ECONTECHMOD. AN INTERNATIONAL QUARTERLY JOURNAL – 2015, Vol. 4, No. 1, 77–85

Mesoscale Modeling of Complex Microfluidic Flows

O. Matviykiv

Department of Computer-Aided Design Systems,
Lviv Polytechnic National University;
79013, Lviv, Bandery st. 12
E-mail: matviykiv@gmail.com

Received January 12.2015; accepted March 15.2015

Abstract. The mesoscale description of multiphase flow in a typical Lab-chip diagnostic device is presented in actual article. The mesoscopic lattice Boltzmann method, which involve evolution equations for the single particle distribution function, was applied for the modeling of complex microfluidic flows. The general D2Q9 lattice Boltzmann formulation, considered multiphase flows, was developed. Three types of boundary conditions were used for the mesoscopic modeling: "ghost-fluid", "bounce-back" and "periodic boundaries". Traditional Dirichlet and Neumann macroscopic boundary conditions were transformed into mesoscopic lattice formulations. Algorithm of fluid flow solution, based on BGK single-relaxation-time scheme was proposed and implemented. The scaling procedure was used for physical parameters convertion into non-dimensional units. Simulation procedure was tested on a fluid flow with single solid particle. The final results showed good consistence with fundamental flow phenomena.
Key words: multiphase, microfluidics, flow, mesoscale, modeling.

INTRODUCTION

A variety of miniaturized microfluidic devices for point-of-care diagnostic, biofluidic assays, biochemical synthesis and single cells analysis have been widely investigated and developed over the last two decades. These devices, called Lab-on-Chips, usually process multicomponent multiphase fluid mixtures, which flows under the hydrodynamic or electrokinetic forces in complex microchannels or microchambers [2, 6, 24]. The precise modeling of microfluidic Lab-chip flows requires an accurate description of multiphase and multicomponent interactions. Well developed classical Navier-Stoks flow models, which are based on the continuum theory, unfortunately can't be applied for simulating Lab-chip devices [9]. Microfluidic flow models should consider various atomistic effects due to the presence of macroscopic time, length and energy scales in phases and components interactions. It is known, that in general applications atomistic models, based on quantum-mechanics and molecular dynamics, are not usable due to its extremely high computational costs. Thus, the majority of recent research activities in the field of microfluidic flows modeling was devoted to the new numerical techniques.

The motion of a fluid mixtures can be described on three different levels (scales) – microscopic, mesoscopic and macroscopic [7, 14, 21]. At macroscopic scale – a Navier-Stokes partial differential equations are used. These equations can be solved by various numerical schemes like finite element method (FEM), finite volume method (FVM), etc. At microscale – on the contrary, a small particles movement simulate by molecular dynamics models. The governing is the Hamilton's equation, where we have to identify the location and velocity of the each involved particle. At mesoscopic scale – which fills the gap between macro– and micro– scales, fluid is considers as a collection of pseudo-particles. There are two main approaches for fluidic flow modeling at mesoscale level [15]: 1) hybrid approach, which combine molecular and continuum models; and 2) simplified approach, which directly formulates mesoscale model from molecular by some transformations. One of the main mathematical methods for mesoscale models is lattice Boltzmann method,

which united microscopic models and mesoscopic kinetic equations [5]. This method uses the modified version of Boltzmann equation, which describes the meso-particles interactions on fixed lattices and directly simulates the fluids movement. The lattice Boltzmann method (LBM) was used here for the modeling a simulating of multiphase and multicomponent microfluidic flow mixtures.

The difficulties associated with Lab-chips microfluidic flows modeling are related to a principal contradiction: the flow equations directed to the macroscopic variables – fluid density (ρ), pressure (p), temperature (T), velocity (u), while the fundamental mechanisms of the basic phenomena are related to the microscopic level. It is well known, that any continuum liquid is an ensemble of particles – molecules, ions and electrons with different positions (r_i) and velocities (v_i), which move under the influence of external forces (electromagnetic fields, pressure, gravity) and internal collision processes (brownian, ionization, charge exchange etc.). Unfortunately, "averaged" macroscopic parameters, which we observe at macroscale level, are averages over the distribution of particle velocities and/or positions. Thus, instead of hydrodynamic equations, more natural fluid flow description might involve evolution equations for the particle distribution function. Applying pseudo-particles instead of real molecules, one can create an alternative mesoscopic description of fluid flows, intermediate between microscopic and macroscopic worlds.

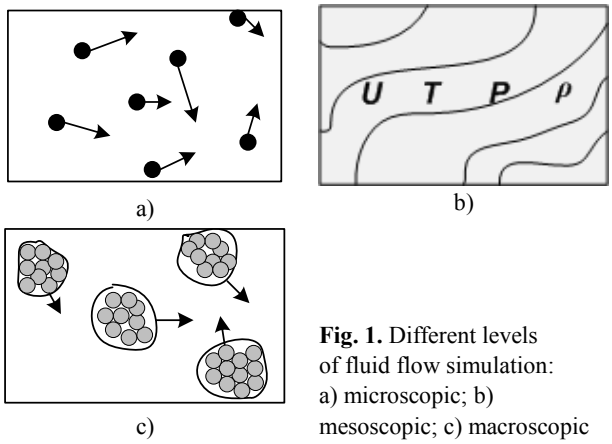


Fig. 1. Different levels of fluid flow simulation: a) microscopic; b) mesoscopic; c) macroscopic

As it is shown, lattice Boltzmann approach describe fluids as substance, consisted of fictive particles. Such particles perform consecutive propagation and collision processes like separate discreet molecules. Due to such particulate nature and local dynamics, mesoscale approach has plenty advantages over classical continuum models, especially in case of complex boundaries, multiphase interactions and in modern tendencies of the parallel calculations. The physics of microfluidics flow and fluid properties can be accurately incorporated into the LBE method, even more accurate than in Navier-Stokes equations [22]. Typical macroscopic parameters, like density ρ and velocity u can be easily calculated as soon as LBE solution will be obtained. Besides, it is particularly suitable for modeling various surface and interfacial phenomena, multiphase bio-flows and porous media flows, typical in microfluidics [1, 8].

MATERIALS AND METHODS

The kinetic Boltzmann equation was used to describe the temporal and spatial variation (evolution) of the particle probability distribution function. It represents the expected mass density of particles located at position r, moving with average velocity v at time t [4, 22]:

$$\frac{\partial f}{\partial t}+v_k\frac{\partial f}{\partial r_k}+\frac{F_k}{m}\frac{\partial f}{\partial v_k}=\Omega(f), \qquad (1)$$

where: $f(\vec{r},\vec{v},t)$ – the distribution function of single-particle position and momentum; F – external bulk force giving rise to the acceleration $a = F/m$; m – the mass of a particle.

Here, at the left-hand side: the first term represents the explicit velocity of the function f variation in time, the second – gives the spatial variation of function f according to particles movement, and the third – describes the effect of any force, acting on the particles. The right-hand side characterize the changes in distribution function, related to particles collisions. This part is highly non-linear, as it depends on actual value of distribution function and on inter-particle forces. Original Boltzmann collision operator $\Omega(f)$ is extremely complex, and usually replaced by a simpler model for computational work [3].

In the Boltzmann , see Eq. (1), the main unknown is mass density distribution function f. In 3D space it depends on seven independent variables: three coordinates, three micro velocity components and time. By using discrete working space into regular lattice, where the particles may have only certain number of allowed velocities q, the number of independent variables can be reduced to four. As a result of such reduction, at any given time each particle may have a discrete velocity from the set $\{v_0, v_1, \ldots, v_{q-1}\}$. Then the Boltzmann equation can be rewritten:

$$\frac{\partial f_i}{\partial t}+v_i\cdot\frac{\partial f_i}{\partial r}+\frac{F_i}{m}\cdot\frac{\partial f_i}{\partial v_i}=\Omega_i\left(\vec{f}\right), \quad i=0,1,\ldots,q-1, \qquad (2)$$

where: i – node index in discrete cell, $f_i \equiv f(\boldsymbol{r},\boldsymbol{v}_i,t)$, and $\tilde{f}$ is a shorthand for the vector $(f_0, f_1, \ldots, f_{q-1})^T$. The continuous operator $\Omega(f)$ is replaced by its discrete analog $\Omega_i(\tilde{f})$, which includes summations.

The solution of the Boltzmann equation defines the macroscopical fluid variables through the velocity moments of f. The Boltzmann equation has very high

mathematical complexity and need to be linearized and simplified. Besides, instead of direct numerical solution, it was proposed to imitate the evolution of particle distribution function $f(r,\vec{v},t)$ over the discretized lattice space. So, in an elementary volume *dr* pseudo-particles move chaotically or under some external forces, collide with each other and change its original speed. Applying basic conservation laws and Chapman-Enskog expansion for the elementary volume *dr*, the change in the distribution function was written [25]:

$$f\left(r+vdt,v+\frac{F}{m}dt,t+dt\right)d^3rd^3v = \\ = f(r,v,t)d^3rd^3v+dJ_{coll}, \quad (3)$$

where: F – external body force, m – particle mass, dJ_{coll} – change in the number of particles due to collisions.

According to the Le Chatelier – Brown principle, in a slight deviation of the physical system f from the stable equilibrium f_0 ($|f\text{-}f_0|<<f_0$), appear internal forces that are trying to return the system to equilibrium. In a simplest approximation, these forces are proportional to the deviation:

$$\frac{df}{dt}=-\frac{1}{\tau}(f-f_0), \quad (4)$$

where: τ – relaxation time or the rate at which f approaches its equilibrium state. The minus indicates that the response of the system on perturbations lead it to equilibrium. In this expression, as an equilibrium function, can be used local Maxwellian $f^{eq}=f^{M}$, which equals:

$$f^{(M)}=\rho\left(\frac{m}{2\pi k_BT}\right)^{D/2}e^{\left(-\frac{m}{2\pi k_BT}(v-u)^2\right)}, \quad (5)$$

where: ρ – particles density, m – particle mass, D – space dimension, v – particle microscopic speed, u – macroscopic fluid velocity, k_B Boltzmann constant.

By merging Eq. (3-5), and applying standard Bathar-Gross-Krook (BGK) collision approximation, where the collisions are considered as a linear process with single relaxation time model (SRT), the final lattice Boltzmann equation was obtained [3]:

$$f\left(r+vdt,v+\frac{F}{m}dt,t+dt\right)-f(r,v,t)=-\frac{\left(f-f^{eq}\right)}{\tau}dt. \quad (6)$$

Rewriting this equation in discrete mode, the evolution of the distribution function, which includes external body force, was obtained:

$$f_i\left(\bar{r}+\bar{v}_i\Delta t,t+\Delta t\right)-f_i(\bar{r},t)= \\ =-\frac{\Delta t}{\tau}\left[f_i(\bar{r},t)-f_i^{eq}(\rho,\bar{u})\right]-\Delta t\frac{F}{m}\cdot\nabla_v f_i(\bar{r},t), \quad (7)$$

where: f_i – related to the mass of particles, that moves from actual position in the direction i in one time step. The left-hand-side described a streaming step, and the right-hand-side – a collision step, which relaxes to its equilibrium value $f_i^{eq}(\rho,u)$ with relaxation time τ. Here f_i^{eq} depends on the macroscopic density and velocity at local domain coordinates and time. This equation don't define which state should be used for simulation: f_i^{eq} – in the point $(r + v_i\Delta t)$ at the time $(t + \Delta t)$ or in the point (r) at the time (t). According to classical computational fluid dynamics experience, it's better to use the up-wind calculation scheme, means: position $(r + v_i)$ at time $(t + \Delta t)$ [10].

As far as the collision process involves calculation of body forces F or other non-linear values, an intermediate step between streaming and collision should be inserted. Thus, the evolution equation (7) is decomposed in two steps: collision and streaming steps [19]:

- collision:

$$f_i^t(\bar{r},t)=f_i(\bar{r},t)-\frac{\Delta t}{\tau}\left[f_i(\bar{r},t)-f_i^{eq}(\rho,\bar{u})\right], \quad (8)$$

- streaming:

$$f_i(\bar{r}+v_i\Delta t,t+\Delta t)=f_i^t(\bar{r},t). \quad (9)$$

were: f_i^t depicts function value in moment t, when particles that came to the node in the direction i, but has not yet collide with other arrived particles, f_i is labeled a post-propagation or pre-collision population, and $f_i^t(r,t)$ – a post-collision or pre-propagation population. Both steps can be schematically presented as following:

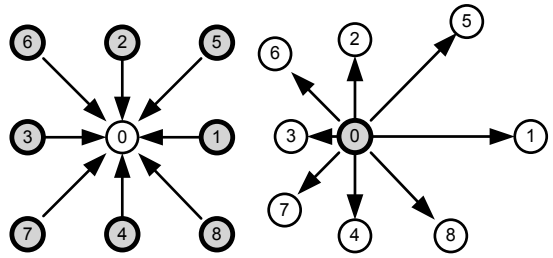


Fig. 2. Schematic representation of the a) streaming step, b) collision step

It can be seen, that in Boltzmann's elementary node there can be either 0 or 1 pseudo-particle, moving at a given speed. After a time interval Δt, each particle will move in the direction of a nearby node (streaming step). If several particles from different directions come into one node, they collide and change their speed according to collision rules (collision step). The collision step is completely local – it involves only the nearest neighbor nodes: the post-collision particles f_i^t move to the next node according to their discrete velocity set. As to the streaming step – it affects

neighbor nodes, but it is uniform and requires regular computational efforts.

By calculating local equilibrium distribution and body forces, the collision step executes and new set of f_i components at each lattice node. Then, particles with f_i components stream into next neighboring lattices. Both – streaming and collision steps sequentially alternates. Appropriate collision rules provides conservation for number/mass, momentum and energy, and the results satisfy continuous Navier-Stokes macroscopic equations.

In case of multiphase fluid flow, Shan and Chen have been formulated original algorithm, where a set of n-discretized LBE should be solved, where each fluid is represented by own equation [1, 20]. Thus, discretized Eq. (7) for the n-th fluid components looks like:

$$f_i^{(n)}\left(\bar{r}+\bar{v}_i\Delta t,t+\Delta t\right)-f_i^{(n)}\left(\bar{r},t\right)=-\frac{\Delta t}{\tau^{(n)}}\Big[f_i^{n}\left(\bar{r},t\right)- \\ -f_i^{(n)eq}\left(\rho,\bar{u}\right)\Big]-\Delta t\frac{F}{m}\cdot\nabla_v f_i^{(n)}\left(\bar{r},t\right), \qquad (10)$$

where: $f_i^{(n)}$, $\tau^{(n)}$ – are traditional single particle distribution function and single relaxation time for n-th fluid component.

The LBM simulation of multi-phase and multi-component transformations can be implemented through calculation of phase boundaries, emerging in the bulk microfluidic mixture. To describe the equation of state, which allows such transitions, it is necessary to introduce forces acted on the mixture components in the neighboring nodes. These forces can also provide the surface tension at the interface between different phases. As far as several fluids/phases may constitute the node component, this ambiguity will be solved according to the measurable dynamics – the node belongs to the fluid, which has the largest mass contribution.

General representative of microfluidic Lab-chip usually contain: injectors, junctions, microchannels, mixers, reactors, separators, filters, etc. Typical principal scheme of such device for diagnostic purposes is shown on Fig. 3 [18]. Mesoscale flow model can be applied in all important regions of Lab-chip device, where complex physical and chemical effects appear. Here, let's consider Ψ-junction (mixer) at the entrance region, where each microchannel input brings different components for multiphase fluid mixture.

The size of this device is 65x35 mm. The typical size for input microchannels can vary within 50..500 microns, depending on the size of bio-particles and contaminants. The length of input Ψ-mixer is up to 2..5 mm. The cross-sections of microchannels usually have rectangular (square) or semicircular shape. Each type of channel has appropriate resistivity parameter, which influence on calculation domain:

$$R_{rect}=12\mu L\big/H^3B;\ R_{circ}=128\mu L\big/\pi D, \qquad (11)$$

where: R_{rect} , R_{circ} – microfluidic resistance for rectangular and semicircular shapes, μ – dynamic viscosity of fluid mixture; L, H, B, D – length, height, width and diameter of the microchannel.

In order to simulate flow effects in a fluid mixture by LBM, a special discretization procedure for selected calculation domain should be conducted. Let's consider, that fluid processes and transformations at income microchannels before Ψ-mixer are less important. Also, 2D domain was chosen for test calculations. These simplifications allow to take only rectangular-body part for simulation microchannel Ψ-mixer. The LBM workspace discretization is executed by creating uniformly located spatial nodes and a set of allowed velocity vectors. Each velocity vector points on possible propagation directions, arises after particles collision. Besides, it contains a zero vector, which describes particles that do not stream and remains in a node. The set of nodes, which outlines the allowed velocity vectors, created an elementary Boltzmann lattice. By applying parallel transformations on the base of Galilean invariance principle, this lattice can be extended on the whole workspace and form required computational grid. That's why, the corresponding equilibrium distribution function and the related coefficients could be derived only after a specific lattice scheme is choose. Here, the D2Q9 lattice scheme was selected for 2D fluid flow in the ψ-junction. Nine discrete velocity vectors for the D2Q9 lattice Boltzmann scheme is shown below in Fig. 4, and the values of corresponding weight factors for calculating equilibrium distribution function are given in the Table 1.

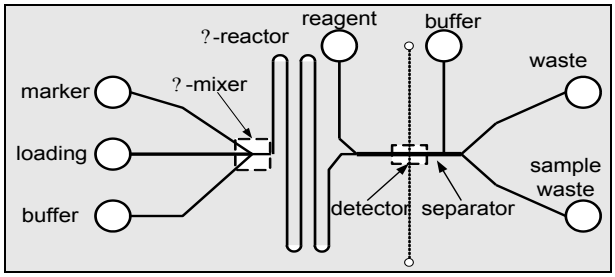


Fig. 3. Principal scheme of Lab-chip diagnostic device [18]

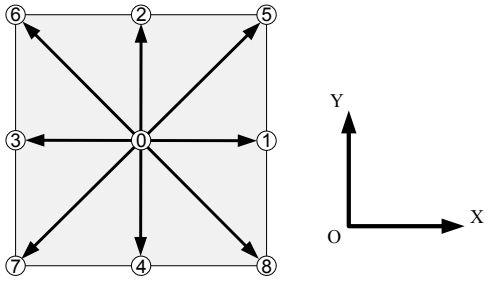


Fig. 4. D2Q9 lattice Boltzmann scheme with discretized velocity vectors

Table 1. Velocity and weight parameters for D2Q9 scheme.

Directions	Elementary vectors (e_i)	Lattice velocities (c_i)	Lattice weights (w_i)
i=0	(0,0)	$c_i = 0$	4/9
i=1,2,3,4	(1,0), (0,1), (-1,0), (0,-1)	$c_i = c_r$	1/9
i=5,6,7,8	(1,1), (-1,1), (-1,-1), (1,-1)	$c_i = c_r\sqrt{2}$	1/36

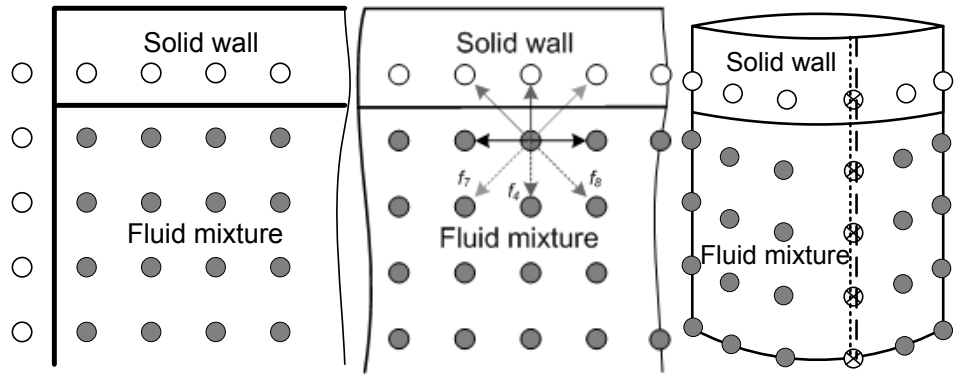


Fig. 5. LBM boundary types for mesoscopic fluid flow implementation:
a) ghost fluid nodes; b) bounce-back scheme; c) periodic boundary

The value of nonzero velocity components is equal to the distance traveled by a mesoparticle along a certain elementary axis during one time step. In general, the velocities c_i are equal to the relation of lattice size δr_i to the time step δt: $c_i = \Delta r_i / \Delta t$. From other side, the speeds in lattice Boltzmann simulation are related to the speed of sound. Theoretically, the discrete time unit can be calculated as $\Delta t = \Delta r / c_s$, where c_s is the speed of sound in the fluid. Besides, the choice of lattice velocity should keep the restriction between the macroscopic velocity u and the sound speed c_s: $u << c_s$ – in most calculations this relation fits $0.01 \le c_s/u \le 0.1$. In case, when $c_s/u \approx 0.1$, the time step Δt is equal: $\Delta t = 0{,}1\Delta r/|u|$.

The next stage in LBM is to define initial and boundary conditions in ψ-junction domain. Discussions of various boundary types can be found in the literature [13, 26]. Three types of boundary schemes were used in actual mesoscopic modeling – "ghost-fluid", "bounce-back" and "periodic boundaries", see Fig. 5. These types were chosen for their simplicity and satisfying accuracy.

In the "ghost-fluid" scheme two types of nodes are defined: interior fluid points are inside a solution domain, and ghost-fluid points are outside the solution domain, mainly inside solid microchannel walls or inside solid particles, which moves in a fluid. Thus, the microchannel solid boundaries lays in the middle between fluid interior nodes and ghost exterior nodes. The ghost nodes are also used in bounce-back boundary scheme for creating distribution functions at solid boundaries. According to the "bounce back" scheme, pseudo-particles comes towards the solid boundary and bounce back along incoming direction – definitely to the node from which they went out, but with opposite directions. The advantages of bounce-back condition is the fact, that it provides mass conservation in the simulation, and ensures no-slip and no-penetration up to 2nd-order of accuracy. The last scheme – periodic boundary condition is used to close the system within the two opposite edges – like input and output are connected to each other. This allows to obtain seamless connection and provides global conservation of mass in the calculation area.

Next, traditional Dirichlet and Neumann macroscopic boundary conditions should be transformed into mesoscopic conditions for LBM grid. As far as LB model deal with the velocity of fluid particles, main boundary conditions should be related to macroscopic velocity u. Once the local velocity values are computed, the single particle distribution function at each point, which streams according to chosen schemes, can be calculated. Generally, the Dirichlet condition postulates no-slip and no-penetration boundary. At first approximation, when boundaries lays between the fluid and ghost nodes, and assuming that velocity varies linearly from one node to another, the relation between adjacent node velocity u_i , macroscopic wall velocity u_w and ghost node velocity u_g can be represented by arithmetic mean:

$$u_g = 2u_w - u_i \,. \tag{11}$$

As to inlet particle distribution function $f_i(0,0)$ – it can be simply calculated from known distribution functions within the flow domain, based on imposed macroscopic parameters for fluid flow velocity and density:

$$f_i\left(r=0,t=0\right) = f_i^{eq}\left(r=0,t=0\right) = f_i^{eq}\left(\rho_0,u_0\right). \tag{12}$$

This equilibrium distribution function f_i^{eq} can be calculated through [11, 12]:

$$f_i^{eq} = \rho w_i \left[1 + \frac{u \cdot c_i}{c_s^2} + \frac{1}{2}\left(\frac{u \cdot c_i}{c_s^2} \right)^2 - \frac{1}{2}\left(\frac{u \cdot u}{c_s^2} \right) \right], \quad (13)$$

where: c_s – lattice sound velocity.

For Neumann condition, one could use a classical central-difference scheme:

$$\left. \frac{\partial u}{\partial y} \right|_{y=y_w} \approx \frac{u_i - u_g}{2\Delta y} = 0 \quad means \quad u_g = u_i . \quad (14)$$

To implement the boundary condition at the outlet (where neither velocity nor density is imposed), one can use the extrapolation of known distribution functions:

$$f_i(r,t) = \frac{1}{2}\left[f_i(r - c_{-i}\Delta t, t) + f_i(r - 2c_{-i}\Delta t, t) \right], \quad (15)$$

where: r – appropriate particle's coordinate.

All macroscopic hydrodynamic boundary conditions, which describes pressure, density, temperature and concentration parameters should be converted into appropriate expressions for the mesoscopic distribution functions.

A schematic LBM grid of a rectangular microchannel (mixer) with appropriate boundary conditions for velocity variable is shown on Fig. 6(a). In case of the presence of multiphase inclusions (gas bubbles, immiscible liquid drops, solid particles or soft agglomerates) in a fluid mixture, the moving boundary condition should be described on the edge of inclusions, see Fig. 6(b).

There are several schemes for reflecting moving boundaries among components in multi-component fluid mixture [17, 23]. The most popular are: classical non-slip bounce-back scheme (for solid particles), slip bounce-back scheme (for two immiscible liquids), bounce-back with modified collision operator (for fluid mixtures), bounce-back operation with non-equilibrium part of distribution function (for complex mixtures), bounce-back operation with Galilean transformation (for universal approach), and others. In this research the modified collision operator, which fits the microfluidic Lab-chip environment most of all. For this reason, the LBM equation for agglomerates is modified by the "solid" area fraction γ in each nodal cell [16, 17].

Single particle distribution function in these nodes becomes:

$$f_i^{(n)}\left(r + c_i\Delta t, t + \Delta t\right) = f_i^{(n)}\left(r,t\right) -$$

$$\frac{\Delta t}{\tau^{(n)}}(1-\beta)\left[f_i(r,t) - f_i^{(n)eq}(r,t) \right] + \beta \cdot f_i^{(n)}, \quad (16)$$

where: β – is coefficient, that depends on the "solid" area fraction (γ_s) in each lattice node:

$$\beta = \frac{\gamma\left(\tau^{(n)} - 1/2\right)}{(1-\gamma) + \left(\tau^{(n)} - 1/2\right)}. \quad (17)$$

All boundary expressions should be formulated and applied in such way, that the relevant hydrodynamic moments recalculated from the distribution functions are consistent with the primary macroscopic hydrodynamic parameters. Thus, in LBM algorithm the macroscopic parameters should be recalculated and harmonized at grid nodes before each collision step.

The major limitation of this LBM algorithm is the requirement of grid discretizing, which should comply to uniform square lattices. Several modifications were proposed in literature, like rectangular or stretched grids, multi-block grid, they do not receive extended application and most of LBM software tools works with regular square lattices. Besides, LBM simulation should be conducted entirely in lattice units, where time step, lattice step are of unit length. So, in order to guarantee the simulation consistent, all physical parameters need to be rescaled to non-dimensional form and then converted to LBM units, (Fig. 7).

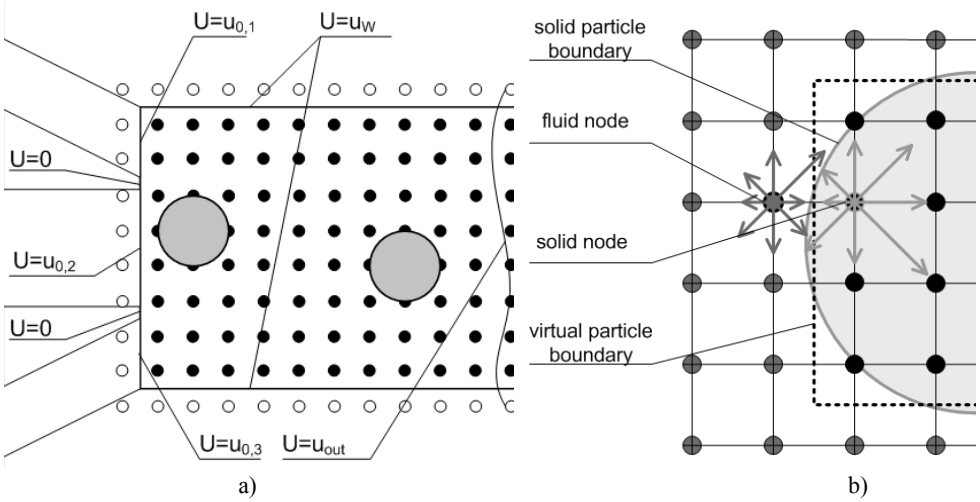


Fig. 6. LBM discretizion grid of microfluidic microchannel domain:
a) black dots – fluid domain; white dots – ghost fluid;
b) fluid and solid LBM nodes in multiphase flow

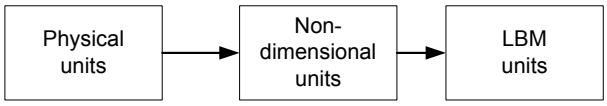


Fig. 7. LBM scaling procedure

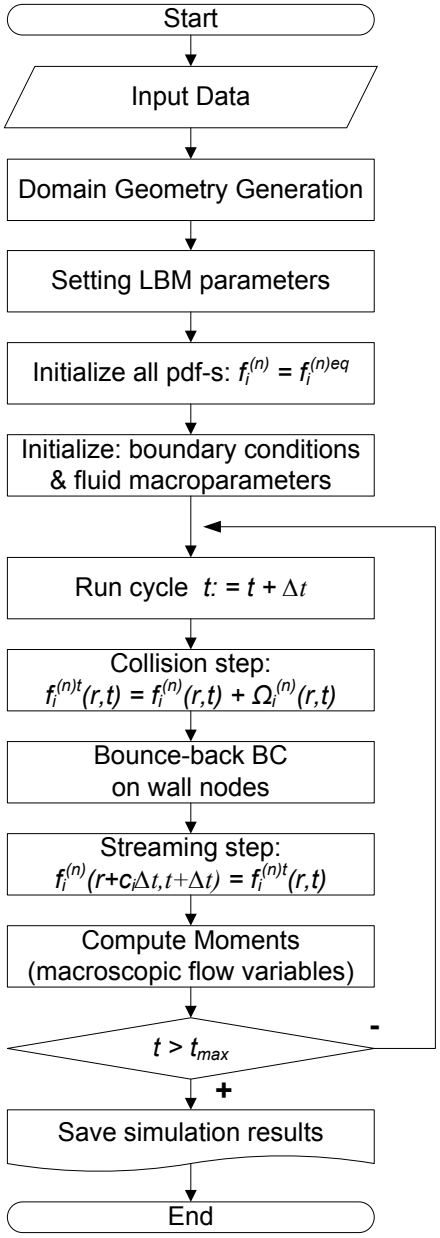


Fig. 8. LBM computational algorithm

The overall solution procedure, allows the LBM fluid flow simulation, can be described by the following collide&stream algorithm, (Fig. 8).

At first, the geometry of computational domain should be defined, and discretized grid (spatial step) according to selected lattice type (velocity scheme) must be formed. Next, appropriate time step, related to the incorporated fluid flow physics (number of fluid components and distribution functions) need to be calculated and verified according to $\Delta t = \Delta r/c$. As far as grid nodes and required distribution functions are known, an initialization procedure should set values for each $f_i^{(n)}$ in all lattice directions at each grid node. Initial boundary conditions should be included into initialization procedure as well according to Eq. (11-15). On this stage "collision step" should be executed and temporary (ready to stream) versions of n distribution functions at each node point for each lattice direction have to be calculated according to Eq. (8). Nodes, which are in touch with various boundaries, should correct their temporary distribution functions by applying appropriate bounce-back and/or periodic conditions, see Eq. (16). Next is "streaming step", where the particle population advects in the direction of corresponding lattice velocity to the neighboring lattice node, see Eq. (9). After that, macroscopic flow variables (u, ρ) can be updated in each grid node by calculating distribution function moments. At last, time step need to be incremented and checked if it reached maximum value.

Usually, the isothermal fluid flow can be characterized by Reynolds number:

$$\mathrm{Re} = \frac{u \cdot L_D}{\nu}, \qquad (18)$$

where: L_D – equivalent hydrodynamic diameter, $L_D = a$ – for microchannel with typical square cross-section (a – appropriate side). In LBM simulation procedure, the mesoscopic and macroscopic Re numbers should be equal [12]. This requirement allows to calculate the number of LBM nodes, and the appropriate size of LBM lattice.

RESULTS AND DISCUSSION

A numerical simulation of two-component fluid flow, which consists of liquid phase (water) and single solid particle was conducted in actual research (see Fig. 9). The microchannel dimensions were chosen as *1.0x1.0x10.0* (*mm*). Reynolds number was estimated in a range $Re = 0.1\text{-}10$. At this research stage, solid particle was represented as a sphere with the radius $r = 10^{-4}\,m$ and density $\rho = 2.59x10^3 kg/m^3$. Viscosity of the fluid μ was set to 10^{-3}Pa s.

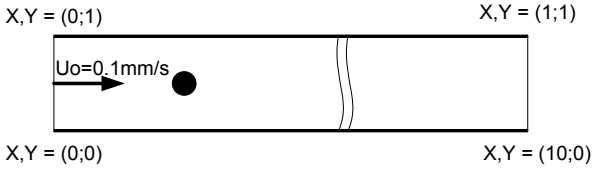


Fig. 9. LBM calculation domain

Flow enters from the left boundary with a prescribed parabolic velocity and exits out the right boundary with a constant pressure $P_{out} = 1,0\ atm$. The

top and bottom boundary are modeled as no-slip walls. In order to reduce calculation costs, the LMB domain was truncated to 500x100 lattices. The carrier fluid enters the domain with a Poiseuille velocity profile.

LBM calculations were conducted to obtain the fluid velocity profile around solid particle and its position. The colorbar legend with appropriate velocity scale is shown at the right. The diameter of particle is much smaller than the domain size and, at the beginning, solid particle moves slower then surrounded fluid.

The development of velocity profile can be presented by graphical screenshots, which were taken in different timestamps (see Fig. 11).

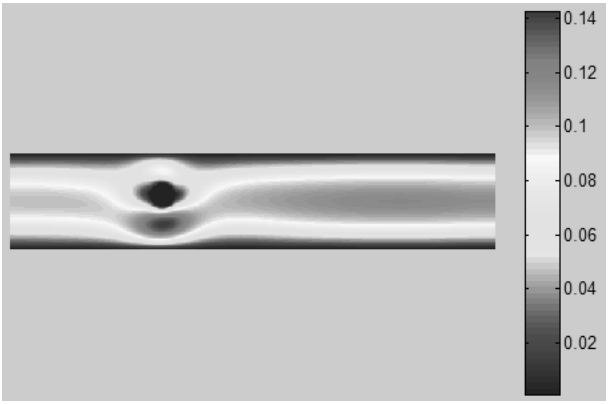
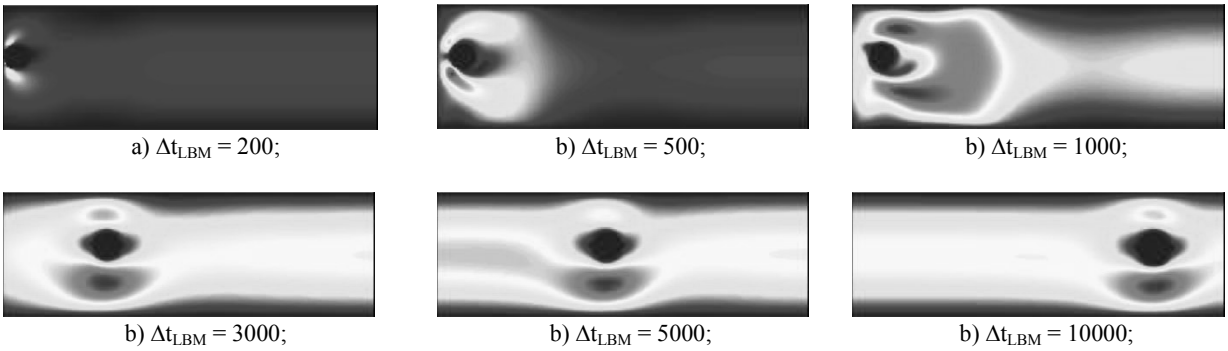


Fig. 10. Fluid velocity profile around particle



a) $\Delta t_{LBM} = 200$; b) $\Delta t_{LBM} = 500$; b) $\Delta t_{LBM} = 1000$;

b) $\Delta t_{LBM} = 3000$; b) $\Delta t_{LBM} = 5000$; b) $\Delta t_{LBM} = 10000$;

Fig. 11. Screenshots of simulated fluid velocity profile

The velocity profile around particle is going to stabilized at Δt_{LBM} approach to 40000 LBM time steps. In areas, remote from particle, velocity profile returns back to initial Poiseuille flow.

CONCLUSIONS

In actual research the mesoscale lattice Boltzmann model of fluid flow was developed and simple numerical simulation experiment with single solid particle was implemented. No-slip, no-penetration and bounce-back boundary conditions were prescribed at each wall using the Ghost-fluid method. Although, mesoscopic LBM gives a remarkable possibility to simulate a broad variety of complex fluid phenomena within single computational procedure. Just small change in the original computational procedure allows significant changes in physics of domain. In further research solid particle will be replaced by immiscible fluid drop, like oil. Next, multiple particles with different sizes may be added and simulated fluid flow behavior. Different obstacles, external forces and non-slip boundary conditions may be included to analyze mixing/separating possibilities.

The most valuable advantages of LBM calculation procedure are: 1) Intrinsic linear scalability in parallel computing that can be efficiently solved, because the collision are calculated locally; 2) Easy dealing with arbitrarily complex geometries: geometric complexity of microfluidic channels is not a challenge, because of the simple solid moving and domain deformation; 3) Efficient inter-phase interaction handling for multiphase flow because phase interaction is inherently included in the particle collisions.

REFERENCES

1. **Aidun C.K. and Clausen J.R. 2010**. Lattice-Boltzmann Method for Complex Flows. Annual Review of Fluid Mechanics, Vol. 42, 439-472.
2. **Ahn C.H. and Choi J.W. 2007**. Microfluidics and Their Applications to Lab-on-a-Chip. Springer Handbook of Nanotechnology, 523-548.
3. **Bhatnagar P.L., Gross E.P. and Krook M. 1954**. A model for collision processes in gases. Physical Review, Vol. 94, Nr 3, 511-525.
4. **Cercignani C. 1988**. The Boltzmann Equation and Its Applications. Ser. Applied Mathematical Sciences, Vol. 67, Springer, New York, 455.
5. **Chen S. and Doolen G.D. 1998**. Lattice Boltzmann method for fluid flow. Annual Review of Fluid Mechanics, Vol. 30, 329-364.
6. **Chin C.D., Linder V. and Sia S.K. 2012**. Commercialization of microfluidic point-of-care diagnostic devices. Lab on a Chip, Issue 12, 2118-2134.
7. **Fedosov D.A. and Karniadakis G.E. 2009**. Triple-decker: Interfacing atomistic-mesoscopic-continuum flow regimes. Journal of Computational Physics, Vol. 228, 1157–1171.

8. **Gad-El-Hak M.2006**. Gas and liquid transport at the microscale. Heat Transfer Engineering, Vol.27, Nr 4, 13–29.
9. **Glatzel T., et.al. 2008**. Computational fluid dynamics (CFD) software tools for microfluidic applications – A case study. Computers & Fluids, Vol. 37, 218–235.
10. **Goranovic G. and Bruus H. 2003**. Simulations in Microfluidics. Ser. Microsystem Engineering of Lab-on-a-Chip Devices, Wiley-VCH Verlag, Weinheim, Germany, 258.
11. **He X. and Luo L.S. 1997**. Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation. Physical Review E, Vol. 56, Nr 6, 6811-6817.
12. **He X. and Luo L.S. 1997**. Lattice Boltzmann model for the incompressible Navier–Stokes equation. Journal of Statistical Physics, Vol. 88, Nr 3-4, 927-944.
13. **Ho C.F. et.al. 2009**. Consistent Boundary Conditions for 2D and 3D Lattice Boltzmann Simulations. Computer Modeling in Engineering & Sciences, Vol. 44, Nr 2, 137-155.
14. **Hu G. and Li D. 2007**. Multiscale phenomena in microfluidics and nanofluidics. Chemical Engineering Science, 62, Nr 13, 3443–3454.
15. **Kalweit M. and Drikakis D. 2011**. Multiscale simulation strategies and mesoscale modelling of gas and liquid flows. IMA Journal of Applied Mathematics, Vol. 76, Nr. 5, 661-671.
16. **Kao P.H. and Yang R.J. 2008**. An investigation into curved and moving boundary treatments in the lattice Boltzmann method. Journal of Computational Physics, Vol. 227, Nr 11, 5671–5690.
17. **Lallemand P. and Luo L.-S. 2003**. Lattice Boltzmann method for moving boundaries. Journal of Computational Physics, Vol. 184, 406-421.
18. **Matviykiv O. et al. 2014**. Multiscale Flow Model for Simulation of Biofluidic Mixtures in Lab-Chip Devices, Proc. of the 21st Int. Conf. "Mixed Design of Integrated Circuits & Systems" (MIXDES'2014), Lublin, Poland, 89-92.
19. **Mohamad A.A. 2011**. Lattice Boltzmann Method. Fundamentals and Engineering Applications with Computer Codes, Springer, London, 178.
20. **Shan X. and Chen H. 1993**. Lattice Boltzmann model for simulating flows with multiple phases and components. Physical Review E, Vol. 47, No. 3, 1815-1820.
21. **Steinhauser M.O. 2008**. Computational Multiscale Modeling of Fluids and Solids. Theory and Applications, Springer-Verlag, Berlin, 427.
22. **Succi S. 2001**. The Lattice Boltzmann Equation for Fluid Dynamics and Beyond, Oxford, University Press, 304.
23. **Wagner A.J. and Pagonabarraga I. 2002**. Lees-Edwards boundary conditions for lattice Boltzmann. Journal of Statistical Physics, Vol. 107, Nr 1, 521-537.
24. **Whitesides G.M. 2006**. Overview The origins and the future of microfluidics. Nature, Vol. 442, Nr 7101, 368-373.
25. **Wolf-Gladrow D.A. 2000**. Lattice-Gas Cellular Automata and Lattice Boltzmann Models: an introduction. Ser. Lecture Notes in Mathematics, Vol. 1725, Springer-Verlag, Berlin, 314.
26. **Ziegler D.P., 1993**. Boundary conditions for lattice Boltzmann simulations. Journal of Statistical Physics, Vol. 71, 1171-1177.