ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computers & Fluids 35 (2006) 805–813

computers & fluids

www.elsevier.com/locate/compfluid

Adjoint lattice Boltzmann equation for parameter identification

M.M. Tekitek [a], M. Bouzidi [b], F. Dubois [a,d], P. Lallemand [c,*]

[a] *Laboratoire de Mathématiques, Université Paris Sud, Bat. 425, 91405 Orsay Cedex, France*
[b] *Université Clermont Ferrand 2, I.U.T., Ave. A. Briand 03107 Montluçon Cedex, France*
[c] *Laboratoire ASCI, Université Paris Sud, Bâtiment 506, 91405 Orsay Cedex, France*
[d] *CNAM, 292, rue Saint-Martin, 75141 Paris, France*

Received 29 July 2004; accepted 17 July 2005
Available online 20 December 2005

Abstract

The lattice Boltzmann equation is briefly introduced using moments to clearly separate the propagation and collision steps in the dynamics. In order to identify unknown parameters we introduce a cost function and adapt control theory to the lattice Boltzmann equation to get expressions for the derivatives of the cost function vs. parameters. This leads to an equivalent of the adjoint method with the definition of an adjoint lattice Boltzmann equation.

To verify the general expressions for the derivatives, we consider two elementary situations: a linearized Poiseuille flow to show that the method can be used to optimize parameters, and a nonlinear situation in which a transverse shear wave is advected by a mean uniform flow. We indicate in the conclusion how the method can be used for more realistic situations.

1. Introduction

In many situations involving fluid flows, one uses a combination of experimental measurements and of numerical simulations in order to obtain a good knowledge of the flow. Experiments can provide accurate data for some observable quantities (e.g., pressure or local velocity) but may not provide other information. Numerical techniques may be used to compute the missing information but only upon detailed knowledge of parameters that may not be readily available (like the viscosity or the boundary conditions). In order to increase the use of a combination of high quality measurements and refined models, the notion of optimal model has been developed. The parameters of the numerical model are chosen by minimizing the value of a cost function that compares the predictions of the model to known experimental results.

This minimization can be simply obtained by a descent method. Therefore it requires the determination of the derivatives of the cost function with respect to the unknown parameters. A general method to compute those derivatives is provided by control theory and is used in many circumstances. Here we adapt this general method to the modeling of fluid flows by the lattice Boltzmann equation (LBE).

We shall briefly recall the framework of moments that allows a very clear distinction of the two steps of LBE: propagation and collision. Then we adapt the derivation of control theory to the case of a discrete model in order to get the adjoint lattice Boltzmann model. We apply the adjoint model to two simple situations (a steady state and a time dependent case). In the first place we consider the linear LBE and apply it to Poiseuille flow in a two-dimensional periodic channel with a uniform body-force. We then include the nonlinear terms in LBE and show how this modifies the adjoint

[*] Corresponding author.
E-mail addresses: Mohamed-Mahdi.Tekitek@math.u-psud.fr (M.M. Tekitek), Bouzidi@asci.fr (M. Bouzidi), Francois.Dubois@math.u-psud.fr (F. Dubois), lalleman@asci.fr (P. Lallemand).

0045-7930/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2005.07.015

equation. As a simple application, we consider a transverse shear wave advected by a uniform flow.

2. Direct model for lattice Boltzmann equation

The lattice Boltzmann equation is a numerical method based on kinetic theory to simulate various hydrodynamic systems. It uses elements of several origins: the classical Boltzmann equation (see e.g. Ref. [1]), the Broadwell models [2,3] with a small number of velocities and more recently the lattice gas automata [4]. In contrast to the continuous Boltzmann equation that deals with distribution functions $\phi(t,r,\xi)$, the LBE method deals with a small number of functions that can be interpreted as populations of fictitious "particles". The dynamics of these "particles" is such that time, space and momentum are discretized. They move at successive discrete times from nodes to nodes of a regular lattice $\mathscr{T} = \{\mathbf{r}_l, 1 \leqslant l \leqslant K\}$, composed by K nodes so that momentum space $\boldsymbol{\xi}$ is discretized into a small set of discrete velocities $\{\mathbf{e}_\alpha | \alpha = 0,1,\ldots,b\}$. The unknown is the distribution $f_\alpha = f_\alpha(\mathbf{r}_l, t)$ which is function of velocity $\mathbf{e}_\alpha$ at location $\mathbf{r}_l$ and at time t. The lattice Boltzmann equations are written as

$$f_\alpha(\mathbf{r}_l + \mathbf{e}_\alpha, t+1) = f_\alpha(\mathbf{r}_l, t) + \Omega_\alpha(f), \tag{1}$$

The term $\Omega_\alpha(f)$ models the collisions. Macroscopic quantities are obtained by taking velocity moments of f as follows:

$$\rho(\mathbf{r}_l, t) = \sum_{\alpha=0}^{\alpha=b} f_\alpha(\mathbf{r}_l, t), \tag{2}$$

$$\rho\mathbf{u}(\mathbf{r}_l, t) = \sum_{\alpha=0}^{\alpha=b} \mathbf{e}_\alpha f_\alpha(\mathbf{r}_l, t), \tag{3}$$

$$e(\mathbf{r}_l, t) = \sum_{\alpha=0}^{\alpha=b} \mathbf{e}_\alpha^2 f_\alpha(\mathbf{r}_l, t), \tag{4}$$

where ρ is the density (mass), $\mathbf{u}$ is the velocity and e is the energy. They will be used later.

From here, for the sake of simplicity we consider the particular two-dimensional LBE model: the nine velocity model without thermal effects [7]. In this model, $K = N_x N_y$ and $\mathscr{T} = \{\mathbf{r}_l \equiv \mathbf{x}_{i,j};\ i = 1,2,\ldots,N_x;\ j = 1,2,\ldots,N_y\}$ is a square lattice, and there are nine discrete velocities (i.e., $b = 8$) shown in Fig. 1 and algebraically given by

$$\mathbf{e}_\alpha = \begin{cases} (0,0), & \alpha = 0, \\ \left(\cos\left((\alpha-1)\frac{\pi}{2}\right), \sin\left((\alpha-1)\frac{\pi}{2}\right)\right), & \alpha = 1,\ldots,4, \\ \left(\cos\left((2\alpha-9)\frac{\pi}{4}\right), \sin\left((2\alpha-9)\frac{\pi}{4}\right)\right), & \alpha = 5,\ldots,8. \end{cases} \tag{5}$$

Eq. (1) describes the evolution of the particle in a one time increment. So in each increment there are two fundamental steps: advection and collision. Now we will describe these two steps.

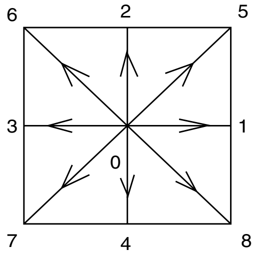

Fig. 1. The velocities for the 9-bit lattice LBE model on a square lattice.

2.1. Advection step

In this step the "particles" move from a lattice node $\mathbf{x}_{i,j}$ to either itself (with the velocity $\mathbf{e}_0 = 0$), one of the four nearest neighbors (with the velocity $\mathbf{e}_\alpha$, $\alpha = 1, \ldots, 4$), or one of the four next-nearest neighbors (with the velocity $\mathbf{e}_\alpha$, $\alpha = 5,\ldots,8$). This step is exact and global in space, since it is the solution to the transport equation given by

$$\frac{\partial f}{\partial t} + \boldsymbol{\xi} \cdot \nabla f = 0.$$

We will represent this step by the operator $\mathscr{A}$ defined by

$$\begin{aligned} \mathscr{A} &: \mathscr{V}_f \to \mathscr{V}_f, \\ \mathscr{F} &\mapsto \mathscr{A}(\mathscr{F}), \end{aligned} \tag{6}$$

where $\mathscr{V}_f \equiv \mathbb{R}^{9\times K}$ and $\mathscr{F} = (f_\alpha(\mathbf{x}_{i,j}, t))^{\mathrm{T}}_{(0\leqslant\alpha\leqslant 8, 1\leqslant i\leqslant N_x, 1\leqslant j\leqslant N_y)}$ is a vector in $\mathscr{V}_f$. Boundary conditions are taken into account through modification of the operator $\mathscr{A}$. Here we shall take either periodic boundary conditions on the outer edges of the fluid domain or the simple "bounce-back" conditions on fluid-solid boundaries.

2.2. Collision step

This step consists in the redistribution of the distribution $\{f_\alpha\}$ at each node $\mathbf{x}_{i,j}$, and it is modeled by the operator $\Omega_\alpha(f)$ in (1). The lattice Boltzmann equation (1) can be rewritten in vector form:

$$\mathbf{f}(\mathbf{x}_{i,j} + \mathbf{e}_\alpha, t+1) = \mathbf{f}(\mathbf{x}_{i,j}, t) + \boldsymbol{\Omega}(\mathbf{f}), \tag{7}$$

where $\mathbf{f} = (f_0, f_1, \ldots, f_8)^{\mathrm{T}}$ and $\boldsymbol{\Omega}(\mathbf{f}) = (\Omega_0(\mathbf{f}), \Omega_1(\mathbf{f}), \ldots, \Omega_8(\mathbf{f}))^{\mathrm{T}}$. We remark that $\mathscr{F} = \left(\mathbf{f}(\mathbf{x}_{i,j}, t)\right)^{\mathrm{T}}_{(1\leqslant i\leqslant N_x, 1\leqslant j\leqslant N_y)} \in \mathscr{V}_f$.

Remark 2.1. If we take the discrete velocities, $\{\mathbf{e}_\alpha | \alpha = 0,1,\ldots,8\}$, with corresponding distribution functions, $\{f_\alpha | \alpha = 0,1,\ldots,8\}$, then we can construct a vector space $\mathbb{V} = \mathbb{R}^9$ based upon the discrete velocity set. So $\mathbf{f} = (f_0, f_1, \ldots, f_8)^{\mathrm{T}}$ is a vector in $\mathbb{V}$. The collision operator acts locally in space $\mathbb{V}$. It will be expressed with the help of the moments.

To describe this operator for each lattice node, we can construct [5] a nine-dimensional vector space $\mathbb{M} = \mathbb{R}^9$ based upon the different moments of $\{f_\alpha\}$. Such that

$$\begin{aligned}&\mathscr{M} : \mathbb{V} \to \mathbb{M},\\ &\mathbf{f} \mapsto \mathscr{M}(\mathbf{f}) = \mathbf{M}\cdot\mathbf{f} = \mathbf{m},\end{aligned} \tag{8}$$

where the orthogonal matrix M is explicitly given by

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1\\ 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1\\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1\\ -4 & -1 & -1 & -1 & -1 & 2 & 2 & 2 & 2\\ 4 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1\\ 0 & -2 & 0 & 2 & 0 & 1 & -1 & -1 & 1\\ 0 & 0 & -2 & 0 & 2 & 1 & 1 & -1 & -1\\ 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \end{pmatrix} \tag{9}$$

and $\mathbf{m} = (\rho, j_x, j_y, e, \epsilon, q_x, q_y, p_{xx}, p_{xy})^{\mathrm{T}}$, where the physical interpretation of the nine moments are respectively: density, x-momentum, y-momentum, energy, energy square, x-heat flux, y-heat flux, diagonal stress and off-diagonal stress. Thus, with the help of the linear transformation $\mathscr{M}$, we can describe the collision operator in moment space $\mathbb{M}$.

In the athermal model the only conserved quantities are density ρ and linear momentum $\mathbf{j} = (j_x, j_y)$. For the other quantities (non-conserved moments) [5], we assume that they relax towards equilibrium values that are nonlinear functions of the conserved quantities. Due to symmetry arguments, the relaxation equations are given by

$$e^* = e - s_2\left[e - \left(\alpha_2\rho + \gamma_2(j_x^2 + j_y^2)\right)\right], \tag{10}$$

$$\epsilon^* = \epsilon - s_3\left[\epsilon - \left(\alpha_3\rho + \gamma_4(j_x^2 + j_y^2)\right)\right], \tag{11}$$

$$q_x^* = q_x - s_5[q_x - (c_1 j_x)], \tag{12}$$

$$q_y^* = q_y - s_5\left[q_y - (c_1 j_y)\right], \tag{13}$$

$$p_{xx}^* = p_{xx} - s_8\left[p_{xx} - \left(\gamma_1(j_x^2 - j_y^2)\right)\right], \tag{14}$$

$$p_{xy}^* = p_{xy} - s_8\left[p_{xy} - (\gamma_3 j_x j_y)\right], \tag{15}$$

where the quantities with and without superscript $*$ are post-collision and pre-collision values, respectively.

Remark 2.2. The relaxation parameters s_i, $i = 2, 3, 8$ are directly linked to the transport coefficients. For the seven other adjustable parameters α_2, α_3, c_1, γ_1, γ_2, γ_3, γ_4, we will fix them as follows:

$$\begin{aligned}&\gamma_1 = \gamma_3 = 1, \quad \gamma_2 = 3, \quad c_1 = -1,\\ &\alpha_2 = -2, \quad \alpha_3 = 1 \quad \text{and} \quad \gamma_4 = -3.\end{aligned} \tag{16}$$

This choice of the parameters γ_1, γ_2, γ_3, c_1 yields Galilean invariance and isotropy. The parameter α_2 is linked to the speed of sound. Two other parameters α_3 and γ_4 are fixed to improve stability. See [7] for the complete derivation of these properties. The relaxation rates s_3 and s_5 play no role in the hydrodynamic behaviour of the model, however they are relevant for stability [7] and for the accuracy of the boundary conditions [11,12].

Now we can contract the collision operator in moment space, with the help of (10)–(15), as follows:

$$\begin{aligned}&\mathrm{C} : \mathbb{M} \to \mathbb{M},\\ &\mathbf{m} \mapsto \mathbf{m}^* = \mathrm{C}(\mathbf{m}).\end{aligned} \tag{17}$$

Thus, we have the collision operator in $\mathbb{V}$, for an initial distribution $\mathbf{f}$, given by

$$\widetilde{\mathbf{\Omega}}(\mathbf{f}) = \mathbf{f} + \mathbf{\Omega}(\mathbf{f}) = M^{-1}\cdot\mathrm{C}(M\cdot\mathbf{f}).$$

So, now we define the global collision operator $\mathscr{C}$ like the advection operator by

$$\begin{aligned}&\mathscr{C} : \mathscr{V}_f \to \mathscr{V}_f,\\ &\mathscr{F} \mapsto \mathscr{C}(\mathscr{F}),\end{aligned} \tag{18}$$

where $\mathscr{C}(\mathscr{F}) = \left(\widetilde{\mathbf{\Omega}}(\mathbf{f})(\mathbf{x}_{1,1}, t), \widetilde{\mathbf{\Omega}}(\mathbf{f})(\mathbf{x}_{1,2}, t), \ldots, \widetilde{\mathbf{\Omega}}(\mathbf{f})(\mathbf{x}_{i,j}, t), \ldots, \widetilde{\mathbf{\Omega}}(\mathbf{f})(\mathbf{x}_{N_x,N_y}, t)\right)^{\mathrm{T}}$.

2.3. Direct model

The net result of the advection and the collision steps is that if $\mathscr{F}_{\text{ini}}$ is the initial state of the system, it evolves according to

$$\begin{cases}\mathscr{F}^0 = \mathscr{F}_{\text{ini}},\\ \mathscr{F}^{k+1} = \mathscr{A}\circ\mathscr{C}\left(\mathscr{F}^k\right) \equiv \Phi\left(\mathscr{F}^k\right), \quad k \in 0, 1, \ldots N-1,\end{cases} \tag{19}$$

where $\mathscr{F}^k$ is the discrete state for particle distribution in space at time k. So $\mathscr{F}^k = (f_\alpha(\mathbf{x}_{i,j}, k))^{\mathrm{T}}_{(0\leqslant\alpha\leqslant 8, 1\leqslant i\leqslant N_x, 1\leqslant j\leqslant N_y)}$. We shall call Eq. (19) the *direct model* which has been shown [6] to behave like the solutions of those of the Navier–Stokes equations in situations where the flow evolves sufficiently slowly in space and time.

3. Adjoint method for identifying parameters

In this section we are interested in identifying some parameters of the lattice Boltzmann scheme, for instance the relaxation parameters s_5 and s_8 by comparing the predictions of the direct model to those derived from some other technique (analytic or numerical solution to the Navier–Stokes equations or from experiments). So, we will use inverse modeling to estimate these parameters. This will be done using the adjoint method, which is directly derived from the optimal control theory [8].

3.1. General discrete theory for adjoint method

To introduce the method, let us consider a steady state laminar Poiseuille flow, with kinematic velocity v, between two plates parallel to Ox, separated by height h, with periodic boundary condition along the flow and a uniform body force (δp) to drive the flow. We know an analytic solution at discrete time k

$$\mathbf{u}(\mathbf{x}_{i,j},k) = \big(u(\mathbf{x}_{i,j},k), v(\mathbf{x}_{i,j},k)\big) = \left(\frac{h\delta p}{2v} y_j\left(1-\frac{y_j}{h}\right), 0\right),$$

$$\text{where } \mathbf{x}_{i,j} = (x_i, y_j) \text{ for } 1 \leqslant i \leqslant N_x,\ 1 \leqslant j \leqslant N_y. \tag{20}$$

We consider the state of the LBE solution for a long enough time so that it has reached steady state computed for the same geometry as the analytic case but with unknown relaxation coefficient $\lambda = (s_5, s_8)$. We note that we neglect any space dependence of the density ρ. So that we will identify from here velocity with momentum $\rho\mathbf{u}$ evaluated by (3). Since we take $\rho \equiv 1$.

We wish to estimate an optimal λ in the sense that it will correspond to the minimum of a cost function. We define the cost-function $J(\lambda)$ in a "natural" way: it is the mean-square difference between the velocity $\widetilde{\mathbf{u}}_k(\lambda, \mathbf{x}_{i,j}, k)$,

$$\widetilde{\mathbf{u}}(\lambda, \mathbf{x}_{i,j}, k) = \big(\widetilde{u}(\lambda, \mathbf{x}_{i,j}, k), \widetilde{v}(\lambda, \mathbf{x}_{i,j}, k)\big), \tag{21}$$

at discrete time $k\delta t$, calculated by LBE (21) and the exact velocity u which is the analytic solution (20). The notation $\widetilde{\mathbf{u}}(\lambda, \mathbf{x}_{i,j}, k)$ corresponds to the fact that the discrete LBE solution $\widetilde{\mathbf{u}}$ depends also on some parameters λ of the model.

We consider the following cost-function $J(\lambda)$, where the first term (with coefficient a) deals with a vision of the error at the final time and the second term (with coefficient b) is a discrete time integration of the error.

$$J(\lambda) = \frac{a}{2}\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}|\widetilde{u}(\lambda, \mathbf{x}_{i,j}, N) - u(\mathbf{x}_{i,j}, N)|^2 + \frac{b}{2}\sum_{k=0}^{N-1}\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}|\widetilde{u}(\lambda, \mathbf{x}_{i,j}, k) - u(\mathbf{x}_{i,j}, k)|^2. \tag{22}$$

$k = k\delta t$ ($\delta t = 1$) is the observation time, $N = N\delta t$ is the time when the steady state is reached, $\lambda = (s_5, s_8) \in \mathbb{R}^2$, and a, b are two real adjustable constants. In the cost-function (22), the term associated with a is used when we simulate a steady state problem. So for the first case of the Poiseuille flow, it is obvious that the exact solution (20) is stationary (i.e., $u(\mathbf{x}_{i,j},k) = u(\mathbf{x}_{i,j})$). We will take $a = 1$, $b = 0$ for the cost function. The term with b is used for the unsteady simulation (e.g., see nonlinear case). The discrete time $N = N\delta t$ ($\delta t = 1$) is the final time where we evaluate the solution. So in this case we will take $a = 0$, $b = 1$.

Now the assimilation process consists in minimizing the cost-function J. We decide to use a gradient method:

$$\lambda^{n+1} = \lambda^n + \widetilde{\omega}\nabla_\lambda J^n(\lambda), \quad \widetilde{\omega} > 0.$$

So we need to estimate the gradient of the cost-function $\nabla_\lambda J(\lambda)$. The adjoint method is used to evaluate the gradient of the cost-function $\nabla_\lambda J$.

First, we rewrite the cost-function:

$$J(\lambda) = \frac{a}{2}\Psi(\mathscr{F}^N, N, \lambda) + \frac{b}{2}\sum_{k=0}^{N-1}\Psi(\mathscr{F}^k, k, \lambda), \tag{23}$$

where $\mathscr{F}^k \in \mathscr{V}_f$ is the solution to LBE and

$$\Psi(\mathscr{F}^k, k, \lambda) = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}|\widetilde{u}(\lambda, \mathbf{x}_{i,j}, k) - u(\mathbf{x}_{i,j}, k)|^2,$$

is the global error at time step k measured with least squares.

Proposition 1. *With the cost-function given by the relation (22) the gradient $\nabla_\lambda J$ can be evaluated as follows:*

$$\nabla_\lambda J = -\sum_{k=0}^{N-1}\mathscr{P}^{k+1}\frac{\partial\Phi}{\partial\lambda}, \tag{24}$$

where $\mathscr{P}^k$ are a set of dual parameters, which are naturally line vectors. The parameters $(\mathscr{P}^k)^{\mathrm{T}}$ belong to the space $\mathscr{V}_f$ introduced in Section 2.1 and are determined by the following backward lattice Boltzmann equation called adjoint lattice Boltzmann equation (ALBE):

$$\begin{cases} \mathscr{P}^N = -\dfrac{a}{2}\dfrac{\partial\Psi}{\partial\mathscr{F}}, \\ \mathscr{P}^k = \mathscr{P}^{k+1}\dfrac{\partial\Phi}{\partial\mathscr{F}} - \dfrac{b}{2}\dfrac{\partial\Psi}{\partial\mathscr{F}} \quad \text{for } k = N-1, N-2, \ldots, 1. \end{cases} \tag{25}$$

Remark 3.1. The $\mathscr{P}^k$ for $k = N-1, N-2, \ldots, 1$ describe an inverse dynamics. So $\mathscr{P}^k$ is a vector defined by $(p_\alpha(\mathbf{x}_{i,j},k))_{(0\leqslant\alpha\leqslant 8, 1\leqslant i\leqslant N_x, 1\leqslant j\leqslant N_y)}$ where $p_\alpha(\mathbf{x}_{i,j},k)$ is the "dual distribution" function of velocity $\mathbf{e}_\alpha$ at location $\mathbf{x}_{i,j}$ and at discrete time k.

Proof. As $\mathscr{F}^k$ is the solution to the direct state (19), we can see this dynamics as a constraint:

$$\begin{cases} \mathscr{F}^0 = 0, \\ \mathscr{F}^{k+1} - \Phi(\mathscr{F}^k) = 0, \quad k = 0, 1, \ldots N-1. \end{cases} \tag{26}$$

And now we can consider the constrained minimization problem of finding the minimum of J given by (23) under the constraint (26).

A classical way to do this is to give a Lagrangian formulation of this problem. So we define a Lagrangian as follows:

$$\mathscr{L} = J + \sum_{k=0}^{N-1} \mathscr{P}^{k+1} \cdot \left(\mathscr{F}^{k+1} - \Phi\left(\mathscr{F}^{k}\right)\right), \tag{27}$$

where the dot $\cdot$ denotes the scalar product in $\mathscr{V}_f$, and $\mathscr{P}^k \in \mathscr{V}_f$ is a Lagrange multiplier related to the constraint (26).

The differentiation of $\mathscr{L}$, reads:

$$\mathrm{d}\mathscr{L} = \mathrm{d}J + \sum_{k=0}^{N-1} \mathscr{P}^{k+1} \left(\mathrm{d}\mathscr{F}^{k+1} - \frac{\partial\Phi}{\partial\mathscr{F}}\mathrm{d}\mathscr{F}^{k} - \frac{\partial\Phi}{\partial\lambda}\mathrm{d}\lambda\right). \tag{28}$$

We note here that the cost-function J does not depend directly on λ, so we have:

$$\mathrm{d}J = \frac{a}{2}\frac{\partial\Psi}{\partial\mathscr{F}}\mathrm{d}\mathscr{F}^{N} + \frac{b}{2}\sum_{k=0}^{N-1}\frac{\partial\Psi}{\partial\mathscr{F}}\mathrm{d}\mathscr{F}^{k}. \tag{29}$$

With a discrete part integration we deduce that:

$$\begin{aligned}\sum_{k=0}^{N-1} \mathscr{P}^{k+1}&\left(\mathrm{d}\mathscr{F}^{k+1} - \frac{\partial\Phi}{\partial\mathscr{F}}\mathrm{d}\mathscr{F}^{k}\right)\\ &= \mathscr{P}^{N}\mathrm{d}\mathscr{F}^{N} - \mathscr{P}^{1}\frac{\partial\Phi}{\partial\mathscr{F}}\mathrm{d}\mathscr{F}^{0}\\ &\quad + \sum_{k=1}^{N-1}\left(\mathscr{P}^{k} - \mathscr{P}^{k+1}\frac{\partial\Phi}{\partial\mathscr{F}}\right)\mathrm{d}\mathscr{F}^{k}.\end{aligned} \tag{30}$$

Now using (28)–(30), we find:

$$\begin{aligned}\mathrm{d}\mathscr{L} = \frac{a}{2}\frac{\partial\Psi}{\partial\mathscr{F}}\mathrm{d}\mathscr{F}^{N} &+ \frac{b}{2}\sum_{k=1}^{k=N-1}\frac{\partial\Psi}{\partial\mathscr{F}}\mathrm{d}\mathscr{F}^{k} + \mathscr{P}^{N}\mathrm{d}\mathscr{F}^{N}\\ &- \mathscr{P}^{1}\frac{\partial\Phi}{\partial\mathscr{F}}\mathrm{d}\mathscr{F}^{0} + \sum_{k=1}^{N-1}\left(\mathscr{P}^{k} - \mathscr{P}^{k+1}\frac{\partial\Phi}{\partial\mathscr{F}}\right)\mathrm{d}\mathscr{F}^{k}\\ &- \sum_{k=0}^{N-1}\mathscr{P}^{k+1}\frac{\partial\Phi}{\partial\lambda}\mathrm{d}\lambda.\end{aligned} \tag{31}$$

Since we do not change the initial condition of the direct model (19), we choose

$\mathrm{d}\mathscr{F}^0 = 0.$

Due to (27), we notice that $\mathrm{d}\mathscr{L} \equiv \mathrm{d}J = \nabla_\lambda J \cdot \mathrm{d}\lambda$ when constraint (26) holds. When we choose the adjoint dynamics, i.e., $\mathscr{P}^k$ be equal to the solution to the backward LB equation (25), we cancel all the terms in factor of $\mathrm{d}\mathscr{F}^k$ in the expression (31) of $\mathrm{d}\mathscr{L}$, so that the expression (24) of $\nabla_\lambda J$ is established. □

We proceed now to compute the adjoint state and the gradient of J.

3.2. Adjoint lattice Boltzmann equation for linear case

Now, we describe the adjoint model (25), which allows us to compute all the $\mathscr{P}^k$. In a first case we consider the steady state laminar Poiseuille flow which is introduced in Section 3.1. This case is a natural way to test this method, since laminar Poiseuille flow is also a solution to the Stokes problem which is linear. So the coefficients $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 0$ and the direct algorithm (19) is simpler (i.e., the operator Φ is linear). Since we have a steady problem we choose $a = 1$ and $b = 0$. So we have an adjoint dynamics $\mathscr{P}^k = \mathscr{P}^{k+1}\frac{\partial\Phi}{\partial\mathscr{F}}$, with initial condition depending on error, where $\frac{\partial\Phi}{\partial\mathscr{F}}$ is a linear operator defined by

$$\begin{aligned}&\frac{\partial\Phi}{\partial\mathscr{F}} : \mathscr{V}_f^{\star} \to \mathscr{V}_f^{\star},\\ &\mathscr{P} \mapsto \mathscr{P}\mathscr{A}\mathscr{C} = (\mathscr{C}^{\mathrm{T}}\mathscr{A}^{\mathrm{T}}\mathscr{P}^{\mathrm{T}})^{\mathrm{T}}.\end{aligned} \tag{32}$$

where $\mathscr{V}_f^{\star}$ is the dual space of linear space $\mathscr{V}_f$. This operator (since the operator $\Phi = \mathscr{A}\mathscr{C}$ is linear in the direct model) is composed of two fundamental steps:

- Transposed advection $\mathscr{A}^{\mathrm{T}}$: which models the transport with "backward" discrete velocity.
- Transposed collision $\mathscr{C}^{\mathrm{T}}$: as in the direct model, this operator is local in space.

So we also need to introduce the following nine "dual moments" of $\mathbf{p}(\mathbf{x}_{i,j},k) = (p_\alpha(\mathbf{x}_{i,j},k))_{0\leqslant\alpha\leqslant8}$ with the help of the matrix M (9):

$\mathbf{m} = (m_0, m_1, \ldots, m_8) = \mathbf{p}\cdot M^{-1}.$

Since there are three conserved moments in the direct model, there are also three conserved quantities in the adjoint model given by

$m_0 + 2m_3 - m_4$ scalar invariant like mass ρ,

$\begin{pmatrix} m_1 + m_5, \\ m_2 + m_6 \end{pmatrix}$ vector invariant like velocity.

The matrix of transposed collision C in the space of moments satisfies:

$$C^{\mathrm{T}} = \begin{pmatrix} 1 & 0 & 0 & -2s_2 & s_3 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & -s_5 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0 & -s_5 & 0 & 0\\ 0 & 0 & 0 & 1-s_2 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1-s_3 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 1-s_5 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 1-s_5 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1-s_8 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1-s_8 \end{pmatrix}.$$

The collision is defined by $\mathscr{P}\cdot\mathscr{C} = \mathscr{P}^*$ where $\mathscr{P} = (\mathbf{p}(\mathbf{x}_{i,j}, k))_{(1\leqslant i\leqslant N_x, 1\leqslant j\leqslant N_y)} \in \mathscr{V}_f^{\star}$, $\mathscr{P}^* = (\mathbf{p}^*(\mathbf{x}_{i,j}, k))_{(1\leqslant i\leqslant N_x, 1\leqslant j\leqslant N_y)} \in \mathscr{V}_f^{\star}$, where $\mathbf{p}^*(\mathbf{x}_{i,j},k) = \mathbf{p}(\mathbf{x}_{i,j},k)\cdot M^{-1}\cdot C\cdot M = ((M)^{\mathrm{T}}\cdot C^{\mathrm{T}}\cdot(M^{-1})^{\mathrm{T}}\cdot(\mathbf{p}(\mathbf{x}_{i,j},k))^{\mathrm{T}})^{\mathrm{T}}$.

3.3. ALBE algorithm for the nonlinear case

When we model the Navier–Stokes equation, the direct algorithm (19) is nonlinear. So $\gamma_i \neq 0$, $i = 1, 2, 3, 4$, and the collision step is nonlinear. In this case the adjoint algorithm (25) is still linear since $\mathscr{P}^k = \mathscr{P}^{k+1}\frac{\partial\Phi}{\partial\mathscr{F}}$, where $\frac{\partial\Phi}{\partial\mathscr{F}}$ is a linear operator. As in the linear case $\frac{\partial\Phi}{\partial\mathscr{F}}$ is defined by (32) and it is composed of two steps: transposed advection $\mathscr{A}^{\mathrm{T}}$ and transposed collision $\mathscr{C}^{\mathrm{T}}$. Only the transposed collision $\mathscr{C}^{\mathrm{T}}$ is different from the linear case. To describe this step let us use the superscript d for quantities related to the direct problem and among them $V^d(\mathbf{x}_{i,j}, t)$ the velocity field. So the transposed matrix collision C^{T} is expressed as follows:

$$C^{\mathrm{T}} = \begin{pmatrix} 1 & 0 & 0 & -2s_2 & s_3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 2\gamma_1 s_2 V_x^d & 2\gamma_3 s_3 V_x^d & -s_5 & 0 & \gamma_3 s_8 V_y^d & 2\gamma_1 s_8 V_x^d \\ 0 & 0 & 1 & 2\gamma_1 s_2 V_y^d & 2\gamma_3 s_3 V_y^d & 0 & -s_5 & \gamma_3 s_8 V_x^d & -2\gamma_1 s_8 V_y^d \\ 0 & 0 & 0 & 1-s_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1-s_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1-s_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1-s_5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1-s_8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1-s_8 \end{pmatrix}.$$

Compared to the linear case, there are just a few additional off-diagonal terms and these terms can be computed from the information in the stored $\mathscr{F}$ field of the direct problem.

4. First numerical experiments for a Poiseuille flow

4.1. Case of a one scalar parameter problem

Our first simple case consists in identifying a single parameter s_8 (i.e., the viscosity $\nu = \frac{1}{3}(\frac{1}{s_8} - \frac{1}{2})$) and the parameter s_5 is supposed to be known. So the unknown parameter λ is equal to s_8 (i.e., $\lambda = s_8 \in \mathbb{R}$). In this case, the discrete exact gradient is given by the help of (31), (32) and (25) as follows:

$$\nabla_\lambda J = J'(s_8) = \mathrm{d}\mathscr{L} = -\sum_{k=0}^{N-1} \mathscr{P}^{k+1}\frac{\partial\Phi}{\partial s_8}. \tag{33}$$

The computation of $J'(s_8)$ requires one integration of the direct model and one integration of the adjoint model. Thus, we may try and apply a descent method in order to find the solution to the minimization problem.

Before making use of the adjoint method, it is necessary to check that we calculate exactly the gradient of the cost-function J. For that purpose we consider a simple determination of the gradient for finite difference quotient.

- Difference quotient of the cost function J

$$J'_{dq}(s_8) = \lim_{\epsilon\to 0}\frac{J(s_8+\epsilon) - J(s_8-\epsilon)}{2\epsilon}. \tag{34}$$

So we can validate the adjoint model if and only if the two quantities (33) and (34) are equal. We have gradient J with adjoint method $J'_{\text{inv}}(s_8 = 0.3) = -4.87278648\times 10^{-7}$.

Table 1 shows that the gradient is well calculated, so for $\epsilon = 10^{-6}$ we have the same quantity for two gradients with a 10^{-8} relative accuracy.

Remark 4.1. Fig. 2 shows that for $\epsilon \leqslant 10^{-10}$ we have machine precision errors and for $\epsilon \geqslant 10^{-3}$ we have convergence errors.

Table 1
Comparison of the proposed determination of gradient with a simple finite difference quotient: relative error: $\frac{|J'_{df}(s_8) - J'_{\text{inv}}(s_8)|}{J'_{\text{inv}}(s_8)}$ for $s_8 = 0.3$ and optimal $s_8 = 0.8$

ϵ	Relative error
10^{-2}	4.89×10^{-3}
10^{-3}	4.84×10^{-5}
10^{-4}	4.83×10^{-7}
10^{-5}	3.85×10^{-9}
10^{-6}	9.44×10^{-10}
10^{-7}	9.19×10^{-10}
10^{-8}	1.30×10^{-9}
10^{-9}	3.20×10^{-9}
10^{-10}	4.56×10^{-8}
10^{-11}	1.00×10^{-7}
10^{-12}	3.63×10^{-6}
10^{-13}	1.99×10^{-5}
10^{-14}	2.10×10^{-4}
10^{-15}	2.23×10^{-3}

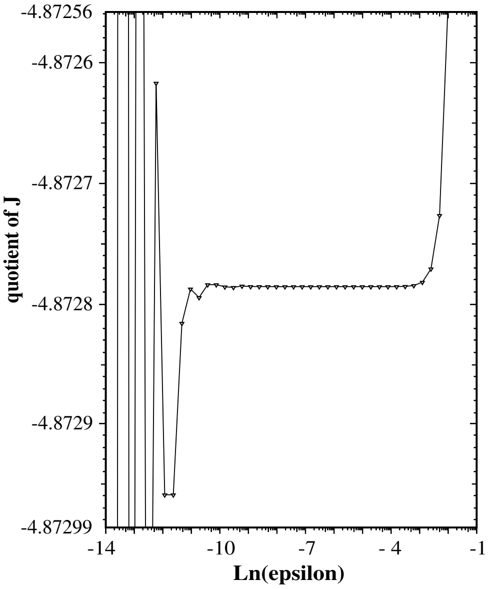

Fig. 2. Difference quotient of the cost function J vs. ln(ϵ).

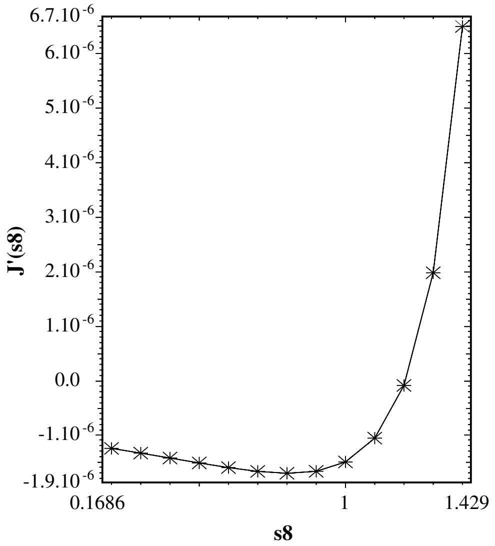

Fig. 3. Profiles of gradient $J'(s_8)$ vs. s_8 calculated by two methods, $J'(s_8)$ with adjoint method (+) and with difference quotient of J (×). Optimal $s_8 = 1.2$.

In Fig. 3, we show that the adjoint method is able to calculate exactly the gradient of cost-function.

Now, we try to identify the parameter s_8 by a descent method with a fixed step:

$$s_8^{n+1} = s_8^n + \widetilde{\omega} J'(s_8)^n, \quad \widetilde{\omega} > 0.$$

Fig. 4 shows the convergence of algorithm to the optimal parameter s_8^*. In this case the first guess is $s_8 = 0.2$ and the optimal one $s_8^* = 0.8$.

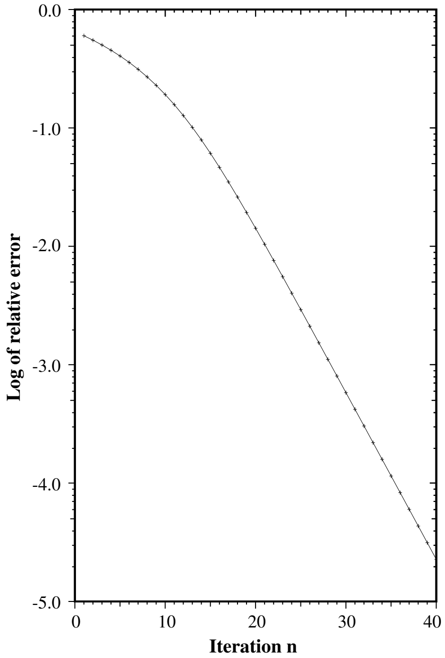

Fig. 4. Log of the error ($|s_8 - s_8^*|$) vs. iteration n.

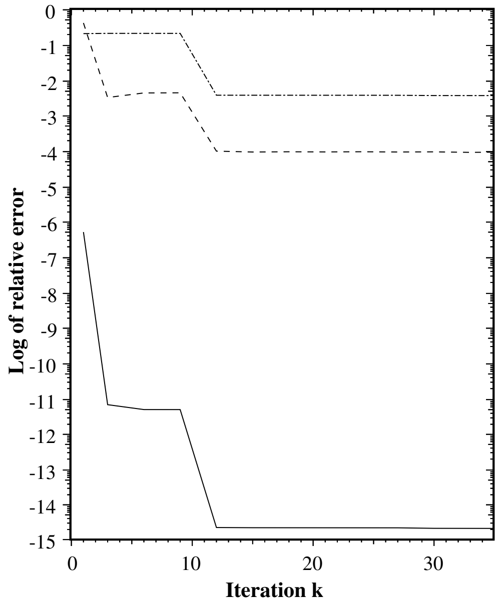

Fig. 5. Log of the cost function $J(s_5, s_8)$ (solid line), log of the error ($|s_5 - s_5^*|$) (dashed-dotted line) and log of the error ($|s_8 - s_8^*|$) (dashed line) vs. iteration k.

4.2. Case of two parameters

In the second case, the two parameters s_8 and s_5 are unknown (i.e., $\lambda = (s_5, s_8)$). So we use the adjoint method to evaluate the discrete gradient $\nabla_\lambda J$. So in this case the gradient is given by the help of (31), (32) and (25) as follows:

$$\nabla_\lambda J = \nabla J(s_5, s_8) = -\sum_{k=0}^{N-1} \mathscr{P}^{k+1} \frac{\partial \Phi}{\partial \lambda}$$
$$= \left(-\sum_{k=0}^{N-1} \mathscr{P}^{k+1} \frac{\partial \Phi}{\partial s_5}, -\sum_{k=0}^{N-1} \mathscr{P}^{k+1} \frac{\partial \Phi}{\partial s_8} \right).$$

We use a descent method with a variable step.

$$\lambda^{n+1} = \lambda^n + \widetilde{\omega}^n \nabla_\lambda J^n(\lambda), \quad \widetilde{\omega}^n > 0,$$

where $\widetilde{\omega}^n$ are calculated by a standard line search [9]. Fig. 5 shows the convergence of algorithm to the optimal parameters $\lambda^* = (s_5^*, s_8^*) = (1.0, 0.8)$. The first guess is $(s_5, s_8) = (1.2, 1.2)$. We find that $|\nabla_{s_8} J| \gg |\nabla_{s_5} J|$ as expected, since it is known that s_8 is related to the viscosity whereas s_5 has only subtle effects on the accuracy of the boundary conditions [11,12].

5. ALBE method for a Navier–Stokes flow

We shall consider a simple case, that of a flow arising from the superposition of a uniform flow with speed $\{V, 0\}$ and a transverse shear wave in a domain with periodic boundary conditions, so that if N_x is the number of lattice points along Ox, one expects to find a time dependent solution, assuming $v \ll V$,

$$v_x = V, \tag{35}$$

$$v_y = v\cos(k(x - Vt) + \phi)\exp(-\nu k^2 t), \tag{36}$$

where the wave vector k is of the form $k = 2m\pi/N_x$ (m integer) and ϕ some phase factor.

Now we will suppose that $\gamma_1 = \gamma_3 = \frac{\gamma_2}{3} = c$ and $\gamma_4 = 3d$. We have introduced two unknown parameters c and d (i.e., $\lambda = (c, d)$) in the expression (16) of γ_i, $i = 1, 2, 3, 4$, having in mind the use of the adjoint method to find their values in order to get a model optimally chosen with respect to a required solution. Once we have solved the adjoint problem, we can determine the derivatives of the cost function, using expressions (24).

We have tested the ability of the adjoint method to determine c and d using the particular cost function with $a = 0$ and $b = 1$ in Eq. (22), which is appropriate for a time dependent problem. We have found that the derivative of the cost function with respect to the parameter d is very small (and probably insignificant due to rounding errors in the numerical simulations). This is expected as the term depending upon d does not show up in the Chapman–Enskog analysis [5] of the problem. It is taken into consideration for the simple reason that in the ordinary BGK-LBE model [10], the equilibrium distributions lead to such a term.

The derivative of the cost function with respect to c obtained by the adjoint method is close to that determined by finite differences as was the case above for the parameter s_8 directly linked to the viscosity. We show in Fig. 6 the convergence of the error function with iteration number (no effort has been made to accelerate convergence).

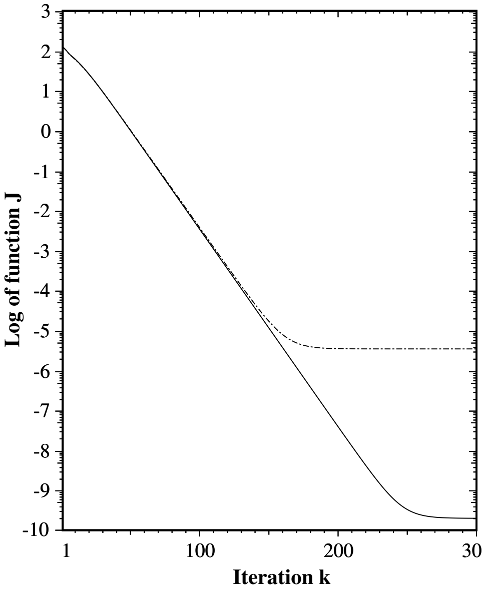

Fig. 6. Log of the cost-function $J(\lambda)$ vs. iteration.

Note that for one case the error levels at a significant value (dashed-dotted line in Fig. 6). That case corresponds to using as the expression given above for the "target function" and using as initial state $\mathscr{F}_0$ the distribution function computed to second order in Chapman–Enskog development. The case that leads to much better convergence (solid line in Fig. 6) uses as "target function" the velocity of a LBE model in which $c = d = 1$. This shows that the initial conditions used in the first case are not satisfactory and that they do not lead precisely to the simple analytic expressions given above. This result could be used to try and determine better initial conditions that lead to a small residual error.

Note more generally that the identification procedure proposed in this paper will not give information on other sources of error (quality of the target or in the numerical model).

6. Conclusion

We have considered the problem of parameter identification for the LBE model in computational fluid dynamics. We have used a gradient method associated with the adjoint methodology applied for discrete time. We have compared this approach with a finite difference methodology and have tested our scheme for two different configurations: a simple linear Poiseuille flow and a more realistic nonlinear model. We have derived the general adjoint model (ALBE). We note that this algorithm is as easy to parallelize as the standard LBE model.

Work is under way to test the ability of the proposed method to determine a large number (p) of unknown parameters. In that case for each set of unknown parameters for which the gradients are required, one needs one direct and one backward computation instead of at least p direct computations using the simple finite quotient determination.

The extension for future work could be the following: determination of the numerical scheme for ALBE model in case of curved boundaries, identification of unknown flow parameters at the boundary or identification of local viscosity for turbulent flows. The extension to three-dimensional flows is straightforward concerning the methodology but the difficulty will be in the larger amount of data to manage.

References

[1] Hirschfelder JO, Curtiss CF, Bird RB. Molecular theory of gases and liquids. New York: Wiley; 1945.

[2] Broadwell JE. Shock structure in a simple discrete velocity gas. Phys Fluids 1964;7:1243–7.

[3] Gatignol R. Théorie cinétique des gaz à répartition discrète de vitesses. Lecture Notes in Phys., vol. 36. New York: Springer-Verlag; 1975.

[4] Frisch U, Hasslacher B, Pomeau Y. Lattice-gas automata for the Navier–Stokes equation. Phys Rev Lett 1986;56:1505–8.

[5] d'Humières D. Rarefied gas dynamics: theory and simulations. In: Shizgal BD, Weaver DP, editors. Prog. Astronaut., vol. 159. Washington (DC): AIAA; 1992.

[6] Chen Sh, Doolen GD. Lattice Boltzmann methods for fluid flows. Ann Rev Fluid Mech 1998;30:3289–364.

[7] Lallemand P, Luo L. Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability. Phys Rev E 2000;61:6546–62.

[8] Lions JL. Contrôle optimal de systèmes gouvernés par des équations aux dérivées partielles. Paris: Dunod; 1986.

[9] Culioli JL. Introduction à l'Optimisation. Paris: Ellipses; 1994.

[10] Qian YH, d'Humières D, Lallemand P. Lattice BGK models for Navier–Stokes equation. Europhys Lett 1992;17:479–84.

[11] Ginzburg I, Adler P. Boundary flow condition analysis for the three-dimensional lattice Boltzmann model. J Phys II France 1994;4:191–214.

[12] Ginzburg I, d'Humières D. Local second order boundary methods for lattice Boltzmann models. J Stat Phys 1996;84:927–71.