INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **58**:1353–1378
Published online 14 April 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1801

Assessment of algorithms for the no-slip boundary condition in the lattice Boltzmann equation of BGK model

Yong Kweon Suh*, †, Jinfen Kang and Sangmo Kang

Department of Mechanical Engineering, Dong-A University, 840 Hadan-dong, Saha-gu, Busan 604-714, Republic of Korea

SUMMARY

Three kinds of algorithms for the lattice Boltzmann equation of the BGK model in the implementation of the no-slip boundary condition on the wall are assessed by using the analytical formula for the slip velocity of the fully developed pressure-driven channel flow. It is shown that the bounce-back algorithm results in the spatial accuracy of 1st order, except for the case when the wall is located at half way between the two grid lines. The interpolation scheme proposed by Yu *et al.* (*Prog. Aerospace Sci.* 2003; **39**:329–367) and the similar one by Bouzidi *et al.* (*Phys. Fluids* 2001; **13**(11):3452–3459) are of 2nd order, but the error increases quadratically with the relaxation time. The extrapolation scheme of Guo *et al.* (*Phys. Fluids* 2002; **14**(6):2007–2010) is also shown to be of 2nd order, and the error level increases linearly with the relaxation time, but it turns out that this scheme is unstable for a certain range of parameter values. Numerical experiments with various parameter sets have been performed to obtain the stability diagram. Three algorithms are then applied to a circular-Couette flow and their performance is also studied in terms of the numerical accuracy and stability. Copyright © 2008 John Wiley & Sons, Ltd.

Received 16 August 2007; Revised 18 February 2008

KEY WORDS: lattice Boltzmann equation; BGK model; no-slip boundary condition; interpolation algorithm; extrapolation algorithm; channel flow; circular-Couette flow

1. INTRODUCTION

Lattice Boltzmann method (LBM) has been employed in the numerical simulation of various flow problems, and nowadays it is highly promising as an alternative to the existing numerical

*Correspondence to: Yong Kweon Suh, Department of Mechanical Engineering, Dong-A University, 840 Hadan-dong, Saha-gu, Busan 604-714, Republic of Korea.
†E-mail: yksuh@dau.ac.kr

Contract/grant sponsor: Ministry of Science and Technology; contract/grant number: 2005-01091

Copyright © 2008 John Wiley & Sons, Ltd.

methods based on the Navier–Stokes (N–S) equations, such as finite volume method (FVM), finite element method (FEM), or its kinds [1]. Most important advantages of this method are as follows: first, we do not have to solve the pressure equation; second, the program code is very simple; and third, it is suitable for flow problems involving complex geometry and multiphase flows. Recent applications of LBM to multiphase flows include deformation, merging and splitting of droplets [2–4], formation, motion, growth and detachment of bubbles [5–8], phase separation and transition [9, 10], and filling in casting process [11]. Very recently, applications are increasingly found in the field of microfluidics, starting from the formulation of the governing equations [12, 13] to the problem of ion transport [14], electro-osmotic flows [15, 16], and mixing [17].

Among several problems with LBM, which are still open to further improvement, the present study is devoted to the problem with the implementation of the no-slip boundary condition. The fundamental difficulties in the boundary-condition implementation with LBM arise from the fact that, in the governing equation of LBM, the dependent variables are the distribution functions, not the velocity components. Therefore, we must specify the boundary values of the distribution functions, before the so-called 'streaming' process, for the links pointing to the fluid region at the nodes inside the wall. Unfortunately in LBM, however, the velocity components are calculated by the moment of the distribution functions, and the number of the functions that contribute to each of the velocity components is 6 for the two-dimensional case. Therefore, determining the distribution-function values explicitly from the velocity constraint at the boundary must be a non-trivial task.

Starting from the simplest scheme, i.e. the bounce-back (B-B) scheme algorithm, several schemes have been so far proposed as the method of no-slip-condition implementation. Inamuro *et al.* [18] proposed to evaluate the unknown distribution functions in the same form as the equilibrium functions with the density and velocity components determined in such a way that the no-slip condition is satisfied on the boundary wall. However, this scheme is applicable only when the wall matches with the lattice links. Noble *et al.* [19] proposed to use not only the velocity equations but also the energy equation to determine the three distribution functions, in two-dimension, on the links heading to the fluid; but their method is again applicable only to the case where lattice nodes are located on the wall. A very simple extrapolation scheme was suggested by Chen *et al.* [20], but its stability was not addressed. After that, various forms of interpolation schemes have been proposed to improve the numerical accuracy and to resolve the stability problem [21–26]. However, it seems that we have much room for improvement in the method of implementation of the no-slip boundary conditions with respect to the numerical accuracy and stability of LBM. Development of accurate and stable algorithms for the no-slip boundary condition should be a prerequisite step toward establishing the feasibility and enlarging the range of LBM applications.

In this study, we analyzed three existing algorithms used as the no-slip boundary condition for the LBM employing the BGK model by applying them to the fully developed pressure-driven channel flow; they are the B-B scheme, the method of Yu *et al.* [26] and the method of Guo *et al.* [27]. It turned out that the numerical error is in most cases caused by the slip velocity on the wall arising from the specific algorithms. To obtain the slip velocity, we employed the distribution-function relations, based on the work of Zou *et al.* [28] and He *et al.* [29], derived under the assumption of fully developed state. The algorithms were further assessed for the case of curved boundaries, i.e. a circular-Couette flow.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

2. LATTICE BOLTZMANN EQUATIONS AND NUMERICAL METHODS

To solve the two-dimensional channel flow both numerically and analytically, we employed the incompressible model of the LBM with nine links, i.e. ID2Q9 [30]. Figure 1 shows square lattices and the sequence of the link numbers used in the present ID2Q9 model. The evolution equation to be solved for the distribution function is

$$f_k(\mathbf{x}+\mathbf{e}_k\Delta t, t+\Delta t) = f_k(\mathbf{x}, t) + \frac{1}{\tau}[f_k^{(\mathrm{eq})}(\mathbf{x}, t) - f_k(\mathbf{x}, t)] \tag{1}$$

where $f_k(\mathbf{x}, t)$ denotes the distribution function of the particle at a discrete space (lattice node) $\mathbf{x}$ at a discrete time t, which is supposed to travel along the link number k during the time interval Δt. τ is the relaxation time. The particle has the velocity $\mathbf{e}_k$, which is defined in such a way that the particle arrives at the neighboring node situated at the other end of the link number k after one time step Δt. As usual we take $\Delta x = \Delta t = 1$, as we take the physical grid size Δx^* and the physical time step Δt^* as the reference quantities in the scaling; further explanation for the scaling of the variables will be given below. The equilibrium function $f_k^{(\mathrm{eq})}(\mathbf{x}, t)$ on the right-hand side of (1) is defined as

$$f_k^{(\mathrm{eq})} = w_k\left[\rho + \rho_0\left(\frac{\mathbf{e}_k\cdot\mathbf{u}}{c_\mathrm{s}^2} + \frac{(\mathbf{e}_k\cdot\mathbf{u})^2}{2c_\mathrm{s}^4} - \frac{\mathbf{u}\cdot\mathbf{u}}{2c_\mathrm{s}^2}\right)\right] \tag{2}$$

where w_k is the weight factor, i.e. $w_0 = \frac{4}{9}$, $w_k = \frac{1}{9}$ for $k = 1$–4 and $w_k = \frac{1}{36}$ for $k = 5$–9, and $c_\mathrm{s} = \frac{1}{\sqrt{3}}$ is the speed of sound. The density ρ and the velocity $\mathbf{u}$ are computed using $\rho = \sum_{k=0}^{8} f_k$ and $\rho_0\mathbf{u} = \sum_{k=1}^{8} \mathbf{e}_k f_k$, where ρ_0 is the reference density; we can also take $\rho_0 = 1$ without loss of generality. The density ρ is a property of the flow. In other words, in the incompressible model, ρ_0 corresponds to the 'real' fluid density, which is not the flow property, and ρ should be considered as a kind of pressure; $p = \rho/3$.

Note that all the variables treated in this paper are based on the lattice unit; the spatial coordinates are scaled by the grid spacing Δx^*, the time by the time step Δt^*, the distribution function by the density ρ_0^*, the velocity by $c^* = \Delta x^*/\Delta t^*$, and the pressure by $\rho_0^* c^{*2}$. The asterisks used here indicate the dimensional variables.

To confirm the analytical formula of the slip velocity, to be given in the following section, we must compare the analytical results with those obtained numerically for the lattice-Boltzmann equation (LBE). In what follows we explain the numerical methods applied to solve (1). Here we confine ourselves to the channel flow; special features for the curved boundary will be addressed in Section 5 upon necessity. We first construct the grid system within the channel. In the streamwise direction the total number of grids is L, and the channel length in the lattice unit is the same as this; this means that the inlet of the channel is situated at half way between the first two stations ($i = 0$ and 1) and the outlet at half way between the last two stations ($i = L$ and $L+1$). We build N grids across the channel section, i.e. between the top and the bottom solid walls. The channel height is then $N - 1 + 2\Delta$ in the lattice unit, where Δ (to be referred to as 'link ratio' hereafter) denotes the distance from each of the walls to the nearest grid line toward the fluid side as shown in Figure 2. We choose the channel's aspect ratio close to 4; $L/N \approx 4$. The Reynolds number is

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

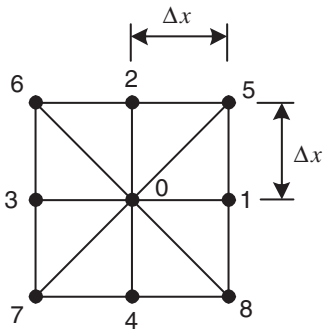

Figure 1. Square lattices for the ID2Q9 model and link numbers.

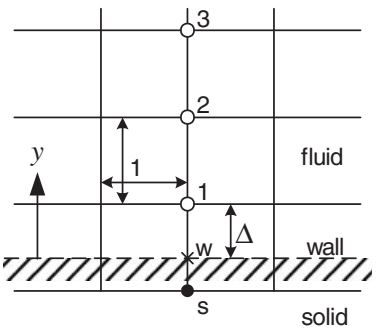

Figure 2. Grid lines and the j-indices of nodes near the bottom wall of the channel.

simply defined as $Re = u_{\max} N/v$, where v is the dimensionless viscosity related to the relaxation time by $v = (2\tau - 1)/6$, and $u_{\max}$ is the velocity at the centerline of the channel.

One cycle of the iterative numerical procedure in solving the LBE (1) is basically composed of three steps. In the first 'collision' step, the time is fixed and the distribution function $\widetilde{f}_k(\mathbf{x}, t)$ is calculated using

$$\widetilde{f}_k(\mathbf{x}, t) = f_k(\mathbf{x}, t) + \frac{1}{\tau}[f_k^{(\mathrm{eq})}(\mathbf{x}, t) - f_k(\mathbf{x}, t)] \tag{3}$$

where the tilde symbol indicates the function just after the collision, i.e. before the streaming. In the second step, the boundary conditions are applied to determine the values of $\widetilde{f}_k(\mathbf{x}, t)$ for the links heading toward the fluid side at the nodes outside of the fluid domain, such as the node 's' in Figure 2. The cycle is closed by the third 'streaming' step, in which each of $\widetilde{f}_k(\mathbf{x}, t)$ is transmitted, with the unit time elapse, to the neighboring node dictated by the link number k:

$$f_k(\mathbf{x} + \mathbf{e}_k, t + 1) = \widetilde{f}_k(\mathbf{x}, t) \tag{4}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

As the inlet boundary conditions we need to specify three distribution functions for the links heading toward the fluid side at the station $i=0$. We simply applied the B-B principle, which can be expressed in terms of the discrete variables as follows:

$$\widetilde{f}^{0}_{1,j}=\widetilde{f}^{1}_{3,j}+6w_1u_{\mathrm{in}}(y_j)$$

$$\widetilde{f}^{0}_{5,j-1}=\widetilde{f}^{1}_{7,j}+6w_5u_{\mathrm{in}}(y_j)$$

$$\widetilde{f}^{0}_{8,j+1}=\widetilde{f}^{1}_{6,j}+6w_8u_{\mathrm{in}}(y_j)$$

where the superscript indicates the station number, i.e. i, and $u_{\mathrm{in}}(y)$ represents the parabolic velocity profile specified at the inlet as

$$u_{\mathrm{in}}(y)=4u_{\max}y(y_{\mathrm{c}}-y)/(N-1+2\Delta)^2 \tag{5}$$

The normal coordinate y has its origin at the bottom wall and its discrete value is given by $y_j=j-1+\Delta$. y_{c} is the value of y at the channel center; $y_{\mathrm{c}}=(N-1+2\Delta)/2$. The maximum velocity $u_{\max}$ is fixed at $u_{\max}=0.1$ unless otherwise stated in the present numerics. Note that the velocity profile (5) also corresponds to the exact solution to the fully developed channel flow. The pressure gradient (or the density gradient in terms of the LBM variables) is automatically determined through the computation.

As the algorithm for the outlet boundary condition, we developed a new method to implement the constant average density and the fully developed flow properties at the outlet. The algorithm can be simply expressed as

$$\widetilde{f}^{L+1}_{k,j}=\widetilde{f}^{L}_{k,j}+w_k(1-\overline{\rho}^{L}) \quad (k=0,1,\ldots,8) \tag{6}$$

where $\overline{\rho}^{L}$ is the average of ρ at the station $i=L$. By employing this simple algorithm we can effectively establish the following properties:

$$u^{L+1}_{j}=u^{L}_{j}, \quad \overline{\rho}^{L+1}=1 \tag{7}$$

where u^{i}_{j} indicates the streamwise velocity component at the discrete point $(i,\ j)$. The target value for the outlet station's average density $\overline{\rho}^{L+1}$ is taken to be the same as ρ_0. The net result is that the upstream density is adjusted automatically in such way that $-\partial\rho/\partial x>0$ provides the given flow rate specified at the inlet.

Implementation of the boundary conditions on the solid walls, which is the main issue of this study, is addressed in the following sections because the specific algorithms directly affect the slip velocity. The numerical value of the slip velocity on the solid wall is computed by using the 2nd-order extrapolation scheme at the central station $i=(L+1)/2$:

$$u_w=\tfrac{1}{2}(1+\Delta)(2+\Delta)u_1-\Delta(2+\Delta)u_2+\tfrac{1}{2}\Delta(1+2\Delta)u_3$$

The normalized slip velocity is then given by $U_w=u_w/u_{\max}$.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

The numerical error is quantified by the root mean square (rms) of the difference between the normalized velocity, obtained from the numerical computation after the steady state has been reached, and that from the analytic solution at the central station

$$\varepsilon = \sqrt{\frac{1}{N}\sum_{j=1}^{N}[u_{\mathrm{in}}(y_j)/u_{\max} - u_j/u_{\max,n}]^2} \tag{8}$$

where $u_{\max,n}$ is the maximum velocity of the numerical solution u_j occurring at the centerline of the channel. If the error is caused purely by shifting of the u-profile as much as the slip velocity u_w, that is, if the numerical solution takes the form $u_j = u_w + u_{\mathrm{in}}(y_j)$, it can be shown that

$$\varepsilon \cong |U_w|/\sqrt{5} \tag{9}$$

The numerical computation reaches a steady state mostly within 10 000 time steps under various parameter sets, and therefore we conducted 30 000 time-step calculations to obtain the steady state while carefully monitoring the results.

3. ANALYTICAL SOLUTION FOR THE SLIP VELOCITY

In this section we derive the formula for the slip velocity expected to occur on the solid walls of the pressure-driven flow within a two-dimensional channel, caused by the implementation of the 'imperfect' algorithms for the no-slip boundary conditions on the walls. Our formulation is based on the work of Zou *et al.* [28] and He *et al.* [29]. In this study we generalize the method to the case when the walls are located at an arbitrary position between two neighboring grids parallel to the walls.

We assume that at the steady fully developed state the distribution function can be decomposed into two parts:

$$f_k = w_k \rho(x) + g_k(y) \tag{10}$$

In other words, we assume that the pressure is independent of y and the streamwise velocity is independent of x. From the above decomposition, we can derive the following formula for the discrete value $g_{k,j}$:

$$g_{1,j} = w_1(3u_j + 3u_j^2) + w_1\tau\delta\rho \tag{11a}$$

$$g_{2,j} = w_2(-1.5u_{j-1}^2)/\tau + (\tau-1)\, g_{2,j-1}/\tau \tag{11b}$$

$$g_{3,j} = w_3(-3u_j + 3u_j^2) - w_3\tau\delta\rho \tag{11c}$$

$$g_{4,j} = w_4(-1.5u_{j+1}^2)/\tau + (\tau-1)g_{4,j+1}/\tau \tag{11d}$$

$$g_{5,j} = w_5(3u_{j-1} + 3u_{j-1}^2)/\tau + (\tau-1)g_{5,j-1}/\tau + w_5\delta\rho \tag{11e}$$

$$g_{6,j} = w_6(-3u_{j-1} + 3u_{j-1}^2)/\tau + (\tau-1)g_{6,j-1}/\tau - w_6\delta\rho \tag{11f}$$

$$g_{7,j} = w_7(-3u_{j+1} + 3u_{j+1}^2)/\tau + (\tau-1)g_{7,j+1}/\tau - w_7\delta\rho \tag{11g}$$

$$g_{8,j} = w_8(3u_{j+1} + 3u_{j+1}^2)/\tau + (\tau-1)g_{8,j+1}/\tau + w_8\delta\rho \tag{11h}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

where $\delta\rho$ denotes the density gradient in the lattice unit, i.e. $\delta\rho=-\mathrm{d}\rho/\mathrm{d}x$; at the fully developed state $\delta\rho$ is constant. The discrete streamwise velocity can be expressed in terms of $g_{k,j}$ as

$$u_j=(g_{1,j}-g_{3,j})+(g_{5,j}-g_{6,j})+(g_{8,j}-g_{7,j}) \tag{12}$$

The slip-velocity formula presented in this paper was obtained by coupling several equations relating the nodal-point velocities near the solid walls. These relations can be grouped into two: one group includes the 'general' formula, which is not affected by the specific algorithm for the no-slip boundary condition, and the other group includes the 'specific' formula, which reflects the detailed algorithm for the no-slip condition. We first derive the 'general' formula in the beginning of this section. The specific formula will be derived in the following subsections. We restrict our attention to the nodes near the bottom wall without loss of generality. First we can show, by employing (11a)–(11h) and (12) in the expression of u_1, u_2, and u_3, the following relation:

$$u_3-2u_2+u_1=\frac{-2\delta\rho}{2\tau-1} \tag{13}$$

In more detail, in the expression of u_2, we just replace the first term on the right-hand side of (12) with $j=2$ by

$$g_{1,2}-g_{3,2}=2u_2/3+2\tau\delta\rho/9 \tag{14}$$

which can be derived from (11a) and (11c) with $j=2$. For u_1, we follow the same way for the first term in (12) with $j=1$, but we use (11e) and (11f) with $j=2$ to treat the second term and $j=1$ to treat the third term. For u_3, we use (11e) and (11f) with $j=3$ for the second term and $j=2$ for the third term. Eliminating the term $(g_{5,2}-g_{6,2})+(g_{8,2}-g_{7,2})$ from these three representations, we obtain (13). Equation (13) is nothing but the discretization, at $j=2$, of the balance equation between the viscous term and the pressure-gradient term in the N–S equations; $\partial^2u/\partial y^2=-2\delta\rho/(2\tau-1)=$constant. In fact, this relation holds at any interior nodal point except the nodes nearest to the walls, i.e. $j=1$ and N. Knowing this constant, we can construct the parabolic velocity profile in terms of the three velocity values u_1, u_2, and u_3 as

$$u=-\frac{\delta\rho}{2\tau-1}(y-1-\Delta)^2+\frac{u_3-u_1}{2}(y-1-\Delta)+u_2 \tag{15}$$

We can assume that, due to the symmetric property of the solution, the slip velocity at the lower wall $y=0$ is the same as that at the top wall $y=N-1+2\Delta$. Applying this assumption to (15) yields the following relation:

$$u_3-u_1=\frac{2(N-3)\,\delta\rho}{2\tau-1} \tag{16}$$

Substitute (16) into (15) to eliminate u_3 and evaluate the result at the wall $y=0$ to obtain a formula of the slip velocity:

$$u_w=-\frac{\Delta(1+\Delta)\delta\rho}{2\tau-1}+(1+\Delta)u_1-\Delta u_2 \tag{17}$$

On the other hand, in terms of the slip velocity we can also express the velocity profile in another form as follows:

$$u=u_w-\frac{\delta\rho}{2\tau-1}y(y-N+1-2\Delta) \tag{18}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

Evaluating this at y_c produces the maximum velocity $u_{\max}$:

$$u_{\max} = u_w + \frac{(N-1+2\Delta)^2\,\delta\rho}{4(2\tau-1)} \tag{19}$$

The normalized slip velocity is defined as

$$U_w = u_w / u_{\max} \tag{20}$$

We need more equations to derive the explicit formula for computing the normalized slip velocity in terms of N, τ, and Δ. The detailed formula as well as the number of these equations required is dependent on the specific algorithm employed for the no-slip boundary conditions on the walls, as mentioned previously.

3.1. B-B boundary condition

The B-B algorithm just returns back the distribution function carried by the particle sent from a neighboring fluid node toward a solid node beyond the wall [30]. Implementation of this algorithm in the code for the LBM is very simple. In addition, we can expect that the mass as well as the momentum is conserved. But, this does not necessarily mean that the no-slip condition is also satisfied on the wall. The slip-velocity formula for the B-B algorithm has been presented by He *et al.* [29], but their expression was found to be erroneous as will be shown in the last part of this subsection.

The B-B boundary condition can be formulated as follows:

$$f_{2,1} = \widetilde{f}_{4,1} = f_{4,1} + \frac{1}{\tau}(f_{4,1}^{(\mathrm{eq})} - f_{4,1}) \tag{21a}$$

$$f_{5,1} = \widetilde{f}_{7,1} = f_{7,1} + \frac{1}{\tau}(f_{7,1}^{(\mathrm{eq})} - f_{7,1}) \tag{21b}$$

$$f_{6,1} = \widetilde{f}_{8,1} = f_{8,1} + \frac{1}{\tau}(f_{8,1}^{(\mathrm{eq})} - f_{8,1}) \tag{21c}$$

We wish to derive the relationship between u_1 and u_2 based on this algorithm. First, we start from the definition of u_1, i.e. from (12) with $j=1$. The first term on the right-hand side can be evaluated as $2u_1/3+2\tau\,\delta\rho/9$. To manipulate the second term, we subtract (21c) from (21b) to obtain

$$g_{5,1} - g_{6,1} = -\frac{1}{6\tau}u_1 + \frac{1-\tau}{\tau}(g_{8,1} - g_{7,1}) \tag{22}$$

Then the equation for u_1 becomes

$$3(2\tau+1)u_1 = 4\tau^2\delta\rho + 18(g_{8,1} - g_{7,1}) \tag{23}$$

On the other hand, similar treatment can be applied to the formulation of u_2 starting from (12) with $j=2$. It leads to the following equation:

$$3\tau^2(2\tau-1)u_2 - 3(\tau-1)u_1 = \tau^2(4\tau^2-4\tau-1)\delta\rho + 18(3\tau^2-3\tau+1)(f_{8,1} - f_{7,1}) \tag{24}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

In the derivation of this formula, (11e) and (11f) with $j=2$ were used to replace the second term on the right-hand side of (12), and (11g) and (11h) with $j=1$ for the third term. Then (22) was used to eliminate $(g_{5,1}-g_{6,1})$. Now we can eliminate $(g_{8,1}-g_{7,1})$ from (23) and (24) to arrive at

$$-u_2+3u_1=\frac{(8\tau^2-8\tau+5)\delta\rho}{3(2\tau-1)} \tag{25}$$

Now, from the five relations (13), (16), (17), (19), and (20) applicable to any type of boundary conditions, and one relation (25) specific for the B-B algorithm, we can derive the following formula for the normalized slip velocity expected to be obtained when the B-B boundary condition is applied:

$$U_w=\frac{K}{3(N-1+2\Delta)^2+K} \tag{26}$$

where K is composed of two terms:

$$K=K_0+K_1N \tag{27}$$

and

$$K_0=2[8\tau^2-8\tau-(6\Delta^2-6\Delta+1)] \tag{28a}$$

$$K_1=6(1-2\Delta) \tag{28b}$$

We note from this result that the spatial accuracy of the B-B algorithm is in general of 1st order because K_1 is non-zero, except for the special case $\Delta=0.5$ at which the algorithm is of 2nd order. This is consistent with the well-known fact that the B-B boundary condition is 2nd order if and only if the boundary is just at half way between the two grid lines. We can also see that there exists a critical τ at which the slip velocity vanishes:

$$\tau_c=\tfrac{1}{4}\{2+\sqrt{6[(2\Delta^2-2\Delta+1)+(2\Delta-1)N]}\} \tag{29}$$

For instance, we obtain $\tau_c=0.933$ independent of N with $\Delta=0.5$, and $\tau_c=2.087$ with $\Delta=0.6$ at $N=31$. We also note that the slip velocity increases with $O(\tau^2)$ for large τ. Figure 3 shows the slip velocity as well as the scaled error ε' given by $\varepsilon'=\sqrt{5}\varepsilon$. Note that the analytical solution is in excellent agreement with the numerical solution. The fact that the scaled error ε' is also in exact agreement with $|U_w|$ (refer to Equation (9)) means that the numerical error from the LBE is caused by the shifting of the velocity profile as much as the slip velocity, as clarified previously. The value of τ_c is in excellent agreement with $\tau_c=2.087$ predicted by (29). Figure 4 shows the plots of τ_c for various values of Δ, as a function of the number of grids N, given by (29). It is seen that τ_c increases as Δ is increased. It also increases with N for $\Delta>0.5$. However, τ_c exists only when $\Delta\geqslant 0.5+1/(4N)$, approximately, which can be derived from (29).

Figure 5 shows the dependence of the magnitude of the slip velocity on the grid resolution. Again, the analytical solutions predict the numerical solutions very accurately except for high relaxation time and low resolution regimes. At $\Delta=0.5$, the slip velocity is at the lowest level and it reveals the 2nd-order behavior. At $\Delta=0$; however, the slip velocity is at a higher level and the accuracy is close to the 1st order. The accuracy level for $\Delta=1$ at high τ is almost the same as that for $\Delta=0$, as can be understood from formulas (26)–(28b).

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

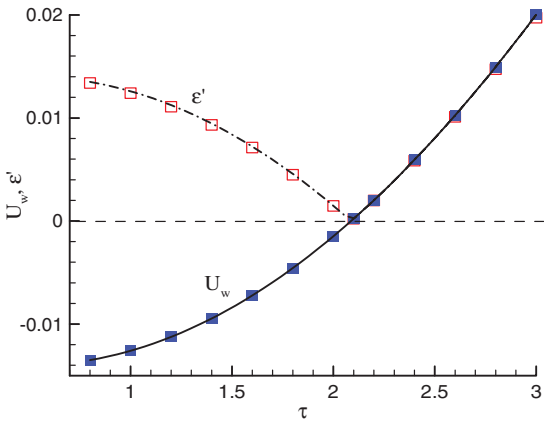

Figure 3. The normalized slip velocity U_w (solid line and filled symbols) and the scaled rms error (dash-dot line and open symbols) given by the analytical formula (lines) and by the numerical simulations (symbols) with the bounce-back algorithm at $N=31$ and $\Delta=0.6$ for the channel flow. The dash-dot line in the region $\tau>2.2$ falls on the solid line.

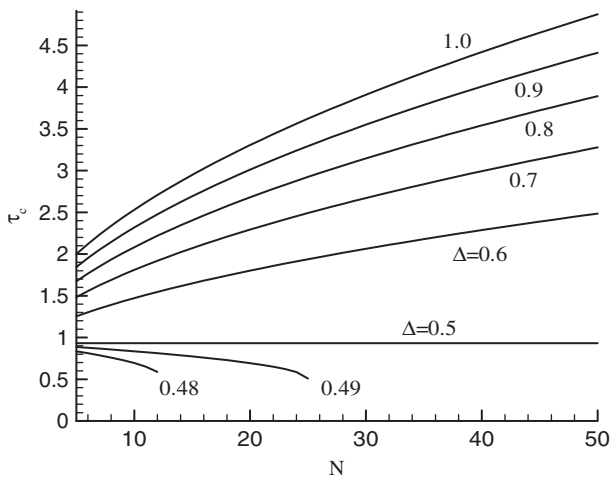

Figure 4. The critical relaxation time (29) as a function of N with various values of Δ in the bounce-back algorithm of the no-slip boundary condition for the channel flow.

It should be, however, pointed out that the formula derived in this study is different from that derived by He *et al.* [29]. In fact, they considered only the case of $\Delta=1$. As an example, for the parameter set $\Delta=1$, $N=31$, and $\tau=5$, the numerical solution to the full LBE produces $U_w=0.0411$. At the same parameter set, formula (26) gives a very close result $U_w=0.0412$, whereas the formula presented by He *et al.* yields a slightly different value $U_w=0.0371$. We conjecture that there must be unknown cause of error in their formulation.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

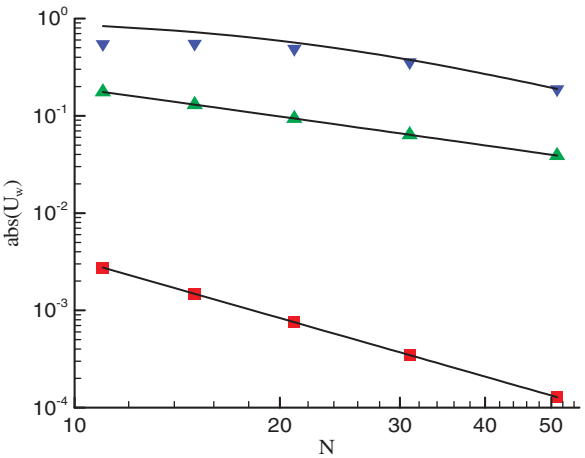

Figure 5. Dependence of the slip-velocity magnitude on the grid resolution given by analytical (solid lines) and numerical (symbols) results for the bounce-back algorithm for the channel flow: squares, $\Delta=0.5$, $\tau=1$; deltas, $\Delta=0$, $\tau=1$; gradients, $\Delta=0$, $\tau=10$.

3.2. Interpolation schemes

The fundamental idea of the various kinds of interpolation schemes is the same as in the B-B algorithm; it returns back the distribution function carried by the particle sent from the neighboring fluid node. In the interpolation scheme, however, the distribution-function value that is to be transmitted back to the fluid node is dependent on the wall position, Δ. Several methods have been proposed regarding the effective calculation of this value. Among them, we select the algorithm of Yu *et al.* [26] because in their algorithm a single formula is used regardless of the value of Δ. In the last paragraph of this subsection, the result for the algorithm of Bouzidi *et al.* [24] is presented.

The interpolation scheme proposed by Yu *et al.* [26] can be expressed as

$$f_{2,1}^{i}=\frac{1}{1+\Delta}[(1-\Delta)f_{4,1}^{i}+\Delta\tilde{f}_{4,1}^{i}]+\frac{\Delta}{1+\Delta}f_{2,2}^{i} \tag{30a}$$

$$f_{5,1}^{i}=\frac{1}{1+\Delta}[(1-\Delta)f_{7,1}^{i}+\Delta\tilde{f}_{7,1}^{i}]+\frac{\Delta}{1+\Delta}f_{5,2}^{i+1} \tag{30b}$$

$$f_{6,1}^{i}=\frac{1}{1+\Delta}[(1-\Delta)f_{8,1}^{i}+\Delta\tilde{f}_{8,1}^{i}]+\frac{\Delta}{1+\Delta}f_{6,2}^{i-1} \tag{30c}$$

We replace tilde variables by non-tilde variables by using (3) and subtract (30c) from (30b) to obtain

$$g_{5,1}-g_{6,1}=\frac{\Delta-\tau}{\Delta+\tau}(g_{7,1}-g_{8,1}) \tag{31}$$

where the identities $f_{5,1}^{i}-f_{6,1}^{i}=g_{5,1}^{i}-g_{6,1}^{i}$ and $f_{7,1}^{i}-f_{8,1}^{i}=g_{7,1}^{i}-g_{8,1}^{i}$ have been used, and we dropped the superscript i considering that they are independent of i. Further, in the derivation of this formula, the term $(f_{7,1}^{(\mathrm{eq})i}-f_{8,1}^{(\mathrm{eq})i})$ was replaced by $-u_1/6$ as can be inferred from (2), and

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

the term $(f_{5,2}^{(\mathrm{eq})i+1}-f_{6,2}^{(\mathrm{eq})i-1})$ was formulated as $(f_{5,2}^{(\mathrm{eq})i+1}-f_{6,2}^{(\mathrm{eq})i-1})=u_1/(6\tau)+(\tau-1)(g_{5,1}-g_{6,1})/\tau$, which can be derived from (10), (11e), and (11f) with $j=2$. On the other hand, in the representation of u_1, i.e. (12) with $j=1$, we replace the first term on the right-hand side by $2u_1/3+2\tau\delta\rho/9$ as before. Now, we substitute (31) into the second term in (12), and the result is $u_1=6\Delta(g_{8,1}-g_{7,1})/(\tau+\Delta)+2\tau\delta\rho/3$. We also follow a similar manner in representing u_2 starting from (12) with $j=2$. Now we eliminate $(g_{8,1}-g_{7,1})$ from these two equations to obtain

$$\Delta(2\tau-1)u_2+[-2(1+\Delta)\tau+\Delta+1]u_1=-\tfrac{1}{3}(4\tau^2-2\tau+3\Delta)\delta\rho \tag{32}$$

Again, we have five general relations (13), (16), (17), (19), and (20) and one specific relation (32) to determine the slip velocity for the algorithm of Yu *et al.* The result can be expressed in the form of (26), but K_0 and K_1 take the following forms:

$$K_0=4(4\tau^2-2\tau-3\Delta^2) \tag{33a}$$

$$K_1=0 \tag{33b}$$

From this result, we can see that this algorithm is of 2nd order regardless of the parameter values chosen. In this case too, there exists a critical τ at which slip velocity vanishes:

$$\tau_c=\tfrac{1}{4}[1+\sqrt{1+12\Delta^2}] \tag{34}$$

For instance, we obtain $\tau_c=0.5$ at $\Delta=0$, $\tau_c=0.75$ at $\Delta=\frac{1}{2}$ and $\tau_c=1.15$ at $\Delta=1$. Therefore, the zero-slip velocity cannot be attained when we choose $\tau>1.15$. On the other hand, we can also define the critical distance in terms of τ as $\Delta_c=\sqrt{2\tau(2\tau-1)/3}$. For instance, we obtain $\Delta_c=0$ at $\tau=0.5$ and $\Delta_c=\sqrt{2/3}$ at $\tau=1$, etc. Figure 6 presents the analytic and numerical results of the slip velocity and the scaled error, as functions of τ, for the algorithm of Yu *et al.* [26]. The agreement between the two results is almost perfect, and the critical relaxation time also matches well with each other. This algorithm's 2nd-order nature is also confirmed by numerical data together with the analytical prediction as shown in Figure 7.

The interpolation scheme suggested by Bouzidi *et al.* [24] is similar to that of Yu *et al.* [26]. The algorithm takes two different forms depending on Δ as follows:

$$f_{\bar{k},1}^{i}=\begin{cases}2\Delta\widetilde{f}_{k,1}^{i}+(1-2\Delta)\widetilde{f}_{k,2}^{i2} & \text{for } \Delta\leqslant\frac{1}{2}\\ \dfrac{1}{2\Delta}\widetilde{f}_{k,1}^{i}+\dfrac{2\Delta-1}{2\Delta}\widetilde{f}_{\bar{k},1}^{i} & \text{for } \Delta>\frac{1}{2}\end{cases} \tag{35}$$

where each of the station numbers $i2=i, i+1, i-1$ and each of the link numbers $k=4, 7, 8$ are matched with the link numbers $\bar{k}=2, 5, 6$, respectively. Following the same procedure as above, we can derive the formula in the same form as (26), where now

$$K_0=\begin{cases}4[4\tau^2-2(1+2\Delta)\tau+\Delta(2-3\Delta)] & \text{for } \Delta\leqslant\frac{1}{2}\\ 4[4\tau^2-2(3-2\Delta)\tau+2-2\Delta-3\Delta^2] & \text{for } \Delta>\frac{1}{2}\end{cases} \tag{36a}$$

$$K_1=0 \tag{36b}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

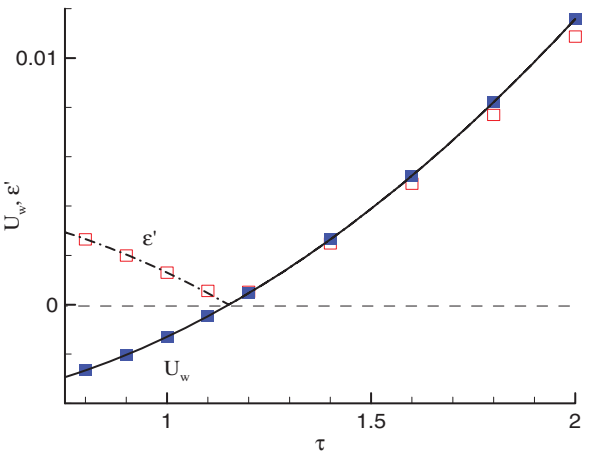

Figure 6. The normalized slip velocity U_w (solid line and filled symbols) and the scaled rms error (dash-dot line and open symbols) given by the analytical formula (lines) and by the numerical simulations (symbols) for the algorithm of Yu *et al.* [26] at $N=31$ and $\Delta=1$ in the channel flow. The dash-dot line in the region $\tau>1.3$ falls on the solid line.

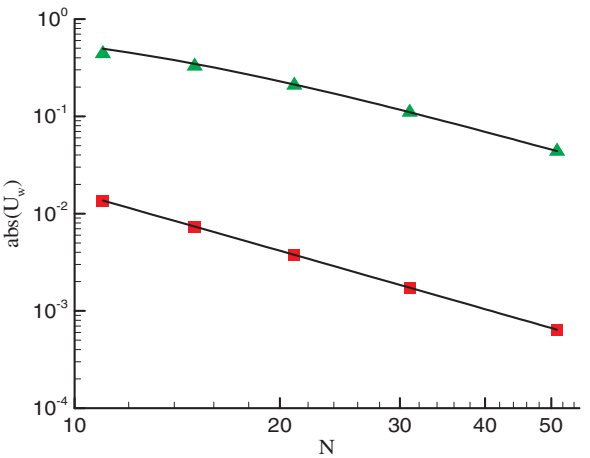

Figure 7. Dependence of the slip-velocity magnitude on the grid resolution given by the analytical (solid lines) and numerical (symbols) results for the algorithm of Yu *et al.* [26] in the channel flow: squares, $\Delta=0.5$, $\tau=1$; deltas, $\Delta=0.5$, $\tau=5$.

Therefore, this algorithm is also of 2nd order. The critical relaxation time is given by

$$\tau_c = \begin{cases} [1+2\Delta+\sqrt{1-4\Delta+16\Delta^2}]/4 & \text{for } \Delta \leqslant \frac{1}{2} \\ [3-2\Delta+\sqrt{1-4\Delta+16\Delta^2}]/4 & \text{for } \Delta > \frac{1}{2} \end{cases} . \tag{37}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

Formula (36a) was verified in this study from the numerical simulation of the LBE. It can be shown that K_0 values given by the algorithm of Bouzidi *et al.*, (36a), evaluated at $\Delta=0$ and 1 are the same as those given by the algorithm of Yu *et al.*, (33a), at the same Δ values. A slight difference exists in K_0 between the two for the other values of Δ. For instance, at $\tau=3$ and $\Delta=\frac{1}{2}$, we obtain $K_0=117$ from (33a) and $K_0=97$ from (36a). As a whole, the two algorithms can be thought of as the same kind, but the algorithm of Yu *et al.* is preferred because it is composed of a single formula.

Although both the interpolation schemes investigated above are of 2nd-order accuracy, the slip velocity increases quadratically with τ similar to that in the B-B scheme, which is the primary disadvantage of these schemes.

3.3. Extrapolation scheme of Guo et al.

Contrary to the other schemes, the one proposed by Guo *et al.* [27] performs the collision process at the solid nodes; the solid node is denoted as '*s*' in Figure 2. The distribution function at the solid node after the collision step is obtained by

$$\tilde{f}_{k,s}=f_{k,s}^{(\mathrm{eq})}+\tilde{f}_{k,s}^{(\mathrm{ne})} \tag{38}$$

The first term on the right-hand side is given from (2) with the nodal velocity components, for the case of the fully developed channel flow, given by the extrapolation scheme

$$u_s=\gamma_1 u_1+\gamma_2 u_2 \tag{39}$$

Their suggestions for the constants γ_1 and γ_2 are

$$\gamma_1=\begin{cases}\Delta-1 & \text{for } \Delta<0.75\\ (\Delta-1)/\Delta & \text{for } \Delta\geqslant 0.75\end{cases} \tag{40a}$$

$$\gamma_2=\begin{cases}-(1-\Delta)^2/(1+\Delta) & \text{for } \Delta<0.75\\ 0 & \text{for } \Delta\geqslant 0.75\end{cases} \tag{40b}$$

The non-equilibrium part $\tilde{f}_{k,s}^{(\mathrm{ne})}$ in (38) is given from the interpolation formula

$$\tilde{f}_{k,s}^{(\mathrm{ne})is}=\beta\tilde{f}_{k,1}^{(\mathrm{ne})i}+(1-\beta)\tilde{f}_{k,2}^{(\mathrm{ne})i2} \tag{41}$$

where the constant β is proposed by them to be determined by

$$\beta=\begin{cases}\Delta & \text{for } \Delta<0.75\\ 1 & \text{for } \Delta\geqslant 0.75\end{cases} \tag{42}$$

The station numbers in (41) are $is=i2=i$ for $k=2$; $is=i-1$ and $i2=i+1$ for $k=5$; and $is=i+1$ and $i2=i-1$ for $k=6$. The non-equilibrium function $\tilde{f}_{k,j}^{(\mathrm{ne})}$ on the right-hand side of (41) is

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

obtained from

$$\tilde{f}_k^{(\mathrm{ne})} = \frac{1-\tau}{\tau}(f_k^{(\mathrm{eq})} - f_k) \tag{43}$$

Then we can express from (41) as follows:

$$f_{5,1} = w_5(\rho + 3u_s + 3u_s^2) + \tilde{f}_{5,s}^{(\mathrm{ne})} \tag{44a}$$

$$f_{6,1} = w_6(\rho - 3u_s + 3u_s^2) + \tilde{f}_{6,s}^{(\mathrm{ne})} \tag{44b}$$

where the tilde variables on the left-hand side are replaced by appropriate non-tilde variables considering the subsequent streaming process. Subtract (44b) from (44a) and apply (41) to replace the non-equilibrium functions at the solid nodes by those at $j=1$ and 2. Then we use (43) to eliminate the non-equilibrium terms at the fluid nodes to derive the equation relating $(g_{5,1}-g_{6,1})$, $(g_{5,2}-g_{6,2})$, u_s, u_1, and u_2. On the other hand, by subtracting (11f) from (11e) with $j=2$ we obtain another equation relating $(g_{5,1}-g_{6,1})$, $(g_{5,2}-g_{6,2})$, u_1, and $\delta\rho$. From these two equations, we can eliminate the term $(g_{5,2}-g_{6,2})$ to obtain the formula relating $(g_{5,1}-g_{6,1})$, u_s, u_1, u_2, and $\delta\rho$. We need two more equations to eliminate $(g_{5,1}-g_{6,1})$ from this. These two equations are given from (12) with $j=1$ and 2, respectively. From the above three equations, we can eliminate $(g_{5,1}-g_{6,1})$ and $(g_{8,1}-g_{7,1})$ to obtain the following simple relation:

$$u_2 - 2u_1 + u_s = \frac{[(6\beta-10)\tau + (5-6\beta)]\delta\rho}{3(2\tau-1)} \tag{45}$$

As before, we use (13), (16), (17), (19), (20), (45), and (39) to obtain the formula of the normalized slip velocity. The result can also be expressed in the form of (26) with the two coefficients expressed as

$$K_0 = \frac{4}{1-\gamma_1-\gamma_2}[(10-6\beta)\tau - 11 + 6\beta - 6\gamma_2 + 3\Delta(1-\Delta)(1-\gamma_1-\gamma_2)] \tag{46a}$$

$$K_1 = \frac{12}{1-\gamma_1-\gamma_2}[1+\gamma_2 - \Delta(1-\gamma_1-\gamma_2)] \tag{46b}$$

Note that the above formula can be applied for an arbitrary choice of β, γ_1, and γ_2. However, when γ_1 and γ_2 satisfy the relation

$$\gamma_2 = \frac{\Delta}{1+\Delta}(2-\gamma_1) - 1 \tag{47}$$

we have $K_1 = 0$ and the algorithm becomes of the 2nd order. In this case formula (46a) reduces to

$$K_0 = 4(1+\Delta)\left[\frac{(10-6\beta)\tau + 6\beta - 5}{2-\gamma_1} - 3\Delta\right] \tag{48}$$

The values of γ_1 and γ_2 determined by formulas (40a) and (40b) as suggested by Guo *et al.* [27] indeed satisfy the criterion (47), and therefore their parameter selection gives the 2nd-order accuracy. However, we should also note that their suggestion corresponds to only one special case, and we can consider infinite sets of γ_1 and γ_2 that lead to the 2nd-order accuracy. Further discussion regarding this is given below. Under the condition of (47), we can derive the formula for the

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

critical relaxation time at which the slip velocity becomes zero:

$$\tau_c = \frac{5-6\beta+3\Delta(2-\gamma_1)}{10-6\beta} \tag{49}$$

Compared with other methods addressed in this study, the method of Guo *et al.* [27] has clear advantages as to the accuracy. The slip velocity is proportional to τ as shown in (48), and therefore we can expect a lower error at higher τ values compared with other schemes. Further, two free parameters, i.e. β and γ_1, can be used to make the error as small as possible; this variant will be referred to as the modified algorithm of Guo *et al.* However, proper choice of these parameters must be related to the numerical instability inherent in the extrapolation schemes.

Figure 8 shows the dependence of the normalized slip velocity and the scaled rms error on the relaxation time for two parameter sets. The analytic data of U_w again predict the numerical data very accurately. The scaled error is also predicted very accurately by the analytic formula, except for a slight difference at high relaxation time for the case with $\beta=1$ and some difference at low relaxation time. The figure also shows a clear picture of the linear dependence of the slip velocity on the relaxation time. As was predicted from the present theory, the method of Guo *et al.* is of the 2nd order as can be seen from Figure 9. Although the algorithm of Guo *et al.* has three free independent parameters, β, γ_1, and γ_2, they proposed to determine these parameter values uniquely by using formulas (40a), (40b), and (42) for a given Δ value. Thus, following their suggestion, we can plot the slip-velocity contour in the parameter space (Δ, τ) as shown in Figure 10. The linear increase in the slip velocity with the relaxation time is again confirmed from this figure. The slip velocity is on the whole at a lower level than the other two algorithms analyzed in this paper, especially at high values of τ. The region $\Delta \geqslant 0.75$, however, shows the magnitude almost twice the one in the other region for the same τ. The reason for using such discontinuous

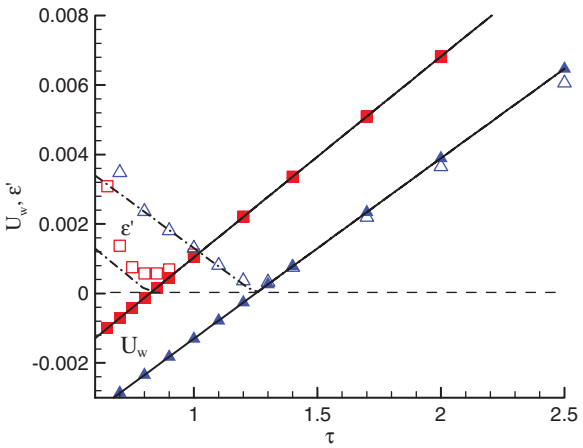

Figure 8. The normalized slip velocity U_w (solid lines and filled symbols) and the scaled rms error (dash-dot lines and open symbols) given by the analytical formula (lines) and by the numerical simulations (symbols) for the algorithm of Guo *et al.* [27] at $N=31$ in the channel flow: squares, $\beta=0.5$, $\Delta=0.5$, $\gamma_1=-0.5$, $\gamma_2=-0.167$; deltas, $\beta=1$, $\Delta=1$, $\gamma_1=0$, $\gamma_2=0$.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

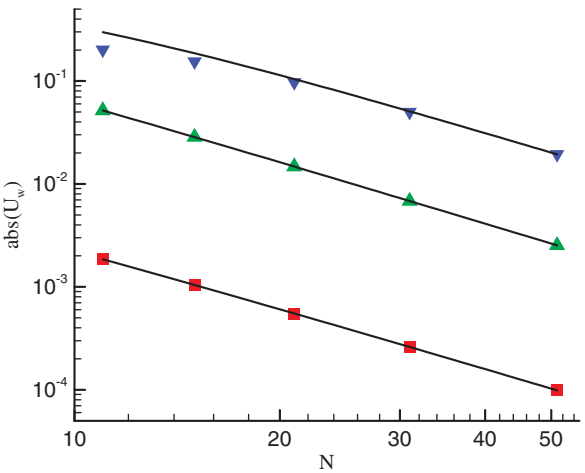

Figure 9. Dependence of the slip-velocity magnitude on the grid resolution given by the analytical (solid lines) and numerical (symbols) results for the algorithm of Guo *et al.* [27] in the channel flow: squares, $\beta=1$, $\Delta=1$, $\gamma_1=\gamma_2=0$, $\tau=1.2$; deltas, $\beta=0.5$, $\Delta=0.5$, $\gamma_1=\gamma_2=0$, $\tau=2$; gradients, $\beta=0.5$, $\Delta=0.5$, $\gamma_1=\gamma_2=0$, $\tau=10$.

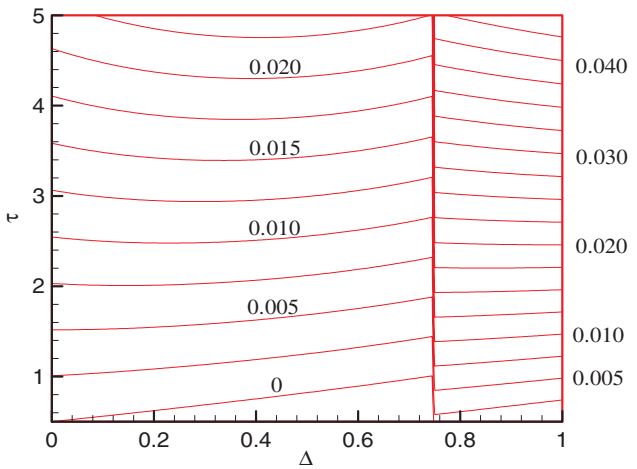

Figure 10. Contour plot of the normalized slip velocity U_w given by the analytic solution for the original algorithm suggested by Guo *et al.* [27] in the channel flow.

parameters was to avoid the numerical instability occurring when the formulas of β, γ_1, and γ_2 in (42), (40a), and (40b), which are applicable for the range $\Delta \geqslant 0.75$, were applied to the case with Δ close to 0.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

4. FURTHER DISCUSSIONS ON EXTRAPOLATION SCHEMES

While performing numerical computations for a variety of parameter values we have found that the algorithm suggested by Guo *et al.* [27] is still unstable for a certain range of parameter values when we follow formulas (40a), (40b), and (42). To study the numerical instability, we make the parameters free from their parameter-selection formula. The relevant parameters associated with the numerical instability are then Δ, τ, β, γ_1, and γ_2. In this study, we assume that γ_2 is determined with (47) in order to maintain the 2nd-order accuracy. Then we are left with four independent parameters. First, we fix $\beta=1$ and test the numerical stability in the parameter space (Δ, γ_1) for various τ values, and the results are shown in Figure 11(a). On the whole, the scheme is the most stable near $\Delta=0.5$. It also shows that the range of γ_1 for the stability is shrunk as τ is increased. To check the dependence of the stability on β, we now fix $\Delta=0$ and obtain the stability range of γ_1 as shown in Figure 11(b). It is seen that the range is almost invariant of β for β less

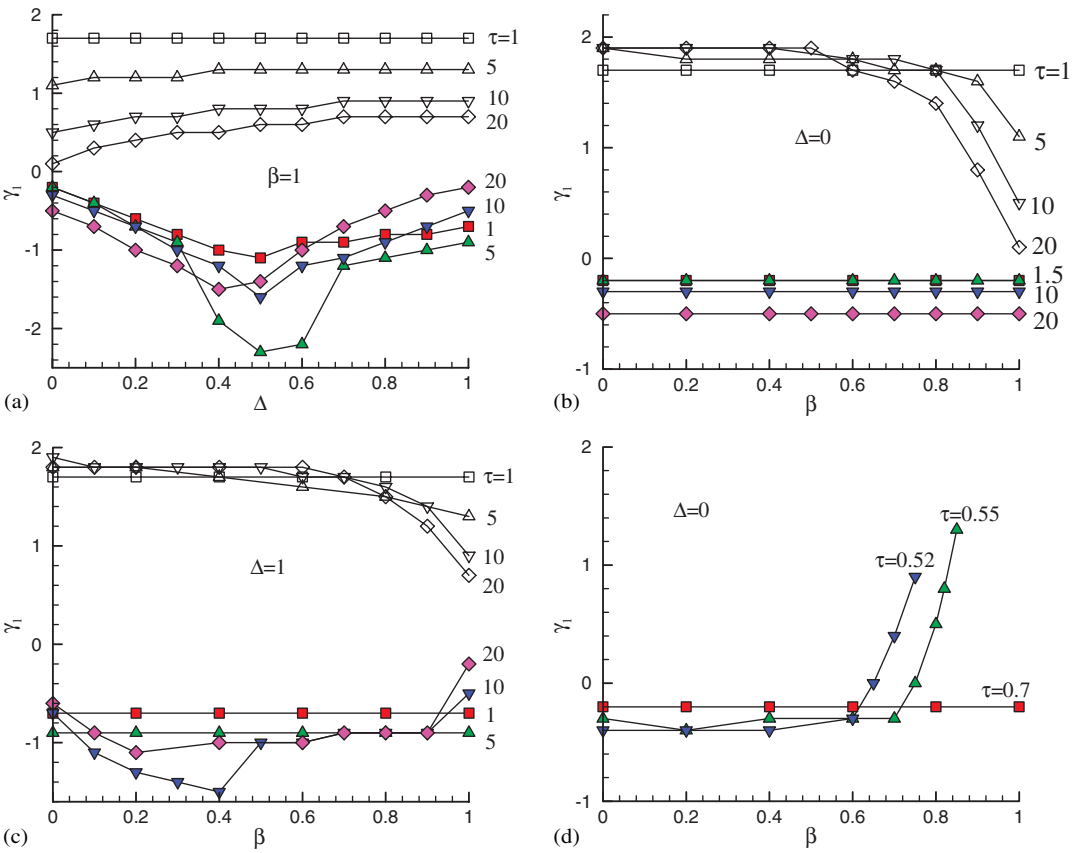

Figure 11. Stability diagrams of the modified algorithm of Guo *et al.* [27] obtained from the numerical test for the channel flow. The open and closed symbols denote the upper and lower limits of γ_1 for the stable computation, respectively. In (a), β is fixed at $\beta=1$ and Δ is changed, whereas in (b), (c), and (d), Δ is fixed and β is changed.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

than 0.6. In the region $\beta>0.6$, the upper limit of γ_1 for the stability decreases rapidly as β is increased. Similar trend was observed for the case of $\Delta=1$, as shown in Figure 11(c), but in this case the lower limit of γ_1 is also increased. Figure 11(d) shows the case with $\Delta=0$ but at lower τ values. This plot shows only the lower limit of γ_1 because the upper limit is almost independent of τ. The sensitive increase in the lower limit is again detected at β larger than 0.6.

So far we have considered not only the accuracy but also the stability aspect of the algorithm of Guo *et al.* [27]. This is the right time to propose a modified scheme for both stable and accurate results. Dependence of the numerical accuracy on the parameters β, τ, and γ_1 can be understood by directly looking into the formula for K_0, i.e. (48). In most cases the accuracy becomes lower at higher τ values. From (48) we can see that, for $\tau>1$, the increase in β leads to a smaller slip velocity. It also reveals that the decrease in γ_1 brings a lower error too. Thus, as for the numerical accuracy, β must be large and γ_1 must be small, if possible. On the other hand, the stability diagrams of Figure 11(a)–(d) indicate that we must consider the limit of these values to avoid numerical instability. If we were allowed to select just one set of parameter values considering these aspects, we may choose

$$\beta=0.6, \quad \gamma_1=-0.2 \tag{50}$$

as a proper parameter set. With this selection, the modified scheme is stable for all possible choices of parameter values if $\tau>0.52$ (or $Re<465$ with $u_{\max}=0.1$ and $N=31$). The magnitude of the slip velocity with this set-up is checked by using the analytic formula, and the result is shown in Figure 12. Comparing this result with Figure 10, we can see that, in the region $\Delta<0.75$, the suggested scheme provides almost the same level of the slip velocity as the original scheme. In the other region, $\Delta\geqslant 0.75$, the proposed scheme yields the level of slip velocity half that of the original scheme. The proposed parameter set (50), however, may not be the most proper for other geometries such as a circular-Couette flow as demonstrated in the following section.

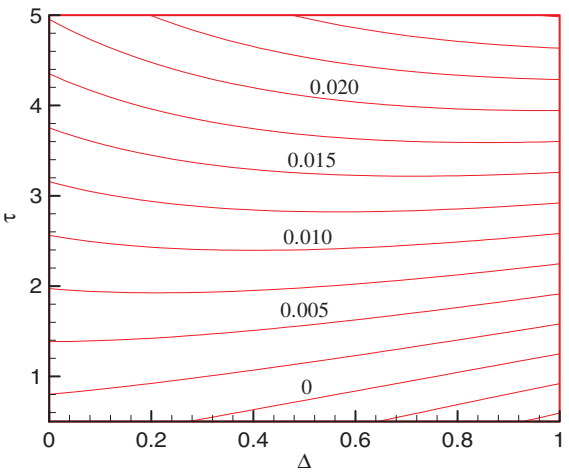

Figure 12. Contour plot of the normalized slip velocity U_w given by the analytic solution with the parameters $\beta=0.6$ and $\gamma_1=-0.2$ proposed in this study for the algorithm of Guo *et al.* [27] in the channel flow.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

5. ASSESSMENT OF ALGORITHMS FOR CIRCULAR-COUETTE FLOW

We applied the algorithms to a circular-Couette-flow problem in order to catch the extent in which our assertion holds for flows with curved boundaries. The flow model comprises by an incompressible fluid confined between a stationary outer circle with radius r_2 and an inner circle with radius r_1 rotating with linear velocity u_0, see Figure 13. The exact solution is given as [27]

$$u_r = 0, \quad u_\theta = \frac{u_0 b}{1-b^2}\left(\frac{r_2}{r} - \frac{r}{r_2}\right) \tag{51}$$

where u_r and u_θ are the radial and azimuthal components of the velocity vector $\mathbf{u} = (u_r, u_\theta)$, respectively, and $b = r_1/r_2$. The number of grids are $N \times N$ and the center of the circles is at $x_c = y_c = N/2$. The Reynolds number is defined here as $Re = u_0(r_2 - r_1)/\nu$ and the rms error is defined as the root mean square of the normalized-velocity-vector difference reading

$$\varepsilon = \sqrt{\frac{1}{N_f}\sum_{i,j} |[\mathbf{u}_{\text{exa}}(x_i, y_j) - \mathbf{u}_{\text{num}}(x_i, y_j)]/u_0|^2} \tag{52}$$

where summation is over all the fluid nodes and N_f stands for the number of fluid nodes.

We tested three algorithms for the no-slip boundary condition at the circles; the B-B scheme, the method of Yu *et al.* [26] and the method of Guo *et al.* [27]. Implementation of the B-B scheme is the simplest because it is independent of the link ratio Δ as shown by (21a)–(21c). Employing the method of Yu *et al.* [26], i.e. (30a)–(30c), is also straightforward. For the method of Guo *et al.* [27], we apply (38) to obtain the distribution function $\widetilde{f}_{k,s}$ at a solid node. Here the non-equilibrium part $\widetilde{f}^{(\text{ne})}_{k,s}$ is computed without difficulty by using the interpolation scheme (41). In computing the

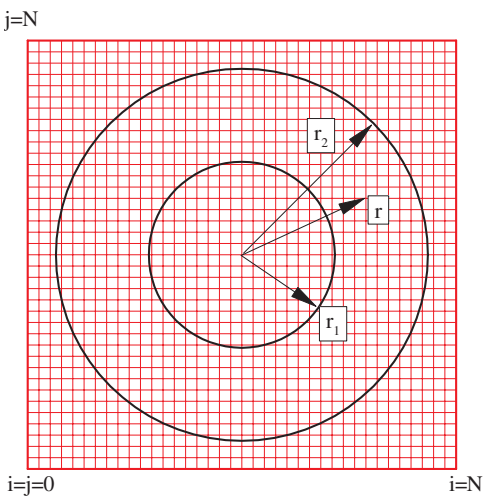

Figure 13. Configuration of the Couette-flow model between concentric circles.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

equilibrium part $f_{k,s}^{(\mathrm{eq})}$ by using (2) we need to know the velocity vector at the solid node $\mathbf{u}_s$. We can use a different value, i.e. $\mathbf{u}_{k,s}$, for a different link given by the following interpolation scheme:

$$\mathbf{u}_{k,s} = (1-\gamma_1-\gamma_2)\mathbf{u}_{k,b} + \gamma_1 \mathbf{u}_{k,f} + \gamma_2 \mathbf{u}_{k,ff} \tag{53}$$

where $\mathbf{u}_{k,b}$ denotes the velocity imposed at the boundary point on the link k, $\mathbf{u}_{k,f}$ the fluid velocity at the nodal point 'f' located at the other end of the link k, and $\mathbf{u}_{k,ff}$ the fluid velocity at the nodal point '*ff*' one more link further away from the node 'f' along the k-direction. However, it turned out that this method yielded lower accuracy and poor stability. Therefore, we applied a single value of $\mathbf{u}_s$ in computing $f_{k,s}^{(\mathrm{eq})}$ regardless of the link. Employed in this study is a weighted-average technique as follows:

$$\mathbf{u}_s = \frac{\sum_k (\mathbf{n}\cdot\mathbf{e}_k)\mathbf{u}_{k,s}/|\mathbf{e}_k|}{\sum_k (\mathbf{n}\cdot\mathbf{e}_k)/|\mathbf{e}_k|} \tag{54}$$

where summation is over all the links that cross the boundary and $\mathbf{n}$ is a unit normal to the boundary drawn from the solid grid point. In the above formulation, the weight factor corresponds to $(\mathbf{n}\cdot\mathbf{e}_k)/|\mathbf{e}_k|$ representing the direction cosine of the link k referring to $\mathbf{n}$. This method turned out to improve both the numerical accuracy and the stability significantly.

Among the various parameters affecting the numerical results, we fix $u_0 = 0.01$ and $b = 0.5$. The outer circle's radius r_2 is determined by $r_2 = N/2 - 2.5$. We also set $N = 52$ as the standard grid. Figure 14(a) shows typical variations of the rms errors upon the change of τ for three algorithms. It reveals that the modified method of Guo *et al.* [27] ('ModGuo' hereafter) with $\beta = 2$ and $\gamma_1 = -1$ gives the most accurate results. On the other hand, the method of Yu *et al.* [26] is accurate only at small τ values. The critical value τ_c for the case of the method of Yu *et al.* (simply 'Yu' hereafter) is much smaller than that of the other methods, being consistent with the previous analysis, e.g. (34). We can also confirm the quadratic increase in ε with τ for both the 'B-B' method and Yu's method. Yu's method becomes less accurate than B-B at $\tau > 1.2$; of course, this is due to the coarse grids used, i.e. $N = 52$. Figure (14b) demonstrates the effect of the parameter set (β, γ_1) on the rms error for ModGuo. With $\beta = 0.6$ being fixed, decrease in γ_1 yields an enhanced accuracy at the supercritical regime ($\tau > \tau_c$). It also provides increased τ_c; this is also in line with the previous analysis as seen from (48). The error level produced by the OriGuo (dashed line) is almost the same as that of ModGuo with $\beta = 0.6$ and $\gamma_1 = -0.2$. On the other hand, with $\gamma_1 = -1$ fixed, increase in β does not affect the change of τ_c but enhances the accuracy. Overall, increase in β (roughly up to $\beta = 2$) and/or decrease in γ_1 (roughly down to $\gamma_1 = -1$) results in a higher accuracy, which is consistent with the previous analysis. However, as an adverse effect, it also causes numerical instability. As a typical case, ModGuo with $\beta = 2$ and $\gamma_1 = -1$ cannot give a stable solution at $\tau = 0.6$. This instability problem becomes more pronounced at higher grid resolutions.

Figure 15 compares the effect of N on ε for different algorithms. Except for B-B, all the algorithms exhibit 2nd-order accuracy. It also indicates that ModGuo with $\beta = 2$ and $\gamma_1 = -1$ produces the most accurate results. On the other hand, we can observe that ε fluctuates significantly with N especially for B-B and Yu. In order to clarify the main reason for such sensitive dependence of ε on N, we fixed $r_2 = 37.5$ and $N = 80$, and changed r_1 from 17.5 to 19. Figure 16 shows the variation of ε with r_1 for the case B-B. The general trend is that ε increases suddenly with a slight increase in r_1 at several points; this is the most pronounced at $r_1 = 18$. It was found from numerical computations for a more localized range near $r_1 = 18$ that a sudden jump occurs while r_1 switches from 18.026 to 18.028; these numbers are very close to the square root of 325. Therefore, we have

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

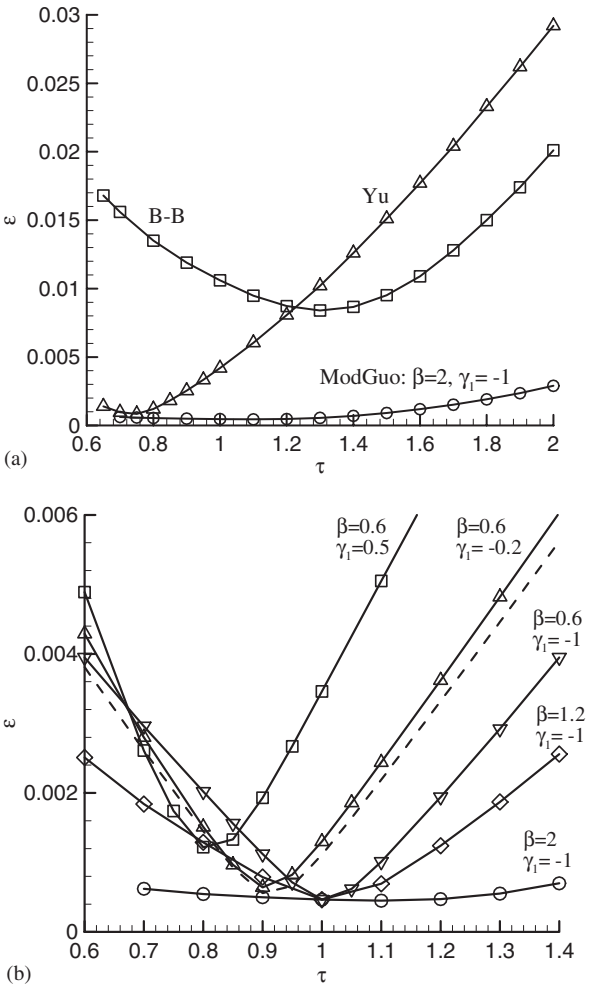

Figure 14. Distribution of the rms error ε *versus* the relaxation time τ for the circular-Couette flow: (a) comparison among three algorithms and (b) effect of β and γ_1 in the modified Guo's method. The dashed line in (b) is obtained from the original method of Guo *et al.* [27].

found from further analysis that, during this switching, 96 'boundary links' (links that intersect the circular boundary) change its link ratio Δ from the values very close to 0 to the ones very close to 1; a fraction of these links among all the boundary links on the inner circle is exceptionally large at this r_1. Such a sudden shift in the histogram of Δ should cause fluctuation of ε observed in Figure 15. This fluctuation becomes weaker when the circles' center is off the grid point, e.g. with $x_c = N/2 - 0.2$ and $y_c = N/2 - 0.3$. It was also found that such a sudden jump in ε is much weaker when r_2 is varied while r_1 is fixed.

As mentioned several times previously, increasing β and decreasing γ_1 cause enhanced accuracy but simultaneously bring an adverse effect, the stability problem. Therefore, we have performed numerical calculations to obtain stability diagrams in the parameter space (β, γ_1) as shown in

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

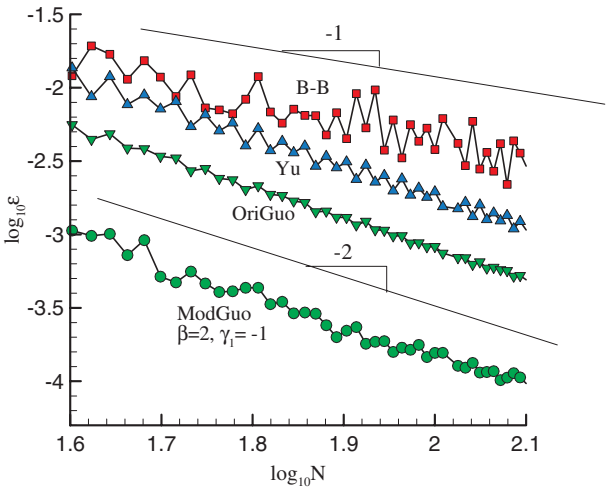

Figure 15. Dependence of the rms error on the grid resolution obtained by four algorithms for the Couette flow.

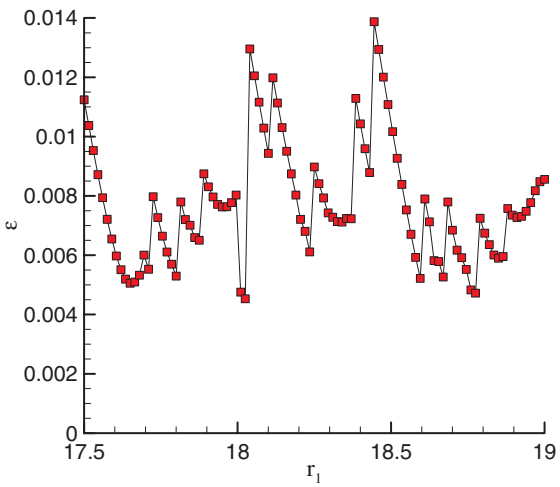

Figure 16. Rms error ε *versus* the inner circle's radius r_1 obtained numerically by using the bounce-back scheme for the Couette flow with $N=80$ and $r_2=37.5$ fixed.

Figure 17(a) for the upper limit values of τ and Figure 17(b) for the lower limit values of τ at $N=102$. The diagrams indicate that stability is mostly determined by β rather than by γ_1. Increase in τ in the supercritical regime (Figure 17(a)) and decrease in τ in the subcritical regime (Figure 17(b)) are shown to reduce the stable region. For the case with $\beta=2$, stability can be attained with the range $\gamma_1=-1.0$–2.0 at $\tau=2.0$ (supercritical) and with the range $\gamma_1=-0.8$–2.0 at $\tau=0.7$ (subcritical).

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

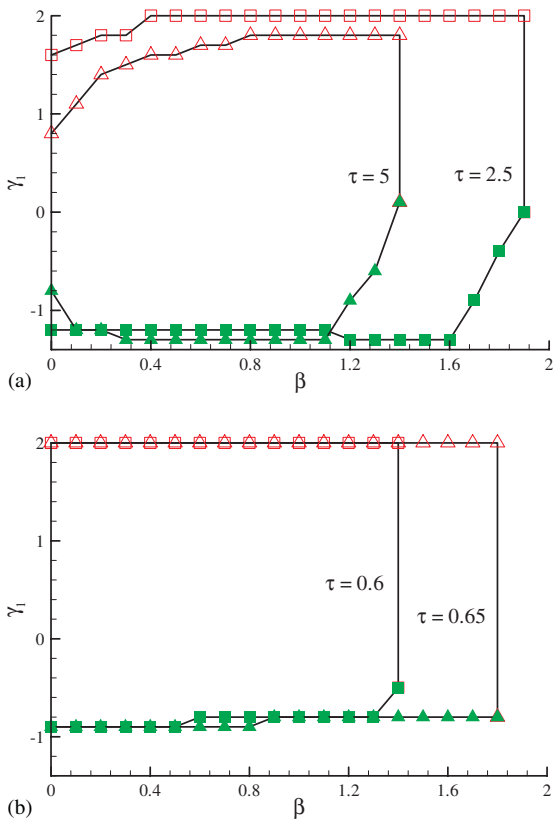

Figure 17. Stability diagrams in the parameter space (β, γ_1) obtained numerically by using the modified algorithm of Guo *et al.* [27] for the circular-Couette flow with $N = 102$: (a) the upper limit values of τ and (b) the lower limit values of τ. Open symbols denote the upper limit of γ_1, and closed symbols denote the lower limit, similar to Figure 11.

6. CONCLUSIONS

It was confirmed from the theoretical and numerical analyses of the simple channel flow and the circular-Couette flow that the numerical accuracy of LBM is dependent on the algorithm used to implement the no-slip boundary condition at the boundary walls. Among various parameters, we investigated the effect of the grid size, the relaxation time, and the link ratio on the numerical accuracy. For the simple channel flow, we derived the analytic formula of the slip velocity for the methods of B-B, Yu, and Guo. It was shown that the numerical error is mainly caused by the slip velocity occurring at the solid boundaries. The slip-velocity formula not only confirms the order of accuracy of the specific method but also quantifies the dependence of accuracy on the relaxation time τ as well as the link ratio Δ. For the case of B-B and Yu, the slip velocity quadratically increases with τ, whereas for the case of Guo it increases linearly. To improve the accuracy, we proposed ModGuo in this study, modified algorithm of Guo *et al.* [27], where the parameters (β, γ_1) are allowed to vary. We have shown from the analysis of the channel flow and

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

from the numerical solutions to the circular-Couette flow that increasing β and decreasing γ_1 result in a significant improvement in the accuracy, in particular at large τ values. On the other hand, we have also shown consistently in both flows that the stable region shrinks when τ is increased, in particular for ModGuo. Therefore, when the accuracy is of our primary importance at relatively large τ, it will be good to use ModGuo with (β, γ_1) close to $(2, -1)$; the parameter set can be tuned if instability occurs.

ACKNOWLEDGEMENTS

This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the National Research Lab. Program funded by the Ministry of Science and Technology (No. 2005-01091).

REFERENCES

1. Chen S, Doolen G. Lattice Boltzmann method for fluid flows. *Annual Review of Fluid Mechanics* 1998; **30**:329–364.
2. Xi H, Duncan C. Lattice Boltzmann simulation of three-dimensional single droplet deformation and breakup under simple shear flow. *Physical Review E* 1999; **59**(3):3022–3026.
3. Inamuro T, Tajima S, Ogino F. Lattice Boltzmann simulation of droplet collision dynamics. *International Journal of Heat and Mass Transfer* 2004; **47**:4649–4657.
4. Zhang J, Kwok DY. A 2D lattice Boltzmann study on electrohydrodynamic drop deformation with the leaky dielectric theory. *Journal of Computational Physics* 2005; **206**:150–161.
5. Takada N, Misawa M, Tomiyama A, Fujiwara S. Numerical simulation of two- and three-dimensional two-phase fluid motion by lattice Boltzmann method. *Computer Physics Communications* 2000; **129**:233–246.
6. Yang ZL, Dinh TN, Nourgaliev RR, Sehgal BR. Numerical investigation of bubble growth and detachment by the lattice-Boltzmann method. *International Journal of Heat and Mass Transfer* 2001; **44**:195–206.
7. Sankaranarayanan K, Kevrekidis IG, Sundaresan S, Lu J, Tryggvason G. A comparative study of lattice Boltzmann methods for bubble simulation. *International Journal of Multiphase Flow* 2003; **29**:109–116.
8. Inamuro T, Ogata T, Ogino F. Numerical simulation of bubble flows by the lattice Boltzmann method. *Future Generation Computer Systems* 2004; **20**:959–964.
9. Shan X, Chen H. Lattice Boltzmann model for simulating flows with multiple phases and components. *Physical Review E* 1993; **46**:1815–1819.
10. Martys N, Douglas J. Critical properties and phase separation in lattice Boltzmann mixtures. *Physical Review E* 2001; **63**:1–18.
11. Ginzburg I, Steiner K. Lattice Boltzmann model for free-surface flow and its application to filling process in casting. *Journal of Computational Physics* 2003; **185**:61–99.
12. Li B, Kwok Y. Discrete Boltzmann equation for microfluidics. *Physical Review Letters* 2003; **90**(12):124502.
13. Li B, Kwok Y. A lattice Boltzmann model for electrokinetic microchannel flow of electrolyte solution in the presence of external forces with Poisson–Boltzmann equation. *International Journal of Heat and Mass Transfer* 2003; **46**:4235–4244.
14. He X, Li N. Lattice Boltzmann simulation of electrochemical systems. *Computer Physics Communications* 2000; **129**:158–166.
15. Hlushkou D, Kandhai D, Tallarek U. Coupled lattice-Boltzmann and finite-difference simulation of electroosmosis in microfluidic channels. *International Journal for Numerical Methods in Fluids* 2004; **46**:507–532.
16. Wang J, Wang M, Li Z. Lattice Poisson–Boltzmann simulations of electro-osmotic flows in microchannels. *Journal of Colloid and Interface Science* 2006; **296**:729–736.
17. Li C, Chen T. Simulation and optimization of chaotic micromixer using lattice Boltzmann method. *Sensors and Actuators B* 2005; **106**:871–877.
18. Inamuro T, Yoshino M, Ogino F. A non-slip boundary condition for lattice Boltzmann simulations. *Physics of Fluids* 1995; **7**(12):2928–2930.
19. Noble DR, Chen S, Georgiadis JG, Buckius RO. A consistent hydrodynamic boundary condition for the lattice-Boltzmann method. *Physics of Fluids* 1995; **7**:203–209.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

20. Chen S, Martinez D, Mei R. On boundary conditions in lattice Boltzmann methods. *Physics of Fluids* 1996; **8**(9):2527–2536.
21. Filippova O, Hänel D. Grid refinement for lattice-BGK models. *Journal of Computational Physics* 1998; **147**: 219–228.
22. Mei R, Luo LS, Shyy W. An accurate curved boundary treatment in the lattice Boltzmann method. *Journal of Computational Physics* 1999; **155**:307–330.
23. Mei R, Shyy W, Yu D, Luo LS. Lattice Boltzmann method for 3-D flows with curved boundary. *Journal of Computational Physics* 2000; **161**:680–699.
24. Bouzidi M, Firdaouss M, Lallemand P. Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Physics of Fluids* 2001; **13**(11):3452–3459.
25. Lallemand P, Luo LS. Lattice Boltzmann method for moving boundaries. *Journal of Computational Physics* 2003; **184**:406–421.
26. Yu D, Mei R, Luo LS, Shyy W. Viscous flow computations with the method of lattice Boltzmann equation. *Progress in Aerospace Science* 2003; **39**:329–367.
27. Guo Z, Zheng C, Shi B. An extrapolation method for boundary conditions in lattice Boltzmann method. *Physics of Fluids* 2002; **14**(6):2007–2010.
28. Zou Q, Hou S, Doolen GD. Analytical solutions of the lattice-Boltzmann BGK model. *Journal of Statistical Physics* 1995; **81**:319–334.
29. He X, Zou Q, Luo LS, Dembo M. Analytic solutions of simple flows and analysis of no-slip boundary conditions for the lattice-Boltzmann BGK model. *Journal of Statistical Physics* 1997; **87**:115–136.
30. He X, Luo LS. Lattice Boltzmann model for the incompressible Navier–Stokes equations. *Journal of Statistical Physics* 1997; **88**:927–944.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld