Journal of Statistical Physics, Vol. 121, Nos. 1/2, October 2005 (© 2005)
DOI: 10.1007/s10955-005-7010-5

Analysis of Lattice Boltzmann Initialization Routines

Alfonso Caiazzo[1]

Received October 12, 2004; accepted June 15, 2005

LB simulations can be affected by the arising of initial layers due to an inconsistent initialization of the discrete LB populations. We present some previously proposed initialization routines built to overcome that problem; using the asymptotic expansion technique, we show how their features can be analyzed and, in some cases, how accuracy and computational efficiency can be improved.

KEY WORDS: lattice Boltzmann equation; initial layer; asymptotic analysis.

1. INTRODUCTION

One of the problems of kinetic methods simulating macroscopic equations is about how to set correctly the initial conditions, usually given in "macroscopic" form, in terms of kinetic variables. Inconsistent choices lead to initial layers that affect numerical simulations, for example reducing the theoretically possible accuracy of the schemes. Born with the same kinetic philosophy, even if evolved and modified (refs. 1 or 2 for detailed overviews), the lattice Boltzmann method (LBM) suffers of the same problem. Some results regarding the setting up of a lattice Boltzmann simulation, in order to avoid initial layers, will be presented. In Section 2 we introduce the initial layers problem together with some basic ideas of asymptotic analysis, providing a simple numerical example. In Section 3, a first initialization routine, presented in ref. 3, is discussed and analyzed, showing its main features; using the asymptotic analysis, we define some modified routines, that improve the previous one in accuracy and efficiency. Each modification step will be supported by numerical tests, based on the presented model. Finally, a special case that allows us to get fourth order initial pressure will be described.

[1] Fraunhofer ITWM, Gottlieb-Daimler-Str., 49, Kaiserslautern D-67663, Germany; e-mail: caiazzo@itwm.fhg.de

2. THE INITIAL LAYERS PROBLEM

The starting point is an initial value incompressible Navier–Stokes problem, with periodic boundary conditions, on a two-dimensional domain $\Omega \subset \mathbb{R}^2$:

$$\begin{aligned} \nabla \cdot \mathbf{u} &= 0, \\ \partial_t \mathbf{u} + \nabla p + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) &= \nu \nabla^2 \mathbf{u} + \mathbf{G}, \\ \mathbf{u}(t=0, \mathbf{x}) &= \mathbf{u}_0(\mathbf{x}) \end{aligned} \tag{1}$$

for a given $\mathbf{u}_0(\mathbf{x}) : \Omega \to \mathbb{R}^2$. Our aim is to solve it using the LBM. Employing a numerical scheme whose variables have a kinetic meaning, the right correspondence between these and the macroscopic quantities has to be set, whenever physical conditions, regarding $\mathbf{u}$ and p, are involved (in problem (1) provided by the initial velocity field $\mathbf{u}_0$). Incorrect coupling can produce *initial oscillatory layers*, with a pure numerical meaning, in the numerical solution.

2.1. Generalities About LBM

The lattice Boltzmann method, as a solver for the incompressible Navier–Stokes equations, was originally built up from the lattice gas cellular automata (LGCA) models (see refs. 4 and 5); however, the numerical schemes provided by the LBM can be also derived[6] from a discretization of a finite discrete-velocity model of the Boltzmann equation

$$\partial_t f_i + \mathbf{c}_i \cdot \nabla f_i = J_i(f), \quad i = 0, \dots, N \tag{2}$$

being $\mathbb{V} = \{\mathbf{c}_i\}_{i=0,\dots,N}$ the finite velocity set. The variable $f_i(t, \mathbf{x})$ represents the particle mass density distribution moving in the direction of $\mathbf{c}_i$ at time t and position $\mathbf{x}$; the functional $J_i(f)$ models the effects of collisions between particles. General theory and detailed derivation of the LBE from the LGCA and from the BE can be found, for example, in ref. 7. To recover the incompressible Navier–Stokes equations in the continuous limit (see refs. 7 and 8), Eq. (2) has to be discretized according to the *diffusive scaling* $\Delta t = \Delta x^2$. In what follows, we describe the LBM in terms of a *dimensionless lattice unities reference system*, where space and time unities are represented by grid size and time step, related by the diffusive scaling. Furthermore, the unity of measure of mass density is fixed setting the reference density of the flow equal to 1.

In the presented results, we will use the particular $D2Q9$ model, with a nine-velocities set $\mathbb{V} = \{\mathbf{c}_0, \dots, \mathbf{c}_8\}$ of two-dimensional vectors; for a more detailed description of such a model, and specific definitions of discrete velocity space, equilibrium distribution and related constants, we refer to ref. 7.

Calling h the physical grid spacing, the LB populations at time $t_n = nh^2$ and on position $\mathbf{x_j} = \mathbf{j}h$ are expressed by functions $\hat{f}_i(n, \mathbf{j}) : \mathbb{N} \times \mathbb{Z}^2 \cap \Omega \to [0, 1]$; the general iteration of the algorithm (LBM with BGK approximation) reads then

$$\hat{f}_i(n+1, \mathbf{j}+\mathbf{c}_i) = \hat{f}_i(n, \mathbf{j}) + \frac{1}{\tau}(f_i^{eq}(\hat{f}) - \hat{f}_i)(n, \mathbf{j}) + \hat{g}_i(n, \mathbf{j}). \tag{3}$$

The equilibrium distribution f^{eq} is function of $\hat{f}$, through the density $\hat{\rho} = \sum_i \hat{f}_i$ and the velocity $\hat{\mathbf{u}} = \sum_i \mathbf{c}_i \hat{f}_i$. We denote with H^{eq} the equilibrium as a function of $\hat{\rho}$ and $\hat{\mathbf{u}}$, composed of a linear ($H^{L(eq)}$) and a quadratic ($H^{Q(eq)}$) part. For the considered $D2Q9$ model:

$$H_i^{eq}(\rho, \mathbf{u}) = f_i^* \left(\rho + c_s^{-2} \mathbf{c}_i \cdot \mathbf{u} + \frac{c_s^{-4}}{2} \left(|\mathbf{c}_i \cdot \mathbf{u}|^2 - c_s^2 \mathbf{u}^2 \right) \right) = H_i^{L(eq)} + H_i^{Q(eq)} \tag{4}$$

(the lattice sound speed c_s and the weights f_i^* depend on the model); more general description of the equilibrium distribution and of its properties can be found, for example, in ref. 9. The last term in (3) is defined as

$$\hat{g}_i(n, \mathbf{j}) = h^3 f_i^* c_s^{-2} \mathbf{c}_i \cdot \mathbf{G}(t_n, \mathbf{x}_j) \tag{5}$$

and takes care of the force term $\mathbf{G}$ (rescaled by h^3 in lattice unities) in the Navier–Stokes equations (1). The relaxation time τ is related to a dimensionless viscosity through $\nu = c_s^2(\tau - 1/2)$.

2.2. Asymptotic Analysis of Periodic LBM

Assuming that the LB solution can be written in the form

$$\hat{f}_i(n, \mathbf{j}) = f_i^{(0)}(nh^2, \mathbf{j}h) + h f_i^{(1)}(nh^2, \mathbf{j}h) + h^2 f_i^{(2)}(nh^2, \mathbf{j}h) + \cdots \tag{6}$$

with coefficients $f_i^{(k)}$ sufficiently smooth and h-independents, we can derive explicitly the $f_i^{(k)}$ by inserting (6) into (3), Taylor expanding and sorting the orders in h; in the *periodic case*, we have (see ref. 8)

$$\begin{aligned} f_i^{(0)} &= f_i^*, \\ f_i^{(1)} &= f_i^* c_s^{-2} \mathbf{c}_i \cdot \mathbf{u}, \\ f_i^{(2)} &= f_i^* c_s^{-2} p + H_i^{Q(eq)}(\mathbf{u}, \mathbf{u}) - \tau f_i^* c_s^{-2} (\mathbf{c}_i \cdot \nabla) \mathbf{c}_i \cdot \mathbf{u}, \\ f_i^{(3)} &= f_i^* c_s^{-2} \mathbf{c}_i \cdot \mathbf{w} - \tau \left((\mathbf{c}_i \cdot \nabla) f_i^{(2)} + \frac{(\mathbf{c}_i \cdot \nabla)^2}{2} f_i^{(1)} + \partial_t f_i^{(1)} \right) + \tau g_i^{(3)}, \end{aligned} \tag{7}$$

where $\mathbf{u}$ and p solve the Navier–Stokes equations and the vector field $\mathbf{w}$ is solution of an inhomogeneous Oseen-type problem. The behavior of the numerical scheme is analyzed through a *truncated expansion*

$$\hat{F}_i = f_i^{(0)} + h f_i^{(1)} + h^2 f_i^{(2)} + h^3 f_i^{(3)}, \tag{8}$$

that can predict the LB solution up to order h^4. Combining Eq. (7), the pressure can be extracted from $\hat{f}$, defining

$$\hat{p} = c_s^2 \frac{\sum_i \hat{f}_i - 1}{h^2}, \tag{9}$$

it expresses a 0-average pressure, related to the higher order density fluctuations (see ref. 9), once removed the constant contribution of the incompressible density (equal to 1 in dimensionless unities). Using the (7) and the prediction (8), it can be shown (computing $\hat{\mathbf{u}}$ and $\hat{p}$ using $\hat{F}$ instead of $\hat{f}$, see ref. 8 for details) that $\hat{\mathbf{u}}$ and $\hat{p}$ are a second order accurate approximation of the Navier–Stokes solution. Moreover, the tensor $\mathbf{S}[\mathbf{u}] \equiv \nabla \mathbf{u} + \nabla \mathbf{u}^T$ (*viscous stress tensor* divided by ν) can be approximated, up to the second order, from the *nonequilibrium* part of $\hat{f}_i$:

$$\hat{\mathbf{S}}[\mathbf{u}] = -\frac{1}{\tau c_s^2 h^2} \sum_i \mathbf{c}_i \otimes \mathbf{c}_i \left(\hat{f}_i - f_i^{eq}(\hat{f}) \right). \tag{10}$$

2.2.1. Lattice Boltzmann Initial Conditions

Coming back to the problem (1), to set up our LB simulation, we have to fix the initial values of the discrete populations f_i, according to the initial macroscopic fields. We present here a list of possible choices, shortly describing them. Par. 2.2 contains some numerical example. The simplest way consists of initializing with *equilibrium values*

$$\hat{f}_i|_{t=0} = H_i^{eq}(1, h\mathbf{u}_0) \tag{11}$$

(the velocity $\mathbf{u}_0$ is scaled by h in lattice unities). In view of (9), a constant initial density $\rho_0 = 1$, gives rise to an initial pressure $p_0 = 0$, but, for general $\mathbf{u}_0$ and $\mathbf{G}_0$, the initial pressure should satisfy

$$\nabla^2 p_0 = -\nabla \cdot (\nabla \cdot (\mathbf{u}_0 \otimes \mathbf{u}_0)) + \nabla \cdot \mathbf{G}_0 \tag{12}$$

(taking the divergence of the Navier–Stokes equation). To include the initial pressure, we could use an *additional Poisson solver*, to get an estimate $\tilde{p}$ from Eq. (12) and define

$$\hat{f}_i|_{t=0} = H_i^{eq}(1 + h^2 c_s^{-2} \tilde{p}, h\mathbf{u}_0). \tag{13}$$

An initialization which leads to better results, including the initial tensor $\mathbf{S}[\mathbf{u}_0]$ was proposed in ref. 9; computing a *numerical approximation* $\tilde{\mathbf{S}}$ we can set

$$f_i = H_i^{eq}(1 + h^2 c_s^{-2} \tilde{p}, h\mathbf{u}_0) - h^2 \tau c_s^{-2} f_i^* \left(\mathbf{c}_i \otimes \mathbf{c}_i : \tilde{\mathbf{S}}\right). \tag{14}$$

Comparing the behavior of different initializations with the results of asymptotic analysis (7), we note that (14) sets correctly the initial values up to the second order part f_i.[2] At this point, it seems that to construct better initializations more and more expensive routines are required; however, the approach presented in ref. 7, analyzed and improved in the present paper, allows to achieve initialization (14) completely within the LB framework.

2.2.2. Test Problems

Our test problems in this article are based on the *periodic Taylor vortex field* on $\Omega = [0, 1] \times [0, 1]$:

$$\begin{aligned} &\mathbf{u}_{TV}(t, x, y) = \frac{1}{2\pi}(-\cos(2\pi x)\sin(2\pi y), \sin(2\pi x)\cos(2\pi y))\exp(-8\pi^2 \nu t), \\ &(\text{pressure } \; p_{TV}(t, x, y) = -\frac{1}{16\pi^2}(\cos(4\pi x) + \cos(4\pi y))\exp(-16\pi^2 \nu t)), \end{aligned} \tag{15}$$

as a solution of different problems:

- (NS): Navier–Stokes, with $\mathbf{G} = 0$;
- (ST): Stokes, adding a *non divergence-free* force $\mathbf{G} = \nabla p_{TV}$.

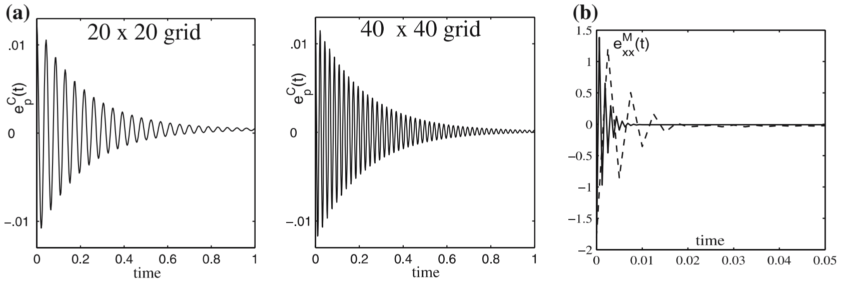

Fig. 1. Qualitative behavior of initial layers. **(a)** Initial layer in the error $e_p^C(t)$ in pressure in the central point $\mathbf{x}_C$ of the unit square (where p takes its maximum), initializing with (11); the exact initial value of pressure is ~ 0.012. Initial layer is of order 0. **(b)** Superimposed oscillatory initial layers in the error of **S[u]**; maximum amplitude oscillations for the component S_{xx} (in $\mathbf{x}_M = (0.25, 0.25)$) are shown, for $h = 0.05$ (dashed line) and $h = 0.025$ (solid) initializing with (13).

Figure 1a shows results of the LBM for the problem (NS), using the initial values (11); the initial discrepancy in pressure produces an initial layer, which does not vanish for $h \to 0$. With initialization (13), the initial layer is still present in **S[u]** (Fig. 1b). We do not see the claimed second order accuracy for p and **S[u]**, because, as remarked in the end of par. 2.1, the coefficients $f_i^{(2)}$ (same order of p and **S[u]**) have wrong value at time $t = 0$, if we use initialization (11) or (13). As a consequence, the hypothesis of smoothness of expansion coefficients is not satisfied by $f_i^{(2)}$ (there is a jump at $t = 0$), and the prediction (8) we derived might differ from numerical results. Initial layers in the order h^2 are removed using (14).

3. LB INITIALIZATION ROUTINES

The following algorithm has been proposed in ref. 3 to initialize LB algorithm according to (14).

It has the same structure as the classical LBM; only, in the collision step, the velocity is kept fixed and equal to $\mathbf{u}_0$ in the equilibrium function.

Algorithm 1.

```
$\rho^0 = 1$
do while $||\rho(n+1,\cdot) - \rho(n,\cdot)|| > \epsilon$ (fixed by tolerance criterion)
  collision: $\hat{f}_i^c(n,\mathbf{j}) = \hat{f}_i(n,\mathbf{j}) + \frac{1}{\tau}(H_i^{eq}(\rho(n,\mathbf{j}), h\mathbf{u}_0(\mathbf{j})) - \hat{f}_i(n,\mathbf{j})) + \hat{g}_i(0,\mathbf{j})$
  advection: $\hat{f}_i(n+1,\mathbf{j}+\mathbf{c}_i) = \hat{f}_i^c(n,\mathbf{j})$
  $\rho(n+1,\cdot) = \sum_i \hat{f}_i(n+1,\cdot)$
end
```

3.1. Asymptotic Analysis

To see how the algorithm acts on the populations, we apply an expansion of the form (6) to the previous procedure. Since in what follows the time t is not the “real” time (the algorithm is used only to initialize the populations, keeping the initial velocity and the initial force fixed), we will call it *pseudotime*, even if we indicate it with the letter t. The frozen quantities are denoted with subscript 0, like $\mathbf{u}_0$ or $\mathbf{G}_0$.

From definition (9), an analogous expansion can be derived for the pressure, which reads

$$\hat{p} = p + hp^{(3)} + h^2 p^{(4)} + h^3 p^{(5)} + \cdots \tag{16}$$

The expressions of the coefficients $f_i^{(k)}$ (up to the third order, compare with (7)) are

$$\begin{aligned} &f_i^{(0)} = f_i^*, \\ &f_i^{(1)} = f_i^* c_s^{-2} \mathbf{c}_i \cdot \mathbf{u}_0, \\ &f^{(2)} = f_i^* c_s^{-2} p + H_i^{Q(eq)}(\mathbf{u}_0, \mathbf{u}_0) - \tau f_i^* c_s^{-2} (\mathbf{c}_i \cdot \nabla) \mathbf{c}_i \cdot \mathbf{u}_0 \\ &f_i^{(3)} = f_i^* c_s^{-2} p^{(3)} - \tau (\mathbf{c}_i \cdot \nabla) f_i^{(2)} - \tau \frac{(\mathbf{c}_i \cdot \nabla)^2}{2} f_i^{(1)} + \tau g_{0,i}^{(3)} \end{aligned} \tag{17}$$

with p satisfying

$$\begin{aligned} &\nabla p + \nabla \cdot (\mathbf{u}_0 \otimes \mathbf{u}_0) = \nu \nabla^2 \mathbf{u}_0 + \mathbf{G}_0 - \frac{1}{\tau} \mathbf{w} \\ &c_s^{-2} \partial_t p + \nabla \cdot \mathbf{w} + \frac{1}{2} (\nabla^2 p + \nabla \cdot (\mathbf{u}_0 \otimes \mathbf{u}_0)) = 0 \end{aligned} . \tag{18}$$

The field $\mathbf{w}$, defined as the first order moment of the coefficient $f_i^{(3)}$, can be recovered from the first equation and inserted into the second, obtaining

$$\partial_t p = \nu (\nabla^2 p + \nabla \cdot (\nabla \cdot (\mathbf{u}_0 \otimes \mathbf{u}_0)) - \nabla \cdot \mathbf{G}_0) + \frac{c_s^2}{2} \nabla \cdot \mathbf{G}_0 \tag{19}$$

with initial condition $p|_{t=0} = 0$. Using an analogous procedure, we obtain for $p^{(3)}$ the PDE

$$\partial_t p^{(3)} = \nu \nabla^2 p^{(3)} \tag{20}$$

with $p^{(3)}|_{t=0} = 0$, that has solution $p^{(3)} \equiv 0$.

3.2. A LB Poisson Solver

Now we analyze more in detail Eq. (19). If $\nabla \cdot \mathbf{G}_0 = 0$, it shows that, at the steady state (in pseudotime), p solves the Poisson equation (12). Hence, we can extract a second order accurate pressure, since (as a consequence of (20), using the expansion (16)), $\hat{p} - p = O(h^2)$. On the other hand, the procedure does not work if the force has non zero divergence, because equation (19) is then different from (12). Figure 2a (right) exemplifies the problem. Looking at the pressure during the initialization algorithm for a (ST) problem (where an additional nondivergence-free force is present) we see an error in the asymptotic value increasing like ν^{-1} (this relationship is also explained by Eq. (19)). Fixing the viscosity and refining the grid (Fig. 2b), the error in the initial pressure (solution of (12)) is not reduced (slope ~ 0 in the double logarithmic plot); the procedure gives an inconsistent pressure.

To cure this anomaly we modify the collision step, replacing $\hat{g}_i$ with

$$g_i^{\nabla} \equiv h^3 c_s^{-2} f_i^* \mathbf{c}_i \cdot \mathbf{G} + h^4 f_i^* \frac{\nabla \cdot \mathbf{G}}{2}, \tag{21}$$

which produces a new term in (19) able to remove the undesired source. Results of this modified routine, shown in Fig. 2b, confirm that the corrected force g_i^{∇} allows to recover a second order accurate initial pressure (slope ~ 2) solving Eq. (12) only by using LB-type iterations.

Other aspects of (modified) Algorithm 1 follow from the analysis performed in par. 3.1. Actually, it is more than a LB Poisson solver, since it does not even require an approximation of $\nabla \mathbf{u}_0$ to set $f_i^{(2)}$ as initial-

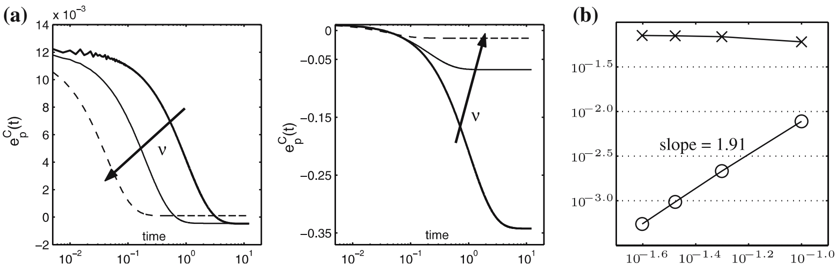

Fig. 2. **(a)** Algorithm 1 applied to vortex solution (20 × 20 grid), for viscosity $\nu_1 = 0.006$ (bold line), $\nu_2 = 5\nu_1 = 0.030$ (solid), $\nu_3 = 5\nu_2 = 0.15$ (dashed) (the arrow in the plots shows the increasing viscosity). Left: error $e_p^C(t)$ of central point pressure in *logarithmic* pseudotime during initialization for (NS) problem. Right: (ST) problem (with additional *non divergence-free* force); the error is increasing like ν^{-1}. **(b)** Double logarithmic plot of maximum error in initial pressure vs. grid size in (ST), with (○) and without (×) the corrected force term g^{∇}.

ization (14) does. However, we observe also that the pseudotime steps needed to reach the steady state of Eq. (19) are a function ($\sim 1/\nu$) of viscosity, i.e. of τ (it can be seen in Fig. 2a, showing the error in pressure approaching a steady value). Therefore, once guaranteed the accuracy in pressure for a general force field, our next aim is to see whether is it possible to reduce the computational effort to get it.

3.3. Accelerated Initialization Routines

We focus on a periodic box without boundaries. The idea is the following: the Poisson equation (12) does not depend on ν; hence, to have a faster procedure, it should be possible to run the algorithm with a higher, *faster*, viscosity (i.e. using a different τ). This allows the pressure to get closer to its limit in less pseudotime steps; unfortunately, the simple increasing of viscosity leads to a wrong initial tensor $\mathbf{S}[\mathbf{u}]$, in which we will still see a zeroth order initial layer (Figs. 3-right and 4a), due to a wrong initialization of the τ-depending term in $f_i^{(2)}$. Looking at the definition of the coefficients (17), we can derive a recipe to correct the initial populations in such a way to remove *completely* the error in $f_i^{(2)}$, even using a different value of τ. Practically, calling $\tilde\tau$ the new relaxation time and r the ratio $\tau/\tilde\tau$, we *isolate* (up to order h) the term we are interested in, subtracting from the output (of the accelerated routine) $\hat f_i^{\tilde\tau}$ the previous order terms, *reconstruct* "by hand" the correct initial second order,

$$\bar f_i^{(2)} = (1-r)\left(f_i^* c_s^{-2}\hat p + H_i^{Q(eq)}(\mathbf{u}_0,\mathbf{u}_0)\right) + r\left(\frac{f_i^{\tilde\tau} - f_i^* - h f_i^* c_s^{-2}\mathbf{c}_i\cdot\mathbf{u}_0}{h^2}\right) \tag{22}$$

and *define* the initial values

$$\bar f_i^{\tilde\tau} = f_i^* + h f_i^{(1)} + h^2 \bar f_i^{(2)} = (1-r)H_i^{eq}(1+h^2 c_s^{-2}\hat p, h\mathbf{u}_0) + r\hat f_i^{\tilde\tau}. \tag{23}$$

Algorithm 2.

```
Given initial data $\mathbf{u}_0$ and force $\mathbf{G}_0$
    compute $\nabla\cdot\mathbf{G}_0$ (at least first order accurately), $g^\nabla$
    run algorithm 1 with $\tau_1$
    compute pressure $p$ and equilibrium from $p$ and $\mathbf{u}_0$
    initialize LBM using (23)
```

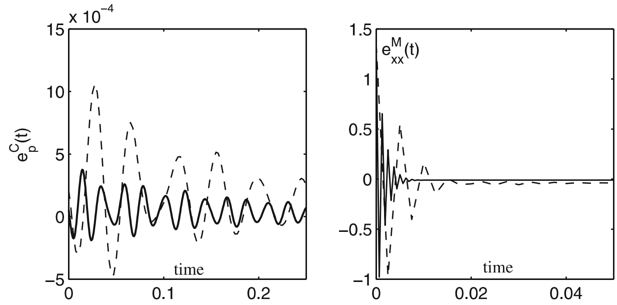

Fig. 3. Same simulation as in Fig.1, the error in p and S_{xx} are now shown after the modified viscosity initialization. It reduces the order of initial layer in pressure (left) but not in the tensor **S[u]** (right); dashed line: 20×20 grid, solid line: 40×40 grid.

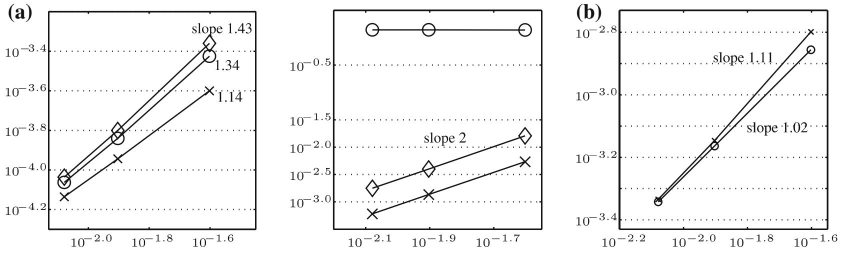

Fig. 4. **(a)** Double logarithmic plot of maximum error in initial pressure (left) and **S[u]** (right) for (NS), versus grid size (viscosity $\sim 3 \cdot 10^{-2}$); the curves are obtained with original-viscosity routine ($\times$), Algorithm 1 run with $\tau = 1$ ($\circ$), and accelerated routine, inclusive of correction ($\diamond$). **(b)** Error in pressure versus grid size for original ($\times$) and accelerated ($\circ$) initialization applied to (ST).

Using Eqs. (22) and (23), the difference between $\bar{f}^{\tilde{\tau}}$ and the correct initial population f can be explicitely written down as

$$f - \bar{f}^{\tilde{\tau}} = h^3\left(-r f^{(3),\tilde{\tau}} + f^{(3)}\right) + O(h^4). \tag{24}$$

This *accelerated initialization procedure* will then remove all the inconsistencies regarding initial conditions from the second order populations, reducing the computational time needed to initialize them. The following numerical tests (Fig. 4) compare results of the original initialization routine, with the accelerated one. The original viscosity is ~ 0.03, with $\tau = 0.59$; as a faster relaxation time we used $\tilde{\tau} = 1$, that allows also to simplify the implementation of the LB collision step. The tolerance criterion in Algorithm 2 is based on the difference between the pressure in two successive pseudotime iterations (related to an approximated $\partial_t p$) and the gain in

CPU time is about 65%. Initial layers in pressure and viscous stress tensor have been compared; we get first order accuracy for $\hat{p}$ and second order (only after correction (23)) for $\hat{\mathbf{S}}$, even if, for pressure, after the accelerated procedure the initial layer amplitude may be slightly bigger. It happens mainly because we modify, and do not correct, the $f_i^{(3)}$; we can explicitly write down the modification occurring in the third order coefficients (as given in (17)) running Algorithm 2, using the evaluation of error (24):

$$f_i^{(3),\tau} - \bar{f}_i^{(3),\tilde{\tau}} = E_i^{(3)} = (\tau - \tilde{\tau})\tau c_s^2 f_i^* (\mathbf{c}_i \cdot \nabla)^2 (\mathbf{c}_i \cdot \mathbf{u}_0). \tag{25}$$

This part is only responsible of the increasing of the amplitude (Fig. 4a) of the initial layer in pressure. This arises even using the original viscosity routine, because the expressions of $f_i^{(3),\tau}$ differ from the exact initial values given in (7); in particular, a term involving $\partial_t \mathbf{u}|_{t=0}$ is missed, affecting our prediction $\hat{F}_i$ from order h^3 (as explained in par. 2.1). Note that in the linear problem, original and accelerated routines lead to similar results, because the difference (25) vanishes, since it contains only quadratic terms (Fig. 4b).

Summarizing the theoretical and numerical results presented so far, we started analyzing an existing initialization algorithm from which a *modified collision* routine, able to initialize correctly LBM up to second order for a *general force field*, has been defined; for a special class of periodic boundary problems, we have proposed a *faster viscosity* routine characterized by a *viscosity-independent CPU-time* needed.

3.4. Special Viscosity

Performing a further step of asymptotic analysis, we can write the equation for the coefficient $p^{(4)}$ of expansion (16):

$$A(\tau)\partial_t p^{(4)} + B(\tau)\partial_t^2 p = \nabla^2 p^{(4)} + \frac{1}{\tau}\phi(\tau)\mathcal{F}(\mathbf{u}_0, \mathbf{G}_0) + \gamma(\tau)\nabla^2(\nabla \cdot \mathbf{G}_0) \tag{26}$$

with initial conditions $p^{(4)} = 0$. The operator $\mathcal{F}$ involves fourth and sixth order derivatives of the initial data. The function $\phi(\tau)$ is a second order polynomial with two real roots, $\tau_{\pm}^* = \frac{1}{2} \pm \frac{1}{\sqrt{6}}$. This means that if $\nabla \cdot \mathbf{G}_0 = 0$, with the special value $\tau_+^* \sim 0.9089$, $p^{(4)}$ vanishes at the steady state, giving a fourth order initial pressure (the coefficient $p^{(5)}$ behaves like the previous odd term $p^{(3)}$, see ref. 8) (see Fig. 5). Note, again, that the third order error has not been removed (it is just "invisible" in pressure) and that the

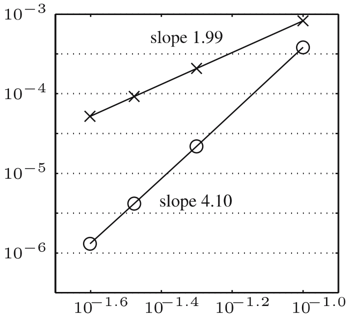

Fig. 5. Double logarithmic plot of maximum error in the initial pressure after initialization routine, for $\tau = \tau_{+}^{*}$ (○) and $\tau = 1$ (×).

fourth order pressure will, in general, become of second order once starting the actual LBE iteration.

ACKNOWLEDGMENTS

I would like to thank Prof. Michael Junk and Dr. Li-Shi Luo for the useful discussions we had.

REFERENCES

1. D. A. Wolf-Gladrow, *Lattice Gas Cellular Automata and Lattice Boltzmann Models* (Springer, Berlin, 2000).
2. D. Yu, R. Mei, L.-S. Luo, and W. Shyy, Viscous flow computations with the method of lattice Boltzmann equation, *Prog. Aerospace Sci.* **39**(5):329–367 (2003).
3. R. Mei, L.-S. Luo, and D. d'Humieres. Consistent Initial Conditions for Lattice Boltzmann Simulations. To appear on *Computers and Fluids*, 2005.
4. U. Frisch, B. Hasslacher, and Y. Pomeau, Lattice-gas automata for the Navier–Stokes equation, *Phys. Rev. Lett.* **56**:1505–1508 (1986).
5. G. R.McNamara, and G. Zanetti, Use of the Boltzmann equation to simulate lattice-gas automata, *Phys. Rev. Lett.* **61**:2332–2335 (1988).
6. X. He, and L.-S. Luo. Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation, *Phys. Rev. E* **56**:6811–6817 (1997).
7. S. Succi. *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond* (Oxford University Press, Oxford 2001).
8. M. Junk, A. Klar, L.-S. Luo. Theory of the lattice-Boltzmann method: Mathematical analysis of the lattice Boltzmann equation, to appear in *J. Comp. Phys.*, 2005.
9. P. A. Skordos, Initial and boundary conditions for the lattice Boltzmann method, *Phys. Rev. E* **48**:4823–4842 (1993).