Acceleration of steady-state lattice Boltzmann simulations for exterior flows

Bo Liu[1] and Arzhang Khalili[1,2,*]
[1] *Max-Planck Institute for Marine Microbiology, 28359 Bremen, Germany*
[2] *Jacobs University Bremen, 28759 Bremen, Germany*

(Received 17 April 2008; revised manuscript received 12 September 2008; published 5 November 2008)

The simulation of a stationary fluid flow past an obstacle by the lattice Boltzmann method (LBM) in two dimensions is discussed. The combination of second-order expressions for far-field boundary conditions and a suitable treatment of the no-slip boundary condition at the obstacle surface with the nested grid-refinement technique can be applied to the LBM, resulting in a highly efficient method for the treatment of exterior flows. Furthermore, via replacing the nested time stepping by local time stepping, the resolution process can be substantially accelerated. A highly accurate drag coefficient was used to make an error assessment for various no-slip boundary conditions imposed on the obstacle surface. The analysis showed that the equilibrium method for treating the no-slip conditions is superior to halfway bounce-back and full-way bounce-back no-slip conditions when the relaxation time $\tau=1$. Also a τ-dependence test was made to evaluate the performance of different methods in the treatment of the no-slip boundary conditions.

DOI: 10.1103/PhysRevE.78.056701 PACS number(s): 47.11.−j

I. INTRODUCTION

The lattice-Boltzmann method (LBM) has been proved to be a promising method in simulating multiphase flows, suspension flows, and fluid flow in porous media [1]. Compared with other methods, the LBM has several advantages, such as its suitability for parallel algorithms, simple structure, simplicity in coding, and the straightforward incorporation of microscopic interactions, which are well demonstrated in the literature [2]. However, a low convergence rate in reaching the steady-state solution is a widely accepted defect of the standard LBM (see Ref. [3] and references therein). Some general methods for achieving higher computational efficiency with the LBM used grid refinement [4–10], local time-stepping methods [11], and code optimization [12–16]. In spite of all these suggestions, the efficiency of the LBM is still a severe issue when applied to low-Reynolds-number exterior flows, which play an increasingly important role in applications, such as marine aggregates [17–20] and sedimentation of small particles in the context of climate prediction [21,22]. A major difficulty in this problem stems from the large extent of the computational domain required. In these cases, the fluid velocity far from the obstacle is assumed to be asymptotically constant, and the forces exerted on the body need to be computed accurately. An accurate and efficient treatment of such flows depends significantly on how the boundary conditions at the far-field as well as at the obstacle surface are implemented. Far field boundary conditions can be approximated by an artificial boundary condition at the truncated boundary. A straightforward approach consists of using the asymptotic velocity condition $\mathbf{u}=\mathbf{u}_\infty$ on the numerical domain boundaries. Although this method is easy to implement, it was found to be inappropriate for the needs of numerical modeling, as it requires the use of excessively large domains [23,24]. Other approaches to this problem use extrapolation schemes on the boundaries so as to ensure a vanishing gradient perpendicular to the boundaries for the velocities [25] or fix the velocity in the upstream boundary and pressure in the downstream boundary. The main drawback of these approaches is that they are insufficient for imposing the asymptotic velocity $\mathbf{u}_\infty$ on the fluid and, therefore, cannot be used on all boundaries. Furthermore, it is difficult to ensure conservation of mass and momentum across domain boundaries. Bönisch *et al.* [23] introduced first-order expressions for the far-field boundary conditions in an explicit form, which depends on the total drag exerted on the obstacle. Using the finite-element method for the calculation of drag at a rectangular obstacle, they demonstrated that the extent of the computational domain, and with this the computational time, can be drastically reduced. While studying the same problem, Latt *et al.* [24] have demonstrated the applicability of first-order expressions for the treatment of the far-field boundary conditions in the LBM. Recently, Bönisch *et al.* [26] extended their previous formulation to a second-order expression and concluded that the extent of the computational domain can be reduced by one order of magnitude, when compared with the first-order expression.

On the other hand, the appropriate treatment of the no-slip boundary conditions on the obstacle surfaces can lead to a significant reduction of the mesh points required for numerical discretization of the obstacle in question. For the treatment of the no-slip boundary condition at the solid surfaces of obstacles, many suggestions have been made in the literature. First, one of the most basic algorithms is the bounce-back rule, which has undergone many modifications such as Bouzidi's rule [27], boundary-fitting method [4] and its improvement [28], link-averaged one-point approach [29], etc. All these schemes reduce to the half-way bounce-back condition (HBBC) once the solid wall is located in the middle of two subsequent grid points [30]. In contrast to this, the multireflection (MR) method [31,32] utilizes a more general closure relation, which is accurate up to third order in the Chapman-Enskog expression. It should be mentioned that the MR method does not reduce to bounce-back, even for rectangular boundaries and their midway locations [30,33].

[*] Corresponding author: akhalili@mpi-bremen.de

Due to its nonlocal properties, it is not considered in this study. The second approach in treating the no-slip boundary condition at solid surfaces consists of setting equilibrium distribution functions [equilibrium method (EM)] at the boundary and its modification with a suitably modified nonequilibrium part of distribution, such as that of the Inamuro method [34], Zou-He method [35], Halliday method [36], Hollis method [37], and Latt's regularized method [38,39]. All these schemes reduce to the equilibrium method, when the relaxation time τ in Bhatnagar-Gross-Krook (BGK) model [40,41] is equal to unity.

Due to the existence of a rectangular geometry in the present study, however, the location of the solid boundary points are aligned either with the lattice middle points or with the lattice points themselves. Therefore, among the methods mentioned in both categories, it would have been sufficient to consider the HBBC and EM when $\tau=1$. However, for the sake of comparison, a τ dependence test has been performed, for which all the methods listed under the second category have been included. Furthermore, for the sake of a consistent comparison, also the full-way bounce-back condition (FBBC) rule has been considered, as in both the FBBC and EM the solid boundary is located at lattice points.

A further important issue in accelerating the LBM for steady-state flows is the kind of time stepping-technique applied. In contrast to the traditionally used nested time stepping, in which two time steps on finer grids have to be performed during one time step on coarser meshes [4–7,24,42], we use in the present study a local time stepping (same time steps on all refinement levels). The reason is that the accuracy of the time interpolation is not crucial in stationary flows, considered here. This itself accelerates the convergence rate of the LBM, drastically.

Many solution methods for the lattice Boltzmann equations have been suggested, such as the single-relaxation-time lattice Boltzmann model (SRT, also known as BGK), two- or multiple-relaxation time (TRT, MRT) models [32,43,44], entropic lattice Boltzmann model (ELB) [45,46], and cascaded lattice Boltzmann model (CLBM) [47]. As the objective of this study is to emphasize the advantage of asymptotic far-field boundaries compared to their conventional treatments, the most simple and popular model, the BGK model, has been considered. A comparative study of different LBM models will be performed in a future study.

In the present study, we demonstrate that the second-order asymptotic expressions for the treatment of the far-field boundary conditions provide a highly efficient method for studying exterior flows.

The paper is organized as follows. Section II briefly describes the LBM model. Section III presents the asymptotic expressions of the boundary conditions. The numerical implementation including the refinement technique, boundary condition treatment, and force evaluation are brought forward in Sec. IV. Following this, in Sec. V, the computational results are discussed. Finally some concluding remarks are made in Sec. VI.

II. LATTICE-BOLTZMANN MODEL FOR THE GENERALIZED NAVIER-STOKES EQUATION

Frequently, the nine-velocity D2Q9 BGK model is used for two-dimensional calculations, leading to flow equations given by [40,41]

$$f_i(\mathbf{x}+\mathbf{e}_i\delta_t,t+\delta_t)-f_i(\mathbf{x},t)=-\frac{1}{\tau}[f_i(x,t)-f_i^{(eq)}(x,t)], \quad (1)$$

where $f_i(x,t)$ is the single-particle distribution function at position $\mathbf{x}$ and time t along the direction represented by the subscript i. Furthermore, δt is the time increment, τ is the nondimensional relaxation time, and $f_i^{(eq)}$ is the equilibrium distribution function, defined as

$$f_i^{(eq)}=\omega_i\rho\left[1+\frac{\mathbf{e}\cdot\mathbf{u}}{c_s^2}+\frac{\mathbf{uu}:(\mathbf{e}_i\mathbf{e}_i-c_s^2\mathbf{I})}{2c_s^2}\right], \quad (2)$$

in which ω_i is the weight, c_s is the speed of sound (set as $c_s^2=1/3$), and $\mathbf{I}$, stands for unit tensor. In the D2Q9 model, the discrete velocities are given by $\mathbf{e}_0=0$ and $\mathbf{e}_i=\lambda_i(\cos\theta_i,\sin\theta_i)c$, with $\lambda_i=1$, $\theta_i=(i-1)\pi/2$ for $i=1-4$ and $\lambda_i=\sqrt{2}$, $\theta_i=(i-5)\pi/2+\pi/4$ for $i=5-8$. Here, $c=\delta_x/\delta_t$ and δ_x is the lattice spacing. The weights are given by $\omega_0=4/9$, $\omega_i=1/9$ for $i=1-4$, and $\omega_i=1/36$ for $i=5-8$.

Accordingly, the fluid density and velocity are given by

$$\rho=\sum_i f_i, \quad (3)$$

$$\rho\mathbf{u}=\sum_i \mathbf{e}_i f_i, \quad (4)$$

respectively.

Through the Chapman-Enskog procedure [48], the following macroscopic equations can be derived from Eq. (1):

$$\frac{\partial\rho}{\partial t}+\boldsymbol{\nabla}\cdot(\rho\mathbf{u})=0, \quad (5)$$

$$\frac{\partial(\rho\mathbf{u})}{\partial t}+\boldsymbol{\nabla}\cdot(\rho\mathbf{uu})=-\boldsymbol{\nabla}p+\nu\boldsymbol{\nabla}\cdot[\rho(\boldsymbol{\nabla}\mathbf{u}+\mathbf{u}\boldsymbol{\nabla})], \quad (6)$$

where $p=c_s^2\rho$ is the pressure and the effective viscosity is defined as

$$\nu=\frac{1}{3}\left(\tau-\frac{1}{2}\right)\delta_t. \quad (7)$$

In the incompressible limit—i.e., $\rho\approx\rho_0=$const—Eqs. (5) and (6) reduce to the generalized Navier-Stokes equation.

III. ASYMPTOTIC EXPRESSIONS FOR THE BOUNDARY CONDITION AT THE TRUNCATED DOMAIN

The solution to the incompressible Navier-Stokes equation can be expanded in a finite series as a function of formal parameters depending on the drag and lift coefficients of the obstacle [23]. It is recognized that at a certain distance from the center, the structure of the flow does not depend on the specific details of the obstacle geometry, but only on the drag

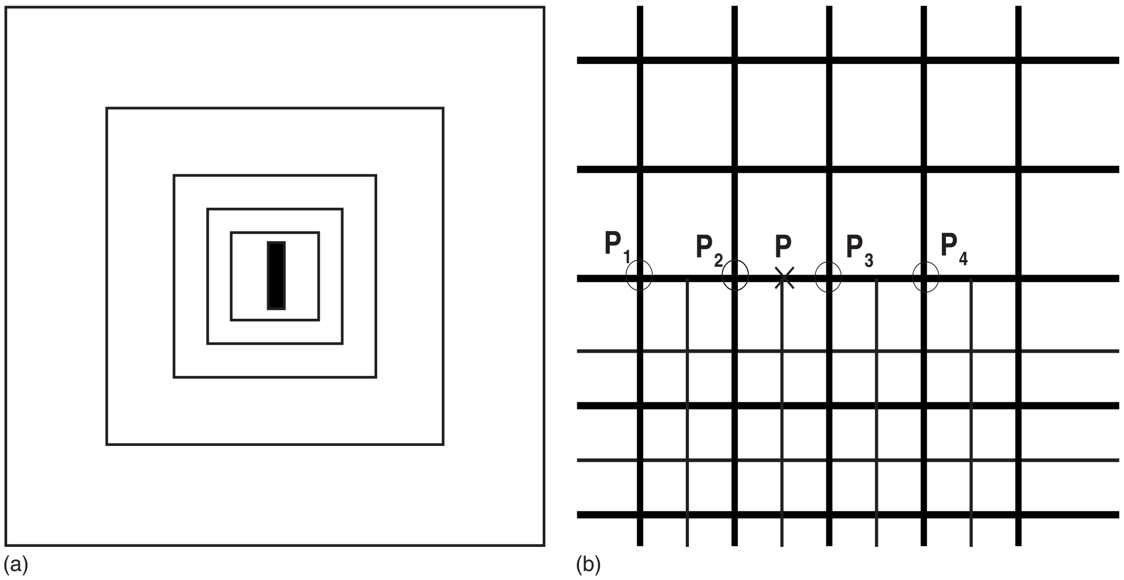

FIG. 1. (a) Structure of the numerical grid close to a rectangular obstacle (black vertical box). The flow goes from left to right. (b) Schematic representation of the interface between two adjacent grids. The values in the hanging node P are obtained from the cubic interpolation of nodes P_1–P_4.

and lift coefficients, resulting in the prescription of an explicit vector field that can be used as a boundary condition on the numerical domain boundaries. The asymptotic expansion (of order N) for the velocity $\mathbf{u}$ can be given as [26]

$$\mathbf{u}_N(x,y) = (u_\infty, 0) + \sum_{n=1}^{N} \sum_{m=1}^{n} \mathbf{u}_{n,m}(x,y). \tag{8}$$

The conventional constant velocity boundary condition can be regarded as the zeroth-order term of the expansion on the velocity field $\mathbf{u}=(u,v)$ at $\mathbf{r}=(x,y)$—i.e., $\mathbf{u}=(u_\infty,0)$. The first-order term reads [23]

$$u(x,y) = u_\infty\left(\frac{d}{\pi}\frac{x}{x^2+y^2} + \frac{b}{\pi}\frac{y}{x^2+y^2} - \theta(x)\frac{d}{\sqrt{\pi}}\frac{1}{\sqrt{x}}e^{-y^2/4x}\right), \tag{9}$$

$$v(x,y) = u_\infty\left(\frac{d}{\pi}\frac{y}{x^2+y^2} - \frac{b}{\pi}\frac{x}{x^2+y^2} - \theta(x)\frac{d}{2\sqrt{\pi}}\frac{y}{x^{3/2}}e^{-y^2/4x}\right), \tag{10}$$

with θ as the Heaviside function [i.e., $\theta(x)=1$ for $x>0$ and $\theta(x)=0$ for $x<0$] and $d=F_x/(2\rho l u_\infty^2)$ as the drag and $b=F_y/(2\rho l u_\infty^2)$ the lift coefficients. The parameters F_x and F_y are the drag and lift force acting on the obstacle, and $l=\nu/u_\infty$ is the viscous length, depending on the kinematic viscosity ν.

The second-order term of the expansion is rather complicated and is taken from Bönisch *et al.* [26]. For the sake of completeness, it is also given in the Appendix.

IV. NUMERICAL IMPLEMENTATION

A. Geometrical configuration and the refinement technique

The geometrical configuration studied is the flow past a rectangular obstacle as shown in Fig. 1(a). The LBM uses a grid with high resolution close to the center. Under this condition, the fluid is subject to sharp pressure and velocity variations. A grid refinement technique is applied with a hierarchy of nested grids that have a successively finer resolution as they approach the system center. This hierarchy of the nested grids is schematically represented in Fig. 1(a), while the interface between coarse and fine grids is shown in Fig. 1(b). In the finest grid (the immediate vicinity of the obstacle), the spatial and temporal step sizes (δ_x and δ_t) have been set to *unity*, while they are multiplied by 2 as we move from finer to coarser grids. This choice causes the velocity $\mathbf{u}$ to be the same from one refinement level to another. The geometry and the input parameters taken here are the same as in previous benchmark studies [23,24,26]: the rectangular obstacle is located in the center with an obstacle height (A) to a width ratio of 5:1, and the Reynolds number is given by $\mathrm{Re}=Au_\infty/\nu=1$. All simulations were run on quadratic domains of varying refinement levels. In each refinement level, the system sizes were 33×33, 65×65, 129×129, and 257×257 corresponding to obstacle heights of $A=20$, 40, 80, and 160, respectively. An exception is given when we used the HBBC scheme, for which the obstacle height were $A=25$, 45, 85, and 165. The relaxation time τ was set to unity, except when the effect of the τ dependence was studied. It should be mentioned that the parameters u, A, and ν are expressed in lattice units. The real physical units or dimensionless units can be obtained by redefining the time and length scales.

B. Treatment of the boundary conditions in the LBM

Two different types of boundary conditions have to be implemented in the LBM, simultaneously. These are asymptotic boundary conditions in the far-field and no-slip boundary conditions at the obstacle surfaces.

Asymptotic boundary conditions, as traditionally implemented [34,35], are subject to numerical instabilities for too coarse grids. Latt *et al.* [38,39] proposed to treat them by local and nonlocal regularized schemes. The former is computed from the first-order stress tensor Π^1, whereas the latter can be obtained from the strain rate $\mathbf{S}$. Latt *et al.* [38,39] also pointed out that the accuracy of the simulation is improved when using the nonlocal scheme. In addition to this, we found an improved stability behavior of the nonlocal regularized method compared to the local one. Therefore, in all simulations of the present study the nonlocal regularized scheme has been chosen.

Due to the existence of a rectangular geometry in the present study, the HBBC, FBBC, and EM were applied when $\tau=1$. When conducting the τ-dependence test, in addition to these, five other methods—namely, those of Inamuro *et al.* [34], Zou and He [35], Halliday *et al.* [36], Hollis *et al.* [37], and Latt's regularized method [38,39]—were considered. Although these methods were already discussed in the literature and recently reviewed by Latt *et al.* [39], yet two issues remained open. The first issue concerns the implementation of the corner nodes. To ensure an equal treatment of the corner points in all methods, the regularized method of Latt [38] has been employed here. The second point is the calculation of the strain rate $\mathbf{S}$, which is needed when using the methods of Halliday *et al.* and Hollis *et al.*. For doing this, both authors make use of finite differences. However, our simulations reveal (data not shown here) that numerical instabilities are encountered for $\tau > 2$. In order to bypass this problem, here we calculate the stress tensor, $\mathbf{S}$, from the first-order stress tensor Π^1 [48].

C. Force evaluation

Two different approaches for the force evaluation exist in the LBM. The first one is the momentum-exchange method, while the second one is the stress-integration method [49]. Usually, the momentum-exchange method is superior to the stress-integration method. The reason is that the former is based directly on the distribution functions, while the latter is derived from further processing of the distribution functions. In addition, the momentum-exchange method uses interpolations, whereas the stress-integration method is based on extrapolations [49]. For this reason, we employed the momentum-exchange method in the present study.

V. RESULTS AND DISCUSSION

A. Grid refinement

As the far-field boundaries depend on the unknown drag and lift coefficients (d and b), it is necessary that they be updated. After their updating, from the far-field boundaries toward the obstacle, spurious pressure waves will propagate through the entire flow field, which partially oscillate back and forth. Due to the very slow decay of these pressure waves, the convergence rate of the LBM is drastically reduced. This has been the reason for different grid-refinement schemes used to accelerate the LBM performance.

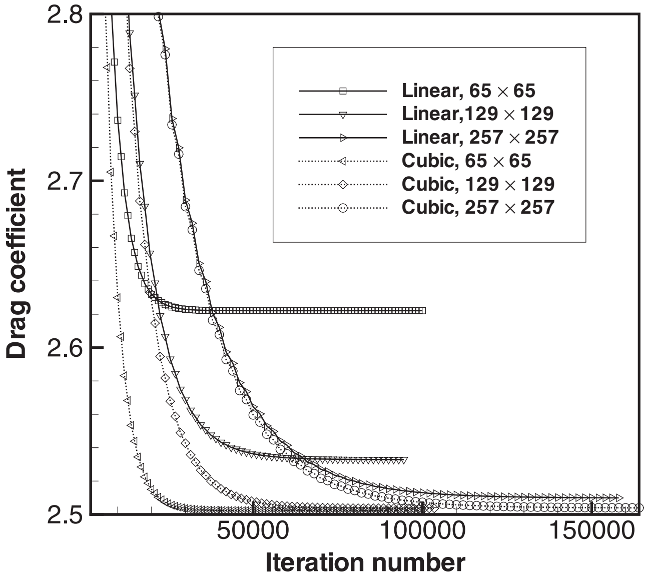

FIG. 2. Convergence histories of drag coefficients d using linear and cubic spline interpolations (see text). The height of the obstacle was $A=45$. A first-order asymptotic boundary condition far away from the center and the HBBC on the surface of the obstacle were used. To explore the effect of interpolation, a total number of 65 $\times$65, 129$\times$129, and 257$\times$257 square nodes were taken in each refinement level. For these arrangements almost equal system sizes to obstacle height ratios were generated by using eight, seven, and six refinement levels, respectively. The figure shows clearly that the cubic spline interpolations are much more accurate than the linear ones.

In grid refinements, there exists an interface between the coarse and the fine grids [see Fig. 1(b)]. The overlap of the interface is due to the fact that missing distributions on one refinement level have to be computed from the adjacent refinement level. Hence, to ensure the continuity of pressure, velocity, and also of their derivatives, the nonequilibrium part of the distribution function, f^{neq} (with $f^{neq}=f-f^{eq}$), has to be rescaled [4–7,42]. We used the method described by Dupuis and Chopard [7], which allows $\tau=1$. When implementing a linear interpolation, the "hanging" node P can be obtained from P_2 and P_3 [see Fig. 1(b)]. In contrast to this, when the linear interpolation is replaced by a cubic spline, the hanging node P can be formulated as a function of P_1, P_2, P_3, and P_4. Without further illustrations, this issue was emphasized in previous studies [4–6,42]. However, the quantification of the differences and the resulting advantages are highlighted here. As a direct comparison of the differences between these two interpolation methods is not possible in exterior flows (lack of analytical solutions), the following, rather nontrivial technique is suggested. By fixing the obstacle height (A) and the total system size (L), the grid refinement levels can be changed, resulting in an automatic alteration of the number of the square nodes in each refinement level. For example, as shown in Fig. 2, we fix the height of the obstacle (A) to 45 and the total system size (L) to 8193$\times$8193, corresponding to an equidistant $\delta x=1$ when no grid refinements take place. From here, we begin to coarsen the spatial spaces in different (2, 3, …, 7, 8) levels, resulting in 4097×4097, $2049\times 2049, \dots, 129\times 129$, 65×65 square nodes, respectively. The interpolation errors, of course, increase when more levels are implemented.

Figure 2 demonstrates clearly that cubic spline interpolations are superior to the linear ones. The reason is that the drag coefficient obtained from linear interpolation depends strongly on the refinement level, whereas it is not the case for cubic splines because spatial asymmetries are drastically reduced. For example, when 65×65 square nodes are taken, the difference between the drag coefficient d obtained from these two interpolation methods is approximately 6%. Therefore, an increased accuracy of the interpolations can be achieved by cubic splines, which are highly recommended in connection with the LBM.

Another important issue in the concept of LBM acceleration is the choice of the time-stepping scheme, which decides the sequences by which the grid levels are updated. From time-stepping scheme and the convergence histories of the drag coefficient, the total iteration number for the calculation can be obtained. The acceleration rate can be easily estimated since CPU time is proportional to the iteration number. In the traditionally used nested time stepping, one time step on the coarser grids is run during two time steps on the finer ones [4–7,24,42], leading to 2^N-1 total time steps for one loop, from which many are, in fact, not necessary. To bypass this problem, a suitable choice of time-stepping scheme can be made by making use of the fact that time interpolation does not matter in stationary flows considered here.

This, namely, allows us to introduce a "local time-stepping scheme," in which a single time step in all refinement levels is applied. By doing so, only a total number of N time steps are required in one loop. In other words, compared to the nested time stepping, in the local time stepping, the computational time is reduced by $(2^N-1)/N$. When using 8 coarsening levels, the local time stepping requires 30 times fewer iterations compared to the nested time stepping (see Fig. 3). It should be mentioned that this observation is independent of the choice of A. The real benefit of the local time stepping becomes pronounced when an increase of N is necessary in the problem of interest. Although the refinement level number is limited to 8 in the BGK model for $\tau=1$, it can be extended up to 12 when the MRT model is employed [9].

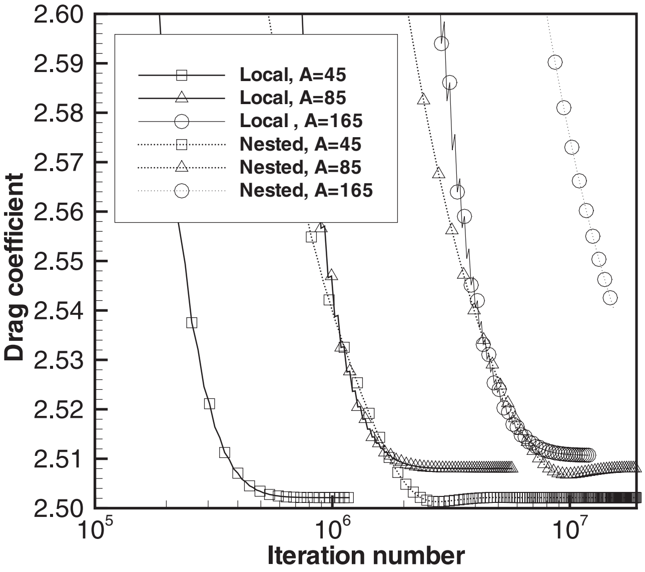

FIG. 3. Convergence histories of the drag coefficients d using local and nested time stepping. The refinement level was fixed to 8. In each refinement level, a total number of 65×65, 129×129, and 257×257 square nodes were used for obstacle heights of $A=45$, $A=85$, and $A=165$, respectively, resulting in almost equal system sizes to obstacle height ratios. A first-order asymptotic boundary condition far away from the center and the HBBC on the surface of the obstacle were used. Due to the extremely large iteration number necessary for the convergence of the drag coefficient for the nested time-stepping method with $A=165$, the computations were terminated. The figure shows clearly that much fewer iteration numbers are required by local time stepping compared with the nested time stepping.

B. Treatment of far-field boundary conditions

Due to the symmetry of the obstacle, the lift coefficient b was equal to zero in all simulations. Figure 4(a) shows the results of a drag coefficient calculation as a function of the system size (L) for an obstacle height of $A=160$ when using an EM scheme. As shown in Fig. 4(a), the drag coefficient obtained by the first-order boundary condition approaches the reference solution from the top, whereas that from the second-order boundary condition approaches the reference solution from the bottom. Such asymptotic behaviors were also found when using finite-volume methods for drag calculation of the same geometry [26]. It should be mentioned that, for a more precise comparison between the results of the drag factor obtained from the BGK model and the reference value of Bönisch *et al.* [26], comparatively large system sizes were necessary. The latter is, however, known to lead to numerical instabilities in the BGK model [9]. Hence, to do away the need for large system sizes, one may extrapolate the system size to infinity. With this technique, we obtained a drag factor of $d=2.5147$. It should be emphasized that this value can be further improved by taking larger obstacle heights A, which lead to a reduction of the discretization error. The relative errors of the drag coefficients with respect to this value are plotted in Fig. 4(b). As depicted in the figure, compared with the zeroth- and first-order asymptotic boundary conditions, the second-order one substantially reduced the computational cost for determining the drag to a given precision. It should be mentioned that the additional computational time needed for the evaluation of the second-order boundary terms is negligible. The drag coefficient d and its relative error obtained from the traditional boundary condition (fixed constant velocity in the upstream boundary and a constant pressure in the downstream boundary) are also plotted in Fig. 4. Obviously, this kind of boundary condition produces only a slight improvement over the zeroth-order boundary condition and cannot compete with the second-order boundary conditions.

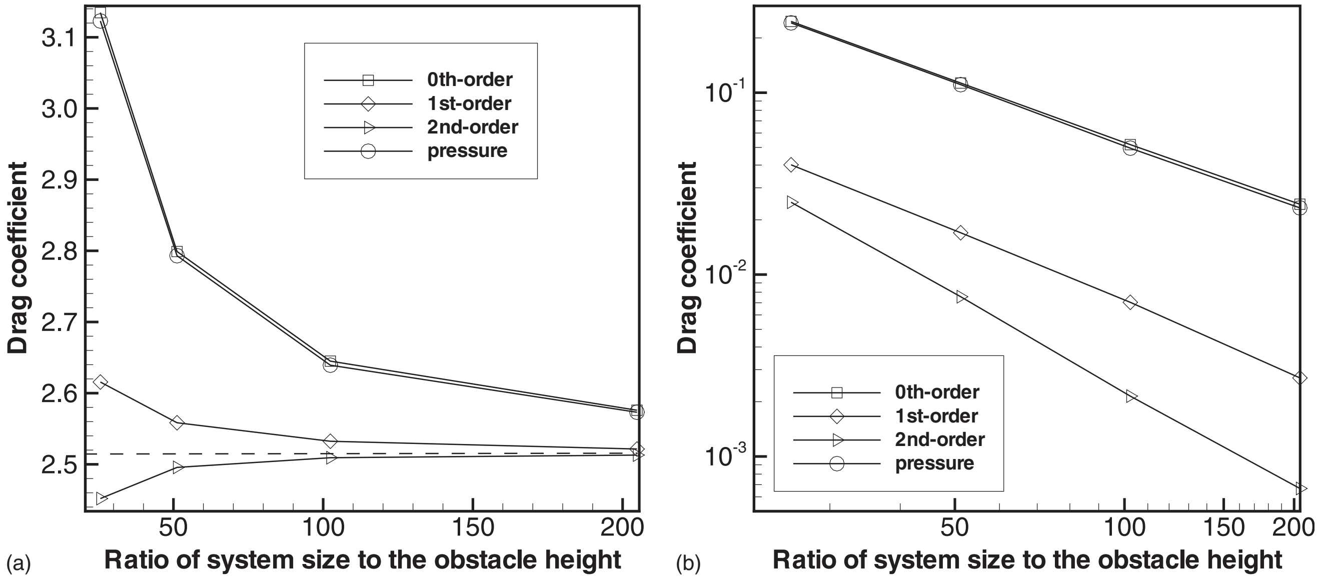

FIG. 4. (a) Drag coefficient d as a function of the system size with asymptotic far-field boundary conditions of the zeroth, first, and second order, as well as traditional boundary condition implementation (fixed constant upstream velocity and constant downstream pressure). The EM scheme is applied on the surface of the obstacles. The height of the obstacle was $A=160$, and a total number of 257×257 square nodes were used in each refinement level. The dashed line in the figure is the reference solution of Bönisch *et al.* [26] with $d=2.5145$. (b) Relative error of the drag coefficient according to the reference solution $d=2.5147$ obtained from the extrapolation of the system size to infinity. Both figures show clearly that second-order asymptotic boundary conditions lead to more accurate results.

C. Treatment of no-slip boundary conditions with $\tau=1$

The effect of no-slip boundary conditions on d is shown in Fig. 5. As can be seen, the drag is overestimated by the FBBC scheme and underestimated by the HBBC, while the drag coefficient d from the EM lies in between and closer to the reference solution $d=2.5145$. Furthermore, since the discretization error from a no-slip boundary condition depends on the obstacle height (A), it is important to study its effect on the drag coefficient d. As can be seen in Fig. 6 the drag obtained from the FBBC scheme strongly depends on A. The dependence of the drag on A obtained for the EM scheme was weaker than that of the other two no-slip boundary conditions. As the trend shows the drag coefficients display an asymptotic behavior for large system sizes and can be determined by extrapolation (extrapolation to infinity). To reduce the truncation error of the far-field asymptotic boundary condition and the discretization error at the obstacle surface, both the system size (L) and obstacle height A had to be extrapolated to the infinity, respectively. This is done in the following manner: Each drag coefficient for a given A in Fig. 7(a) corresponds to a value that was obtained using the extrapolation procedure discussed above. In Fig. 7(a) one can also see that the drag coefficients obtained from implementation of different no-slip conditions tend toward the reference value, but with different speed. The quantification of the relative errors made by all three no-slip boundary conditions with respect to d is made in Fig. 7(b). As expected, the lowest relative errors are associated with the EM scheme. For simplicity, the dotted lines are added to the figure representing the -1 and -2 exponents of the power-law functions of the relative errors. As can be seen, the scaling behavior of relative errors follows a first-order trend for both bounce-back conditions. Within the obstacle height (A) ranges studied here, however, the EM scheme behaved partially linear and partially quadratic. The cause of this might be due to the combined effect of first-order accurate pressure and second-order accurate velocity [30].

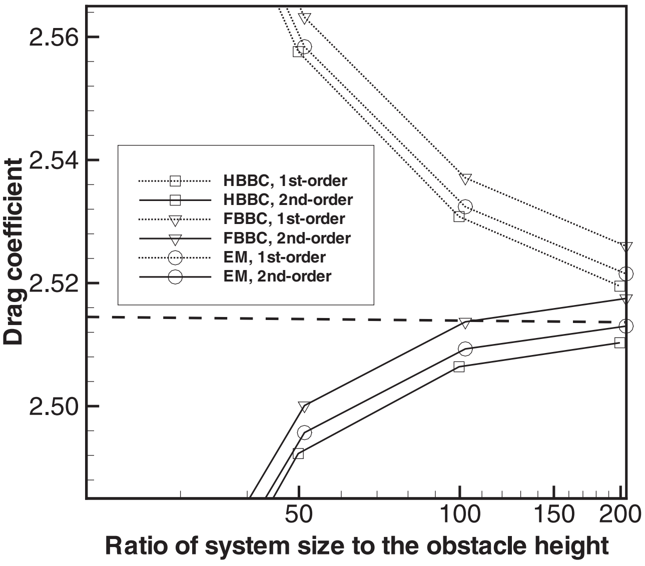

FIG. 5. Drag coefficient d as a function of the system size with first- and second-order asymptotic boundary conditions far away from the center. The FBBC, HBBC, and EM schemes were applied at the obstacle surfaces. The height of the obstacle is $A=160$ (for the HBBC, $A=165$), and 257×257 square nodes were used in each refinement level. The dashed line is the reference solution of Bönisch *et al.* [26] with $d=2.5145$. The figure shows that first-order asymptotic boundary conditions require larger system sizes than the second-order one.

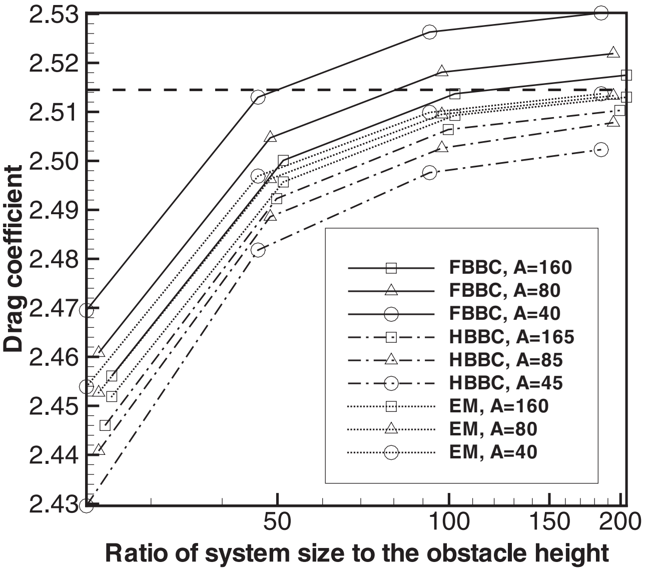

FIG. 6. Drag coefficient d as a function of the system size and obstacle height. The FBBC, HBBC, and EM schemes are applied at the obstacle surface. Here second-order asymptotic far-field boundary conditions have been used for all simulations. The dashed horizontal line represents the reference solution with $d=2.5145$ of Bönisch *et al.* [26]. As can be seen in the figure, the drag coefficient d depends on both the system size and obstacle height. To eliminate the discretization error at the obstacle surface and truncation error of the asymptotic boundary condition, both the system size and obstacle height have to be extrapolated to infinity. As demonstrated, the discretization error obtained from the EM scheme is much lower than those produced by the FBBC and HBBC for all obstacle heights.

D. Effect of the variable τ on the drag coefficient

As mentioned already in the literature [33,50], the BGK model is sensitive with respect to the choice of relaxation time τ. Hence, in order to evaluate the use of different methods, a τ-sensitivity analysis has been performed for calculation of the drag coefficient. To do this, various no-slip boundary condition treatment techniques, such as those suggested by Inamuro *et al.* [34], Zou and He *et al.* [35], Halliday *et al.* [36], Hollis *et al.* [37], and Latt [38], along with the HBBC, FBBC, and EM schemes, were applied. As illustrated in Fig. 8, the drag coefficients obtained from these methods reduce monotonically with larger τ. When τ increased, the error in the drag coefficient produced by the FBBC method decreased first and increased after $\tau\approx 4.2$. Note that this value will change when the obstacle height is varied. In contrast to this, the drag coefficient calculated using the HBBC led to a continuous decrease and, within the τ range studied, could not reach the reference value at all. Furthermore, all equilibrium-based methods turned out to match with the reference value around $\tau\approx 1$.

As shown by the dot-diamond line in Fig. 8, compared to other methods of this category, the coefficient is ill behaved in the entire τ range, except for $\tau=1$. The reason is that the EM takes into account only the zeroth-order contribution to the hydrodynamics, while other EM-based methods (solid lines in the figure) do consider the first-order term too. Among these methods, the Zou-He method shows a better accuracy in the larger-τ region, while others differ little.

VI. CONCLUSIONS

Enhancement of the accuracy and the convergence rate of the lattice Boltzmann method is a severe issue when it is

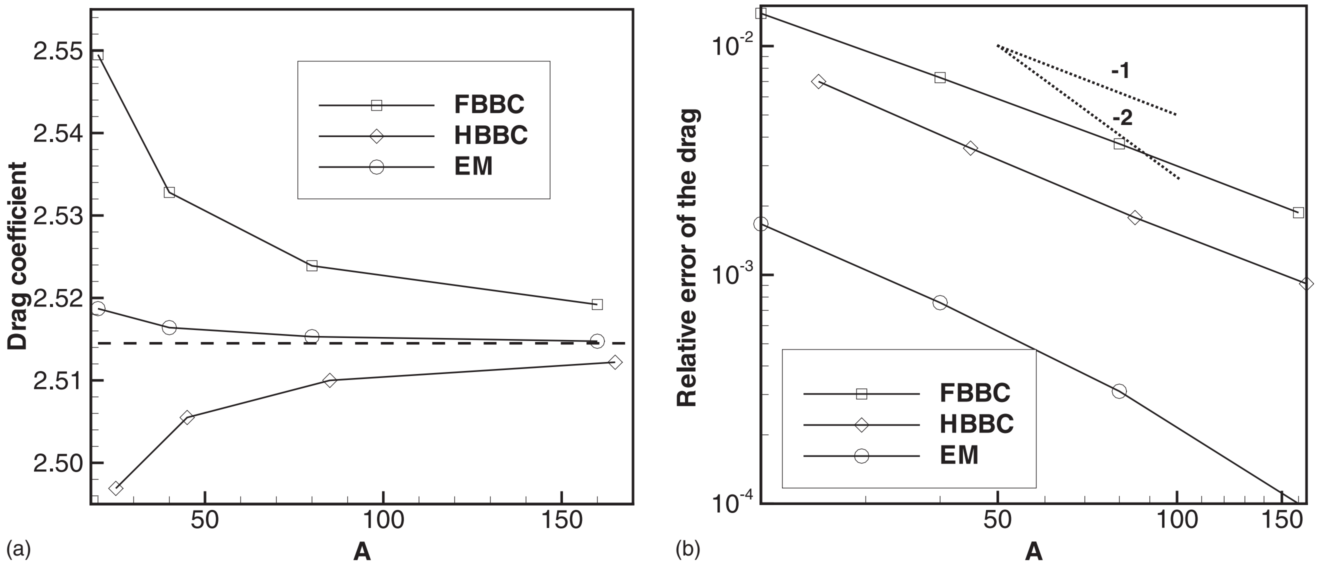

FIG. 7. (a) Drag coefficient d as a function of the obstacle height A with the HBBC, FBBC, and EM schemes. The drag coefficients were obtained from extrapolation of the system size to infinity. The dashed line is the reference value of $d=2.5145$ [26]. (b) Relative errors of the drag coefficient computed from different schemes. The two dotted lines represent a power-law function with the slopes of −1 and −2. As shown, the drag coefficients obtained from all three no-slip boundary condition types are likely to converge toward the reference value. However, as demonstrated, the convergence speed of the EM scheme is superior to the full-way bounce-back and halfway bounce-back conditions.

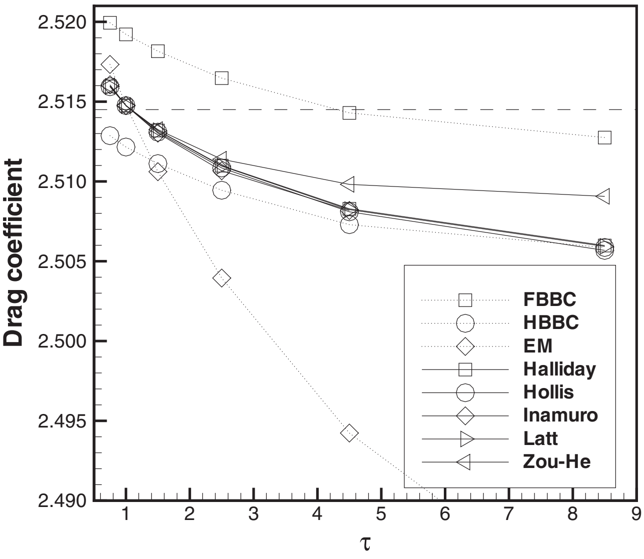

FIG. 8. Drag coefficient d as a function of relaxation time τ. The FBBC, HBBC, EM, Halliday, Hollis, Inamuro, Latt, and Zou-He schemes were implemented at the obstacle surface. The obstacle height A was 160 (in all cases, except in the HBBC with 165). All computations have adopted 257×257 square nodes in each refinement level. Second-order asymptotic far-field boundary conditions have been applied. The drag coefficients were obtained from extrapolation of the system size to infinity. The dashed horizontal line represents the reference solution of Bönisch *et al.* [26] with $d=2.5145$. As can be seen in the figure, the drag coefficients d obtained from various no-slip boundary conditions are τ dependent in the BGK model.

applied to exterior flows. Here, we demonstrate that the combination of second-order expressions for far-field boundary conditions and a suitable treatment of the no-slip boundary condition at rectangular obstacle surfaces results in a highly efficient and accurate method when coupled with nested grid refinement. Compared with the zeroth- and first-order asymptotic boundary conditions, the second-order one substantially reduced the computational cost for determining the drag to a given precision. For the treatment of the no-slip boundary conditions, the equilibrium method was found to be superior to bounce-back boundary conditions for a rectangular obstacle when $\tau=1$. From the τ-dependence test, the Zou-He method was found to be more accurate in large-τ regions. In addition, a further acceleration of the LBM in steady-state flows may be obtained by a local time-stepping technique.

ACKNOWLEDGMENTS

We would like to thank Dr. J. Latt and Dr. P. Wittwer for fruitful discussions and an anonymous referee for critical and constructive comments. One of the authors (A.K.) thanks the Max Planck Society for providing financial support of this work.

APPENDIX: ASYMPTOTIC EXPANSION FOR u OF ORDER *N*

We define the dimensionless coordinates $\tilde{x}=x/l$, dimensionless vector field $\mathbf{u}$, and a dimensionless pressure p given by

$$\tilde{\mathbf{u}}(\tilde{x}) = \mathbf{u}(x)/u_\infty, \tag{A1}$$

$$\tilde{p}(\tilde{x}) = p(x)/(\rho u_\infty^2). \tag{A2}$$

After dropping the tilde, we have the following asymptotic expansion (of order N) for the velocity $\mathbf{u}$, which can be given by

$$\mathbf{u}_N(x,y) = \mathbf{e}_1 + \sum_{n=1}^{N}\sum_{m=1}^{n} \mathbf{u}_{n,m}(x,y), \tag{A3}$$

where $\mathbf{e}_1=(1,0)$. To first order we have

$$u_{1,1}(x,y) = u_{1,1,E}(x,y) - \theta(x)\frac{d}{\sqrt{\pi}}\frac{1}{\sqrt{x}}e^{-y^2/4x}, \tag{A4}$$

$$v_{1,1}(x,y) = v_{1,1,E}(x,y) - \theta(x)\frac{d}{2\sqrt{\pi}}\frac{y}{x^{3/2}}e^{-y^2/4x}, \tag{A5}$$

with θ the Heaviside function [i.e., $\theta(x)=1$ for $x>0$ and $\theta(x)=0$ for $x<0$], and

$$u_{1,1,E}(x,y) = \frac{d}{\pi}\frac{x}{x^2+y^2} + \frac{b}{\pi}\frac{y}{x^2+y^2}, \tag{A6}$$

$$v_{1,1,E}(x,y) = \frac{d}{\pi}\frac{y}{x^2+y^2} - \frac{b}{\pi}\frac{x}{x^2+y^2}. \tag{A7}$$

To second order we have

$$u_{2,1}(x,y) = \theta(x)\frac{bd}{2\pi^{3/2}}\frac{\ln x}{x}\frac{y}{\sqrt{x}}e^{-y^2/4x}, \tag{A8}$$

$$v_{2,1}(x,y) = \theta(x)\frac{bd}{2\pi^{3/2}}\frac{1}{x^{3/2}}\left[\ln x\left(-1+\frac{1}{2}\frac{y^2}{x}\right)+2\right]e^{-y^2/4x}, \tag{A9}$$

and

$$u_{2,2}(x,y) = u_{2,2,E}(x,y) + \theta(x)d^2\frac{1}{x}f'\left(\frac{y}{\sqrt{x}}\right) + \lambda\,\theta(x)f_\infty d^2\frac{3}{8}\frac{1}{x^2}\left[\left(1 + \frac{|y|}{\sqrt{x}}\right)\left(1-\frac{1}{2}\frac{y^2}{x}\right) + \frac{|y|}{\sqrt{x}}\right]e^{-y^2/4x}, \tag{A10}$$

$$v_{2,2}(x,y) = v_{2,2,E}(x,y) + \theta(x)\frac{d^2}{2}\frac{1}{x^{3/2}}\left\{\left[f\left(\frac{y}{\sqrt{x}}\right) - f_\infty\,\mathrm{sgn}(y)\right] + \frac{y}{\sqrt{x}}f'\left(\frac{y}{\sqrt{x}}\right)\right\} + \lambda\,\theta(x)f_\infty d^2\frac{3}{4}\frac{1}{x^{5/2}}\left[\left(1 + \frac{|y|}{\sqrt{x}}\right)\frac{y}{\sqrt{x}}\left(1-\frac{1}{8}\frac{y^2}{x}\right) + \frac{1}{4}\frac{y^2}{x}\,\mathrm{sgn}(y)\right]e^{-y^2/4x}, \tag{A11}$$

where

$$u_{2,2,E}(x,y)=f_{\infty}\frac{d^2}{2}\frac{|y|}{r^2}\left(\frac{1}{r_2}-\frac{r_2}{r}\right), \tag{A12}$$

$$v_{2,2,E}(x,y)=f_{\infty}\frac{d^2}{2}\frac{\mathrm{sgn}(y)}{r}\left(-\frac{1}{r_2}-\frac{x}{r_2 r}+\frac{x r_2}{r^2}\right), \tag{A13}$$

with $r=\sqrt{x^2+y^2}$, $r_2=\sqrt{2r+2x}$, $\lambda=1$, $f_{\infty}=-1/\sqrt{2\pi}$, and

$$f(z)=-\frac{1}{\sqrt{2\pi}}\mathrm{erf}\left(\frac{z}{\sqrt{2}}\right)+\frac{1}{2\sqrt{\pi}}\mathrm{erf}\left(\frac{z}{2}\right)e^{-z^2/4}, \tag{A14}$$

where "erf" is the error function—i.e., $\mathrm{erf}(z)=2/\sqrt{\pi}\int_0^z \exp -\zeta^2 d\zeta$.

The material mentioned here has been taken from the work of Bönisch *et al.* [26].

[1] H. Chen, Phys. Rev. E **58**, 3955 (1998).

[2] S. Succi, *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond* (Oxford University Press, Oxford, 2001).

[3] Z. Guo, T. S. Zhao, and Y. Shi, Phys. Rev. E **70**, 066706 (2004).

[4] O. Filippova and D. Hänel, J. Comput. Phys. **147**, 219 (1998).

[5] D. Yu, R. Mei, and W. Shyy, Int. J. Numer. Methods Fluids **39**, 99 (2002).

[6] B. Crouse, E. Rank, M. Krafczyk, and J. Tölke, Int. J. Mod. Phys. B **17**, 109 (2003).

[7] A. Dupuis and B. Chopard, Phys. Rev. E **67**, 066707 (2003).

[8] D. Yu and S. Girimaji, Physica A **362**, 118 (2006).

[9] J. Tölke, S. Freudiger, and M. Krafczyk, Comput. Fluids **35**, 820 (2006).

[10] M. Rheinländer, J. Stat. Phys. **121**, 49 (2005).

[11] T. Imamura, K. Suzuki, T. Nakamura, and M. Yoshida, J. Comput. Phys. **202**, 645 (2005).

[12] T. Pohl, M. Kowarschik, J. Wilke, K. Iglberger, and U. Rüde, Parallel Process. Lett. **13**, 549 (2003).

[13] A. C. Velivelli and K. M. Bryden, Concurrency Comput.: Pract. Exper. **16**, 1415 (2004).

[14] G. Wellein, T. Zeiser, S. Donath, and G. Hager, Comput. Fluids **35**, 910 (2006).

[15] R. Argentini, A. F. Bakker, and C. P. Lowe, FGCS, Future Gener. Comput. Syst. **20**, 973 (2004).

[16] K. Mattila, J. Hyväluoma, T. Rossia, M. Aspnäs, and J. Westerholm, Comput. Phys. Commun. **176**, 200 (2007).

[17] T. Kiørboe, H. Plough, and U. H. Thygesen, Mar. Ecol.: Prog. Ser. **211**, 1 (2001).

[18] T. Kiørboe and U. H. Thygesen, Mar. Ecol.: Prog. Ser. **211**, 15 (2001).

[19] T. Kiørboe and G. A. Jackson, Limnol. Oceanogr. **46**, 1309 (2001).

[20] S. Bhattacharyya, S. Dhinakaran, and A. Khalili, Chem. Eng. Sci. **61**, 4451 (2006).

[21] M. B. Baker, Science **276**, 1072 (1997).

[22] H. R. Pruppacher and J. D. Klett, *Microphysics of Clouds and Precipitation* (Kluwer, Dordrecht, 2000).

[23] S. Bönisch, V. Heuveline, and P. Wittwer, J. Math. Fluid Mech. **7**, 85 (2005).

[24] J. Latt, Y. Grillet, B. Chopard, and P. Wittwer, Math. Comput. Simul. **72**, 169 (2006).

[25] D. Yu, R. Mei, and W. Shyy, Prog. Comput. Fluid Dyn. **5**, 3 (2003).

[26] S. Bönisch, V. Heuveline, and P. Wittwer, J. Math. Fluid Mech. **10**, 45 (2008).

[27] M. Bouzidi, M. Firdaouss, and P. Lallemand, Phys. Fluids **13**, 3452 (2001).

[28] R. Mei, L.-S. Luo, and W. Shyy, J. Comput. Phys. **155**, 307 (1999).

[29] M. Junk and Z. Yang, Phys. Rev. E **72**, 066701 (2005).

[30] M. Junk and Z. Yang, J. Stat. Phys. **121**, 3 (2005).

[31] I. Ginzburg and D. d'Humières, Phys. Rev. E **68**, 066614 (2003).

[32] I. Ginzburg, F. Verhaeghe, and D. d'Humières, Comm. Comp. Phys. **3**, 427 (2008).

[33] C. Pan, L.-S. Luo, and C. T. Miller, Comput. Fluids **38**, 898 (2006).

[34] T. Inamuro, M. Yoshino, and F. Ogino, Phys. Fluids **7**, 2928 (1995).

[35] Q. Zou and X. He, Phys. Fluids **9**, 1591 (1997).

[36] I. Halliday, L. Hammond, and C. Care, J. Phys. A **35**, L157 (2002).

[37] A. Hollis, I. Halliday, and C. Care, J. Phys. A **39**, 10589 (2006).

[38] J. Latt, Ph.D. thesis, University of Geneva, 2007.

[39] J. Latt, B. Chopard, O. Malaspinas, M. Deville, and A. Michler, Phys. Rev. E **77**, 056703 (2008).

[40] Y. H. Qian, D. D'Humières, and P. Lallemand, Europhys. Lett. **17**, 479 (1992).

[41] Y. H. Qian and S. A. Orszag, Europhys. Lett. **21**, 255 (1993).

[42] O. Filippova and D. Hänel, J. Comput. Phys. **165**, 407 (2000).

[43] I. Ginzburg, F. Verhaeghe, and D. d'Humières, Comm. Comp. Phys. **3**, 519 (2008).

[44] P. Lallemand and L.-S. Luo, Phys. Rev. E **61**, 6546 (2000).

[45] B. M. Boghosian, J. Yepez, P. V. Conveney, and A. Wager, Proc. R. Soc. London, Ser. A **457**, 717 (2001).

[46] F. Tosi, S. Ubertini, S. Succi, and I. V. Karlin, J. Sci. Comput. **30**, 369 (2007).

[47] M. Geier, A. Greiner, and J. G. Korvink, Phys. Rev. E **73**, 066705 (2006).

[48] B. Chopard and M. Droz, *Cellular Automata Modeling of Physical Systems* (Cambridge University Press, Collection Aléa, 1998).

[49] R. Mei, D. Yu, W. Shyy, and L.-S. Luo, Phys. Rev. E **65**, 041203 (2002).

[50] X. He, Q. Zou, L.-S. Luo, and M. Dembo, J. Stat. Phys. **87**, 115 (1997).