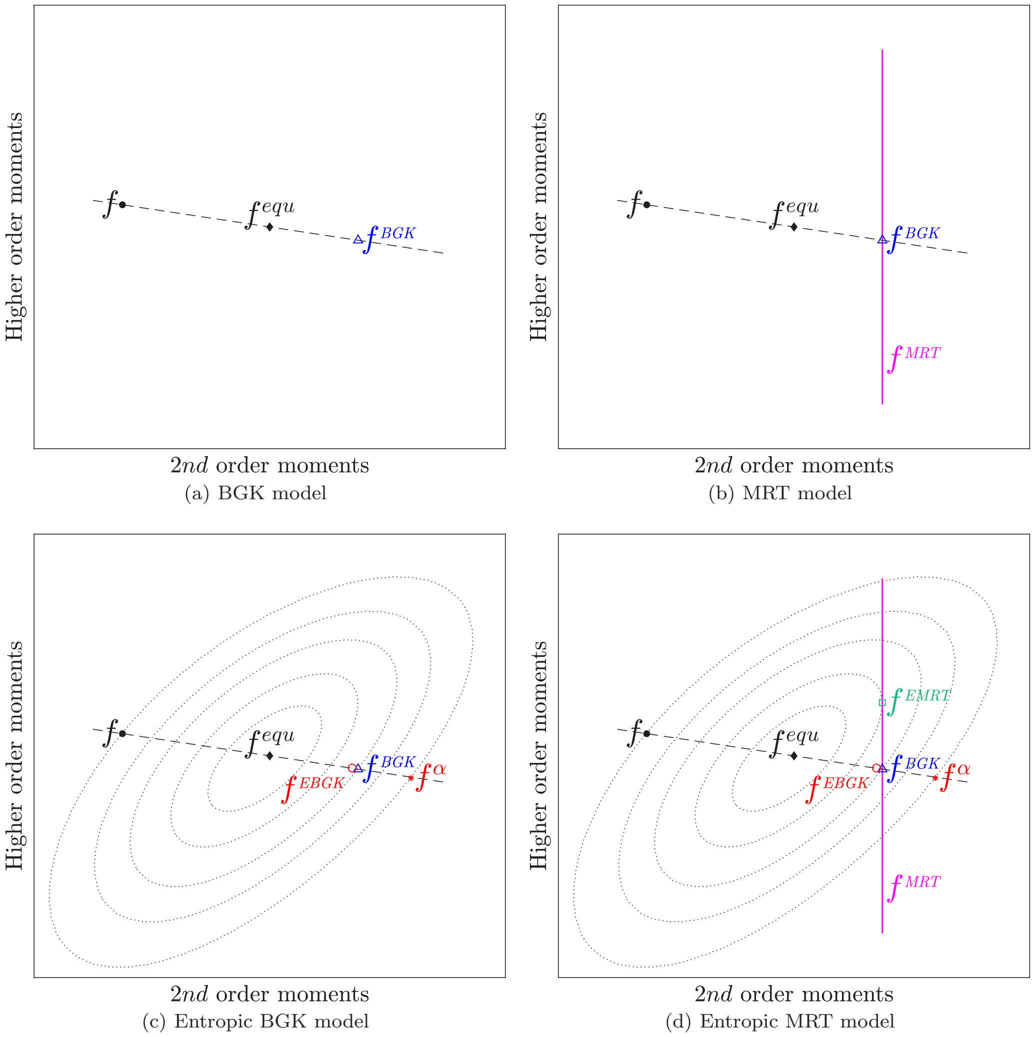
where $\boldsymbol{f}^{\mathrm{BGK}}$ is the postcollision state of the BGK model, and other terms have the same meaning as Eq. (1). There is only a single relaxation parameter ω in the BGK model and the distribution function $\boldsymbol{f}$ would relax to the equilibrium states $\boldsymbol{f}^{\mathrm{equ}}$ at the same rate.

The diagram of collision models is shown in Fig. 2. For visualization, we represent the vectors $\boldsymbol{f} \in \mathbb{R}^9$ in this 2D plane. The distribution function $\boldsymbol{f}$ is about to experience the collision step, and $\boldsymbol{f}^{\mathrm{equ}}$ is the local equilibrium state. Because the collision step has to satisfy the conservation of mass and momentum, we assume that all points in the plane have the same density and velocities as $\boldsymbol{f}$. In other words, no matter what collision model we choose, its postcollision state has to be in this plane. We choose the "second-order moments" and "higher-order moments" as two coordinates. In this case, any point on the axis of "second-order moments" represents $(\Pi_{xx}, \Pi_{yy}, \Pi_{xy}) \in \mathbb{R}^3$ and on axis of "higher-order moments" represents $(Q_{xyy}, Q_{yxx}, Q_{xxyy}) \in \mathbb{R}^3$, respectively.

The collision diagram of the BGK model is straightforward [Fig. 2(a)]. The postcollision state $\boldsymbol{f}^{\mathrm{BGK}}$ has to lie somewhere on the line connecting $\boldsymbol{f}$ and $\boldsymbol{f}^{\mathrm{equ}}$, and its position depends on the specific value of relaxation parameter ω (12). The value of ω usually is greater than 1 and it leads to a over-relaxation in the collision step. We will provide a possible explanation of why it could cause the numerical instability issue from the perspective of entropy function in the following section.

B. Multiple relaxation time collision model

The idea behind the MRT model is that different moments could relax to the equilibrium states at different rates [4]. This is different from the BGK model, where there is only a single relaxation parameter ω. In order to recover the desired



(a) BGK model

(b) MRT model

(c) Entropic BGK model

(d) Entropic MRT model

FIG. 2. The schematic of collision models. The distribution function $\boldsymbol{f}$ (solid dot) is going to experience collision step, and its equilibrium state is $\boldsymbol{f}^{\text{equ}}$ (solid diamond). The BGK collision model is shown in (a), where $\boldsymbol{f}^{\text{BGK}}$ (open triangle) is the postcollision state. It lies on the point of the line (dashed line) connecting $\boldsymbol{f}$ and $\boldsymbol{f}^{\text{equ}}$ depending on the relaxation parameter ω (12). The MRT collision model is shown in (b), where the vertical line $\boldsymbol{f}^{\text{MRT}}$ (solid line) is the possible postcollision states. The entropic BGK collision model is shown in (c). With entropy function (5), we draw a few contours (dotted line) of the entropy levels. The state $\boldsymbol{f}^{\alpha}$ (asterisk) is related to the solution α in Eq. (13), and $\boldsymbol{f}^{\text{EBGK}}$ (open dot) is the postcollision state of the EBGK model. The entropic multiple relaxation time model is shown in (d), where the postcollision state $\boldsymbol{f}^{\text{EMRT}}$ (open square) is the solution of the entropy maximization problem (15).

viscosity, we need to fix the relaxation parameters to the same value as ω up to the second-order moment. At the same time, it is free to choose the relaxation parameters of the higher-order moments. The possible postcollision states of MRT are the vertical line of $\boldsymbol{f}^{\text{MRT}}$ in Fig. 2(b). It shows that the postcollision state $\boldsymbol{f}^{\text{BGK}}$ is also on this line and we consider it as a limiting case of $\boldsymbol{f}^{\text{MRT}}$ (i.e., the relaxation parameters of the higher-order moments are also equal to the value of ω). The problem of MRT model is that it is difficult to find the physical interpretation for the higher-order moments. In other words, we do not know much about which point we should pick as the postcollision state along $\boldsymbol{f}^{\text{MRT}}$ line.

C. Entropic BGK collision model

The derivation of entropy function (5) introduces the second law of thermodynamics into the LBM [5,6]. It expands our understanding of the collision step. We draw the contours of the entropy level in the collision diagram [see Fig. 2(c)]. The local equilibrium state $\boldsymbol{f}^{\text{equ}}$ is the point with maximum entropy value in the plane (because all the points share the same fluid density ρ and velocities $\boldsymbol{u}$, this is equivalent to the constraints of fixed fluid density and velocities).

The entropic collision model is initially built on the BGK model (EBGK) [5,7]. The key idea behind the EBGK model is to let the collision step always increase the local entropy value. To realize this idea, it first looks for the auxiliary point $\boldsymbol{f}^{\alpha}$, which shares the same entropy value of $\boldsymbol{f}$ along the line connecting $\boldsymbol{f}$ and $\boldsymbol{f}^{\text{equ}}$. As shown in Fig. 2(c), the auxiliary point $\boldsymbol{f}^{\alpha}$ could be considered as the boundary: if the postcollision state $\boldsymbol{f}^{\text{post}}$ falls into the segment between $\boldsymbol{f}$ and $\boldsymbol{f}^{\alpha}$, it would guarantee that the collision would increase the local entropy value. We need to solve the following equation to find $\boldsymbol{f}^{\alpha}$:

$$H(\boldsymbol{f}) = H(\boldsymbol{f} + \alpha(\boldsymbol{f}^{\text{equ}} - \boldsymbol{f})), \tag{13}$$

where α is a scalar and solving the value of α is equivalent to finding the auxiliary state $\boldsymbol{f}^{\alpha}$. With the solution of α, the entropic BGK model rescales the relaxation parameter ω to get the postcollision state as follows:

$$\boldsymbol{f}^{\mathrm{EBGK}} = \boldsymbol{f} + \frac{\alpha}{2}\omega(\boldsymbol{f}^{\mathrm{equ}} - \boldsymbol{f}), \tag{14}$$

where $\boldsymbol{f}^{\mathrm{EBGK}}$ is the postcollision state shown in Fig. 2(c).

Practically, the value of variable α is very close to the value of 2 (thus the value of $\frac{\alpha}{2}$ is close to the value of 1). That is the reason why the postcollision state of the EBGK model $\boldsymbol{f}^{\mathrm{EBGK}}$ is very similar to the BGK model $\boldsymbol{f}^{\mathrm{BGK}}$ [see Fig. 2(c)]. By rescaling the relaxation parameter ω, the entropic collision model exhibits excellent stability performance. However, the EBGK model could cause a potential accuracy issue. Because the value of α is not exact the value of 2, rescaling the single relaxation parameter ω would dynamically modify the local viscosity. As it is shown in Fig. 2(d), the postcollision state $\boldsymbol{f}^{\mathrm{EBGK}}$ is moving along the line connecting $\boldsymbol{f}$ and $\boldsymbol{f}^{\mathrm{equ}}$ rather than along the line of $\boldsymbol{f}^{MRT}$.

Although it is difficult to prove the numerical stability issue for the LBM mathematically, the entropy function could provide a possible explanation for the BGK model when Re is large, as shown in Eq. (2), we have $\omega \approx 2$. On the other hand, in Eq. (13), the value of variable α is also very close to 2 in practice. Therefore, in flows with a high Reynolds number, we often have $\omega \approx \alpha$, which means the postcollision state $\boldsymbol{f}^{\mathrm{BGK}}$ would be very close to the auxiliary point $\boldsymbol{f}^{\boldsymbol{\alpha}}$. We see this by comparing the $\boldsymbol{f}^{\alpha} = \boldsymbol{f} + \alpha(\boldsymbol{f}^{\mathrm{equ}} - \boldsymbol{f})$ [see the term in the right-hand side of Eq. (13)] and Eq. (12). Because the auxiliary point $\boldsymbol{f}^{\boldsymbol{\alpha}}$ is the boundary for entropy value to increase [see the contours in Fig. 2(c)], if the postcollision state $\boldsymbol{f}^{\mathrm{BGK}}$ is very close to this boundary, it is quite possible for the postcollision state $\boldsymbol{f}^{\mathrm{BGK}}$ to go beyond it in some situations, and thus cause entropy value to decrease during the collision. This might explain why the BGK model would experience the numerical stability issue for large Re flow simulation.

On the other hand, if the Re is small or higher resolution grids are applied, the value of relaxation parameter ω in the BGK model would be much smaller than the value of 2 and the postcollision state $\boldsymbol{f}^{\mathrm{BGK}}$ would be far away from the boundary of the auxiliary point $\boldsymbol{f}^{\boldsymbol{\alpha}}$. In this case, it is unlikely that the postcollision states $\boldsymbol{f}^{\mathrm{BGK}}$ would go beyond that boundary, and the entropy value would most likely increase during the collision. Then, the BGK model might not experience the numerical stability issue because the collision step is consistent with the physics.

D. Entropic multiple relaxation time collision model

The problem with the EBGK collision model is that it is built on the BGK model, where there is only a single relaxation parameter ω. Could we build a collision model based on the idea of a multiple relaxation time model while still taking advantage of the entropy function? That is the idea behind the EMRT collision model [11,12]. In the EMRT model, the relaxation parameters up to the second-order moments are the same as those in the MRT model (i.e., to recover the desired viscosity). At the same time, it leaves the entropy function (5) to determine the optimal relaxation parameters for the higher-order moments. The problem with the MRT model is that the higher-order moments lack the physical interpretation, while the EMRT model looks for the physics in the entropy function. More specifically, we would like the postcollision state of EMRT to be the state with maximum local entropy value under the constraints of the fixed fluid density ρ, velocities u_α and the second-order moments tensor $\Pi_{\alpha\beta}$.

As it is shown in Fig. 2(d), the postcollision state $\boldsymbol{f}^{\mathrm{EMRT}}$ is the point where the vertical line of $\boldsymbol{f}^{\mathrm{MRT}}$ is tangent to some contour of entropy level. Because the postcollision state of the EMRT model would always fall into the line of $\boldsymbol{f}^{\mathrm{MRT}}$, it is able to recover the desired viscosity from the perspective of Chapman-Enskog expansion. It is worth mentioning that the EMRT model treats the entropy function differently from it does in the EBGK model. Setting the postcollision state as the maximum entropy state can not guarantee that the entropy value would always increase during the local collision step. However, the EMRT model does "make the best effort" to get the entropy of the postcollision state as large as possible under certain constraints.

In order to realize the idea of the EMRT model, the critical step is to solve the constrained entropy maximization problem. However, it is a challenging task. In the following section, we will discuss the solutions to this problem in detail.

V. ENTROPY MAXIMIZATION PROBLEM

In order to get the postcollision states $\boldsymbol{f}^{\mathrm{post}}$ in the EMRT model, we need to solve the entropy maximization problem under certain constraints. We convert it to the equivalent minimization problem in terms of H function as

$$\begin{aligned}
\text{minimize} \quad H(\boldsymbol{f}^{\mathrm{post}}) &= \sum_{i=0}^{8} f_i^{\mathrm{post}} \ln\left(\frac{f_i^{\mathrm{post}}}{w_i}\right),\\
\text{subject to} \quad \rho &= \sum_{i=0}^{8} f_i^{\mathrm{post}},\\
\rho u_\alpha &= \sum_{i=0}^{8} f_i^{\mathrm{post}} v_{i\alpha},\\
\rho \Pi_{\alpha\beta}^{\mathrm{post}} &= \sum_{i=0}^{8} f_i^{\mathrm{post}} v_{i\alpha} v_{i\beta},
\end{aligned} \tag{15}$$

where $\alpha, \beta = x, y$ and the H function is given in Eq. (5) (we will use the term "entropy maximization" and "minimization of H function" interchangeably in the following). The constraints on fluid density ρ and velocities u_α (u_x, u_y) are due to the conservation of mass and momentum during the collision. In order to recover the desired viscosity, we need to fix the relaxation parameters of the second-order moments, which means we need to impose the constraints on the second-order moment tensor for the postcollision states $\Pi_{\alpha\beta}^{\mathrm{post}}$ (where Π_{xx}^{post} and Π_{yy}^{post} are the diagonal components and Π_{xy}^{post} is the off-diagonal component, respectively). As we discussed earlier, the relaxation parameters of the second-order moments would be set to be the value of ω, so it is easy to derive the equations of the second-order moments tensor for the postcollision state $\Pi_{\alpha\beta}^{\text{post}}$ as

$$\Pi_{\alpha\beta}^{\text{post}} = \Pi_{\alpha\beta} + \omega\big(\Pi_{\alpha\beta}^{\text{equ}} - \Pi_{\alpha\beta}\big), \tag{16}$$

where $\Pi_{\alpha\beta}$, $\Pi_{\alpha\beta}^{\text{equ}}$, and $\Pi_{\alpha\beta}^{\text{post}}$ are the second-order moments for the precollision state, equilibrium state, and postcollision state, respectively. If we take the second-order moments of the equilibrium state $\boldsymbol{f}^{\text{equ}}$ (3), we get

$$\Pi_{\alpha\beta}^{\text{equ}} = c_s^2\delta_{\alpha\beta} + u_\alpha u_\beta, \tag{17}$$

where c_s is the speed of sound and $\delta_{\alpha\beta}$ is the Kronecker delta function. The relaxation parameter ω is determined by the desired viscosity (2).

We could also convert it to the equivalent unconstrained version if we look at the entropy maximization problem (15) from the perspective of moment representation. With the transformation matrix M and its inverse M^{-1}, we transform the postcollision states between the ordinary representation $\boldsymbol{f}^{\text{post}}$ and its moment representation $\boldsymbol{m}^{\text{post}}$ (9). Given the fluid density ρ, velocities u_α, and second-order moment tensor $\Pi_{\alpha\beta}^{\text{post}}$, if we pick any value for three higher-order moments Q_{xyy}^{post}, Q_{yxx}^{post}, and Q_{xxyy}^{post} (8), we would know everything about postcollision states $\boldsymbol{f}^{\text{post}}$ (and hence the entropy value). Therefore, the equivalent unconstrained version of the entropy maximization problem could be defined as

$$\begin{aligned}\text{minimize}\ &\sum_{i=0}^{8} f_i^{\text{post}}\big(Q_{xyy}^{\text{post}}, Q_{yxx}^{\text{post}}, Q_{xxyy}^{\text{post}}\big)\\ &\times \ln\left(\frac{f_i^{\text{post}}\big(Q_{xyy}^{\text{post}}, Q_{yxx}^{\text{post}}, Q_{xxyy}^{\text{post}}\big)}{w_i}\right),\\ \text{given}\quad &\rho,\ u_\alpha \text{ and } \Pi_{\alpha\beta}^{\text{post}},\end{aligned} \tag{18}$$

where the values of fluid density ρ, velocities u_α, and second-order moments $\Pi_{\alpha\beta}^{\text{post}}$ are fixed. Our target is to find the optimal values of Q_{xyy}^*, Q_{yxx}^*, and Q_{xxyy}^* such that the H function could reach the minimum value. For simplicity, we will discard the superscript of "post."

A. Two special cases

Although it is difficult to solve the entropy maximization problem in its general form (15) and (18), we could find the exact solutions to this family of problems for two special cases. Let us discuss these two particular cases and hope their solutions can provide some insights into the general problem.

In the first case, we would like to find the distribution functions with maximization entropy value under the constraints only on the fixed fluid density ρ and velocities u_x and u_y as

$$\begin{aligned}\text{minimize}\quad & H(\boldsymbol{f}) = \sum_{i=0}^{8} f_i \ln\left(\frac{f_i}{w_i}\right),\\ \text{subject to}\quad & \rho = \sum_{i=0}^{8} f_i,\\ & \rho u_x = \sum_{i=0}^{8} f_i v_{ix},\ \rho u_y = \sum_{i=0}^{8} f_i v_{iy}.\end{aligned} \tag{19}$$

The Lagrange multiplier could solve this problem [23], and the solutions are the local equilibrium states as

$$f_i^{\text{equ}}(\rho,\boldsymbol{u}) = \rho w_i \prod_{\alpha=x,y} (2-\varphi(u_\alpha))\left(\frac{2u_\alpha+\varphi(u_\alpha)}{1-u_\alpha}\right)^{v_{i\alpha}}, \tag{20}$$

where $\varphi(u_\alpha) = \sqrt{3u_\alpha^2+1}$ and $\alpha = x, y$. For f_i^{equ} to be positive, $|u_\alpha| < c_s = 1/\sqrt{3}$ has to be satisfied, which agrees with the small Mach number assumption. If we take the Taylor expansion about u_α, the most commonly used polynomial form of equilibrium states (3) would be recovered.

The second particular case is more interesting. We would like to find the distribution functions with the maximization entropy value under the constraints not only on fluid densities ρ and velocities u_x and u_y but also on the diagonal elements Π_{xx} and Π_{yy} in second-order moment tensor

$$\begin{aligned}\text{minimize}\quad & H(\boldsymbol{f}) = \sum_{i=0}^{8} f_i \ln\left(\frac{f_i}{w_i}\right),\\ \text{subject to}\quad & \rho = \sum_{i=0}^{8} f_i,\\ & \rho u_x = \sum_{i=0}^{8} f_i v_{ix},\ \rho u_y = \sum_{i=0}^{8} f_i v_{iy},\\ & \rho\Pi_{xx} = \sum_{i=0}^{8} f_i v_{ix}v_{ix},\ \rho\Pi_{yy} = \sum_{i=0}^{8} f_i v_{iy}v_{iy},\end{aligned} \tag{21}$$

where the difference between this particular case and the general entropy maximization problem (15) is that it does not impose any constraint on the off-diagonal element Π_{xy} in second-order moments. Asinari and Karlin [24] discussed this problem. They referred to the solution as "generalized Maxwellian." It is "among the only few analytic results on the relevant distribution functions in the classical kinetic theory." We could also apply the Lagrange multiplier to solve this problem. The solution [24] is

$$\begin{aligned}f_i^* = \rho w_i \prod_{\alpha=x,y} &\frac{3(1-\Pi_{\alpha\alpha})}{2}\left(\sqrt{\frac{\Pi_{\alpha\alpha}+1u_\alpha}{\Pi_{\alpha\alpha}-1u_\alpha}}\right)^{v_{i\alpha}}\\ &\times\left(\frac{2\sqrt{\Pi_{\alpha\alpha}^2-u_\alpha^2}}{1-\Pi_{\alpha\alpha}}\right)^{v_{i\alpha}^2},\end{aligned} \tag{22}$$

where w_i is the weight with the same values as in (4). The solution of $\boldsymbol{f}^*$ (22) looks complicated, but if we transform it into the moment representation $\boldsymbol{m}^*$ (8), the solution would look more elegant,

$$\boldsymbol{m}^* = \left(1, u_x, u_y, \Pi_{xx}, \Pi_{yy}, \underbrace{u_xu_y}_{\Pi_{xy}}, \underbrace{u_x\Pi_{yy}}_{Q_{xyy}}, \underbrace{u_y\Pi_{xx}}_{Q_{yxx}}, \underbrace{\Pi_{xx}\Pi_{yy}}_{Q_{xxyy}}\right)^{\mathsf{T}}, \tag{23}$$

where the elements of 1, u_x, u_y, Π_{xx}, and Π_{xy} are recovered as expected (since they are the constraints on this particular case). The off-diagonal element of the second-order moment $\Pi_{xy} = u_xu_y$ and higher-order moments $Q_{xyy} = u_x\Pi_{yy}$, $Q_{yxx} = u_y\Pi_{xx}$, and $Q_{xxyy} = \Pi_{xx}\Pi_{yy}$ are all the function of constraints.

Let us go back to the entropy maximization problem in its general case (15) and (18). It would have a different story if we still apply the Lagrange multiplier to solve it. Constructing the Lagrangian $\mathcal{L}$ and taking the partial derivative is not difficult. However, the generated system equations become too complicated to solve when we introduce the constraint on the diagonal element Π_{xy}.

B. Numerical solutions

In order to solve the entropy maximization problem (15) and (18), we apply a numerical method. For example, we could apply Newton's method [14] to solve it. Newton's method is an iterative method, where it would update the solutions based on the previous guess in each iteration. Newton's method would make a quadratic approximation to the target function based on the function value, gradient, and Hessian matrix from the previous guess, and then take the solutions minimizing that quadratic approximation as the next guess.

It is easier to apply Newton's method in the unconstrained version of the entropy maximization problem (18), where we consider the H function as a function of three variables Q_{xyy}, Q_{yxx}, and Q_{xxyy} [we will write it as the vector form as $\boldsymbol{Q} = (Q_{xyy}, Q_{yxx}, Q_{xxyy})^\intercal$]. The update algorithm is

$$\boldsymbol{Q}^{(k+1)} = \boldsymbol{Q}^{(k)} - [\nabla^2 H(\boldsymbol{Q}^{(k)})]^{-1} \nabla H(\boldsymbol{Q}^{(k)}), \quad k = 0, 1, 2, \ldots, \tag{24}$$

where $\boldsymbol{Q}^{(k)} \in \mathbb{R}^3$ is the numerical solution of kth iteration, and $\boldsymbol{Q}^{(0)}$ is the initial guess. The term $\nabla^2 H(\boldsymbol{Q}^{(k)})$ and $\nabla H(\boldsymbol{Q}^{(k)})$ are the Hessian matrix $\mathcal{H}$ and gradient $\boldsymbol{g}$ of the target entropy function (18), respectively. For the Hessian matrix, we need to get its inverse form. Practically, this term could be rewritten as the solution of the linear system of equations $\mathcal{H}\boldsymbol{x} = \boldsymbol{g}$.

The issue with numerical solutions is that it is expensive. Several iterations are needed for convergence, and the linear system equations must be solved for each iteration (the Hessian matrix $\mathcal{H}$ is a 3×3 matrix for the D2Q9 model). Therefore, a more efficient solution would be more attractive because we need to solve the entropy maximization problem at each computational node for each time step.

VI. PERTURBATION THEORY

Instead of seeking the numerical solutions to the entropy maximization problem (15) and (18), we propose to apply perturbation theory [17,18] to solve it. Perturbation theory is a powerful mathematical tool for solving complex problems. It explores the asymptotic behavior of the equations at the limit as a small parameter goes to zero. The intuition behind perturbation theory is that it converts a complex problem into an infinite set of more manageable subproblems at different scales. We gradually derive and solve them. Although deriving and solving the infinite set of subproblems might be as hard as solving the original problem, the power of perturbation theory is that solving only a few of them might provide enough information to approximate the exact solution at the scales that we are interested in. The advantage of an asymptotic solution is that it presents the solutions in the analytical form (hence very efficient compared with numerical solutions). On the other hand, it could give a surprisingly good approximation of the exact solutions. Therefore, the perturbation theory looks much more attractive in solving the entropy maximization problem than the numerical methods.

Typically, solving a problem by perturbation theory consists of three steps: (1) Find the solvable "unperturbed problem" by inserting a small parameter ϵ into the original problem. (2) Derive and solve the subproblems corresponding to the coefficients in different order of small parameter ϵ. (3) Sum the series. We will discuss all these steps in detail.

The first step in perturbation theory is to discover the unperturbed problem. We introduce a small parameter ϵ into the original problem. If we set this small parameter ϵ to 0, the original problem would reduce to the problem defined as the "unperturbed problem." It has to be solvable as a valid candidate for the unperturbed problem. Sometimes, it is not apparent to insert the small parameter ϵ and discover the unperturbed problem. As discussed by Holmes [18], a good understanding of the physics in the original problem might help decide to insert the small parameter ϵ. However, it always requires several guesses (probably some luck).

How should we insert the small parameter ϵ into the entropy maximization problem? Let us start with a slightly modified version of the original problem. Suppose we are asked to solve the entropy maximization problem [it is easier to discuss the equivalent unconstrained version (18)], while the constraints on the off-diagonal element Π_{xy} happens to be the value of $u_x u_y$ in the second-order moment. For this purposely designed problem, how should we choose the values for the higher-order moments Q_{xyy}, Q_{yxx} and Q_{xxyy} so that the entropy function could get the maximum value? If we go back to the second particular case (21) and examine its solutions more carefully, it would provide some insights. A reasonable guess about the solutions to the higher-order moments might be

$$Q_{xyy} = u_x \Pi_{yy}, \quad Q_{yxx} = u_y \Pi_{xx}, \quad Q_{xxyy} = \Pi_{xx}\Pi_{yy}, \tag{25}$$

where they have precisely the same values as the solutions of "generalized Maxwellian" (23) in the second particular case.

This guess is indeed the solution to the purposely designed problem! Let us prove it. If we look at the second particular case (21) from the perspective of the moment representation, its solutions $\boldsymbol{m}^*$ (23) tells us that given the fixed density ρ, fluid velocities u_x and u_y, and diagonal elements of the second-order moments Π_{xx} and Π_{yy}, the distribution functions would get the maximum entropy states when its off-diagonal element and the higher-order moments take the values as $\Pi_{xy} = u_x u_y$, $Q_{xyy} = u_x \Pi_{yy}$, $Q_{yxx} = u_y \Pi_{xx}$, and $Q_{xxyy} = \Pi_{xx}\Pi_{yy}$ simultaneously. In other words, for any other choice on off-diagonal element and higher-order moments, it would always have less entropy value than the maximum value.

In our purposely designed entropy maximization problem, if the constraint on the off-diagonal element Π_{xy} happens to be the value of $u_x u_y$ (by definition), we claim that, for any other choice on the higher-order moments, it would never get a greater entropy value than when we simultaneously choose the values in Eq. (25). Why? Suppose someone claims that when $Q_{xyy} = X$, $Q_{yxx} = Y$, and $Q_{xxyy} = Z$, it would have larger entropy value than our solutions. If it is true, then the solutions $\Pi_{xy} = u_x u_y$, $Q_{xyy} = X$, $Q_{yxx} = Y$, and $Q_{xxyy} = Z$ would have greater entropy value than the solutions (23) in the second particular case. This leads to a contradiction. Therefore, the values in Eq. (25) must be the solutions to this purposely designed problem.

However, we know how to solve the entropy maximization problem (21) only when the constraint on the off-diagonal element Π_{xy} happens to be the value of $u_x u_y$. What if the constraint on off-diagonal element Π_{xy} is an arbitrary value? The answer is simple: for any value of constraint on the off-diagonal element Π_{xy}, we assume it is being perturbed around the value of $u_x u_y$ as

$$\Pi_{xy} = u_x u_y + \epsilon, \tag{26}$$

where a small parameter ϵ is naturally introduced, we discover the unperturbed problem of the entropy maximization problem (18). When we set $\epsilon = 0$, the maximization problem would reduce to the "unperturbed" problem, to which we already know the solutions. The original problem could be fully represented by the introduction of this small parameter ϵ, i.e., for any value of Π_{xy}, we consider it as being "perturbed" around the value of $u_x u_y$ by a small amount of ϵ.

After inserting a small parameter ϵ, we are ready to explore the asymptotic behavior of the entropy maximization problem (15) and (18). In order to apply perturbation theory, we need to assume that all the variables of the higher-order moments Q_{xyy}, Q_{yxx}, and Q_{xxyy} as well as the introduced Lagrange multipliers $\lambda_1, \lambda_2, \dots, \lambda_6$ are also the function of small parameter ϵ. More specifically, we assume they have the polynomial form as

$$V(\epsilon) \sim V^{(0)} + V^{(1)}\epsilon + V^{(2)}\epsilon^2 + V^{(3)}\epsilon^3 + \cdots , \tag{27}$$

where V represents the variables of the higher-order moments Q_{xyy}, Q_{yxx}, and Q_{xxyy} as well as the introduced Lagrange multipliers $\lambda_1, \lambda_2, \dots, \lambda_6$. We substitute the off-diagonal element Π_{xy} with the value given in Eq. (26), and rewrite the system equations generated by the Lagrange multipliers as

$$\begin{aligned}
&\ln\left(\frac{\rho}{w_0}\right) + \ln[1 - \Pi_{xx} - \Pi_{yy} + Q_{xxyy}(\epsilon)] \sim \lambda_1(\epsilon),\\
&\ln\left(\frac{\rho}{2w_1}\right) + \ln[u_x + \Pi_{xx} - Q_{xyy}(\epsilon) - Q_{xxyy}(\epsilon)] \sim \lambda_1(\epsilon) + \lambda_2(\epsilon) + \lambda_4(\epsilon),\\
&\ln\left(\frac{\rho}{2w_2}\right) + \ln[u_y + \Pi_{yy} - Q_{yxx}(\epsilon) - Q_{xxyy}(\epsilon)] \sim \lambda_1(\epsilon) + \lambda_3(\epsilon) + \lambda_5(\epsilon),\\
&\ln\left(\frac{\rho}{2w_3}\right) + \ln[-u_x + \Pi_{xx} + Q_{xyy}(\epsilon) - Q_{xxyy}(\epsilon)] \sim \lambda_1(\epsilon) - \lambda_2(\epsilon) + \lambda_4(\epsilon),\\
&\ln\left(\frac{\rho}{2w_4}\right) + \ln[-u_y + \Pi_{yy} + Q_{yxx}(\epsilon) - Q_{xxyy}(\epsilon)] \sim \lambda_1(\epsilon) - \lambda_3(\epsilon) + \lambda_5(\epsilon),\\
&\ln\left(\frac{\rho}{4w_5}\right) + \ln[u_x u_y + \epsilon + Q_{xyy}(\epsilon) + Q_{yxx}(\epsilon) + Q_{xxyy}(\epsilon)] \sim \lambda_1(\epsilon) + \lambda_2(\epsilon) + \lambda_3(\epsilon) + \lambda_4(\epsilon) + \lambda_5(\epsilon) + \lambda_6(\epsilon),\\
&\ln\left(\frac{\rho}{4w_6}\right) + \ln[-u_x u_y - \epsilon - Q_{xyy}(\epsilon) + Q_{yxx}(\epsilon) + Q_{xxyy}(\epsilon)] \sim \lambda_1(\epsilon) - \lambda_2(\epsilon) + \lambda_3(\epsilon) + \lambda_4(\epsilon) + \lambda_5(\epsilon) - \lambda_6(\epsilon),\\
&\ln\left(\frac{\rho}{4w_7}\right) + \ln[u_x u_y + \epsilon - Q_{xyy}(\epsilon) - Q_{yxx}(\epsilon) + Q_{xxyy}(\epsilon)] \sim \lambda_1(\epsilon) - \lambda_2(\epsilon) - \lambda_3(\epsilon) + \lambda_4(\epsilon) + \lambda_5(\epsilon) + \lambda_6(\epsilon),\\
&\ln\left(\frac{\rho}{4w_8}\right) + \ln[-u_x u_y - \epsilon + Q_{xyy}(\epsilon) - Q_{yxx}(\epsilon) + Q_{xxyy}(\epsilon)] \sim \lambda_1(\epsilon) + \lambda_2(\epsilon) - \lambda_3(\epsilon) + \lambda_4(\epsilon) + \lambda_5(\epsilon) - \lambda_6(\epsilon),
\end{aligned} \tag{28}$$

where w_i is the weights given in Eq. (4). We write the variable of the higher-order moment as $Q_{xyy} = Q_{xyy}(\epsilon)$ to emphasize it is a function of small parameter ϵ, and the same applies to Q_{yxx}, Q_{xxyy} and λ_i.[1]

As a next step, we need to gradually derive the subproblems from the system equations (28) and solve the corresponding coefficients of variables (27) in different orders. Collecting the ϵ^0 terms could derive the system equations for the constant terms. The solutions to this subproblem are

$$Q^{(0)}_{xyy} = u_x \Pi_{yy}, \quad Q^{(0)}_{yxx} = u_y \Pi_{xx}, \quad Q^{(0)}_{xxyy} = \Pi_{xx}\Pi_{yy}, \tag{29}$$

where $Q^{(0)}_{xyy}$, $Q^{(0)}_{yxx}$, and $Q^{(0)}_{xxyy}$ are the same solutions to unperturbed problems (25). That is what we expect. Collecting the ϵ^0 terms is equivalent to setting $\epsilon = 0$ and the problem would reduce to the unperturbed version. We always use constant terms to check if any mistake is made in the perturbation method. Solving the system equations in constant terms could also give the solutions of the $\lambda^{(0)}_i$, which is required for the subproblem in the future. For simplicity, we will not present the solutions of coefficients for $\lambda^{(k)}_i$ in the paper.

[1] We need to replace all the equal signs "=" with the asymptotic sign "∼" in the system equations. The asymptotic sign "∼" makes the equations less rigorous (i.e., the left and right-hand sides do not have to be precisely equal anymore) but more powerful. This step is essential in the perturbation theory.

In order to collect ϵ^1 terms, we need to take the Taylor expansion of Eq. (28) around $Q^{(0)}_{xyy}$, $Q^{(0)}_{yxx}$, and $Q^{(0)}_{xxyy}$. For example, the second term of the first equation in Eq. (28) could be expanded as

$$\begin{aligned}\ln[1-\Pi_{xx}-\Pi_{yy}+Q_{xxyy}(\epsilon)] &= \ln\left(1-\Pi_{xx}-\Pi_{yy}+Q^{(0)}_{xxyy}\right)\\ &\quad+\frac{Q^{(1)}_{xxyy}\epsilon+Q^{(2)}_{xxyy}\epsilon^2+\cdots}{\left(1-\Pi_{xx}-\Pi_{yy}+Q^{(0)}_{xxyy}\right)}\\ &\quad+\cdots, \end{aligned}\tag{30}$$

where $Q^{(1)}_{xxyy}, Q^{(2)}_{xxyy}, \ldots$ are the coefficients corresponding to the different order of small parameter ϵ (27). With the help of Taylor expansion, we get the system equations for ϵ^1 terms. By solving this subproblem, we get

$$\begin{aligned} Q^{(1)}_{xyy} &= \frac{u_y(\Pi_{yy}-1)}{u_y^2-\Pi_{yy}},\\ Q^{(1)}_{yxx} &= \frac{u_x(\Pi_{xx}-1)}{u_x^2-\Pi_{xx}},\\ Q^{(1)}_{xxyy} &= \frac{u_xu_y(\Pi_{xx}-1)(\Pi_{yy}-1)}{\left(u_x^2-\Pi_{xx}\right)\left(u_y^2-\Pi_{yy}\right)}, \end{aligned}\tag{31}$$

where $Q^{(1)}_{xyy}$, $Q^{(1)}_{yxx}$, and $Q^{(1)}_{xxyy}$ are the coefficients corresponding to the first order of small parameter ϵ.

Similarly, if we collect ϵ^2 terms, we get the system of equations for ϵ^2 terms. By solving it, we get

$$\begin{aligned} Q^{(2)}_{xyy} &= \frac{u_x\psi(x)\left(\Pi_{yy}^2-u_y^2\right)(\Pi_{yy}-1)}{2\left(u_x^2-\Pi_{xx}\right)^2\left(u_y^2-\Pi_{yy}\right)^3},\\ Q^{(2)}_{yxx} &= \frac{u_y\psi(y)\left(\Pi_{xx}^2-u_x^2\right)(\Pi_{xx}-1)}{2\left(u_y^2-\Pi_{yy}\right)^2\left(u_x^2-\Pi_{xx}\right)^3},\\ Q^{(2)}_{xxyy} &= \frac{(\Pi_{xx}-1)(\Pi_{yy}-1)\left[\phi(x)\phi(y)-u_x^2u_y^2\psi(x)\psi(y)\right]}{2\left(u_x^2-\Pi_{xx}\right)^3\left(u_y^2-\Pi_{yy}\right)^3}, \end{aligned}\tag{32}$$

where the term $\psi(\alpha)=2u_\alpha^2-3\Pi_{\alpha\alpha}+1$ and $\phi(\alpha)=(2u_\alpha^2-\Pi_{\alpha\alpha})(u_\alpha^2-\Pi_{\alpha\alpha})$ for $\alpha=x,y$.

If we keep doing it, we get the coefficients of $\epsilon^3, \epsilon^4, \ldots,$ terms. As the order of ϵ increases, the subproblems become more and more complicated. Practically, it is too tedious to derive and solve these system equations manually. We apply the symbolic programming technique to do the tasks. In this paper, the computation is performed by writing the programs in Maple (Maplesoft, Waterloo, Ontario) [25].

After solving the coefficients for the higher-order moments Q_{xyy}, Q_{yxx}, and Q_{xxyy}, the last step is to sum the coefficients to get the asymptotic solutions for the entropy maximization problem (18). Sometimes, summing the series is not as simple as adding up each term, which is a rich topic in the perturbation theory [17]. Therefore, we will discuss two practical considerations of the series summation.

The first consideration is what if the coefficients derived by the perturbation theory are divergent. If the series are divergent, it does not make sense to add them to approximate the summation naively. The result would not give us much useful information. On the other hand, there is nothing wrong with the divergent series, and they are just what they are. The issue is we implicitly represent the summation by adding up the Taylor-like series when they are divergent. In order to address this issue, we need to figure out a way to extract useful information from the divergent series. In this case, we apply the technique of Padé approximants [17] to represent the asymptotic solutions.

The second consideration is how many coefficients should be included in the asymptotic solutions. As the order of small parameter ϵ increases, the corresponding coefficients become more and more complicated [see the solutions in Eq. (29), (31), and (32) for comparison]. If we have to calculate so many coefficients that the computational overhead in perturbation theory might outweigh Newton's method, then the perturbation theory would not be attractive to solve the entropy maximization problem.

We conduct a numerical experiment to test the accuracy of asymptotic solutions. In the entropy maximization problem (18), we set the constraints as $\rho=1.0$, $u_x=0.05$, $u_y=-0.03$, $\Pi_{xx}=0.35$, and $\Pi_{yy}=0.40$. A specific value of ϵ would represent a specific constraint on off-diagonal element Π_{xy} as given in Eq. (26). The asymptotic solutions of the higher-order moment Q_{xyy}, Q_{yxx}, and Q_{xxyy} would be calculated by adding terms up to first, second and third order of ϵ, respectively. We also present the Padé summation $P^1_2(\epsilon)$ for a reference

$$P^1_2(\epsilon)=\frac{a_0+a_1\epsilon}{1+b_1\epsilon+b_2\epsilon^2},\tag{33}$$

where a_i and b_i are determined by taking the Taylor expansion and comparing the coefficients with the polynomials (27) for the same order of ϵ [17]. It is worth mentioning that Padé summation P^1_2 requires the same number of coefficients as third-order polynomials. We take the numerical solutions by Newton's method (24) as the reference solution and quantify the deviation between asymptotic solutions and Newton's method as

$$\mathcal{E}_f=\frac{\|\boldsymbol{f}^{\text{asym}}-\boldsymbol{f}^{\text{ref}}\|_2}{\|\boldsymbol{f}^{\text{ref}}\|_2},\tag{34}$$

where $\|\cdot\|_2$ is the Euclidean norm in $\mathbb{R}^9$ and $\boldsymbol{f}^{\text{asym}}, \boldsymbol{f}^{\text{ref}}\in\mathbb{R}^9$ are the asymptotic solutions and the reference solutions by Newton's method, respectively.

The test results are shown in Fig. 3, where the log scale is taken in the y coordinate. We set ϵ to be within the range $(-0.06, 0.06)$, which is much broader than the interest of flow simulation in this paper. As expected in the asymptotic solutions, the smaller the parameter ϵ is, the better approximation the asymptotic solutions would make. We observe it clearly for all the solutions in Fig. 3(a). When the problem reduces to the "unperturbed" form at $\epsilon=0$, the asymptotic solutions become the exact solutions. In this case, the value of $\mathcal{E}_f(0)$ actually represents the relative error of the solution by Newton's method. It is too small to be shown in current limit values in y coordinate.

Another important observation in Fig. 3(a) is that the first-order asymptotic solutions give much larger error than other solutions. We see the gap between them for all the values of ϵ (except $\epsilon=0$). The maximum deviation of the first-order solution is about 0.017 when ϵ reaches the boundary. We argue that the first-order solution seems not to be a bad

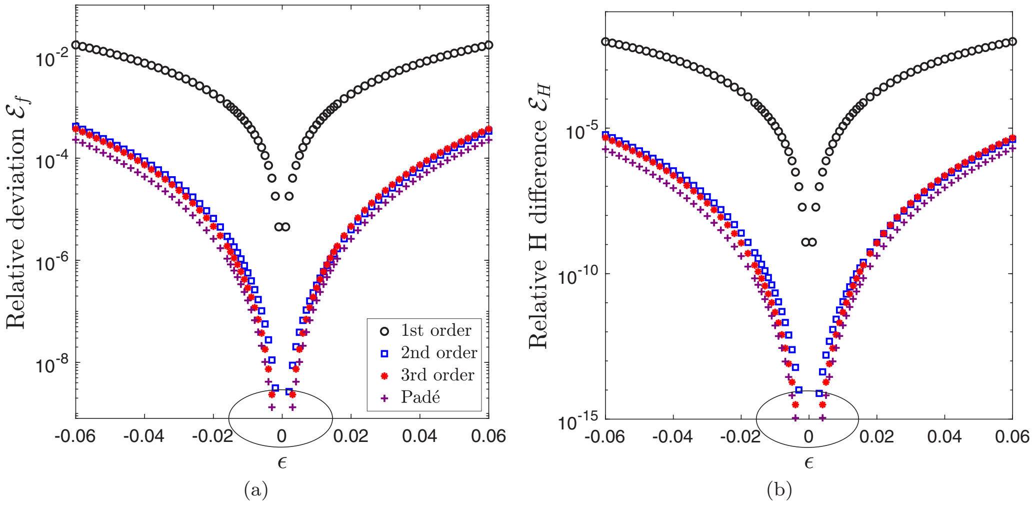


FIG. 3. The test results on the accuracy of asymptotic solutions. The experiment is conducted with the constraints of $\rho = 1.0$, $u_x = 0.05$, $u_y = -0.03$, $\Pi_{xx} = 0.35$, and $\Pi_{yy} = 0.40$. A specific value of ϵ would represent a specific constraint on the off-diagonal element Π_{xy} (26). The asymptotic solutions are calculated by adding the terms up to first, second, and third order, as well as Padé summation (33), respectively. We take the solutions by Newton's method as the reference solution. The deviation of $\boldsymbol{f}^{\text{asym}}$ norms (34) is shown in (a) and the H value difference (35) is shown in (b), where the log scale is taken in the y coordinate. The ellipse presents a more realistic range $(-0.015, 0.015)$ for ϵ.

approximation. However, this numerical test is not based on actual simulation data. It is difficult to tell whether the value of 0.017 is good or bad to approximate the maximum entropy state for the EMRT model. We will show more results on the deviations based on the actual simulation data in the following section.

We also provide the relative H value difference in Fig. 3(b) for a reference,

$$\mathcal{E}_H = \frac{H^{\text{asym}} - H^{\text{ref}}}{|H^{\text{ref}}|}, \tag{35}$$

where $|\cdot|$ represents the absolute value, and the scaler H^{asym} and H^{ref} are the H function (5) of the asymptotic solutions and the reference solution by Newton's method, respectively. The relative H value difference $\mathcal{E}_H$ could also show how well the approximation the asymptotic solution is making. The less $\mathcal{E}_H$ is, the larger entropy value the asymptotic solution would get. We observe a similar pattern in the H value difference $\mathcal{E}_H$ to the deviation $\mathcal{E}_f$.

They present better approximations for asymptotic solutions of the second and third orders and Padé summation. Even though Padé summation exhibits a slightly lower deviation, we must consider the practical balance between "accuracy" and "efficiency" when choosing the asymptotic solution. For the simulations in this paper, we measure the value of ϵ, and $(-0.015, 0.015)$ is a more realistic range. The solution for the second order is much more efficient than the third order and Padé summation. Also, we do not think it would make much difference concerning the accuracy, especially within the range $(-0.015, 0.015)$. Therefore, we choose to truncate the asymptotic solutions up to the second order as

$$Q = Q^{(0)} + Q^{(1)}\epsilon + Q^{(2)}\epsilon^2, \tag{36}$$

where Q represents the higher-order moments Q_{xyy}, Q_{yxx}, and Q_{xxyy}. The coefficients of $Q^{(0)}$, $Q^{(1)}$, and $Q^{(2)}$ are given in Eqs. (29), (31), and (32), respectively. The value of ϵ is determined by Eq. (26).

It is interesting to discuss the small parameter ϵ in detail and to see if we could find some physical interpretation of it. When we insert the small parameter ϵ into the entropy maximization problem, we assume that the off-diagonal element Π_{xy}^{post} of the second-order moments is perturbed around the value of $u_x u_y$ (26) for the postcollision state. In order to recover the desired fluid viscosity, we need to fix the relaxation parameter of the second-order moments to the value of ω. Then the off-diagonal element would be computed as $\Pi_{xy}^{\text{post}} = \Pi_{xy} + \omega(\Pi_{xy}^{\text{equ}} - \Pi_{xy})$ (16) during the collision step. Also, for the off-diagonal element of the equilibrium state Π_{xy}^{equ}, it happens to be the value of $u_x u_y$ (17). If we put all these equations together, we write the small parameter ϵ as

$$\epsilon = (1 - \omega)\Pi_{xy}^{\text{non-equ}}, \tag{37}$$

where ω is the relaxation parameter and $\Pi_{xy}^{\text{non-equ}} = \Pi_{xy} - \Pi_{xy}^{\text{equ}}$ represents the nonequilibrium part of the off-diagonal element in the second-order moment.

When taking the Chapman-Enskog expansion on the LBM, a critical derivation on the nonequilibrium part of the second-order moment $\Pi_{\alpha\beta}^{\text{non-equ}}$ is that it is related to the rate of strain tensor [12,19,20,22] as

$$\Pi_{\alpha\beta}^{\text{non-equ}} \approx -\frac{c_s^2}{\omega}(\partial_\alpha u_\beta + \partial_\beta u_\alpha), \tag{38}$$

where c_s is the speed of sound and $\partial_\alpha u_\beta$ represents the partial derivative of fluid velocity u_α with respect to β, where $\alpha, \beta = x, y$. This equation does not only help us recover the N-S equations but also develops the initial and boundary conditions in the LBM. Krüger *et al.* [22,26] presented a thorough study on this equation and took its measurements in the simulation data. We will discuss it further in the following section.

Let us focus only on the off-diagonal element and put Eqs. (37) and (38) together; it could get

$$\epsilon \approx \frac{(\omega - 1)c_s^2}{\omega}(\partial_x u_y + \partial_y u_x), \tag{39}$$

where all terms have the same meaning as Eq. (38). It shows that the value of small parameter ϵ is proportional to the velocity gradient. Under the small Mach number assumption, it is reasonable to assume that the value of ϵ would be small in the simulation. Thus the asymptotic solution can yield a good approximation to the exact solution of the maximum entropy state.

We summarize the steps, together with the corresponding equation number, to calculate the asymptotic solutions to the entropy maximization problem (15) in Algorithm 1 in the Appendix, which could serve as practical guidance for implementing the collision step of the EMRT model.

VII. FLOW SIMULATION

Two standard benchmark flows, the Taylor-Green vortex and the doubly periodic shear layer, are tested. To avoid the errors introduced by boundary conditions, we apply periodic boundary conditions to the flow simulations in this paper.

A. Initial conditions

The initial and boundary conditions play an essential role in the LBM. Even though we apply the periodic boundary conditions, the errors could also be introduced into flow simulation through initial conditions. In LBM, the fluid pressure is represented by the fluid density as $p = \rho c_s^2$, where c_s is the speed of sound. In order to initialize the distribution functions $\boldsymbol{f}(\boldsymbol{x}, 0)$, the initial pressure $p(\boldsymbol{x}, 0)$ [i.e., the density $\rho(\boldsymbol{x}, 0)$] is required. Usually, we only have the initial velocity field of the flow, and the initial pressure field is missing. A simple and common solution is to initialize the fluid density ρ to be a constant value of 1 everywhere. However, this is equivalent to setting the initial pressure constant everywhere, and the information about the pressure gradient is lost [19,27]. In fact, given the initial velocity field, we can get the initial pressure field by solving the Poisson equations. Because the computational domain is doubly periodic and grids are uniform in this study, applying the spectral method [28] to solve the Poisson equation by fast Fourier transform is very convenient.

With initial fluid density ρ and velocity $\boldsymbol{u}$, we initialize the distribution function $\boldsymbol{f}$ to its equilibrium state $\boldsymbol{f}^{\text{equ}}$. However, we lose the nonequilibrium part of the initial distribution functions $\boldsymbol{f}^{\text{non-equ}}$. As we discussed above, the nonequilibrium part of the second-order moments $\Pi_{\alpha\beta}^{\text{non-equ}}$ is related to the rate of strain tensor (38). If we neglect the nonequilibrium parts, some information about the velocity gradient would be lost in the initialization. It is straightforward to derive the velocity gradient from the initial velocity field. Therefore, we combine the equilibrium (17) and nonequilibrium parts (38) to recover the second-order moments $\Pi_{\alpha\beta}$.

With the value of ρ, $\boldsymbol{u}$ and $\Pi_{\alpha\beta}$, the information about initial distribution functions $\boldsymbol{f}(\boldsymbol{x}, 0)$ is still not complete, and the nonequilibrium part of the higher-order moment is missing. How should we recover the initial distribution functions $\boldsymbol{f}(\boldsymbol{x}, 0)$ from the partial information? Skordos [19] proposed approximating the initial distribution functions $\boldsymbol{f}(\boldsymbol{x}, 0)$ based on the Chapman-Enskog expansion (see Ref. [20] for the discussion on the approximation).

Different from Skordos's solution, we propose applying the idea of the EMRT model and letting the entropy function determine the missing information on the higher-order moments. More specifically, the distribution function $\boldsymbol{f}(\boldsymbol{x}, 0)$ could be initialized to the maximum entropy state under the constraints of fixed ρ, $\boldsymbol{u}$, and $\Pi_{\alpha\beta}$. In order to implement this idea, we just need to solve the same entropy maximization problem (15) and (18) as we discussed in the paper, which could be solved efficiently with perturbation methods. It is worth mentioning that the generalized KBC stabilizer [15] is not working because the precollision state is missing in the initial conditions. We show the test results on different initial conditions and examine how they could affect the accuracy of flow simulations in the following.

B. Taylor-Green vortex

The Taylor-Green vortex is a classical benchmark problem, and it has the analytic solutions [27] to the incompressible N-S equations as

$$\begin{aligned} u_x(x, y, t) &= -U\cos(k_x x)\sin(k_y y)e^{-k^2\nu t}, \\ u_y(x, y, t) &= \frac{k_x}{k_y}U\cos(k_y y)\sin(k_x x)e^{-k^2\nu t}, \\ p(x, y, t) &= -\frac{1}{4}U^2\left(\cos(2k_x x) + \left(\frac{k_x}{k_y}\right)^2\cos(2k_y y)\right)e^{-2k^2\nu t} \\ &\quad + P_0, \end{aligned} \tag{40}$$

where U is the characteristic flow velocity and it should be a small value due to the small Mach number assumption. We set $k^2 = k_x^2 + k_y^2$, where k_x and k_y are the wave numbers in the x and y directions, respectively. The flow viscosity is ν, and P_0 is an arbitrary constant set for the pressure. The computational domain is scaled to be $0 \leqslant x, y \leqslant 1$ and the simulation time is chosen to be $t_s = 1/U$.

In order to explore the influence of initial conditions, we apply two different initialization procedures: (1) the distribution functions $\boldsymbol{f}(\boldsymbol{x}, 0)$ is initialized only to the equilibrium states $\boldsymbol{f}^{\text{equ}}$ and (2) the nonequilibrium parts are also considered (38). Because the initial pressure field is known at $t = 0$ (40), there is no need to solve the Poisson equation in this problem. We choose $\text{Re} = 1000$, $U = 0.02$, and $k_x = k_y = 2\pi$. The simulations would run on 16×16 to 512×512 grids for both BGK and EMRT models. The ℓ^2 norm for the relative error is used to quantify the accuracy of u_x as

$$\mathcal{E}_{u_x} = \sqrt{\frac{\sum |u_{\text{exact}} - u_{\text{sim}}|^2}{\sum |u_{\text{exact}}|^2}}, \tag{41}$$

where u_{sim} is the simulation result and u_{exact} is the exact solutions (40). The summation is applied to all the grid nodes.

The results of convergence rates at time $t = t_s$ are shown in Fig. 4. For the initial conditions with only the equilibrium part, both BGK and EMRT models have a second-order convergence rate and reach a limit value on 256×256 grids.

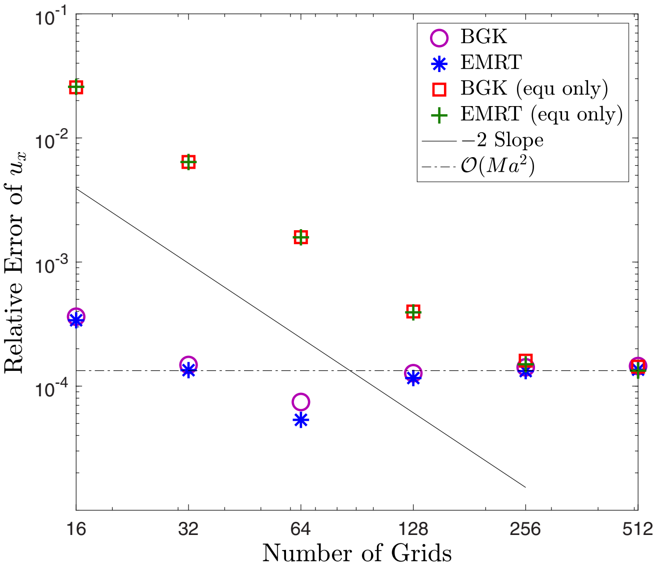


FIG. 4. The convergence rate of Taylor-Green vortex for relative error on u_x (41) at time $t = t_s$. The flow parameters are set as $U = 0.02$, Re = 1000, and $k_x = k_y$. "EMRT" and "BGK" represent the EMRT and BGK collision models, respectively. The initial density is consistent with the initial pressure field. For the initial conditions, "equ only" means the distribution function is only initialized to its equilibrium state. Otherwise, the nonequilibrium part is also considered, and the distribution function would be initialized to the maximum entropy state under constraints on fluid density, velocity, and second-order moments. The values of Ma^2 and −2 slope are provided for a reference.

However, when the nonequilibrium part is also considered for the initial conditions, the simulations become more accurate, and the relative error could reach the limit value just on 32×32 grids. This shows the critical role the initial conditions play in the LBM. We always include the nonequilibrium part for the initial conditions in the following simulations. In practice, we also observe that there is some difference between performing the "streaming" or "collision" step first right after initializing the distribution functions $\boldsymbol{f}(\boldsymbol{x}, 0)$. We choose to perform the collision step first in the simulation of this paper (it would not affect the simulation results if the distribution functions are initialized to the equilibrium states since performing a collision step to equilibrium states would still get the equilibrium states). When the simulation reaches the limit, the accuracy could not be improved further by increasing the grid numbers. We observe that this limit has the value of about Ma^2, which is the expected accuracy in the LBM.

Further, we implement the EMRT model by asymptotic solutions, the KBC stabilizer, and Newton's method, respectively. As to the KBC stabilizer, Karlin *et al.* developed a family of models based on the choice of moment representation [12]. But we implement only the model (i.e., the KBC C model), which takes the same moment representation as we choose for asymptotic solutions in this paper. Also, we do not implement the generalized KBC stabilizer because we believe Newton's method would be a better choice if solving a 3×3 matrix is affordable. Therefore, we mean the KBC C model by the KBC stabilizer [12], where the higher-order moments share the same relaxation parameter in the following.

The wave numbers k_x and k_y would also be set to different values, and the resolution would be increased up to 2048×2048 grids. The results of convergence rates on fluid velocity u_x are shown in Fig. 5(a) ($k_x = k_y = 2\pi$) and Fig. 6(a) ($2k_x = k_y = 4\pi$), respectively. The relative errors on u_x could reach the expected limit value of Ma^2 on both wave number settings. We observe the results from asymptotic solutions and Newton's method are almost indistinguishable for all the simulations. This shows that asymptotic solutions could give a good approximation to the maximum entropy states. Compared to BGK models, the EMRT model by different implementations could render a similar accuracy.

As discussed in Sec. VI, the equation of the nonequilibrium part in the second-order moment (38) is a crucial derivation when we take the Chapman-Enskog expansion to the LBM [12,19,20]. It shows the relationship between a local variable $\Pi_{\alpha\beta}^{\text{non-equ}}$ and a nonlocal variable rate of strain tensor $\frac{1}{2}(\partial_\alpha u_\beta + \partial_\beta u_\alpha)$ in the simulation. Equation (38) does not only help us recover the N-S equations but also provides some insights into developing the initial and boundary conditions to the LBM. The numerical scheme of the LBM presents the feature of "locality," and it is interesting to check how well the local variable $\Pi_{\alpha\beta}^{\text{non-equ}}$ could approximate the velocity gradient. Krüger *et al.* [22,26] presented an extensive study on this problem. They argued that the relative error on the shear stress tensor is expected to be second-order convergence in the BGK model but is sensitive to some error sources (e.g., the boundary conditions). Slightly different from Krüger *et al.*, we measure the ℓ^2 norm for the relative errors of $\Pi_{\alpha\beta}^{\text{non-equ}}$ on both BGK and EMRT models as

$$\mathcal{E}_{\Pi_{\alpha\beta}^{\text{non-equ}}} = \sqrt{\frac{\sum \left|\Pi_{\alpha\beta}^{\text{non-equ}} - \frac{-c_s^2}{\omega}(\partial_\alpha u_\beta + \partial_\beta u_\alpha)\right|^2}{\sum \left|\frac{-c_s^2}{\omega}(\partial_\alpha u_\beta + \partial_\beta u_\alpha)\right|^2}}, \quad (42)$$

where $\Pi_{\alpha\beta}^{\text{non-equ}}$ is computed locally on the computational node, and the velocity gradient is computed by analytic solutions (40). The summation is performed on all computational nodes.

The relative error of $\Pi_{xx}^{\text{non-equ}}$ is shown in Fig. 5(b) for $k_x = k_y$ case. The error of $\Pi_{yy}^{\text{non-equ}}$ is symmetric to $\Pi_{xx}^{\text{non-equ}}$ and $\Pi_{xy}^{\text{non-equ}}$ would vanish by this setting. We observe the second-order convergence on both BGK and EMRT models. As different implementations of EMRT models, they present similar relative errors on $\mathcal{E}_{\Pi_{xx}^{\text{non-equ}}}$. The relative errors $\mathcal{E}_{\Pi_{xx}^{\text{non-equ}}}$ would reach the limit on 128×128 grids; however, the limit value is not Ma^2 as it is on u_x. We also measure the relative errors on $\Pi_{\alpha\beta}$ for $2k_x = k_y$ case. The results of $\Pi_{xx}^{\text{non-equ}}$, $\Pi_{yy}^{\text{non-equ}}$ and $\Pi_{xy}^{\text{non-equ}}$ are shown in Figs. 6(b), 6(c) and 6(d), respectively. The second-order convergence could also be observed on all three components. A slightly different pattern happens on the relative errors of $\Pi_{yy}^{\text{non-equ}}$ by the KBC stabilizer, where it presents higher errors than others before 256×256 grids and converges to the limit value at higher resolutions [see Fig. 6(c)]. The thorough investigation on $\mathcal{E}_{\Pi_{\alpha\beta}^{\text{non-equ}}}$ is challenging since the error sources come from the

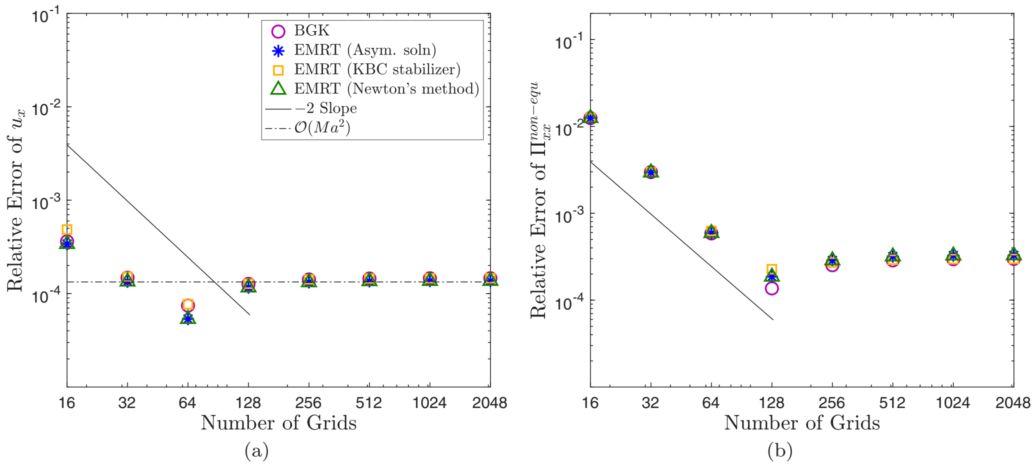


FIG. 5. The convergence rate of the Taylor-Green vortex for wave number $k_x = k_y = 2\pi$ at time $t = t_s$. The flow parameters are set as $U = 0.02$, Re = 1000. "EMRT" and "BGK" represent the EMRT and BGK collision model, respectively. The EMRT model is implemented by asymptotic solutions, KBC stabilizer, and Newton's method, respectively. The relative errors are calculated on (a) u_x (41) and (b) $\Pi_{xx}^{\text{non-equ}}$ (42), respectively. The values of Ma^2 and −2 slope are provided accordingly for a reference.

higher-order terms being ignored, and we leave it for a future study.

This flow problem does not have a stability issue; the EMRT and BGK model simulation results are almost indistinguishable. This demonstrates that the EMRT models could render a similar accuracy performance to the BGK model, which is the most commonly used model in the LBM.

C. Doubly periodic shear layer

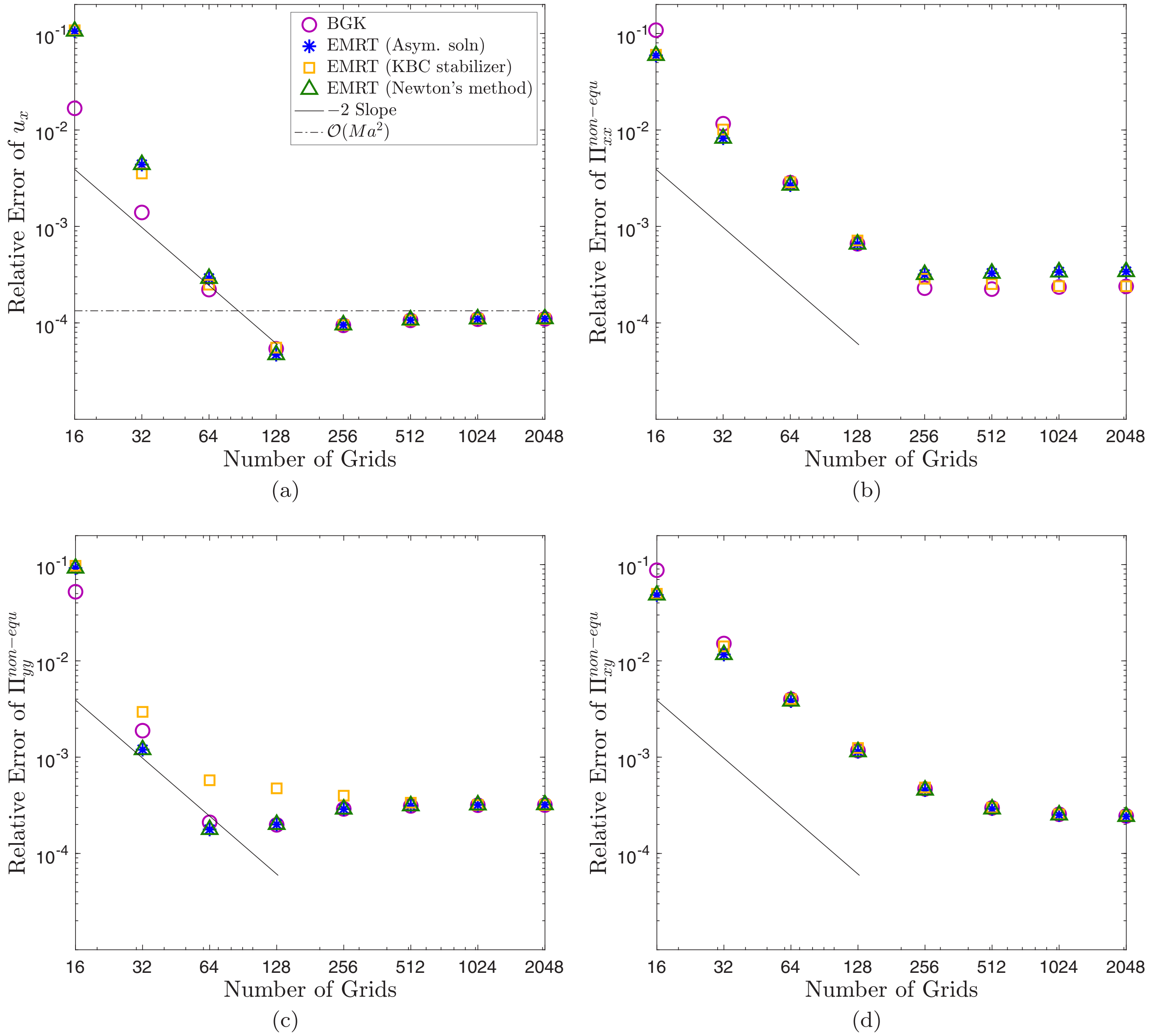
Minion and Brown [29] studied the doubly periodic shear layer flow by running different traditional numerical methods (non-LBM type). When the simulation is under-resolved, the "spurious, nonphysical vortices" could be produced in some results. Several investigators have studied this benchmark flow using LBM with different collision models [12,21]. They sometimes observed spurious vortices when the flow was under-resolved. The initial velocity field of the doubly periodic shear layer is

$$u_x = \begin{cases} U\tanh[\lambda(y-0.25)], & y \leqslant 0.5,\\ U\tanh[\lambda(0.75-y)], & y > 0.5,\end{cases}$$
$$u_y = \delta U\sin[2\pi(x+0.25)], \tag{43}$$

where U is the characteristic flow velocity, which is small due to the small Mach number assumption. The parameter λ controls the thickness of the shear layers, and δ is the small initial disturbance. The computational domain is scaled to be $0 \leqslant x, y \leqslant 1$ and the simulation time is chosen to be $t_s = 1/U$. Unlike the Taylor-Green vortex, this problem does not have an analytical solution.

To test the behavior of the EMRT model on under-resolved flow, we set parameter $\lambda = 80$ (it is referred to as a thin layer [29] where there are about five nodes in 64×64 grids and 11 nodes in 128×128 grids within the velocity transition region). For other parameters, we set $\delta = 0.05$, Re = 30 000. The simulations would run on 64×64 to 4096×4096 grids for both BGK and EMRT models. The EMRT model would be implemented by asymptotic solutions, KBC stabilizer, and Newton's method, respectively. Figure 7 shows the contour of vorticity at time $t = t_s$ (only to 512×512 grids). For the BGK model, the simulations fail on 64×64 and 128×128 grids. It yields a converged solution on 256×256 grids [Fig. 7(g)]. However, we observe the spurious vorticity on this resolution, which shows it might still be under-resolved. The spurious vorticity would disappear on higher resolutions for the BGK model [Fig. 7(k)]. On the other hand, the EMRT model by different implementations could produce a converged solution even for 64×64 grids [Figs. 7(a), 7(b) and 7(c)]. The results of EMRT by asymptotic solution and Newton's method are almost indistinguishable (see second and fourth columns in Fig. 7). Some spurious vorticity is observed on lower resolution and would disappear for 128×128 grids [Figs. 7(d) and 7(f)]. The KBC stabilizer could survive on 64×64 grids, but we observe the spurious vorticity on 128×128 grids (see the KBC C model simulation results in Ref. [12]). In this flow simulation, the EMRT collision model demonstrates much better stability than the BGK model.

Let us measure the relative errors of $\Pi_{\alpha\beta}^{\text{non-equ}}$ with respect to the rate of strain tensor (38). Because there is no analytical solution to this problem, we apply the spectral method [28] to calculate the velocity gradient based on the simulation data (42). We choose the characteristic velocity U to be 0.01, 0.02, and 0.04, which represent different Ma, respectively. The results are shown in Fig. 8. The expected second-order convergence rate could be observed for all the elements of $\Pi_{\alpha\beta}^{\text{non-equ}}$. The error on $\Pi_{\alpha\beta}^{\text{non-equ}}$ would converge to a limit value as the resolution increases. By looking at Fig. 8 horizontally, we observe that the limit value would increase with the increase of Ma for each element of $\Pi_{\alpha\beta}^{\text{non-equ}}$. As to the off-diagonal element of $\Pi_{xy}^{\text{non-equ}}$ (see the third row in Fig. 8), the BGK and KBC stabilizer models converge to a similar limit, and the limit value would increase as Ma increases. However, the EMRT models by asymptotic solutions and Newton's method would converge to a different limit, and we could still not



(a)

(b)

(c)

(d)

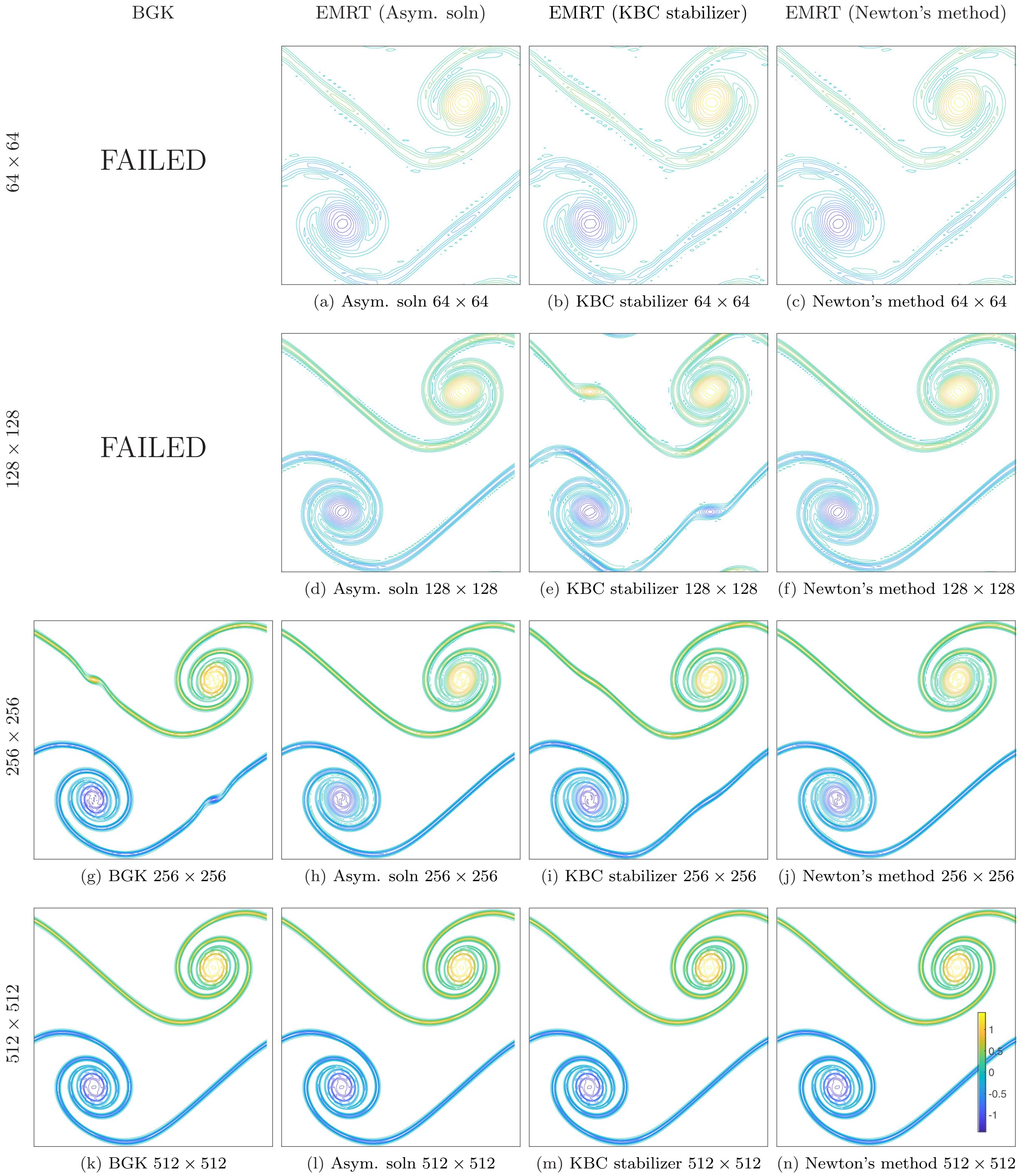
FIG. 6. The convergence rate of the Taylor-Green vortex for wave number $2k_x = k_y = 4\pi$ at time $t = t_s$. The flow parameters are set as $U = 0.02$, Re $= 1000$. "EMRT" and "BGK" represent the EMRT and BGK collision model, respectively. The EMRT model is implemented by asymptotic solutions, KBC stabilizer, and Newton's method, respectively. The relative errors are calculated on (a) u_x (41), (b) $\Pi_{xx}^{\text{non-equ}}$, (c) $\Pi_{yy}^{\text{non-equ}}$, and (d) $\Pi_{xy}^{\text{non-equ}}$ (42), respectively. The values of Ma^2 and -2 slope are provided accordingly for a reference.

observe the limit value even on 4096×4096 grids. This shows that the relaxation of higher-order moments does have an impact on the approximation on $\Pi_{\alpha\beta}^{\text{non-equ}}$ (38). As we mentioned above, the error sources of the approximation (38) depend on the higher-order terms being ignored [22]. The mathematical analysis of the error terms is very complicated, and we hope our observation could provide some insights for future study.

D. Discussion on different EMRT implementations

Let us compare the asymptotic solution, KBC stabilizer, and Newton's method for the EMRT model. We focus on how efficient the implementations are and how well the approximation they make to the maximum entropy state.

In order to compare the efficiency of different methods, we conduct numerical experiments on doubly periodic shear layers. The simulation would be run on 256×256 grids for $T = 6400$ time steps (we choose characteristic velocity $U = 0.04$ and other flow parameters are the same as in the last subsection). We measure the cumulative CPU times of collision steps in the simulation (the streaming steps are not considered). Each test is repeated 20 times, and we take their average as the result of collision CPU times (the standard derivation is also provided). We also take the measurements on the BGK model as a reference.



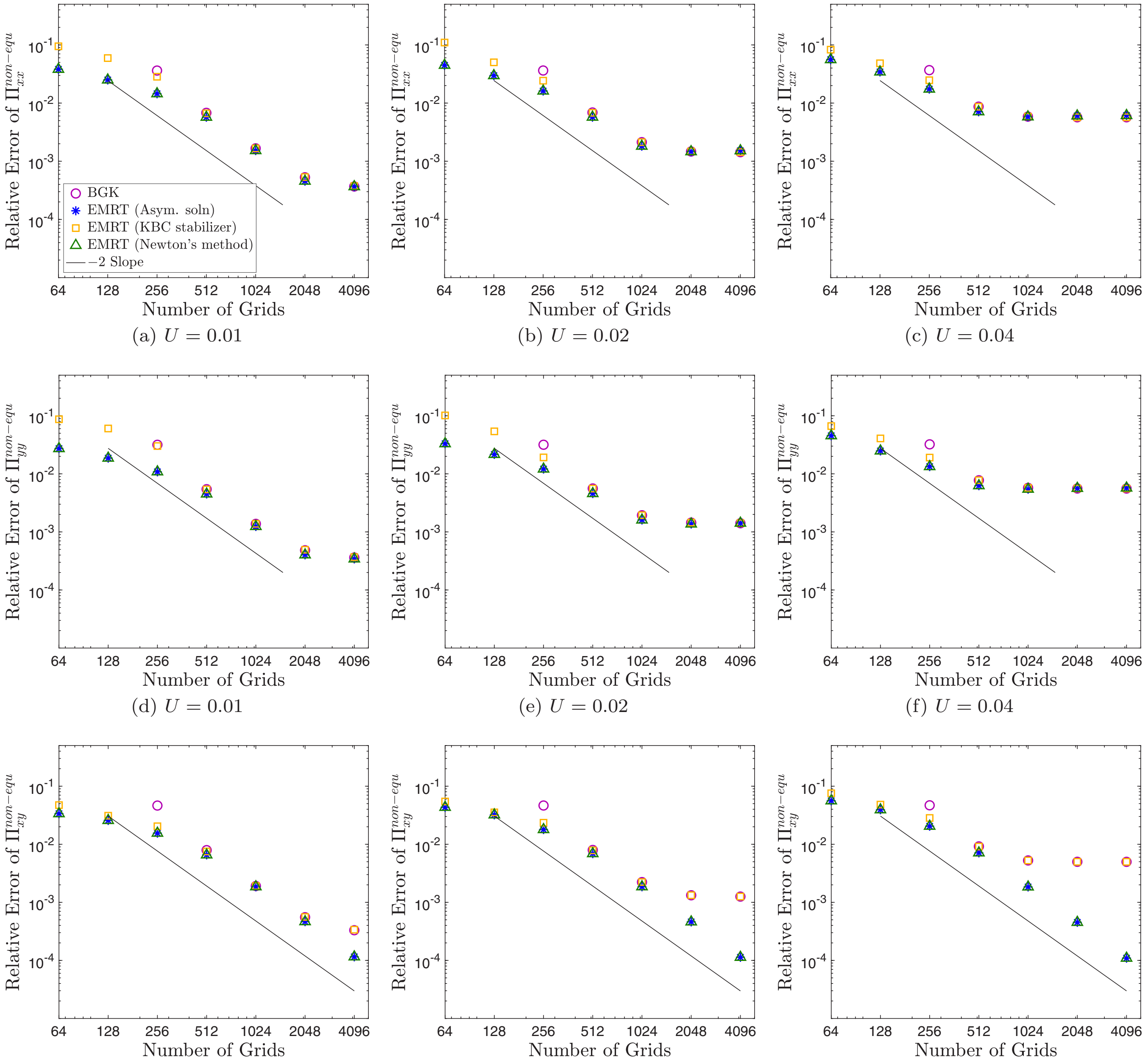
(a) Asym. soln 64 × 64 (b) KBC stabilizer 64 × 64 (c) Newton's method 64 × 64

(d) Asym. soln 128 × 128 (e) KBC stabilizer 128 × 128 (f) Newton's method 128 × 128

(g) BGK 256 × 256 (h) Asym. soln 256 × 256 (i) KBC stabilizer 256 × 256 (j) Newton's method 256 × 256

(k) BGK 512 × 512 (l) Asym. soln 512 × 512 (m) KBC stabilizer 512 × 512 (n) Newton's method 512 × 512

FIG. 7. The plots of vorticity contour of doubly periodic shear layer at time $t = t_s$. The flow parameters are set as $\lambda = 80$, $\delta = 0.05$, $\mathrm{Re} = 30\,000$, and $U = 0.02$. The first column shows the results of BGK model, and the other columns show EMRT model. The EMRT model is implemented by asymptotic solution (second column), KBC stabilizer (third column), and Newton's method (fourth column), respectively. The rows represent the results on 64×64, 128×128, 256×256, and 512×512 grids, respectively. The simulation diverges for the BGK model with 64×64, 128×128 resolution.



(a) $U = 0.01$

(b) $U = 0.02$

(c) $U = 0.04$

(d) $U = 0.01$

(e) $U = 0.02$

(f) $U = 0.04$

(g) $U = 0.01$

(h) $U = 0.02$

(i) $U = 0.04$

FIG. 8. The plots of relative errors on approximation $\Pi^{\text{non-equ}}_{\alpha\beta} \approx -\frac{c_s^2}{\omega}(\partial_\alpha u_\beta + \partial_\beta u_\alpha)$ (38). The flow parameters are set as $\lambda = 80$, $\delta = 0.05$, Re = 30 000. The first, second, and third columns show the results of characteristic velocity $U = 0.01$, $U = 0.02$, and $U = 0.04$, respectively. The different characteristic velocity U represents different Ma in the simulation. The first, second, and third rows show the relative error of $\Pi^{\text{non-equ}}_{xx}$, $\Pi^{\text{non-equ}}_{yy}$, and $\Pi^{\text{non-equ}}_{xy}$, respectively. The simulations run on the BGK and EMRT model. The EMRT model is implemented by the asymptotic solution, KBC stabilizer, and Newton's method, respectively. The plot of −2 slope is provided for a reference.

Table I shows the cumulative CPU times of the collision steps. The measured results are also normalized by the CPU times of the BGK model. We set the initial guess for Newton's method to be the solution to unperturbed problems (25). This initialization could make Newton's method converge very fast. We find that it only takes two iterations to converge for the most of nodes (we set a relative error of 1.0×10^{-8} as the convergence criterion) and even a single iteration could make the solution close to the maximum entropy state. The results show that the BGK is the most efficient model, which we expect. For various implementations of the EMRT model, the asymptotic solution is about 11.7 times faster than Newton's method in the collision steps. It is a little slower (1.6 times slower) than the BGK collision model. It is interesting to note that the asymptotic solution is even faster than the KBC stabilizer. This might not be that surprising: although the collision step of the KBC stabilizer seems to only happen to the regular representation, we still need to convert the distribution function to the moment representation to evaluate the optimal relaxation parameter in the higher-order moment [11,12]. We

TABLE I. The measurements of collision steps on doubly periodic shear layer. The simulations are running on 256×256 grids for $T = 6400$ time steps. For the EMRT model, we implement it with asymptotic solutions, KBC stabilizer, single iteration in Newton's method, and converged solutions in Newton's method, respectively. The CPU times are the average of 20 measurements (the standard deviation is provided for a reference) and normalized with the BGK model. In order to evaluate the deviation $\mathcal{E}_f$ (34) in real simulation data, we first run the BGK model and keep all the precollision states at each node and each time step. Then we calculate the postcollision states for different models, and evaluate deviation $\mathcal{E}_f$ with respect to postcollision states with Newton's method. We also evaluate if the deviation $\mathcal{E}_f$ of postcollision state for different models is smaller than it is in the BGK model and present it as a percentage (we define a smaller deviation $\mathcal{E}_f$ as it is at least 1.0×10^{-10} smaller than the BGK model to avoid some truncation errors in memory). The same evaluation of the deviation $\mathcal{E}_f$ is also performed by running the EMRT model to get simulation data.

				Deviation $\mathcal{E}_f$ in $\Omega\times[0, T]$			
		Cumulative collision CPU times		BGK-based samples		EMRT-based samples	
		Measured (sec)	Normalized	Average	Percentage	Average	Percentage
BGK		17.02 ± 0.14	1.00	2.28×10^{-5}	–	4.30×10^{-5}	–
EMRT	Asymptotic solution	26.98 ± 1.45	1.59	9.08×10^{-12}	100%	8.94×10^{-12}	100%
	KBC stabilizer	36.65 ± 0.23	2.15	2.20×10^{-5}	57.4%	2.43×10^{-5}	99.3%
	Single iteration in NT	175.26 ± 0.51	10.30	1.41×10^{-9}	100%	1.40×10^{-9}	100%
	Newton's method	316.59 ± 0.27	18.59	–	–	–	–

should clarify that the results on cumulative CPU times of collision steps (Table I) are only for demonstration purposes. It is difficult to take an accurate measurement because several factors could affect the results and vary from execution to execution. For example, we choose the Eigen C + + library to solve the 3×3 matrix when implementing Newton's method [30]. The efficiency of Newton's method might be improved by taking a more optimized matrix solver. Also, the results are based on a single CPU core and would not directly reflect general simulation wall clock times by parallel computation. In practice, we observe that the streaming steps would take almost the same time as the BGK model's collision steps when running on a single CPU core. However, it would introduce extra cost into the streaming steps on parallel computation, where the communication between cores might be expensive and cannot be ignored.

When we evaluate the accuracy of asymptotic solutions in Sec. VI, the numerical experiment (Fig. 3) is not based on the actual simulation data. We collect the simulation data from the doubly periodic shear layer and perform similar experiments. We first simulate using the BGK model and extract 256×256×6400 precollision distribution states $\boldsymbol{f}(\boldsymbol{x}, t)$. For each precollision distribution state $\boldsymbol{f}(\boldsymbol{x}, t)$, we compute its postcollision state of EMRT model by asymptotic solutions, KBC stabilizer, Newton's method with single iteration, and Newton's method with converged solution, respectively. With the desired relaxation parameter ω (2), the postcollision states by different methods share the same density, fluid velocities, and second-order moments. The only difference happens to the higher-order moments, making the postcollision states have different entropy values. We use the same equation (34) to quantify the deviation $\mathcal{E}_f$ between a model and Newton's method, where it replaces $\boldsymbol{f}^{\text{asym}}$ with the corresponding postcollision state. Also, we take the postcollision state by BGK model as a baseline, where the deviation $\mathcal{E}_f$ shows how far away the most commonly used collision model is from the "exact" maximum entropy state. We measure if the EMRT model by different implementations could have a smaller deviation $\mathcal{E}_f$ than BGK model in postcollision state. The results in Table I shows that asymptotic solution could give a good approximation to the maximum entropy state (the average deviation $\mathcal{E}_f$ is at 10^{-12} order). Taking single step in Newton's method is also able to give good approximation (the average deviation $\mathcal{E}_f$ is at 10^{-9} order), but it is about 6.5 times slower than asymptotic solution. This tells us that the asymptotic solution is a much better choice to implement the EMRT models than Newton's method. Surprisingly, the postcollision state by the KBC stabilizer has just a slightly smaller deviation $\mathcal{E}_f$ than BGK model (their average deviations $\mathcal{E}_f$ are at the same order of magnitude, i.e., 10^{-5}). In BGK model, the relaxation parameter in higher-order moments is always the same value as ω; but KBC stabilizer would dynamically rescale ω based on the approximation to instantaneous critical point. The purpose of looking for an optimal relaxation parameter in higher-order moments is to maximize the entropy value of the postcollision state, but the results in Table I show there are just 57.4% of postcollision states that have a smaller deviation $\mathcal{E}_f$ by KBC stabilizer than BGK model. The KBC stabilizer might not drive the postcollision state to the maximum entropy state more significantly than BGK model, but it demonstrates much better stability performance for LBM. It raises an interesting question about what role the entropy value (or the maximum entropy state) plays in the collision step, and we leave it to future study. The asymptotic solution and Newton's method always have a smaller deviations $\mathcal{E}_f$ than BGK model. If we run the simulation by EMRT model and extract all the precollision states $\boldsymbol{f}(\boldsymbol{x}, t)$ to perform same experiments. The results (see the column of EMRT-based samples in Table I) are very similar to BGK-based experiments. Although the deviation $\mathcal{E}_f$ by KBC stabilizer is still at the same order as BGK model, 99.3% of postcollision states of KBC stabilizer have a smaller deviation $\mathcal{E}_f$ than of BGK model.

Finally, it is helpful to compare different implementations of the EMRT model from the perspective of the "bases" on

Algorithm 1 The algorithm to implement the collision step of the EMRT model by the asymptotic solutions.

```
Suppose $f(x, t)$ is the distribution function at node $x$ and time step $t$, which is about to experience the collision step. The postcollision state $f^{\text{post}}(x, t)$ of the EMRT model by asymptotic solutions could be calculated as
Calculate the local fluid density $\rho$, velocities $u_x$ and $u_y$, second-order moments $\Pi_{xx}$, $\Pi_{yy}$, and $\Pi_{xy}$ (6);
Calculate the second-order moments of equilibrium state $\Pi_{xx}^{\text{equ}}$, $\Pi_{yy}^{\text{equ}}$, and $\Pi_{xy}^{\text{equ}}$ (17);
Calculate the second-order moments of postcollision state $\Pi_{xx}^{\text{post}}$, $\Pi_{yy}^{\text{post}}$, and $\Pi_{xy}^{\text{post}}$ (16), where the relaxation parameter $\omega$ could be
    derived by the desired viscosity (2);
Calculate the value of small parameter $\epsilon = \Pi_{xy}^{\text{post}} - u_x u_y$ (26);
Calculate the asymptotic solutions for the higher-order moments $Q_{xyy}^{\text{post}}$, $Q_{yxx}^{\text{post}}$, and $Q_{xxyy}^{\text{post}}$ based on Eqs. (29), (31), (32), and (36);
Calculate the postcollision state $f^{\text{post}}$ by transforming back from its moment representation (9b).
```

which their solutions are built. The asymptotic solution is based on the "unperturbed" form (23), which is also referred to as "generalized Maxwellian." It is the exact solution to the entropy maximization problem under relaxed constraints [24]. Therefore, the asymptotic solution's starting point is close to the maximum entropy state. The KBC stabilizer [11,15] takes the Taylor expansion on the derivative of entropy function around the local equilibrium state. We consider it as something built on a local equilibrium state. Krämer *et al.* [16] take the Taylor expansion of the entropy function around the global equilibrium state (i.e., the equilibrium state under the unit fluid density and zero velocities) and make a quadratic approximation. This method is built on the global equilibrium state.

VIII. CONCLUSION

In this study, we propose to apply perturbation theory to solve the entropy maximization problem (15) and (18) accurately and efficiently. The advantage of asymptotic solutions is that it presents the solutions in the analytical form, and they can provide an excellent approximation to the exact solutions. We develop an efficient algorithm to realize the idea of the EMRT collision model. The flow simulation results show that EMRT could render similar accuracy to the most commonly used BGK model and exhibits much better stability performance. Also, we measure the relative errors of $\Pi_{\alpha\beta}^{\text{non-equ}}$ with respect to the rate of strain tensor (38). We observe the expected second-order convergence rate and find that the limit value of relative errors would increase with the increase of Ma. Interestingly, we find that the error of off-diagonal element $\Pi_{xy}^{\text{non-equ}}$ would converge to different limit values between the BGK model (and KBC stabilizer), where the higher-order moments share the same relaxation parameter, and EMRT models by asymptotic solutions and Newton's method, where the higher-order moments have different relaxation parameters. It shows the relaxation of higher-order moments does have some impact on the approximation of $\Pi_{\alpha\beta}^{\text{non-equ}}$. The mathematical analysis of error sources is complicated and we hope our observation could provide some insights into future work.

Some information about the nonequilibrium part in the higher-order moments is missing for initialization and boundary nodes. We need to figure out how to recover it and transform it back to the distribution functions. We propose to use the same idea of the EMRT model and assume that the distribution functions would reach the maximum entropy state under certain constraints for imposing the initial and boundary conditions. In order to realize this idea, it just needs to solve the same entropy maximization problem (15) and (18) we have discussed in detail in this paper. We demonstrate only the initial conditions in this study and will explore more about the boundary conditions in the future.

Finally, this paper discusses only the 2D case for the EMRT model. We will expand the asymptotic solutions to the 3D case in future work.

ACKNOWLEDGMENT

Y.Y. acknowledges support from the National Science Foundation under Award DMS 1753031.

APPENDIX

We summarize the steps, together with the corresponding equation number, to calculate the postcollision state $f^{\text{post}}(x, t)$ of the EMRT model by asymptotic solutions in Algorithm 1, which could serve as practical guidance to implement the EMRT collision step.

[1] S. Chen and G. D. Doolen, Lattice Boltzmann method for fluid flows, Annu. Rev. Fluid Mech. **30**, 329 (1998).

[2] H. Chen, S. Chen, and W. H. Matthaeus, Recovery of the Navier-Stokes equations using a lattice-gas Boltzmann method, Phys. Rev. A **45**, R5339 (1992).

[3] Y.-H. Qian, D. d'Humières, and P. Lallemand, Lattice BGK models for Navier-Stokes equation, Europhys. Lett. **17**, 479 (1992).

[4] D. d'Humieres, in *Rarefied Gas Dynamics Theory and Simulations*, edited by B. D. Shizgal and D. P. Weaver, Progress

in Astronautics and Aeronautics Vol. 159 (AIAA, Washington, DC, 1992).

[5] I. V. Karlin, A. Ferrante, and H. C. Öttinger, Perfect entropy functions of the lattice Boltzmann method, Europhys. Lett. **47**, 182 (1999).

[6] S. Ansumali, I. V. Karlin, and H. C. Öttinger, Minimal entropic kinetic models for hydrodynamics, Europhys. Lett. **63**, 798 (2003).

[7] S. S. Chikatamarla and I. V. Karlin, Entropic lattice Boltzmann method for turbulent flow simulations: Boundary conditions, Physica A **392**, 1925 (2013).

[8] F. Tosi, S. Ubertini, S. Succi, and I. V. Karlin, Optimization strategies for the entropic lattice Boltzmann method, J. Sci. Comput. **30**, 369 (2007).

[9] T. Yasuda and N. Satofuka, An improved entropic lattice Boltzmann model for parallel computation, Comput. Fluids **45**, 187 (2011).

[10] M. Atif, P. K. Kolluru, C. Thantanapally, and S. Ansumali, Essentially Entropic Lattice Boltzmann Model, Phys. Rev. Lett. **119**, 240602 (2017).

[11] I. V. Karlin, F. Bösch, and S. S. Chikatamarla, Gibbs' principle for the lattice-kinetic theory of fluid dynamics, Phys. Rev. E **90**, 031302(R) (2014).

[12] F. Bösch, S. S. Chikatamarla, and I. V. Karlin, Entropic multi-relaxation models for simulation of fluid turbulence, ESAIM: Proc. Surveys **52**, 1 (2015).

[13] T. L. Wilson, M. Pugh, and F. Dawson, Stabilization of the lattice Boltzmann method using information theory, arXiv:1801.03863.

[14] S. Boyd and L. Vandenberghe, *Convex Optimization* (Cambridge University Press, Cambridge, 2004).

[15] I. V. Karlin, F. Bösch, S. S. Chikatamarla, and S. Succi, Entropy-assisted computing of low-dissipative systems, Entropy **17**, 8099 (2015).

[16] A. Krämer, D. Wilde, K. Küllmer, D. Reith, and H. Foysi, Pseudoentropic derivation of the regularized lattice Boltzmann method, Phys. Rev. E **100**, 023302 (2019).

[17] C. M. Bender and S. A. Orszag, *Advanced Mathematical Methods for Scientists and Engineers I Asymptotic Methods and Perturbation Theory* (Springer, New York, 1999).

[18] M. H. Holmes, *Introduction to Perturbation Methods* (Springer, New York, 2013).

[19] P. A. Skordos, Initial and boundary conditions for the lattice Boltzmann method, Phys. Rev. E **48**, 4823 (1993).

[20] J. Latt, B. Chopard, O. Malaspinas, M. Deville, and A. Michler, Straight velocity boundaries in the lattice Boltzmann method, Phys. Rev. E **77**, 056703 (2008).

[21] P. J. Dellar, Bulk and shear viscosities in lattice Boltzmann equations, Phys. Rev. E **64**, 031203(R) (2001).

[22] T. Krüger, F. Varnik, and D. Raabe, Shear stress in lattice Boltzmann simulations, Phys. Rev. E **79**, 046704 (2009).

[23] S. Ansumali, Minimal kinetic modeling of hydrodynamics, Ph.D. thesis, ETH Zurich (2004).

[24] P. Asinari and I. V. Karlin, Generalized Maxwell state and H theorem for computing fluid flows using the lattice Boltzmann method, Phys. Rev. E **79**, 036703 (2009).

[25] L. Bernardin, P. Chin, P. DeMarco, K. O. Geddes, D. E. G. Hare, K. M. Heal, G. Labahn, J. P. May, J. McCarron, M. B. Monagan *et al.*, *Maple Programming Guide* (Maplesoft, a division of Waterloo Maple Inc., 1996-2021).

[26] T. Krüger, F. Varnik, and D. Raabe, Second-order convergence of the deviatoric stress tensor in the standard Bhatnagar-Gross-Krook lattice Boltzmann method, Phys. Rev. E **82**, 025701(R) (2010).

[27] R. Mei, L.-S. Luo, P. Lallemand, and D. d'Humières, Consistent initial conditions for lattice Boltzmann simulations, Comput. Fluids **35**, 855 (2006).

[28] L. N. Trefethen, *Spectral Methods in MATLAB* (SIAM, Philadelphia, 2000).

[29] M. L. Minion and D. L. Brown, Performance of under-resolved two-dimensional incompressible flow simulations, II, J. Comput. Phys. **138**, 734 (1997).

[30] G. Guennebaud, B. Jacob, P. Avery, A. Bachrach, S. Barthelemy, C. Becker, D. Benjamin, C. Berger, A. Berres, J. L. Blanco *et al.*, Eigen v3, http://eigen.tuxfamily.org (2010).