Lattice Boltzmann model for coffee percolation

Luciano Misici, Silvia Palpacelli,

Riccardo Piergallini, Renato Vitolo

Dipartimento di Matematica e Informatica,
Università di Camerino,
via Madonna delle Carceri, 62032 Camerino, Italy
Phone: +39 0737 402549, E-mail: renato.vitolo@unicam.it

Abstract - A Lattice Boltzmann model is constructed for the simulation of water percolation through coffee. The model describes the dynamics of two fluids (water and air) through a layer of porous medium (coffee), where the porous medium may be eroded by the flow of water. Surface tension between the fluids is included, and a multi-relaxation scheme is used to improve numerical stability at low viscosities. Moreover, a probabilistic algorithm is included to model erosion, transport and deposition of coffee particles.

Keywords— **cellular automata, porous media, coffee percolation.**

I. INTRODUCTION

In this paper we describe a model for the simulation of water percolation through a coffee bed. The physical process of interest takes place inside a professional espresso machine (typically used in Italian bars) and runs as follows. The coffee is contained inside a cylindrical domain bounded by a bottom filter. The percolation consists of two phases. In the first (wetting) phase the water enters the domain from the inlet, which is placed on top of the cylinder, and falls down under the effect of gravitational force and of a low pressure gradient. Thereby it wets the coffee and pushes away the air. In the second phase a strong pressure gradient is applied to the water from above.

During this process (and particularly in the second phase), small particles of coffee can detach from larger grains and can be transported by the fluid. Eventually they can be deposited on other grains or on the bottom filter.

A lattice Boltzmann scheme is used to model the dynamics of the two fluids (water and air). We first extend the basic D3Q19 lattice Boltzmann model [QIA 92] to the case of two fluids. Our approach is similar to that of [INA 04], in that we use an indicator function to model fluid interface dynamics [MIS 06]. On top of this, a surface tension term for the two fluids is introduced in the model, as in [KEH 02]. A multi-relaxation time scheme [D'H 02] is used to improve numerical stability at low viscosity.

Moreover, we take into account transport of coffee particles. Coffee grains are modeled by lattice cells which are allowed either to dissolve completely or to 'lose' subparticles in the water, according to a probabilistic rule. The subparticles in turn may cluster up into a new solid grain. The algorithm we developed for particle transport is based on ideas of [CHO 00].

II. LATTICE BOLTZMANN TWO-FLUID MODEL

Denote by $\boldsymbol{x} \equiv (x, y, z)$ the position of a node on a cubic 3D lattice, and by t the time. In Lattice Boltzmann models, velocity space is discretized by selecting only a finite number l of possible velocities $\boldsymbol{c}_i$ with $i = 0, ..., l-1$. Each lattice node is linked by the $\boldsymbol{c}_i$'s to some of its nearest neighbors. A mass distribution function $f_i(\boldsymbol{x}, t)$ is associated to each of the links $\boldsymbol{c}_i$ and represents the mass fraction that is moving with velocity $\boldsymbol{c}_i$ from the node $\boldsymbol{x}$ at time t.

Flow density $\rho(\boldsymbol{x}, t)$ and velocity $\boldsymbol{v}(\boldsymbol{x}, t)$, are computed from the $f_i(\boldsymbol{x}, t)$ as follows:

$$\rho(\boldsymbol{x}, t) = \sum_{i=0}^{l-1} f_i(\boldsymbol{x}, t), \tag{1}$$

$$\rho(\boldsymbol{x}, t)\boldsymbol{v}(\boldsymbol{x}, t) = \sum_{i=0}^{l-1} \boldsymbol{c}_i f_i(\boldsymbol{x}, t). \tag{2}$$

The model considered here is the D3Q19, where each node has $l = 19$ links (see [QIA 92] for notation and terminology). In particular, velocities $\boldsymbol{c}_i$ are given by:

- the node self:
$\boldsymbol{c}_0 = (0, 0, 0)$,
- neighbors at distance 1:
$\boldsymbol{c}_{1,2} = (\pm 1, 0, 0)$,
$\boldsymbol{c}_{3,4} = (0, \pm 1, 0)$,
$\boldsymbol{c}_{5,6} = (0, 0, \pm 1)$,
- neighbors at distance $\sqrt{2}$:
$\boldsymbol{c}_{7,8,9,10} = (\pm 1, \pm 1, 0)$,
$\boldsymbol{c}_{11,12,13,14} = (\pm 1, 0, \pm 1)$,
$\boldsymbol{c}_{15,16,17,18} = (0, \pm 1, \pm 1)$.

We now sketch a description of our two-fluid model, see [MIS 06] for details and theoretical discussion. The presence of two fluids (*red* and *blue* fluid) in our model is characterized by two distribution functions: $f_i(\boldsymbol{x}, t)$, $i = 0, \ldots, l-1$ is the usual mass distribution, and $\alpha_i(\boldsymbol{x}, t)$ is a 'color' distribution from which an indicator function

$\phi(\boldsymbol{x},t)$ is computed as

$$\phi(\boldsymbol{x},t) = \begin{cases} 1 & \text{if } \alpha(\boldsymbol{x},t) \geq 1, \\ 0 & \text{if } \alpha(\boldsymbol{x},t) \leq 0, \\ \frac{\sin((\alpha(\boldsymbol{x},t)-\frac{1}{2})\pi)+1}{2} & \text{otherwise,} \end{cases} \quad (3)$$

where $\alpha(\boldsymbol{x},t) = \sum_{i=0}^{l-1} \alpha_i(\boldsymbol{x},t)$. Red and blue nodes are those for which $\phi(\boldsymbol{x},t) = 1$ and $\phi(\boldsymbol{x},t) = 0$, respectively. On fluid interface nodes we have $0 < \phi(\boldsymbol{x},t) < 1$.
The $\alpha_i(\boldsymbol{x},t)$ evolve according to the LBGK equation

$$\alpha_i(\boldsymbol{x}+\boldsymbol{c}_i, t+1) - \alpha_i(\boldsymbol{x},t) = \\ -\omega_\alpha(\alpha_i(\boldsymbol{x},t) - \alpha_i^{eq}(\boldsymbol{x},t)), \quad (4)$$

where ω_α is a relaxation parameter, $\boldsymbol{c}_i$ are the D3Q19 lattice velocities and α_i^{eq} are the equilibrium functions given by

$$\alpha_i^{eq}(\boldsymbol{x},t) = \frac{\alpha(\boldsymbol{x},t) f_i(\boldsymbol{x},t)}{\rho}.$$

By this choice of the equilibrium functions a continuity-like equation is obtained for α in the macroscopic limit, see [MIS 06] for an explanation.
The evolution of the $f_i(\boldsymbol{x},t)$ is performed by a lattice Boltzmann equation with either single or multi-relaxation time (SRT and MRT, respectively, see [D'H 02]). The relaxation parameter $\omega(\boldsymbol{x},t)$ determining the kinematic viscosity of the fluid is set for each node and at each time step depending on the value of $\phi(\boldsymbol{x},t)$:

$$\omega(\boldsymbol{x},t) = \omega_r \phi(\boldsymbol{x},t) + \omega_b(1-\phi(\boldsymbol{x},t)),$$

where ω_r and ω_b denote the relaxation parameters of each fluid. In the SRT case, the LBGK equation is used for $f_i(\boldsymbol{x},t)$:

$$f_i(\boldsymbol{x}+\boldsymbol{c}_i, t+1) - f_i(\boldsymbol{x},t) = \\ -\omega(\boldsymbol{x},t)(f_i(\boldsymbol{x},t) - f_i^{eq}(\boldsymbol{x},t)). \quad (5)$$

Here the equilibrium distribution functions are

$$f_i^{eq}(\boldsymbol{x},t) = W_i \rho \left[1 + 3(\boldsymbol{c}_i \cdot \boldsymbol{v}) + \frac{9}{2}(\boldsymbol{c}_i \cdot \boldsymbol{v})^2 - \frac{3}{2}\boldsymbol{v}^2\right], \quad (6)$$

for $i = 0, \ldots, l-1$, where $\rho = \rho(\boldsymbol{x},t)$ and $\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{x},t)$ are computed as in (1) and (2) respectively, whereas the weights W_i depend on the lattice. For the D3Q19 model we have

$$W_i = \begin{cases} 1/3, & i = 0, \\ 1/18, & i = 1, ..., 6, \\ 1/36, & i = 7, ..., 18. \end{cases}$$

From the kinetic equation (5) and with the above expression for the equilibrium distributions, if a region is occupied by only one fluid, using multi-scale analysis one can recover the Navier–Stokes equations with the usual [QIA 92] relation between the kinematic viscosity and the relaxation parameter (either ω_r or ω_b) of the fluid:

$$\nu_{r,b} = \frac{1}{c_s^2}\left(\frac{1}{\omega_{r,b}} - \frac{1}{2}\right),$$

where c_s^2 is the lattice sound speed (for the D3Q19 model $c_s = 1/\sqrt{3}$).
Furthermore, a surface tension term is introduced (like in [KEH 02]), whose actual form depends on the relaxation time scheme adopted. In the SRT scheme, this term is added to the equilibrium distributions $f_i^{eq}(\boldsymbol{x},t)$ and has the form

$$S_i = W_i \frac{9}{2}\sigma\left(\frac{F_a F_b}{\|F\|} - \delta_{ab}\|F\|\right)\left(c_{ia}c_{ib} - \frac{\delta_{ab}}{3}\right),$$

where Latin indices represent the spatial coordinates and summation over a and b is implicitly assumed. The parameter σ measures the surface tension strength, while F is the gradient of ϕ.
In the MRT case the surface tension term is added to the equilibrium functions of certain linear combinations (called moments) of the $f_i(\boldsymbol{x},t)$. These moments are computed from the $f_i(\boldsymbol{x},t)$ according to the rules described in [D'H 02].
In the present setting red and blue fluid are water and air, respectively. In particular, we have:

- if $\phi(\boldsymbol{x},t) = 1$, then $\boldsymbol{x}$ is occupied by water,
- if $\phi(\boldsymbol{x},t) = 0$, then $\boldsymbol{x}$ is occupied by air,
- if $0 < \phi(\boldsymbol{x},t) < 1$, then $\boldsymbol{x}$ is on the interface between water and air.

III. A CELLULAR AUTOMATA FOR EROSION, TRANSPORT AND DEPOSITION OF SUBPARTICLES

The coffee is modeled as a randomically generated porous medium, consisting of solid nodes. The remaining (fluid) nodes evolve by the rules described in the previous section. In the lattice Boltzmann part of our algorithm, coffee nodes obey to the same bounce-back rule as the solid nodes on the walls of the cylinder.
In order to study the interaction between the water flow and the coffee, we allow the coffee nodes to evolve as well, according to a three-stage algorithm. All coffee nodes initially contain a fixed number of subparticles. The water flow may erode subparticles away from a coffee node, it may transport subparticles, and it may deposit them onto another coffee node or over the bottom filter:

Erosion: if one of the fluid nodes that are neighbors of a coffee node is *wet* (*i.e.*, partly occupied by water), a number of subparticles may jump from the coffee node to the wet node; a small fraction of coffee nodes may dissolve completely, thereby becoming fluid nodes;

Transport: subparticles contained inside a wet node travel along with the water flow, with the same velocity of the water contained in the node;

Deposition: if the velocity of a wet node is directed towards a neighboring solid node, then the subparticles (possibly) contained in the wet node may jump to the coffee node; moreover, if a sufficient number of subparticles accumulates inside a wet node, the latter turns into a coffee node.

In the next three sections, each of the above three stages is described in more detail.

A. Erosion

Roughly speaking, the erosion algorithm is performed on a coffee node only if the neighboring fluid nodes contain a sufficient percentage of water. To be more precise, recall that, if $\boldsymbol{y}$ is a fluid node, then $\phi(\boldsymbol{y}, t)$ is the 'color' function indicating whether node $\boldsymbol{y}$ is predominantly occupied by air or water (compare (3)). Then a fluid node is declared *wet* if $\phi(\boldsymbol{y}, t)$ is larger than a fixed threshold ϕ_{wet}.
Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be two neighboring nodes, where $\boldsymbol{x}$ is a coffee and $\boldsymbol{y}$ is a wet node. In few words, the erosion algorithm consists in examining the velocity $\boldsymbol{v}$ of the fluid inside $\boldsymbol{y}$: if $\boldsymbol{v}$ is directed towards $\boldsymbol{x}$ then the water flow may detach a number of subparticles from $\boldsymbol{x}$ and bring them to $\boldsymbol{y}$, according to a probabilistic rule. Subparticles can not move from solid to solid nodes.
Now, the details of the algorithm are given. Denote by $n(\boldsymbol{x}, t)$ the number of subparticle located in the node $\boldsymbol{x}$ (be it solid or fluid) at time t, and by $s(\boldsymbol{x}, t)$ the indicator

$$s(\boldsymbol{x}, t) = \begin{cases} 1 & \text{if } \boldsymbol{x} \text{ is a fluid node at time } t \\ 0 & \text{if } \boldsymbol{x} \text{ is a solid node at time } t. \end{cases}$$

Let N be the maximum number of subparticles that can be contained in a node. At the initial step we fix $n(\boldsymbol{x}, 0) = N$ for coffee nodes, while $n(\boldsymbol{x}, 0) = 0$ for fluid nodes.
Let $\boldsymbol{x}$ be a coffee node. Notice that $\boldsymbol{x}$ cannot be wet itself, therefore we resort to estimating the average water content $\phi_{av}(\boldsymbol{x}, t)$ of the neighboring fluid nodes:

$$\phi_{av}(\boldsymbol{x}, t) = \frac{\sum_{i=1}^{18} s(\boldsymbol{x} + \boldsymbol{c}_i)\phi(\boldsymbol{x} + \boldsymbol{c}_i)}{\sum_{i=1}^{18} s(\boldsymbol{x} + \boldsymbol{c}_i)}.$$

The erosion process is performed on coffee node $\boldsymbol{x}$ only if the neighboring fluid nodes are wet in average, *i.e.*, if $\phi_{av}(\boldsymbol{x}, t) > \phi_{wet}$.

We define two classes of coffee nodes: those that may dissolve completely in the water, thereby becoming fluid nodes, and those that may lose up to a fixed maximum number N_{er} of subparticles. If a node of the latter type has lost all of the N_{er} subparticles, then the erosion algorithm is no longer applied to it, unless some subparticles are subsequently deposited on it by the deposition algorithm.
The erosion process is modeled by associating a probability $p^{er}(\boldsymbol{x}, t)$ to each coffee node $\boldsymbol{x}$. Initially, $p^{er}(\boldsymbol{x}, t)$ is set to 1 and it is decreased each time the minimum number of subparticles inside a non-fluidizable coffee node is reached. The following rule is used: the probability $p_i^{er}(\boldsymbol{x}, t)$ for coffee node $\boldsymbol{x}$ to release subparticles to fluid node $\boldsymbol{x} + \boldsymbol{c}_i$, with $i \in \{1, \ldots, 18\}$, is given by:

$$p_i^{er}(\boldsymbol{x}, t) = \max\left(0, s(\boldsymbol{x} + \boldsymbol{c}_i, t)p^{er}(\boldsymbol{x}, t)\frac{\boldsymbol{v}(\boldsymbol{x} + \boldsymbol{c}_i, t) \cdot \boldsymbol{c}_j}{\|\boldsymbol{c}_j\|}\right), \quad (7)$$

where $\boldsymbol{c}_j$ is the direction opposite to $\boldsymbol{c}_i$.
Equation (7) means that the fluid in node $\boldsymbol{x} + \boldsymbol{c}_i$ is able to detach subparticles from node $\boldsymbol{x}$ only if the fluid velocity in $\boldsymbol{x} + \boldsymbol{c}_i$ has a positive projection on $\boldsymbol{c}_j$ (the latter is the direction from $\boldsymbol{x} + \boldsymbol{c}_i$ to $\boldsymbol{x}$).
The number $n_i^{er}(\boldsymbol{x}, t)$ of subparticles that may jump onto $\boldsymbol{x} + \boldsymbol{c}_i$ (with probability $p_i^{er}(\boldsymbol{x}, t)$) is proportional to the projection of $\boldsymbol{v}(\boldsymbol{x} + \boldsymbol{c}_i, t)$ along $\boldsymbol{c}_j$:

$$n_i^{er}(\boldsymbol{x}, t) = \frac{\boldsymbol{v}(\boldsymbol{x} + \boldsymbol{c}_i, t) \cdot \boldsymbol{c}_j / \|\boldsymbol{c}_j\|}{v_{max}} N_{max}(\boldsymbol{x}).$$

Here v_{max} is a constant representing the largest modulus of the velocities involved in the problem, whereas $N_{max}(\boldsymbol{x})$ is the maximum number of subparticles that may be detached from the node. The constant $N_{max}(\boldsymbol{x})$ is fixed at the beginning, depending on whether $\boldsymbol{x}$ is fluidizable or not:

Fluidizable: these coffee nodes can dissolve completely, thereby becoming fluid. Therefore,

$$N_{max}(\boldsymbol{x}) = N.$$

Fluidizable nodes are a fixed percentage of the total number of coffee nodes.

Non–fluidizable: these are allowed to lose up to N_{er} subparticles, hence we set

$$N_{max}(\boldsymbol{x}) = N_{er}.$$

Fluidizable coffee nodes turn into fluid nodes whenever the number of subparticles contained in them decreases below a threshold value N_f. The reverse process is also allowed to occur by the deposition algorithm, see below for details.

B. Transport

We now describe how the eroded subparticles are transported by the water flow. The transport algorithm is essentially the same proposed in [CHO 00], with minor modifications to adapt it to coffee percolation.

Consider a subparticle located on a fluid node $\boldsymbol{x}$. In one time step the subparticle should ideally jump to position $\boldsymbol{x}+\boldsymbol{v}(\boldsymbol{x},t)$, where $\boldsymbol{v}(\boldsymbol{x},t)$ again denotes the fluid velocity. However, the 'landing point' $\boldsymbol{x}+\boldsymbol{v}(\boldsymbol{x},t)$ typically does not coincide with a lattice node. Enforcing this by discretizing the landing point on the lattice would yield physically uncorrect transport velocities of the subparticles. Therefore we resort to a probabilistic rule: subparticles may jump to any of the neighboring fluid nodes with a probability which is proportional to $\boldsymbol{v}(\boldsymbol{x},t)$. Moreover, the rule is designed in such a way that subparticles can move to a fluid node only if the latter contains some water.

The algorithm details follow. Given $\boldsymbol{v}(\boldsymbol{x},t)$ we need to compute the destination node for each of the $n(\boldsymbol{x},t)$ subparticles located in $\boldsymbol{x}$. Here, subparticles can jump to *any* of the 27 nearest neighbors of $\boldsymbol{x}$: they are not bound to move on the 19 directions $\boldsymbol{c}_i$, $i=0,\ldots,18$, of the D3Q19. Hence, also neighbors at distance $\sqrt{3}$ are involved, pointed to by the additional directions

$$\boldsymbol{c}_{19,20,21,22,23,24,25,26} = (\pm 1, \pm 1, \pm 1).$$

Denote by $p_i(\boldsymbol{x},t)$ the probability that a subparticle in $\boldsymbol{x}$ moves to neighbor $\boldsymbol{x}+\boldsymbol{c}_i$ at time t, where $i=0,\ldots,26$. At first, $p_i(\boldsymbol{x},t)$ is computed only for $i=1,\ldots,6$ depending on the projection of flow velocity $\boldsymbol{v}(\boldsymbol{x},t)$ on direction $\boldsymbol{c}_i$ and on the *average water content* in the direction $\boldsymbol{c}_i$. The latter is defined as a weighted average of $\phi(\boldsymbol{y},t)$ where $\boldsymbol{y}$ ranges in all the fluid nodes $(\boldsymbol{x}+\boldsymbol{c}_j)$ such that $\boldsymbol{c}_i \cdot \boldsymbol{c}_j > 0$:

$$\phi_i(\boldsymbol{x},t) = k_1 \phi(\boldsymbol{x}+\boldsymbol{c}_i,t) + k_2 \sum_{j=7}^{18} \phi(\boldsymbol{x}+\boldsymbol{c}_j,t)(\boldsymbol{c}_i \cdot \boldsymbol{c}_j) + k_3 \sum_{j=19}^{26} \phi(\boldsymbol{x}+\boldsymbol{c}_j,t)(\boldsymbol{c}_i \cdot \boldsymbol{c}_j),$$

where $k_1 = 4/9$, $k_2 = 1/9$ and $k_3 = 1/36$. In this way, nearest neighbors give larger contributions. Hence, for $i = 1,\ldots,6$, $p_i(\boldsymbol{x},t)$ is computed as

$$p_i(\boldsymbol{x},t) = \phi_i(\boldsymbol{x},t)\max(0, \boldsymbol{v}(\boldsymbol{x},t)\cdot \boldsymbol{c}_i).$$

The remaining probabilities $p_i(\boldsymbol{x},t)$ with $i=7,\ldots,26$ are defined as linear combinations of the $p_i(\boldsymbol{x},t)$'s with $i=1,\ldots,6$. In practice, they are not explicitly computed and the algorithm runs as follows. For each of the $n(\boldsymbol{x},t)$ subparticles contained in $\boldsymbol{x}$ a boolean variable $\mu_i^k(\boldsymbol{x},t)$, $k=1,\ldots,n(\boldsymbol{x},t)$, is computed:

$$\mu_i^k(\boldsymbol{x},t) = \begin{cases} 1 & \text{with probability } p_i(\boldsymbol{x},t) \\ 0 & \text{with probability } 1-p_i(\boldsymbol{x},t), \end{cases}$$

with $i=1,\ldots,6$. Then the destination node of subparticle k retained in node $\boldsymbol{x}$ at time t is given by

$$\boldsymbol{x} + \sum_{i=1}^{6} \mu_i^k(\boldsymbol{x},t)\boldsymbol{c}_i.$$

The transport algorithm also interacts with the deposition scheme. As a result, the transport procedure is not necessarily applied to each of the $n(\boldsymbol{x},t)$ subparticles located in $\boldsymbol{x}$ at time t. In particular, in certain situations the subparticles can not move to one or more of the destination nodes, see the next section for details. In this case, a fraction of these 'blocked' subparticles may accumulate in $\boldsymbol{x}$, preluding to the creation of a new solid node.

The above process is implemented by defining a variable $p_m(\boldsymbol{x},t) \in [0,1]$, which indicates the fraction of subparticles that are allowed to move. Specifically, the transport algorithm is executed on $n(\boldsymbol{x},t)p_m(\boldsymbol{x},t)$ subparticles. Initially, $p_m(\boldsymbol{x},t)$ is set to 1 for all fluid nodes. Then at each time step $p_m(\boldsymbol{x},t)$ is set to a value such that $1-p_m(\boldsymbol{x},t)$ is proportional to the number of 'blocked' subparticles.

C. Deposition

During its motion a subparticle may jump onto a solid node. This can only happen if the destination node is not *full* of subparticles, *i.e.*, if it contains less than N subparticles (due to the erosion mechanism). If the destination node is full, the subparticle remains in the fluid node and starts to 'accumulate' there. This opens the possibility for the formation of a new coffee node.

Let $\boldsymbol{x}$ be the fluid node from which subparticles are moving and $\boldsymbol{y}$ their solid destination node. We distinguish two cases:

The solid node is not full: $n(\boldsymbol{y},t) < N$. All incoming subparticles are added to the node $\boldsymbol{y}$.

The solid node is full: $n(\boldsymbol{y},t) = N$. In this case incoming subparticles can not be deposited on $\boldsymbol{y}$, so they remain in $\boldsymbol{x}$. We keep track of their deposition in $\boldsymbol{x}$ by adding their number to a variable $n_{acc}(\boldsymbol{x},t)$ (initially set to 0 for all nodes). This variable is then used to decrease the fraction of moving subparticles of $\boldsymbol{x}$. Indeed the percentage $p_m(\boldsymbol{x},t)$ of subparticles that can be transported is computed as:

$$p_m(\boldsymbol{x},t) = \max\left(0,\ 1 - \frac{n_{acc}(\boldsymbol{x},t)}{N}\right).$$

In this way, the larger is the number of accumulated subparticles in $\boldsymbol{x}$, the harder is, for subparticles in $\boldsymbol{x}$, to move.

When the number of subparticles retained by a fluid node exceeds a given threshold value N_s, the fluid node becomes a coffee node.

D. Interaction between subparticles and fluid models

In the present setting, the subparticle component of the model (erosion, transport and deposition algorithm) affects the fluid dynamics only in one way: because of the induced change in the domain configuration. In particular, fluid nodes can turn into coffee nodes and also a fixed percentage of coffee nodes can become fluid.
We fix threshold values N_f and N_s (see at the end of the previous section and of Sec. III-A). The first is the number of subparticles below which a fluidizable coffee node turns fluid; the second is the number of subparticles a fluid node has to exceed in order to turn solid.

From fluid to solid: consider a fluid node $\boldsymbol{x}$. If $n(\boldsymbol{x}, t) > N_s$, then $\boldsymbol{x}$ becomes a coffee node and the evolution rule for Lattice Boltzmann algorithm is switched to bounce back.

From solid to fluid: Let $\boldsymbol{x}$ be a fluidizable coffee node. If $n(\boldsymbol{x}, t) < N_f$ then $\boldsymbol{x}$ becomes a fluid node. This means that the kinetic equation (5) (or the corresponding MRT version) is subsequently used to update the mass distribution functions f_i and the (4) to update the color distribution functions α_i. Mass distribution functions f_i are set to the value $f_i^{eq}(\boldsymbol{x}, t)$ in (6) with $\rho = \rho_0$ (the initial value of ρ in all of the domain) and $\boldsymbol{v} = (0, 0, 0)$. Similarly, the color distribution functions α_i are set to $\alpha_i^{eq}(\alpha)$ with $\alpha = \phi_{av}$.

Cycle of a non–fluidizable solid node: Consider a non–fluidizable coffe node $\boldsymbol{x}$. Initially it contains N subparticles and it may lose no more than N_{er} subparticles by the erosion procedure, with $p^{er}(\boldsymbol{x}, 0) = 1$. If the mininum number of subparticles $N - N_{er}$ is reached, $\boldsymbol{x}$ remains a coffee node (*i. e.* it evolves according to the bounce back rule) but it does not execute the erosion procedure. If other subparticles deposit onto it ($n(\boldsymbol{x}, t) > N - N_{er}$) it may be eroded again, but the probability $p^{er}(\boldsymbol{x}, t)$ is halved each time the minimum number of subparticles $N - N_{er}$ is reached.

We remark that the presence of subparticles in the water flow (transport algorithm) does not affect the viscosity of the fluid in the present setting. This question will be addressed in future research, by using experimental measurements of the viscosity alteration.

IV. EXPERIMENTAL MEASUREMENTS, AND PARAMETER SETTING FOR THE SIMULATIONS

To prepare a 'good' espresso is not an easy task. Many variables can affect the quality of the coffee, such as water pressure and temperature, the amount of ground coffee, how hard it is pressed and (not less important) the practice of the barman. Evidently some of these parameters cannot be controlled in a scientific way, but, for those that can, a few optimal conditions have been empirically determined:

water temperature : inside the machine it should be about 88°C;

pressure : it should be a function of time with profile as shown in Fig. 1;

quantity of coffee : 7 g;

duration : the whole process should take 30 seconds;

production : at the end, we should have about 25 ml of coffee in the cup.

In the rest of this section we report data and results of experimental measurements concerning espresso machines, performed at the Nuova Simonelli company.

The pressure profile in Fig. 1 highlights two stages in the process: in the first, which is called *pre–brewing* (or wetting) phase, water pressure is set to 1 bar and remains at this level during 4 seconds, then, in the second phase, it increases linearly up to 9 bar in about 6 seconds and keeps this value until the end of the process.

The pre–brewing phase is very important in order to obtain a good coffee. Indeed, the water, falling slowly through the coffee, melts the essential oils and other subtances that give taste and aroma. During this phase, however, not all of the coffee bed is wet by the water. It has been measured that the percentage of coffee reached by the water is bounded between 60% and 64%.

It has been experimentally measured that the porosity of a coffee bed typically is about 60% and that the imposed pressure at inlet induces a corresponding water velocity at inlet, which initially is $v_1 = 0.011\ m/s$ and then it increases up to $v_2 = 0.037\ m/s$, see Table I.

Commercial filters used in espresso machines have two shapes (cylinder and cone frustum) and two sizes (one- or two-cups filters). We only consider cylindrical filters with radius $R = 2.1$ cm and $R = 3.0$ cm for single and double coffee respectively, whereas height $h = 0.9$ cm in both cases. The bottom filter consists of a metal plate with circular holes that allow the coffe to exit and fall in the cup. The number and dimensions of holes may vary, depending on the filter sizes and shapes described above.

Experimentally estimated parameters, such as water and air viscosity, inlet velocity and so on, are given in Table I. The corresponding Lattice Boltzmann parameters used in the numerical simulations (see next section) are reported in Table II.

For what concerns particle erosion, transport and deposition the parameters used in the Lattice Boltzmann simulations are reported in Table III, where P_f is the percentage of fluidizable coffee nodes with respect to the total number of coffee nodes.

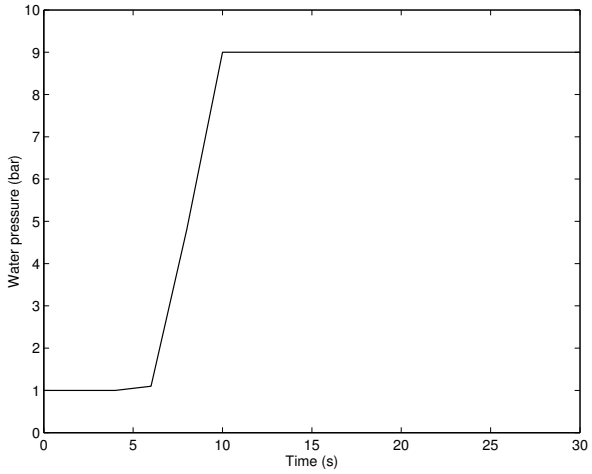

Fig. 1. Pressure profile at inlet as a function of time.

TABLE I
EXPERIMENTAL PARAMETERS IN PHYSICAL UNITS.

ν_a	ν_w	v_1	v_2	σ	T
(m^2/s)	(m^2/s)	(N/m)	(m/s)	(m/s)	(s)
17×10^{-6}	10^{-6}	0.011	0.037	0.067	30

V. NUMERICAL SIMULATION RESULTS

Two Lattice Boltzmann simulations were performed with the parameter setting described in the previous section, by considering two filter types, for single and double cup. The spatial discretization is the same in both simulations, resulting in a subdivision of the radius into 47 and 67 lattice units for single and double filter respectively. At the bottom of the domain ($z = 0$), squared holes are placed. The number and dimensions of the holes are choosen in agreement with the filter type examined.

With the above setting our model gives a good prediction of the interface motion. Indeed, during the pre–brewing phase we get a 63% of coffee bed reached by the water (see Fig. 2), in agreement with the experimental data.

Moreover, the volume of coffee coming out from the filter as a function of time has been experimentally measured. In order to compare these measurements with the values predicted by the model, we need to compute the coffee flow throught the bottom filter:

$$Q(t) = \int_S [\rho(\boldsymbol{x},t)\boldsymbol{v}(\boldsymbol{x},t)\cdot\boldsymbol{n}]\,ds \simeq -\sum_{k=1}^{M} \rho(\boldsymbol{x}_k,t)v_z(\boldsymbol{x}_k,t)\delta x^2.$$

Here S is the portion of the bottom filter occupied by the holes and $\boldsymbol{n}$ is the unit outward normal vector to the surface S. In the discrete approximation of $Q(t)$, M is the total number of non–solid lattice nodes placed at the bottom of the filter and δx is the minimal physical distance between two adjacent lattice nodes.

In Fig. 3 and Fig. 4 a comparison between experimental and model results is shown for single and double filter respectively. In particular, asterisks mark averages on a set of measurements, the vertical bars show the minimum and maximum measured values and solid line denotes the results of the corresponding Lattice Boltzmann simulation.

Concerning erosion, transport, and deposition of subparticles, a suitable experimental variable that could be measured is the variation of the 'coffee grain concentration', *i.e.*, the volume fraction occupied by the coffee grains inside horizontal slices of the coffee bed, which is a function of height (distance from the bottom of the filter). Initially, coffee concetration is almost uniform, but the water flow tends to transport coffee particles toward the bottom of the filter causing an accumulation. Accurate experiments to determine the magnitude of this variation are still in course. However, preliminary results indicate the formation of a rather compact layer next to the bottom of the filter, while the coffee concentration is lower near the top.

In Fig. 5 simulation results for a single cup filter are shown. Three layers, of thickness 3 mm each, are considered and the variation of coffee concentration in each layer is computed as function of time. In the case of a double cup filter

TABLE II
PARAMETERS, IN LATTICE UNITS, USED IN THE SIMULATIONS.

ω_a	ω_w	v_1	v_2	σ	T	ω_α
1.991	1.859	0.00367	0.0123	1.6×10^{-4}	2×10^5	1.95

TABLE III
PARAMETER SETTING FOR THE SUBPARTICLES ALGORITHM.

N	N_{er}	P_f	N_s	N_f
100	10	10%	75	25

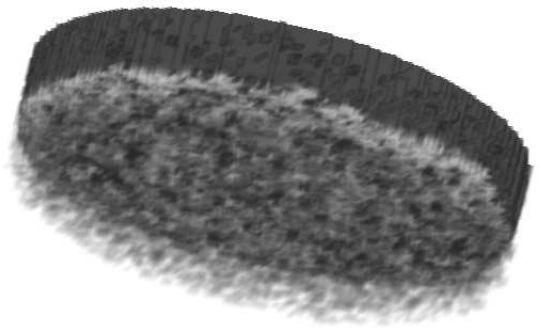

Fig. 2. Level of the water in the coffee bed at the end of the pre–brewing phase (*i. e.* at time $t = 4$ seconds) in a single cup filter.

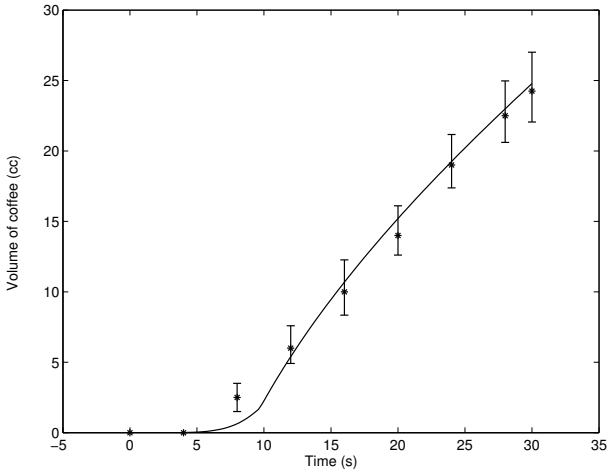

Fig. 3. Volume of coffee exiting from single cup filter.

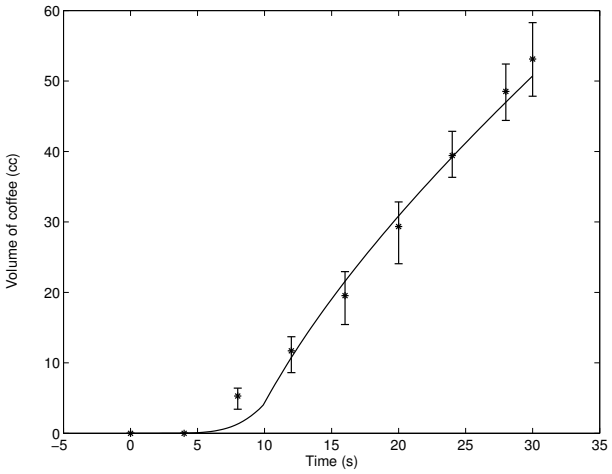

Fig. 4. Volume of coffee exiting from double cup filter.

the model results show a similar behavior.

In conclusion, the model correctly describes the main dynamical features of percolation through a coffee bed. The simulation results show a promising agreement with experimental data. Further experimental work is still needed to validate the numerical scheme and accurately set free parameters. The present model may provide a valid tool to investigate new or alternative settings for the physical parameters and configuration of the espresso machines, such as different filter shapes or inlet pressure profiles.

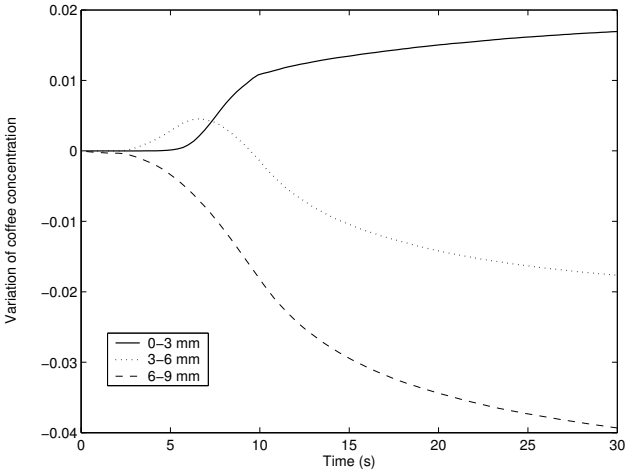

Fig. 5. Variation in time of coffee concentration inside three horizontal layers of the coffee bed for a single cup filter.

ACKNOWLEDGEMENTS

The authors are grateful to Fondazione CARIMA and Nuova Simonelli for the research grant "Modellizzazione matematica di alcuni fenomeni fisici relativi all'estrazione del caffè espresso".

REFERENCES

[CHO 00] CHOPARD B., MASSELOT A., DUPUIS A., *A lattice gas model for erosion and particles transport in a fluid*, vol. 129, p. 167–176, Comput. Phys. Comm., 2000.

[D'H 02] D'HUMIÈRES D., GINZBURG I., KRAFCZYK M., LALLEMAND P., LUO L. S., *Multiple-relaxation-time lattice Boltzmann models in three dimensions*, vol. 360, p. 437–451, R. Soc. Lond. Philos. Trans. Ser. A Math. Phys. Eng. Sci., 2002.

[INA 04] INAMURO T., OGATA T., TAJIMA S., KONOSHI N., *A lattice Boltzmann method for incompressible two-phase flows with large density difference*, vol. 198, p. 628–644, J. Comput. Phys., 2004.

[KEH 02] KEHRWALD D., Numerical Analysis of Immiscible Lattice BGK, PhD thesis, University of Kaiserlautern, 2002.

[MIS 06] MISICI L., PALPACELLI S., *A lattice Boltzmann approach for immiscible fluids with very different viscosities*, vol. 16, n. 9, J. Modern Phys. C. (to appear), 2006.

[QIA 92] QIAN Y. H., D'HUMI'ERES D., LALLEMAND P., *Lattice BGK models for Navier Stokes equations*, vol. 17, p. 479–484, Europhys. Lett., 1992.