Optimizing lattice Boltzmann simulations for unsteady flows

A.M. Artoli, A.G. Hoekstra *, P.M.A. Sloot

Section Computational Science, University of Amsterdam, Kruislaa 403, 1098 SJ Amsterdam, The Netherlands

Abstract

We present detailed analysis of a lattice Boltzmann approach to model time-dependent Newtonian flows. The aim of this study is to find optimized simulation parameters for a desired accuracy with minimal computational time. Simulation parameters for fixed Reynolds and Womersley numbers are studied. We investigate influences from the Mach number and different boundary conditions on the accuracy and performance of the method and suggest ways to enhance the convergence behavior.
© 2005 Elsevier Ltd. All rights reserved.

PACS: 47.11.+j; 87.19.UV

1. Introduction

For years to come, a necessity for efficient and robust computational fluid dynamics (CFD) solvers will be demanded by scientists working at the edge of available computer power. It has been understood by many authors that transport phenomena can be studied from a kinetic theory point of view [1,2] where an alternative description is given through the Boltzmann equation for the probability density function $f(t, \vec{x}, \vec{\xi})$ for particles of velocity $\vec{\xi}$ at point $\vec{x}$ and time t. The Boltzmann equation was mainly developed for ideal gases, but in the limit of small mean free path

---

* Corresponding author. Tel.: +20 525 7543; fax: +20 525 7490.
   E-mail addresses: artoli@science.uva.nl (A.M. Artoli), alfons@science.uva.nl (A.G. Hoekstra), sloot@science.uva.nl (P.M.A. Sloot).
   URL: http://www.science.uva.nl/research/scs (A.G. Hoekstra).

between molecular collisions, a gas may be considered as a continuum fluid. The main advantage over the Navier–Stokes equations is the capability of the Boltzmann solution to capture complex fluid behavior such as multicomponent flow and flow in porous media. After the link between the Boltzmann equation and hydrodynamics was well established (see e.g. Ref. [3]), the need for efficient numerical solvers was large, since the Boltzmann equation is hard to solve in general. Perturbation methods such as the Chapmann–Enskog and Hilbert techniques are common numerical approaches taken. After simplifying the collision operator, solutions are obtained as asymptotes to the Boltzmann equation. However, these solutions are not readily applicable to flows of common engineering interest.

A new way of solving the Boltzmann equation has been realized soon after Frisch et al. [4] introduced the lattice gas cellular automata (LGCA), an automata that is capable of capturing complex fluid nature by obeying local conservation laws. A few shortcomings of the LGCA were recognized and intensively investigated. Those are, among others, the lack of Galilean invariance, statistical noise, low Reynolds number (high viscosity) and exponential complexity of the collision operator [5–7]. The lattice Boltzmann method was introduced to remove these shortcomings [8]. The idea is simply to replace the Boolean LGCA occupation numbers n_i with ensemble-averaged populations $f_i = <n_i>$. The system has become capable to capture many features in the evolving nature [9–11]. A little later, the lattice Boltzmann equation has been realized as a direct discretization of the simplified Boltzmann equation [12], independent of lattice-gas automata, as will be demonstrated subsequently.

2. The lattice Boltzmann equation

We start from the Boltzmann equation with the BGK approximation for the collision operator

$$\frac{\mathrm{d}f}{\mathrm{d}t} + \vec{\xi} \cdot \nabla f = -\frac{1}{\lambda}(f - g) \tag{1}$$

which describes the time evolution of the probability distribution function $f \equiv f(\vec{x}, \vec{\xi}, t)$ of a single particle moving with a microscopic velocity $\vec{\xi}$ and relaxing with a collision relaxation time λ to a Maxwellian equilibrium distribution

$$g = \frac{\rho}{(2\pi RT)^{\frac{3}{2}}} \exp\left(-\frac{(\vec{\xi} - \vec{u})^2}{2RT}\right) \tag{2}$$

which is a function of the ideal gas constant R, the macroscopic quantities ρ, $\vec{u}$ and T, and the microscopic velocity $\vec{\xi}$, with ρ the density, $\vec{u}$ the fluid velocity and T its temperature. The macroscopic quantities are computed from the microscopic moments of f. Approximating the formal integral of the Boltzmann equation over a time step δ_t and ignoring the leading terms of $\mathrm{O}(\delta_t^2)$ by the use of Taylor expansion, yields a first order in time discretization for the Boltzmann equation [12] given by

$$f(\vec{x} + \vec{\xi}\delta_t, t + \delta_t) - f(\vec{x}, t) = -\frac{1}{\tau}[f(\vec{x}, t) - g(\vec{x}, t)] \tag{3}$$

where $\tau = \frac{\lambda}{\delta_t}$ is the dimensionless relaxation time.

Discretization of velocity is needed to approximate the hydrodynamic moments and thereafter compute the equilibrium distribution. A lattice structure that satisfies conservation laws and guarantees the symmetry needed by the Navier–Stokes solutions is obtained. Different lattice structures can be used [12], such as the three dimensional Cartesian grid with 19 discrete velocities (referred to as D3Q19) used in this paper. D3Q19 has 6 particles moving in the principal directions with a constant speed $v = \delta_x/\delta_t$, 12 particles moving along the diagonals with a speed $\sqrt{2}v$ and a rest particle in the center of a cube. The equilibrium distribution function is approximated to f_i^{eq} by a truncated velocity expansion up to a desired accuracy (usually to the second order in u^2)

$$g \approx f_i^{\text{eq}} = \rho w_i \left(1 + \frac{3}{v^2} \vec{e}_i \cdot \vec{u} + \frac{9}{2v^4} (\vec{e}_i \cdot \vec{u})^2 - \frac{3}{2v^2} \vec{u} \cdot \vec{u} \right) \tag{4}$$

where $\vec{e}_i$ is the velocity of the particle along the direction i, with $i = 0, 1, 2, \ldots, z$, where z is the number of particle distributions of the model ($z = 19$ for D3Q19) and w_i a weight coefficient. The hydrodynamic density ρ and the macroscopic velocity $\vec{u}$ are determined in terms of the particle distribution functions from

$$\rho = \sum_i f_i = \sum_i f_i^{(\text{eq})} \tag{5}$$

and

$$\rho \vec{u} = \sum_i \vec{e}_i f_i = \sum_i \vec{e}_i f_i^{(\text{eq})} \tag{6}$$

Eq. (3) then becomes

$$f_i(\vec{x} + \vec{e}\delta_t, t + \delta_t) - f_i(\vec{x}, t) = -\frac{1}{\tau} [f_i(\vec{x}, t) - f_i^{(\text{eq})}(\vec{x}, t)] \tag{7}$$

which is usually known as the lattice Boltzmann BKG equation (LBGK).

It can be shown that the LBGK is equivalent to a second order discretization of the Navier–Stokes equations [12] if the viscosity is defined by

$$v = Cv^2 \delta_t \left(\tau - \frac{1}{2} \right) \tag{8}$$

and the pressure as

$$p = \rho c_s^2 \tag{9}$$

where C is a lattice-dependent coefficient, equal to 1/3 for D3Q19, and

$$c_s = v\sqrt{C} \tag{10}$$

is the "pseudo" speed of sound in the system, which is also lattice dependent. We can also prove that the α component of the symmetric stress tensor on a surface perpendicular to β-axis is given by [11,13]

$$\sigma_{\alpha\beta} = -\rho c_s^2 \delta_{\alpha\beta} - \left(1 - \frac{1}{2\tau}\right) \sum_{i=0}^{z} f_i^{(1)} e_{i\alpha} e_{i\beta} \tag{11}$$

where $f_i^{(1)}$ is the non-equilibrium part of the distribution function and $\delta_{\alpha\beta}$ is the unit tensor. Here, we emphasize that the quantity $f_i^{(1)} e_{i\alpha} e_{i\beta}$ is usually computed during the collision process. Therefore, the stress tensor components can be obtained independent from the velocity fields. This strongly enforces the lattice Boltzmann method when compared to other CFD methods which all estimate the stress tensor components from the simulated velocity field. It should be noted that the stress tensor components can be computed with the lattice Boltzmann equation with second order accuracy. The benefit over other CFD techniques in computing the stress tensor is clear when dealing with flow in complex geometry of irregular cross-sections or flows characterized by large velocity gradients. The lattice Boltzmann equation can model macroscopic fluid flows with second order accuracy in both space and time [e.g. 14,15]. However, this accuracy is commonly degraded by four major sources. These are the boundary conditions, the compressibility error, the discretization error and the momentum error. In this study, we investigate the accuracy and performance of the lattice Boltzmann method for unsteady flow simulations.

The aim of this study is to optimize the method for best accuracy within a minimum possible computational time. Different ways to refine the computational lattice are studied and their influences on both accuracy and convergence are investigated. The study concentrates on the influence of the Mach number and boundary conditions on the accuracy and performance. We will first present and discuss results of unsteady flow simulations in a rigid tube, investigate effects from different lattice refinement techniques on the accuracy and performance of the method and suggest ways to enhance the performance.

3. Simulations

We have conducted a number of different simulation studies for unsteady flow in a 3D tube. First, we simulate systolic flow in a tube and validate the results against the analytical Womersley solutions. We then compare the bounce-back rule (BBL) with the body-fitted Bouzidi boundary conditions with first (BBC1) and second (BBC2) order interpolations at different Mach numbers. Finally, different ways of reducing the errors by grid refinement techniques are presented and investigated under their accuracy and convergence behavior.

3.1. Systolic flow in a straight rigid tube

We compute time-dependent flow in a 3D tube. In all simulations, the pressure gradient in the tube is computed from a measured systolic pressure at the entrance of the human abdominal aorta, using Fourier transformations, up to the 8th harmonic. Simulation parameters are set to obtain a maximum Reynolds number $Re = \frac{UD}{\nu} = 590$ and a Womersley parameter $\alpha = R\sqrt{\frac{\omega}{\nu}} = 16$, where $R = D/2$ is the radius of the tube, $\omega = 2\pi/T$ is the angular frequency and $T = 1/f$ is the period, with f the frequency. Pressure boundary conditions are used for the inlet and the outlet [11] boundaries and, for the walls, either the bounce back on the links (BBL) or the Bouzidi [16] boundary conditions with first (BBC1) and second order (BBC2) interpolations are used.

Obtained velocity profiles over a complete period are compared to the analytical Womersley solutions [24]

$$u(y,t) = \sum_{m=1}^{8} \left[-\frac{A_m}{\rho m \omega} e^{-im\omega t} \left(1 - \frac{J_0\left[\sqrt{mb}y\right]}{J_0\left[\sqrt{mb}R\right]} \right) \right] \tag{12}$$

where J_0 is the zeroth order Bessel function of the first type, A_m is the amplitude of pressure gradient and $b = -i\omega/\nu = -im(\alpha/R)^2$ for the mth Fourier harmonic. The obtained shear stress is compared to the one derived from the above equation.

The computational mesh is a simple Cartesian lattice. The diameter of the tube is represented by D lattice nodes. The length of the tube is always taken to be $3 \times D$. We start by setting $D = 74$ lattice nodes. First, BBL is used to simulate systolic flow in the tube. The simulation parameters are set to yield the required Womersley and Reynolds numbers which are kept fixed. For this simulation, $T = 2000$, and $\tau = 0.55$, giving $\alpha = 16$ and $Re = 590$. Fig. 1(a) shows a density plot of obtained velocity profiles over the whole period. The corresponding shear stress values are contoured in Fig. 1(b). From this figure we observe that the velocity has high gradients very close to the walls (in the region $3R/4 < y < R$) and gets flat toward the center of the tube. The shear stress vanishes in the center and is high and oscillating near to the walls.

Samples of obtained velocity profiles at different times of the systolic cycle are shown in Fig. 2 compared to the real part of the analytical Womersley solutions. As clearly shown in this figure, the agreement with the analytical solution is quite good. We define the relative error in velocity at each time-step as

$$E_u = \frac{\sum_{i=0}^{n} |\mathbf{u}_{th}(\mathbf{x}_i) - \mathbf{u}_{lb}(\mathbf{x}_i)|}{\sum_{i=1}^{n} |\mathbf{u}_{th}(\mathbf{x}_i)|} \tag{13}$$

where $\mathbf{u}_{th}(\mathbf{x}_i)$ is the analytical solution for the axial velocity and $\mathbf{u}_{lb}(\mathbf{x}_i)$ is the velocity obtained from the LBE simulations. The overall relative error is averaged over the period and will be
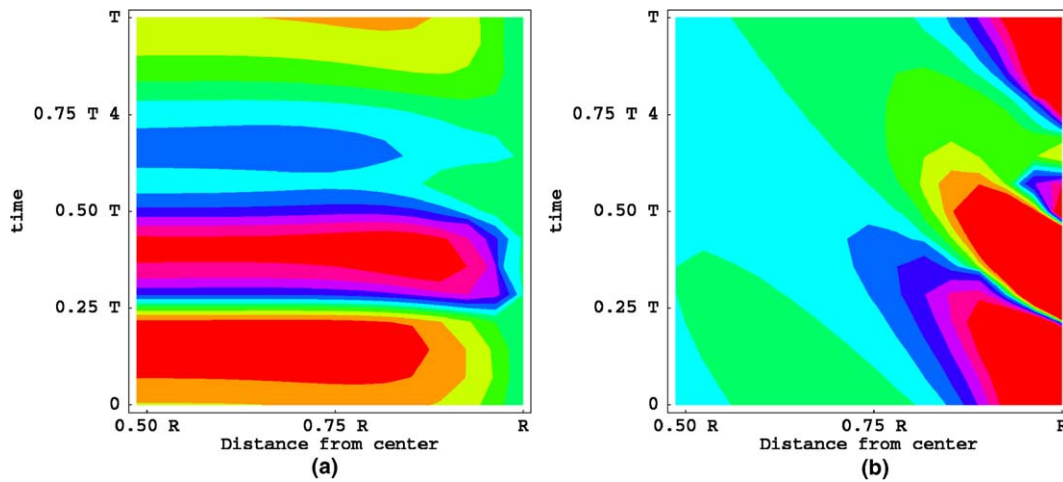

Fig. 1. Contour density plots of obtained (a) velocity and (b) shear stress during a full systolic cycle with $T = 2000$, Mach = 0.05, $Re = 590$ and $\alpha = 16$. Dark areas stands for high values of velocity and shear stress in lattice units.
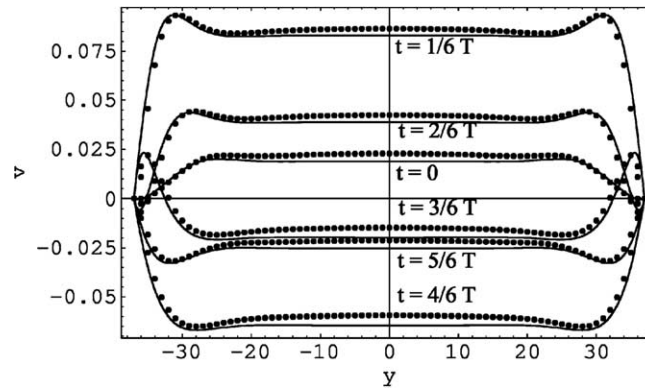
Fig. 2. Samples of obtained velocity profiles (dots) in lattice units during the systolic cycle in a 3D tube, compared to the analytical Womersley solutions (lines), using the bounce-back rule at Mach = 0.05, Re = 590 and α = 16. The time during systole for the curves are from 0 to $\frac{5}{6}T$ with steps of $\frac{1}{6}T$.

referred to as the average error. The bounce back on the links yields an average error of 0.12 at a Mach number of 0.05 for this specific simulation. This indicates that, even with the bounce-back rule, acceptable accuracy can be obtained for "engineering applications".

Using the same simulation parameters, we have conducted another set of simulations after replacing the bounce back on the links with Bouzidi boundary conditions. The agreement with analytical solutions enhances significantly, as shown in Fig. 3. The average error reduces to approximately 0.03. As shown in Fig. 4, similar agreement is achieved for the shear stress. It is to be noted that the Bouzidi boundary condition does not perform better than the bounce back at points with large velocity gradients, as it uses interpolation.

We also note that it is possible to go for higher Mach numbers with Bouzidi boundary conditions while still having better accuracy than those with BBL at a considerably lower Mach number. Even with a 10 times higher Mach number, the error associated with the Bouzidi boundary
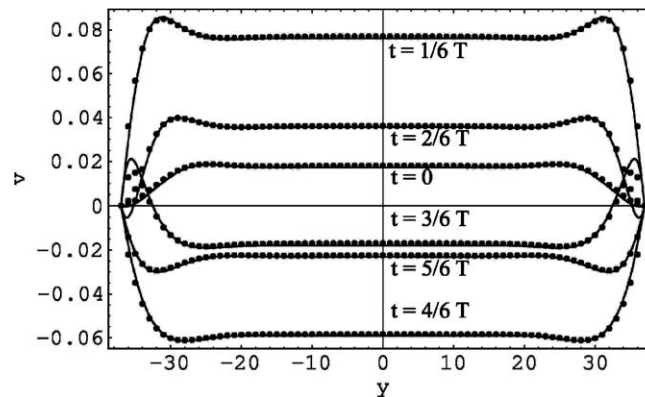

Fig. 3. Samples of obtained velocity profiles (dots) in lattice units during the systolic cycle in a 3D tube, compared to the analytical Womersley solutions (lines) using Bouzidi boundary conditions at Mach = 0.05, Re = 590 and α = 16. The time during systole for the curves are from 0 to $\frac{5}{6}T$ with steps of $\frac{1}{6}T$.
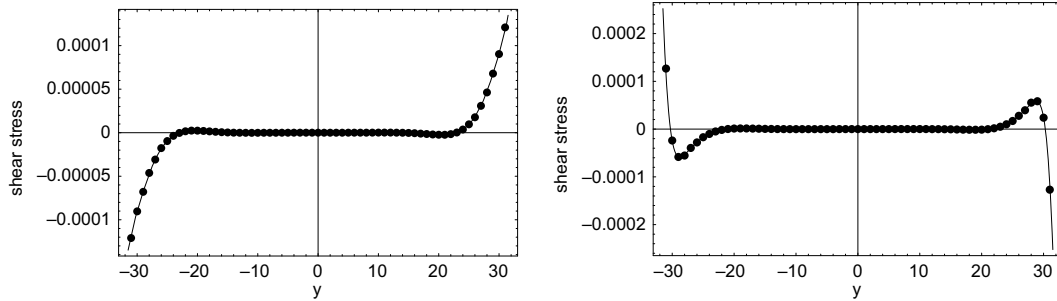
Fig. 4. Samples of obtained (dots) shear stress (in lattice units) near the end of diastole (left) and at mid systole (right), compared to the analytical Womersley solution (lines) for the 3D tube benchmark, using Bouzidi boundary conditions.

conditions is still better than that associated with BBL. Using a Bouzidi boundary condition therefore enhances the stability of the system and can reduce the simulation time since it allows higher Mach numbers than the simple bounce-back rule [21].

3.2. On search for faster convergence

In order to study the accuracy and convergence behavior for the same physical problem under fixed Reynolds and Womersley number, we need to tune our simulation parameters in a special way. The standard lattice Boltzmann equation has several free parameters, but for pulsating flow in rigid tubes, it is most convenient to tune the lattice viscosity v, the Mach number M, the diameter D of the tube and the period T of the pulse. Changes in any of these parameters will result in changes in the space and time resolutions of the grid. For LBE simulations the error behavior is influenced by the Mach number, $M = \frac{U}{c_s}$, and the Knudsen number $\epsilon \sim (2\tau - 1)/D$ [e.g. 17–19]. The effect of these numbers appears as a compressibility error, which is approximately [20]

$$\phi = \delta_x^2 Re^2 v^2 v^2 \partial_t \rho \tag{14}$$

which increases with increasing Reynolds and Mach numbers, since $Re = MDc_s/v$. As a function of Womersley number, the compressibility error may be written as

$$\phi = \frac{2\delta_x^2 c_s v^2}{\pi} \frac{\alpha^2 MD}{St} \partial_t \rho \tag{15}$$

where $St = Df/U$ is the Strouhal number. In this simulation, the dimensionless hydrodynamic numbers (Re, α and St) are all fixed. This implies that the grid must be refined, and/or the density gradients shall be small in order to reduce the compressibility error. Time-dependent flow with LBGK involves high density gradients, since the pressure and the density are tied-up together. Therefore, we are only left with the grid refinement option. There are three different ways to do this:

(1) *fixed M method:* the diameter D, the period T and the viscosity v are changed while keeping fixed the Mach number;
(2) *fixed τ method:* the diameter D, the period T and the Mach number M are changed while keeping fixed the lattice viscosity v, and

(3) *fixed D method*: the viscosity *v*, the period *T* and the Mach number *M* are all changed while keeping fixed the diameter *D*.

The effects of these changes on the grid resolution are shown in Table 1, in which we assume a factor of *n* change in one of the parameters and compute the corresponding changes in the other parameters in order to keep fixed Reynolds and Womersley numbers. From this table, we can predict the computational efficiency of each approach. For instance, the fixed *M* approach involves *n* times decrement in δ_x (which increases the number of grid points n^3 times) and *n* times reduction in δ_t which scales the total simulation time by n^4 per systolic cycle. The fixed τ method scales the simulation time per cycle as n^5 and the fixed *D* method scales it as *n*. Although it is easy to see that the last method is faster while the second one is more accurate, a combination of accuracy and performance is not trivial. The fixed *M* method does not involve reduction of the Mach number, which is a major contribution to accuracy enhancement when considering time-dependent flows and, therefore, it is not attractive in this study. Accordingly, we have performed 2 sets of simulations corresponding to the fixed τ and the fixed *D* methods. They are discussed below.

3.2.1. Accuracy and performance with the fixed D method

We have selected three sets of parameters to study the error behavior produced by this technique. First, we have performed a reference simulation at $M = 0.5$, $D = 74$, $\tau = 1$, $T = 200$, and we took the magnitude of the pressure gradient as $G = 0.0011$. This pressure gradient will result in the wanted Mach number. Then, two other parameter sets are selected with the aid of Table 1. The pressure gradient must be reduced by a factor n^2 to give the correct Mach number scaling. The errors associated with each boundary condition at three different Mach numbers are shown in Table 2, in which $n = 1$ represents the reference simulation. In all simulations, the system is initialized from rest and the simulation ends after 40 complete systolic cycles. The simulations were performed on 8 nodes of a Beowulf cluster using slice decomposition. The mean time per iteration is about 0.45 s using the bounce back and 0.47 using the Bouzidi boundary conditions on 8 processors. The total execution time for 40 complete cycles is therefore $18T$ for the bounce back and $18.8T$ for the Bouzidi boundary conditions. For low Mach numbers, this difference in execution time scales as hours (4.5 h for $n = 100$).

From this table we notice that for BBC1 with $M = 0.5$, the agreement with the analytical solutions is still better than that for a 10 times smaller Mach number with BBL.

It is also apparent that Bouzidi boundary conditions are more stable than the bounce-back rule at high Mach numbers, but the situation reverses for lower Mach numbers.

Table 1
Relative changes in simulation parameters under fixed Reynolds and Womersley numbers with respect to a factor of *n* change in one of the parameters of a reference simulation

Lattice parameter	D'/D	v'/v	T'/T	U'/U	δ_x'/δ_x	δ_t'/δ_t	M'/M
Fixed *D*	1	$1/n$	n	$1/n$	1	$1/n$	$1/n$
Fixed τ	n	1	n^2	$1/n$	$1/n$	$1/n^2$	$1/n$
Fixed *M*	n	n	n	1	$1/n$	$1/n$	1

The prime refers to the new simulation.

Table 2
Average error for unsteady flow in a tube with $D = 74$ using BBL, BBC1 and BBC2 boundary conditions at three different Mach numbers

	n		
	1	10	100
Eav, BBL	Not stable	0.120	0.027
Eav, BBC1	0.0627	0.0352	0.0253
Eav, BBC2	0.0615	0.0102	Not stable

In addition, we have noticed that BBC1, which interpolates data up to the first fluid node is more stable than the second order interpolation scheme (BBC2) which interpolates data using two neighboring fluid nodes. This may be attributed to effects from interpolation in a region of large velocity gradients in the case of BBC2.

From this set of simulations, we conclude that it would be faster, more stable and more accurate to use a Bouzidi boundary condition than the simple bounce back.

3.2.2. Accuracy and performance with the fixed τ method

Given the n^5 scaling of computing time per systole (see Table 1), the fixed τ method is most preferably studied by first trying to reduce the computing time as much as possible, and then observing the associated errors. These errors can then be compared to those in the fixed D method, after which one can decide which scaling method to use. In order to reduce the simulation time, it is necessary to have a large time-step in a coarse grid at a high Mach number. To attain that, we use the fixed τ method to perform a set of simulations in which the period is set to minimum possible value that leads to a stable solution on the coarsest grid. Then the corresponding values for the pressure gradient and the relaxation parameter are set to yield the desired Womersley and Reynolds numbers. The convergence behavior is studied by grid refinement in both δ_x and δ_t, as explained in Table 1. The obtained average errors associated with the three used boundary conditions are plotted in Fig. 5. We notice that the accuracy of the bounce-back rule approaches that of the Bouzidi boundary conditions as the grid is refined. As this method results in reducing δ_x, δ_t and the pressure gradient, both accuracy and performance are significantly enhanced, since all parameters influencing the error are under control. Moreover, with this method, solutions with periods smaller than the fixed D method are stable and therefore the simulation time is less, but it scales as n^5.

The convergence behavior as a function of time for this method is shown in Fig. 6, which shows the absolute error (difference between the analytical and obtained velocity profiles) at different simulation times. From this figure, we observe that the method converges to a reasonable accuracy after 40 complete periods, similar to the fixed D method, but with a major computational gain, since δ_t is larger. This figure also illustrates that the error localizes near to the walls, where large gradients exist, and the accuracy does not enhance noticeably near to the walls on the same grid. The accuracy close to the wall may be enhanced by considering local grid refinement or running the simulation at lower Mach numbers. Table 3 lists the error dependence as a function of simulation time for BBL, BBC1 and BBC2 boundary conditions for a tube with $D = 65$ lattice nodes. The error is reasonably comparable to that obtained by using the fixed D method. The
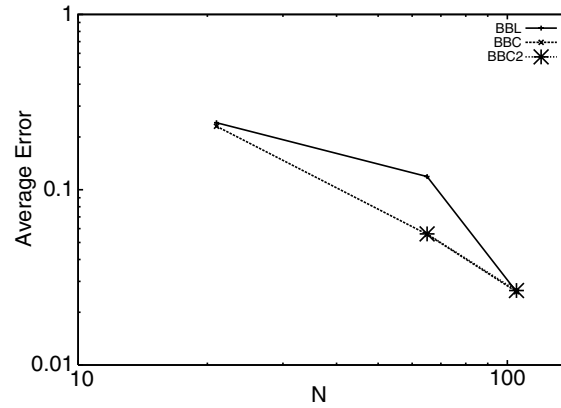
Fig. 5. Convergence behavior obtained by reducing the grid spacing n times, time-step n^2 times and increasing the period n^2 times, for the BBL, BBC and BBC2 boundary conditions as a function of grid points representing the diameter of the tube. The relaxation parameter is kept constant and the body force is reduced n^3 times to return the same Reynolds and Womersley parameters at $Re = 590$ and $\alpha = 16$.
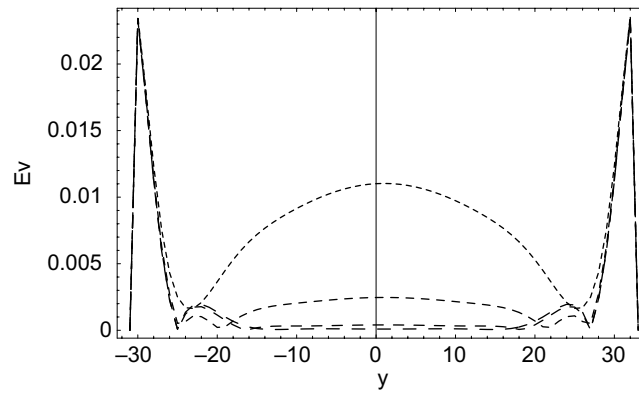

Fig. 6. Absolute error (Ev), δE, computed for the velocity field at $t = 20T$ (top curve), $30T$, $40T$ and $50T$ (bottom curve). The diameter of the tube is represented by 65 nodes and the period is $T = 360$ sampling points. The average errors are tabulated in Table 3. The bounce back on the links shows similar behavior with larger spread (see text).

Table 3
Temporal relative error, $E_u(T)$ for BBL and BBC1 and BBC2 boundary conditions with $D = 65$ lattice nodes, $T = 360$, $Re = 590$ and $\alpha = 16$

	Time					
	0	10T	20T	30T	40T	50T
Eu(T), BBL	0.9950	0.2520	0.0698	0.0279	0.0200	0.0200
Eu(T), BBC1	0.9950	0.2769	0.0615	0.0280	0.0200	0.0197
Eu(T), BBC2	0.9950	0.2747	0.2520	0.0866	0.0350	0.0560

mean deviation of the error is 0.02 for the bounce back on the links and two times larger for the Bouzidi boundary conditions.

In conclusion, this method is computationally more feasible than the fixed D method. It is recommended to use it while keeping D and T to their minimum values that returns stable solutions.

3.3. Performance analysis

Convergence of the lattice Boltzmann method to steady state is significantly affected by two local processes: initialization and boundary conditions. In this section, we focus on the influence of these processes on the convergence behavior.

3.3.1. Performance and walls boundary conditions

As mentioned before, boundary conditions need to be defined at walls, inlets and outlets. For the walls, two categories of boundary conditions can be recognized; bounce backs and Bouzidi boundaries. The bounce-back rule is a very efficient boundary condition since it only involves a single memory swapping process for each relevant distribution on each node on the surface of the simulated object. For Bouzidi boundaries, the exact position of the walls is determined at least once if the boundary is fixed and need to be computed dynamically for moving boundaries. This is by itself more costly than using the bounce-back rule. In addition, using Bouzidi boundary conditions involves first or second order interpolation or extrapolation for velocity, distribution functions or density or a combination of some or all of them. As demonstrated earlier in this article, use of a Bouzidi boundary condition enhances the accuracy but is computationally more intensive compared to the simple bounce back at the same Mach number. To gain the accuracy of a Bouzidi boundary condition and a performance similar to the bounce back, an accelerating technique may be applied. The Mach number annealing technique has been recently introduced by the authors [21] to accelerate unsteady flow simulations. This annealing process is based on using the bounce-back rule to start the simulation at relatively high Mach number and wait until the system converges, then reduce the Mach number to a lower value M' to obtain a desired accuracy. The system directly adapts to the new parameters defining the new Mach number. We have performed a number of simulations using this annealing technique. The gain in computational time can be determined as follows. We assume that after simulating 40 full systoles, the simulation has converged (see e.g. Fig. 6 and Table 3). We have also observed this in simulations of real arteries (data not shown). We therefore perform our simulations by first running 40 systoles, and then simulate one final systole in which we measure the wanted observables (velocity fields, shear stress tensor). To obtain the same Reynolds and Womersley numbers, the period and the kinematic viscosity must be modified. Calling T_0 the period on the high Mach number and T_n the period on small, annealed Mach number we write $T_n = mT_0$ and $v_n = v_0/m$. The gain in computational time obtained by Mach number annealing then is

$$\text{Gain} = \frac{41m}{40 + m} \tag{16}$$

where $m = M/M'$ is the annealing factor. For details, we refer to Ref. [21].

3.3.2. Inlet and outlet conditions

For LBE simulations, a number of inlet and outlet conditions are available. The most commonly used are periodic boundary conditions, in which distributions leaving the simulation box

at the outlet are re-injected at the inlets and vice-versa. Periodic boundary conditions involve only memory swapping operations. Although they are fast, and accurate, they can only be used for periodic geometry. For non-periodic geometry, inlets and outlets need to be treated differently in the following manner:

- Velocity and pressure: assign one and compute the other [22], assign both (only for inlets), extrapolate or assume no flux normal to the walls (only for outlets).
- Unknown distributions: compute explicitly [22], set to their equilibrium, copy from nearest neighbors, interpolate or extrapolate.

For the first item, if the velocity or the pressure are computed one from the other, at least 15 additions and two multiplications are needed per node on the boundary and therefore is at least 15 times more expensive than periodic boundary conditions. Extrapolation and no-flux schemes are far better in terms of accuracy and performance than computing velocity or pressure from one another, but they are only suitable for the outlets. A reasonable choice for time-dependent flow in irregular geometry is then to assign pressure and compute velocity at the inlet, no-flux at the outlets and set the unknown distributions to their equilibrium values. If the outlets are far enough from inflow, copying from upstream would be the most efficient outlet condition.

3.3.3. Initial conditions

Time-dependent flow involves large density fluctuations. Although it increases compressibility errors, this reduces the initialization influence on convergence behavior. However, the way in which the simulation box is initialized has little effect on the final flow fields [23]. Since the Boltzmann equation assumes that the system is not far from equilibrium, a correct and reasonable initialization technique is to compute the equilibrium distributions from expected macroscopic parameters and set each distribution to its equilibrium with a small perturbation, We have adopted this initialization process in all our simulations. It is noted that more sophisticated initialization techniques such as second order interpolations from the boundaries may be useful but they complicate the standard LBE scheme.

4. Conclusions

An aortic pressure is used as an inlet condition to drive the flow in a 3D rigid tube and the Womersley solution is recovered to an acceptable accuracy. Different ways to change the spatial resolution in the simulations where studied, and the resulting errors and convergence behavior were discussed. The influence of walls, inlet and outlet boundary conditions on accuracy and performance is studied in detail. We have shown that the lattice Boltzmann BGK method is an accurate and efficient method as a solver for time-dependent flows. Different methods for performing time-dependent flows at fixed simulation parameters are tested in terms of accuracy and performance. We have demonstrated three different grid-refinement scenarios for error analysis of unsteady flow simulations: fixed spatial resolution (the fixed D method), fixed lattice viscosity (the

fixed τ method) and fixed Mach number (the fixed M method). Our study allows users of the LBGK method to find optimal simulation parameters for their specific (time periodic) flow problem.

We have shown that it would be more accurate and relatively stable to use Bouzidi boundary conditions at high Mach numbers, or still use the bounce back on the links at lower Mach numbers. It is to be noted that the Bouzidi boundary conditions become very expensive when used for moving boundaries, since the wall–fluid distance has to be computed at each time step. However, the cost of the bounce-back rule for moving boundaries is the same for that for static ones. We claim that the bounce back on the links could be more efficient than the Bouzidi boundary conditions if used at low Mach numbers when the Mach number annealing technique is used. Since Bouzid boundary conditions are complex and computationally intensive for unstructured and dynamic geometry, the bounce back on the link with Mach number annealing is a promising technique for studying such kind of applications.

Acknowledgement

References

[1] Caflisch RE. Fluid dynamics and the Boltzmann equation. In: Lebowitz JL, Montroll EW, editors. Non-equilibrium phenomena I: The Boltzmann equation. North-Holland Publishing Company; 1983. p. 195–226.
[2] Ramaswamy S. Adv Phys 2001;50:297.
[3] Cercingnani C. J Quant Spec Rad Trans 1971;197:973.
[4] Frisch U, Hasslacher B, Pomeau Y. Phys Rev Lett 1986;56:1505.
[5] Higuera FJ, Succi S, Benzi R. Europhys Lett 1989;9:345.
[6] Rothmann DH, Zaleski S. Lattice-gas cellular automata, simple models of complex hydrodynamics. Cambridge: Cambridge University Press; 1997.
[7] Rivet JP, Boon JP. Lattice gas hydrodynamics. Cambridge, UK: Cambridge University Press; 2001.
[8] McNamara GR, Zanetti G. Phys Rev Lett 1988;61:2332.
[9] Chen S, Doolen GD. Ann Rev Fluid Mech 1998;30:329.
[10] Succi S. The lattice Boltzmann equation for fluid dynamics and beyond. Oxford, UK: Oxford University Press; 2001.
[11] Artoli AM, Hoekstra AG, Sloot PMA. Int J Mod Phys C 2002;13:1119.
[12] He X, Luo LS. Phys Rev E 1997;56:6811.
[13] Hou S, Zou Q, Chen S, Doolen G, Cogley AC. J Comput Phys 1995;118:329.
[14] Filippova O, Hänel D. Int J Mod Phys C 1998;9:1271.
[15] Mei RW, Luo LS, Shyy W. J Comput Phys 1999;155:307.
[16] Bouzidi M, Firdaouss M, Lallemand P. Phys Fluids 2001;13:3452.
[17] Zou Q, Hou S, Chen S, Doolen GD. J Stat Phys 1995;81:35.
[18] He XY, Luo LS. J Stat Phys 1997;88:927.
[19] Guo ZL, Shi BC, Wang NC. J Comput Phys 2000;165:288.

[20] Holdych DJ, Noble DR, Georgiadis JG, Buckius RO. J Comput Phys 2004;193:595.
[21] Artoli AM, Hoekstra AG, Sloot PMA. Int J Mod Phys C 2003;14:835.
[22] Zou Q, He X. Phys Fluids 1997;9:1591.
[23] Skordos PA. Phys Rev E 1993;48:4823.
[24] Womersley JR. J Physiol 1955;127:553.