14th World Congress on Computational Mechanics (WCCM)
ECCOMAS Congress 2020)
19–24 July 2020, Paris, France
F. Chinesta, R. Abgrall, O. Allix and M. Kaliske (Eds)

COMPARISON OF LAGRANGIAN-EULERIAN AND EULERIAN-EULERIAN APPROACHES FOR PARTICLE LADEN FREE SURFACE FLOW BY MEANS OF LATTICE BOLTZMANN METHOD

VIMMR J. [1], BUBLÍK O.[2] AND HEIDLER V.[3]

[1] Department of Mechanics, Faculty of Applied Sciences, University of West Bohemia,
Technická 8, 301 00 Pilsen, Czech Republic,
jvimmr@kme.zcu.cz, http://www.kme.zcu.cz

[2] NTIS - New Technologies for the Information Society, Faculty of Applied Sciences,
University of West Bohemia,
Technická 8, 301 00 Pilsen, Czech Republic,
obublik@kme.zcu.cz, http://www.kme.zcu.cz

[3] NTIS - New Technologies for the Information Society, Faculty of Applied Sciences,
University of West Bohemia,
Technická 8, 301 00 Pilsen, Czech Republic,
heidler@kme.zcu.cz, http://www.kme.zcu.cz

Key words: lattice Boltzmann method, fluid-particle interaction, free surface flow, immersed boundary method, Lagrangian-Eulerian model, Eulerian-Eulerian model

Abstract. The aim of this study is a comparison of Lagrangian-Eulerian and Eulerian-Eulerian numerical approach for the simulation of fluid-particles interaction. Within the study the immersed particles are restricted to have spherical shapes and are equal or smaller than the resolution of the computational mesh. The interaction between fluid and particles is performed using the immersed boundary method and the free surface flow of an incompressible fluid is simulated using the lattice Boltzmann method. Both approaches are compared within two test problems. Firstly, the swarm of particles falling in the fluid, and secondly, casting of the fluid with dispersed particles into a mold. Both tests showed good qualitative and quantitative agreement of mentioned approaches.

1 INTRODUCTION

The study of flows with immersed particles is of high importance across various fields in industry. Often the free surface of the fluid need to be tracked simultaneously. An important example of this is the gravity casting. In this manufacturing process, for the improvement of the mechanical properties of the cast, the fillers are added into the filling material. In order to address these issues the free surface flow solver has to be extended by the simulation of interaction between the fluid and the rigid particles. Many publications included [1, 2, 3] shows that the lattice Boltzmann (LB) method is capable to accurately simulate the free surface flows problems in complex geometries and its algorithm is straightforward, accurate, and effective. For this reason, the lattice Boltzmann method seems to be an appropriate choice

for the real casting simulations.

The modelling of fluid-particles interaction can be performed in many ways. In this work, specifically, the immersed boundary method (IBM) is adopted [4]. The traditional IBM assumes the mesh resolution much smaller than particle size and impose zero-velocity boundary condition on the particle surface afterwards [5]. In contrast of this, presented paper assumed particles to be much smaller than the grid resolution in order to capture the huge number of tiny particles in simulations. As long as the particle size is smaller than the grid resolution, multiple particles may occur within one cell. For correcting the fluid-particle interaction behaviour in this situations a fluid-particle slip velocity is introduced. This modification is known as particulate immersed boundary method (PIBM) [6, 7, 8]. Generally, the approach in which the fluid-particle interaction is realized only through the aerodynamic force is called two-way coupling. But in the cases of non-dilute systems, where the size of the particles is comparable to the size of the cells, the so called four-way coupling approach is necessary [9]. To put simply, in this approach, the local fluid volume fraction is used for modification of Navier-Stokes equations known as volume-averaged Navier Stokes (VANS) equations [10].

The particles submerged in a fluid can be described using the Lagrangian or the Eulerian approaches. In the Lagrangian approach, each of the particles has own coordinates and own equation of motion [11]. In the case of the Eulerian approach, the particles phase is described as a continuous function of particles concentration and the time evolution of concentration is described by the convection-diffusion equation [12]. The Lagrangian approach is more suitable for the simulations of a smaller number of large particles, while Eulerian approach is appropriate for the simulations of a huge number of small particles.

In this study we are focusing on spherical shaped particles, which sizes are comparable or smaller than the fluid mesh resolution. Moreover, the volume fraction of particles within a cell is relatively small. From this reason, the two-way coupling is sufficient. The aim of this work is to compare Lagrangian-Eulerian and Eulerian-Eulerian approaches on two test problems.

2 LATTICE BOLTZMANN METHOD FOR FREE SURFACE FLOW

The LBM is based on the Boltzmann equation (BE) describing the time evolution of a distribution function $f = f(\mathbf{x},\mathbf{u},t)$, $\mathbf{x} = [x_1,x_2,x_3]^T$, $\mathbf{u} = [u_1,u_2,u_3]^T$ in a phase space

$$\frac{\partial f}{\partial t} + \mathbf{u}\cdot\frac{\partial f}{\partial \mathbf{x}} = \Omega(f), \tag{1}$$

where $\Omega(f)$ is the complex collision operator, representing the inter-particle collisions. In LBM the BE is discretized firstly in velocity space by the means of enabling the molecules to posses only a small fraction of discrete velocities. Within this work the D3Q19 velocity model is considered and the microscopic velocity vectors $\mathbf{e}_\alpha$ are depicted in Fig. 1 (left). The further discretization of BE in space and time yields a well known LBM scheme

$$f_\alpha(\mathbf{x}+\mathbf{e}_\alpha\Delta t, t+\Delta t) = f_\alpha(\mathbf{x},t) - \mathbf{M}^{-1}\mathbf{S}(\tau)\mathbf{M}(f_\alpha - f_\alpha^{eq}) + f_\alpha^g(\mathbf{x},t), \tag{2}$$

where f_α corresponds to the microscopic velocity direction vector $\mathbf{e}_\alpha$ and $f_\alpha^g(\mathbf{x},t) = 3w_\alpha\rho\,(\mathbf{e}_\alpha\cdot\mathbf{g})$ is the force term added for incorporation of the gravity acceleration $\mathbf{g}$ or other body forces, like fluid-particle interaction force. Note that the collision operator at the right-hand side of BE is approximated using the

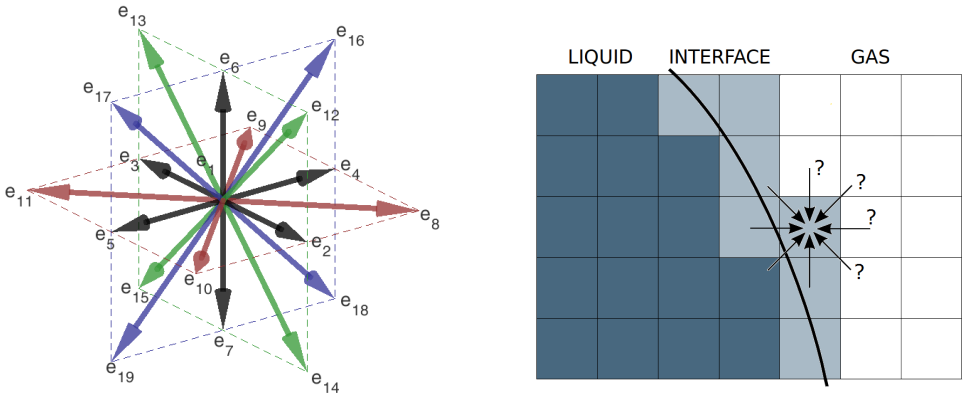

Figure 1: D3Q19 lattice Boltzmann velocity model (left). Liquid-gas interface, unknown values of functions f_α coming from the gas (right).

multiple relaxation time (MRT) operator [13]. The transformation matrix $\mathbf{M}$ is a linear mapping of the distribution functions f_α and f_α^{eq} into the moment space. For the fluid dynamics simulations discussed below, the equilibrium distribution functions f_α^{eq} are defined as

$$f_\alpha^{eq}(\rho,\mathbf{u}) = w_\alpha \rho \left(1 + 3(\mathbf{e}_\alpha \cdot \mathbf{u}) + \frac{9}{2}(\mathbf{e}_\alpha \cdot \mathbf{u})^2 - \frac{3}{2}(\mathbf{u}\cdot\mathbf{u})\right), \tag{3}$$

where w_α are the microscopic velocity vector weights, $\rho = \sum_\alpha f_\alpha(\mathbf{x}_i,t)$ is the density and for the macroscopic velocity can be written $\mathbf{u} = \frac{1}{\rho}\sum_\alpha f_\alpha(\mathbf{x}_i,t)\mathbf{e}_\alpha$, [14]. The collision matrix $\mathbf{S}(\tau)$ is a diagonal matrix containing relaxation rates of individual moments and the parameter τ is related to the kinematic viscosity ν as follows

$$\tau = \frac{1}{2} + 3\nu. \tag{4}$$

The extension of LBM for free surface flow modelling is rather straightforward. In the following, the algorithm introduced in [2], which resembles the volume of fluid (VOF) method, is considered. The free surface is tracked with help of the distribution functions f_α (corresponding to mass transfer in the α direction). In order to resolve the free surface, three types of grid cells are distinguished, see Fig. 1 (right), and the mass m and the fluid fraction $\chi = m/\rho$ is defined for them :

- $\chi = 1$ for liquid (completely filled with liquid)
- $\chi = 0$ for gas (empty cell - neglected in calculations)
- $\chi \in (0,1)$ for interface (partly filled with liquid)

The mass change inside the interface cells is computed as $\Delta m(\mathbf{x},t+\Delta t) = \sum_\alpha \Delta m_\alpha(\mathbf{x},t+\Delta t)$ which yields

$$\Delta m(\mathbf{x},t+\Delta t) = \sum_\alpha (f_{\bar\alpha}(\mathbf{x}+\mathbf{e}_\alpha\Delta t,t) - f_\alpha(\mathbf{x},t))\frac{\chi(\mathbf{x}+\mathbf{e}_\alpha\Delta t,t)+\chi(\mathbf{x},t)}{2}, \tag{5}$$

where $\bar\alpha$ denotes the opposite direction to α. The total mass of fluid in the interface cell can be determined as $m(\mathbf{x},t+\Delta t) = m(\mathbf{x},t) + \Delta m(\mathbf{x},t+\Delta t)$. After calculating the mass $m(\mathbf{x},t+\Delta t)$, the basic steps of LBM algorithm, i.e. collision and propagation steps, are performed. The unknown distribution functions

$f_{\bar{\alpha}}(\mathbf{x}, t+\Delta t)$ after the propagation step at interface cells are reconstructed using the equilibrium functions $f_{\alpha}^{eq}(\rho_a, \mathbf{u})$ and $f_{\bar{\alpha}}^{eq}(\rho_a, \mathbf{u})$. The $\mathbf{u}$ is the macroscopic velocity at the interface cell and the density ρ_a depends on the external pressure p_a and on the surface tension σ as follows

$$\rho_a = 3(p_a + 2\kappa\sigma), \tag{6}$$

where $\kappa = \nabla \cdot \frac{\nabla\chi}{|\nabla\chi|}$ is the surface curvature. The distribution function is then computed as

$$f_{\bar{\alpha}}(\mathbf{x}, t+\Delta t) = f_{\alpha}^{eq}(\rho_a, \mathbf{u}) + f_{\bar{\alpha}}^{eq}(\rho_a, \mathbf{u}) - f_{\alpha}(\mathbf{x}, t). \tag{7}$$

At the end of the currently executed time iteration, the type of each interface cell is refreshed based on the following conditions

$$m(\mathbf{x}, t+\Delta t) \;\geq\; \rho(\mathbf{x}, t+\Delta t) + \varepsilon \longrightarrow \text{liquid cell}, \tag{8}$$

$$m(\mathbf{x}, t+\Delta t) \;\leq\; -\varepsilon \longrightarrow \text{gas cell}, \tag{9}$$

where ε is a chosen threshold ($\varepsilon = 0.05$ in this work). In general, liquid and gas cells cannot lie next to each other. Thus when an interface cell becomes a liquid cell, all adjacent gas cells must be changed to interface cells, and vice versa. The algorithm is described in detail in [2].

3 IMMERSED BOUNDARY METHOD

Within this study, only the particles smaller than mesh resolution are considered, see Fig. 2 (left). As long as the particle size is smaller than the grid resolution, multiple particles may occur within one cell. The velocity of the particles may then differ from the velocity of the fluid inside the cell. In order to address this issues the interaction between fluid and particles is realized by the immersed boundary method as follows. Firstly, the fluid velocity field $\mathbf{u}$ is interpolated into an arbitrary point of particle phase $\mathbf{X}$, see Fig. 2 (middle), using the formula

$$\mathbf{u}_p(\mathbf{X}_p) = \sum_{i=1}^{N} \mathbf{u}_i(\mathbf{x}) D(\|\mathbf{X}_p - \mathbf{x}_i\|), \tag{10}$$

where $D(\mathbf{r})$, $\mathbf{r} = \|\mathbf{X} - \mathbf{x}\|$ is the kernel function and Ω_R is the circular area with the centre in $\mathbf{X}$, where values of $D(\mathbf{r})$ are non-zeros. In this study we consider kernel function as

$$D(\mathbf{r}) = \delta(X - x)\delta(Y - y)\delta(Z - z), \tag{11}$$

where $\delta(r)$ has form

$$\delta(r) = \begin{cases} \frac{1}{4}\left(1 + \cos(\frac{\pi|r|}{2})\right), & |r| \leq 2, \\ 0, & |r| > 2. \end{cases} \tag{12}$$

If we consider, that the particle-phase at the point $\mathbf{X}$ moves with the velocity $\mathbf{U}_p$ then the fluid flow induce drag force acting on the particle

$$\mathbf{F}_p^D = \frac{1}{2}\rho_F C_D(Re) A \|\mathbf{U}_p - \mathbf{u}_p\|(\mathbf{U}_p - \mathbf{u}_p) = \frac{1}{2}\rho_F C_D(Re) A \|\mathbf{U}_{slip}\|\mathbf{U}_{slip}, \tag{13}$$

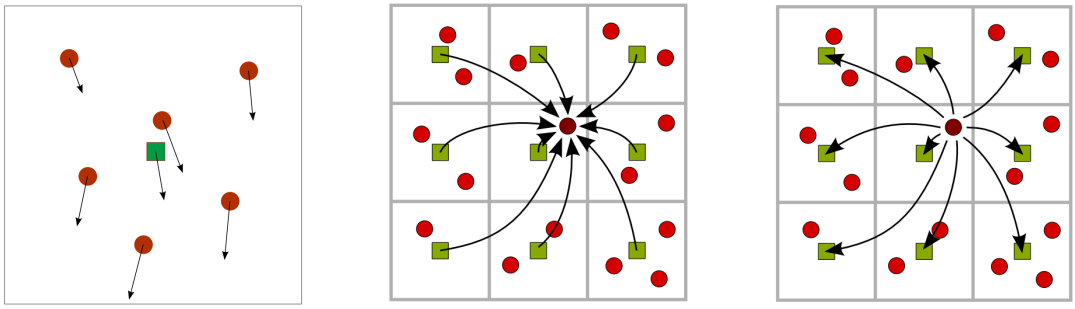

Figure 2: Particles are considered smaller than grid resolution (left). Fluid velocity interpolation into the particle position (middle). Particles interaction forces distribution into the fluid cells (right).

where $\mathbf{U}_p$ is the particle velocity, $\mathbf{U}_{slip} = \mathbf{U}_p - \mathbf{u}_p$ is the slip velocity, $A = \pi d^2/4$ is particle cross-section, ρ_F is the fluid density and $C_D(Re)$ is the drag coefficient dependent on the Reynolds number [15]

$$C_D(Re) = \begin{cases} \frac{24}{Re}\left(1 + 0.15\,Re^{0.687}\right), & Re \leq 1000, \\ 0.44, & Re > 1000, \end{cases} \tag{14}$$

where Reynolds number is defined by the relation

$$Re = \frac{d\,\|\mathbf{U}_{slip}\|}{\nu}, \tag{15}$$

where d is the particle diameter and ν is the kinematic viscosity of the fluid.

The drag coefficient (14) is not defined when slip velocity $\|\mathbf{U}_{slip}\|$ is equal to zero because of definition of the Reynolds number. From this reason, we express the product of drag coefficient with slip velocity to avoid division by zero

$$C_D(Re)\|\mathbf{U}_{slip}\| = \begin{cases} \frac{24\nu}{d}\left(1 + 0.15\,Re^{0.687}\right), & Re \leq 1000, \\ 0.44\|\mathbf{U}_{slip}\|, & Re > 1000. \end{cases} \tag{16}$$

Considering the other forces acting the particle, like gravity force $\mathbf{F}_p^G$, buoyancy force $\mathbf{F}_p^B$ and particle-particle interaction force $\mathbf{F}_p^{PP}$, the balance equation for single particle can be described as

$$m_p \frac{d\mathbf{U}_p}{dt} = \mathbf{F}_p^D + \mathbf{F}_p^G + \mathbf{F}_p^B + \mathbf{F}_p^{PP}, \tag{17}$$

where $m_p = \rho_p V_p$ is the particle weight, ρ_p is the particle density and V_p is the spherical particle volume. Due to the action and reaction principle, the negative value of the total force on the right hand side of the equation (17) is transfer back into the fluid, see Fig. 2 (right), using the relation

$$\mathbf{f}^P(\mathbf{x}_i) = -\sum_{p=1}^{N_P} \mathbf{F}_p^D(\mathbf{X}_p) D(\|\mathbf{X}_p - \mathbf{x}_i\|), \tag{18}$$

where N_P is the number of particles in the vicinity of $\mathbf{x}_i$ and $\mathbf{f}^P(\mathbf{x}_i)$ is the total force from the particles acting on the cell $\mathbf{x}_i$.

4 EULERIAN AND LAGRANGIAN DESCRIPTION OF PARTICLE PHASE

To simulate the dynamics of the disperse particles, two different approaches are used. Firstly, the Lagrangian description, where each of particles has own position vector $\mathbf{X}_p$ and velocity vector $\mathbf{U}_p$. The movement of one particle can be described by the equation (17) together with the kinematic equation

$$\frac{d\mathbf{X}_p}{dt} = \mathbf{U}_p. \tag{19}$$

The gravity and buoyancy forces acting on the particles are defined as

$$\mathbf{F}_p^G = m_p\,\mathbf{g}, \qquad \mathbf{F}_p^B = -\rho_f V_p\,\mathbf{g}. \tag{20}$$

For the evaluation of inter-particle force the spring connection between particles is considered. The gap between particles is $h = L-(d_1+d_2)/2$, where $L = \|\mathbf{X}_1-\mathbf{X}_2\|$, see Fig. 3. The particle interaction force is then computed as

$$\mathbf{F}_p^{PP} = \begin{cases} k\,h\,\mathbf{n} + b\,u_n\,\mathbf{n}, & \text{if } h<0, \\ 0, & \text{if } h\geq 0, \end{cases} \tag{21}$$

where $\mathbf{n} = (\mathbf{X}_2-\mathbf{X}_1)/\|\mathbf{X}_2-\mathbf{X}_1\|$ is the unit vector in the direction of the center line of the particles, $u_n = (\mathbf{U}_2-\mathbf{U}_1)\mathbf{n}$ is the difference between particles speeds projected into direction of the vector $\mathbf{n}$, k is the spring stiffness constant and b is the damping constant.

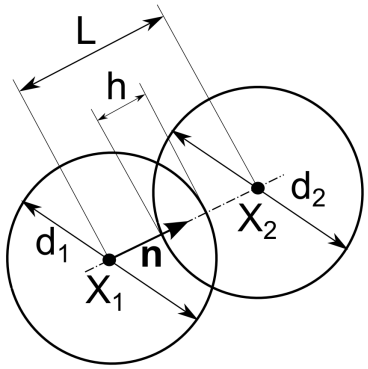

Figure 3: Sketch of two particle colission.

The second approach is the Eulerian description of the particles phase. In this approach the particles phase is considered to be homogeneous, as well as the fluid phase. The particles are not tracked individually, but the concentration function $\varphi(\mathbf{x})$ is used instead. This approach basically averages all particles occupying a fluid cell into one representative particle. The scalar concentration is discretized on the same grid as fluid and is defined as $\varphi(\mathbf{x}) = N/\Delta\mathbf{x}^3$, where N is the number of particles occupying the volume $\Delta\mathbf{x}^3$. The time evolution of $\varphi(\mathbf{x})$ is then computed with the convection-diffusion equation as follows

$$\frac{\partial\varphi}{\partial t} + \nabla\cdot(\mathbf{U}_P\varphi) = \nabla\cdot(D\nabla\varphi), \tag{22}$$

where D is a constant diffusion coefficient and $\mathbf{U}_P$ is the particle velocity computed using the evaluation of equation (17) in the cell centres. The force acting on the fluid $\mathbf{f}^P(\mathbf{x}_i)$ in equation (18) have to be

multiplied by the concentration. The value of diffusion coefficient D in our numerical simulations is set to $D = 10^{-6}\,\mathrm{m^2\,s^{-1}}$ according to work in [12].

The equation (22) can be solved using lattice Boltzmann method. In addition to the fluid solver a separate set of distribution functions, g_α, is introduced. To recover the convection-diffusion equation it is enough to use a smaller set of discrete velocities then for fluid solver, specifically D3Q7. The discrete scheme in LBM framework is

$$g_\alpha(\mathbf{x}+\mathbf{e}_\alpha\Delta t, t+\Delta t) = \varphi(\mathbf{x},t) + (1-\frac{1}{\tau_g})\phi_\alpha(\mathbf{x},\mathbf{t}), \tag{23}$$

where

$$\phi_\alpha(\mathbf{x},\mathbf{t}) = \frac{3(\mathbf{e}_\alpha - \mathbf{u}(\mathbf{x},t))}{\rho(\mathbf{x},t)} \sum_{\overline{\alpha}} \mathbf{e}_{\overline{\alpha}} f_{\overline{\alpha}}(\mathbf{x},t)(g_{\overline{\alpha}}(\mathbf{x},t) - \varphi(\mathbf{x},t)), \quad \varphi(\mathbf{x},t) = \frac{\sum_\alpha f_\alpha(\mathbf{x},t) g_\alpha(\mathbf{x},t)}{\rho(\mathbf{x},t)}, \tag{24}$$

where the relaxation parameter is not linked to the viscosity as in (4), but to the diffusivity $\tau_g = 1/2 + 3D$. The scalar φ serves as its own equilibrium, therefore no expression of equilibrium function is needed here. It should be pointed out that the collision operator is approximated using the regularized collision operator [16]. It ensures that only the first-order non-equilibrium moments contribute to scalar diffusion in the hydrodynamic range and is rather fast.

5 THE TEST PROBLEMS

The first test case serves as a validation of implemented Lagrangian-Eulerian fluid-particle interaction approach by simulating one particle falling inside a fluid under the influence of gravity force. The block domain with dimensions 0.05m, 0.05m and 0.15m in x, y and z directions is used. For its discretization 20 cells were chosen along x-axes, leading to grid spacing step $\Delta x = 0.0025\,\mathrm{m}$. The gravity force orientation is in the negative direction of z axes. The physical parameters of the fluid are the density $\rho = 1000\,\mathrm{kg\,m^{-3}}$ and the kinematic viscosity $\nu = 10^{-5}\,\mathrm{m^2\,s^{-1}}$. Two cases with the different diameter of the particle are used for the purpose of comparison. In both cases, the initial position of the particle is $X_0 = [0.025, 0.025, 0.1]$. In the first case, the particle diameter $d_P = 0.0025\,\mathrm{m}$ and the density $\rho_P = 3000\,\mathrm{kg\,m^{-3}}$ is set. In the second case, the particle diameter is reduced to $d_P = 0.001\,\mathrm{m}$. The Reynolds numbers related to the particle diameter and the settling velocity are $Re = 52.9$, in the first case and $Re = 7.1$, in the second case. Figs. 4 and 5 show the z-coordinate of the particle (left) and the z-component of the particle velocity (right). The red line represents the solution of ordinary differential equation

$$m_p \frac{d^2 z}{dt^2} = -(m_p - \rho_F V_p)\, g - \frac{1}{2} c_D \rho_F A_p u_p^2, \tag{25}$$

where m_p, V_p, A_p are the weight, volume and cross-section of the particle and ρ_F, c_D are fluid density and drag coefficient.

The second test case deals with a swarm of particles falling inside a fluid under the influece of the gravity force. The computational domain have a block shape with square base $0.1\,\mathrm{m} \times 0.1\,\mathrm{m}$ (x and y directions) and height $H = 0.2\,\mathrm{m}$ (z direction), see Fig. 7. The computational domain is discretized by uniform computational grid with 60 cells along x direction, which corresponds to grid spacing $\Delta x = 0.001667\,\mathrm{m}$. The computational domain is filled by the fluid with the density $\rho = 1000\,\mathrm{kgm^{-3}}$ and kinematic viscosity $\nu = 10^{-4}\,\mathrm{m^2\,s^{-1}}$. Inside each grid cell having distance d from a point $\mathbf{x}_0 = [H/2, H/2, 3H/2]$ smaller

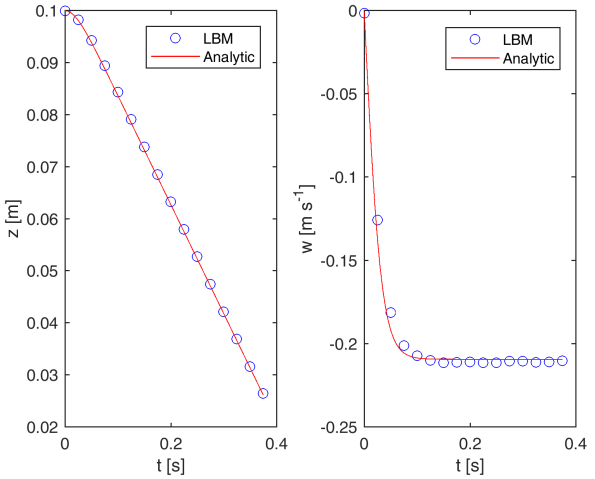

Figure 4: The time evolution of particle z-axis position (left) and particle z-axis velocity (right) for particle of diameter $d_P = 0.0025\,\mathrm{m}$ and particle Reynolds number $Re = 52.9$.

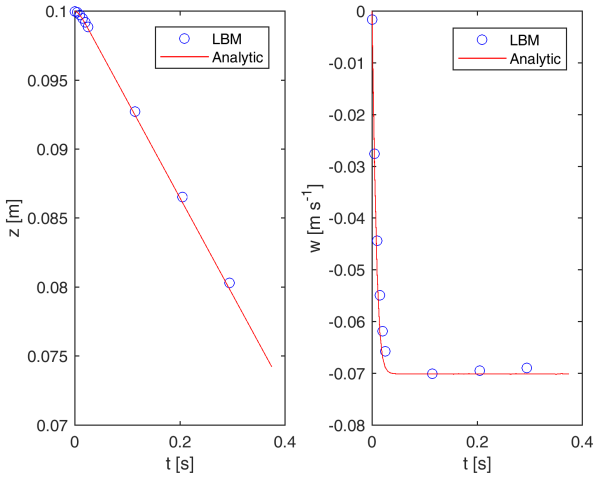

Figure 5: The time evolution of particle z-axis position (left) and particle z-axis velocity (right) for particle of diameter $d_P = 0.001\,\mathrm{m}$ and particle Reynolds number $Re = 7.1$.

then the radius $r = 0.03\,\mathrm{m}$, eight uniformly distributed particles are placed. The diameter of each particle is $d_P = 5 \times 10^{-4}\,\mathrm{m}$ and the density $\rho_P = 1500\,\mathrm{kg\,m^{-3}}$, see Fig. 7. In other words 195712 particles have to be tracked in the case of Lagrangian description. In the Eulerian description, on the other hand, particles are represented by the value of the concentration function, initialized to 8 (eight particles inside a cell). The Fig. 6 shows the time evolution of the contour line of the concentration. The lines are plotted for a chosen value 2 and for Lagrangian (red) and Eulerian (black) description of the particles phase.

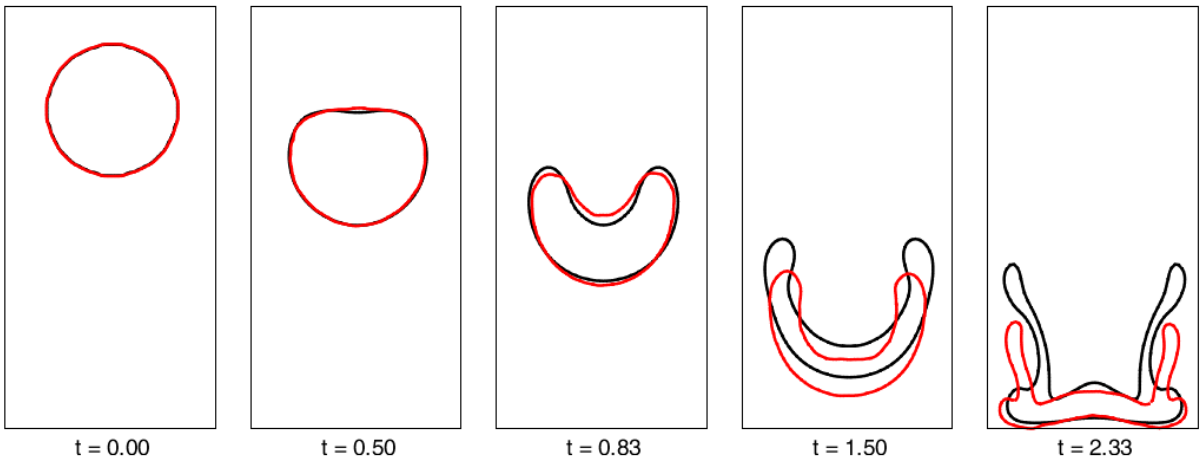

Figure 6: The time evolution of the contour line of the particle concentration plotted for value 2. The red line coresponds to Lagrangian description while black line coresponds to Eulerian description.

The last test case deals with a free surface flow into a mold of complex geometry. The geometry consists of rectangular blocks 10x10x10 mm. The dimensions of the mold are shown in Fig. 8. The computational

domain is composed of rectangular blocks 10x10x10 mm for an effective parallelization. Each of the block is uniformly covered by 12x12x12 cells, which lead to $\Delta x = 0.833\,\text{mm}$. The reservoir on the top is fulfilled by the fluid, see Fig. 8 - blue color. The fluid parameters are the density $\rho = 1260\,\text{kg}\,\text{m}^{-3}$ and the kinematic viscosity $\nu = 5.0794\,10^{-4}\,\text{m}^2\,\text{s}^{-1}$. The reservoir is also filled by the particles, see Fig. 8 - red color. The particles diameter and density were set to $d_P = 10^{-4}\,\text{m}$ and $\rho_P = 1200\,\text{kg}\,\text{m}^{-3}$. There are 8 particles uniformly placed in each cell from particle-filled area (2x2x2). This lead to 108288 particles which have to be tracked in Lagrangian description. Given the size of the cell, we obtain initial particles distribution approximately 0.72 percent of the cell volume. At time $t = 0$, the gate is released, and the fluid starts moving downwards due to the gravity force. The time evolution of the fluid for Lagrangian and Eulerian particles description, for the purpose of comparison, are illustrated at Figs. 9.

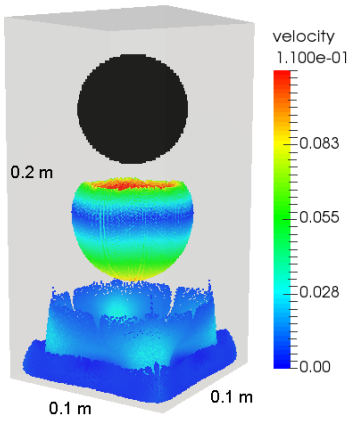

Figure 7: The setup of the second test problem. The initial position of the particles swarm is coloured black. The time evolution of the particles is captured by two states coloured according to a velocity.

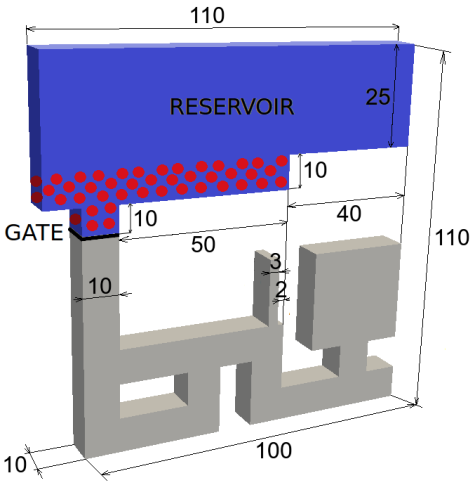

Figure 8: The sketch and dimensions of the mold.

6 CONCLUSION

The first test case deals with a motion and settling velocity of one particle falling inside a fluid. The comparison of obtained results with the ODE solutions (25) showed, that the presented method is able to sufficiently simulate the particle motion for different Reynolds numbers. Based on this test, it can be assumed the method will achieve valid results even for multiple particles assuming a low particles volume fraction. In the second case, the Lagrangian and Eulerian descriptions of the particles were compared. The results show the difference between concentration contours during the simulation time. Since the method calculating the fluid flow and particles velocity is same, the differences are only due to the different description of the particle motion. The Lagrangian approach exactly follows the physics. For this reason, we will consider the Lagrangian calculation as a ground-true solution. In the case of Eulerian description, the individual particles are averaged and replaced by a continuous function and the interaction force is evaluated only in the cell centres. Furthermore, the simulation of the transport equation is influenced by the artificial diffusivity, leading to spread out of the concentration function. For these reasons, there is a different evolution of the concentration contours over time. On the other hand,

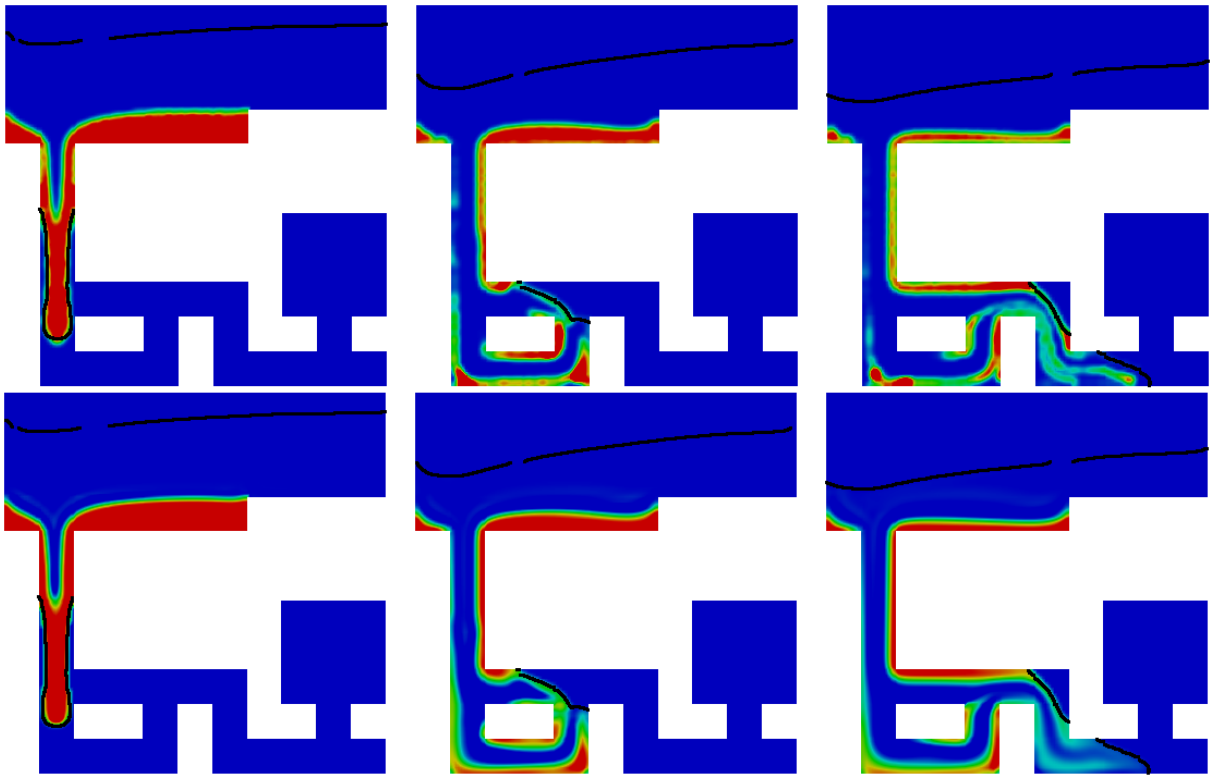

Figure 9: Comparison of Lagrangian (top) and Eulerian (down) description of the particles concentration at the selected times 0.16s, 1s and 2.17s. The black line shows the surface level.

it should be remembered that this is a very sensitive problem and that the differences have quantitative rather than qualitative character. In the last test example, gravity casting of the fluid with dispersed particles into a mold was considered. Figs. 9 show the comparison between Lagrangian and Eulerian particles description at the selected times. In spite of the free-surface flow in relatively complex domain, the results for the Lagrangian and Eulerian approach are both in qualitative and quantitative agreement. The Lagrangian approach is in principle much more accurate but its computational demands increases with the number of modeled particles. In these cases, the use of the Eulerian approach is a necessity.

7 ACKNOWLEDGEMENT

The authors appreciate the kind support by the grant GA 18-25734S "Experimental and computational modelling of free surface flow of non-Newtonian fluids with dispersed particles" of the Czech Science Foundation.

REFERENCES

[1] Vimmr, J. and Lobovský, L. and Bublík, O. and Mandys T. *Experimental validation of numerical approach for free surface flows modelling based on lattice Boltzmann Method.* In Proceedings of the 6th European Conference on Computational Mechanics: Solids, Structures and Coupled Problems,

ECCM 2018 and 7th European Conference on Computational Fluid Dynamics, ECFD 2018, (2020), pp. 2211–2222.

[2] Thürey, N. *Physically based Animation of Free Surface Flows with the Lattice Boltzmann Method*. Doctoral thesis, Erlangen, (2007).

[3] Ginzburg, I. and Steiner, K. Lattice Boltzmann model for free-surface flow and its application to filling process in casting. *J. Comp. Phys.* (2003) **185**:61–99.

[4] Peskin, C.S. The immersed boundary method. *Acta Numer.* (2002) **11**:479–517.

[5] Feng, Z.-G. and Michaelides, E. E. The immersed boundary-lattice Boltzmann method for solving fluid-particles interaction problems. *J. Comp. Phys.* (2004) **195**:602–628.

[6] Zhang, H. and Trias, F.X. and Oliva, A. and Yang, D. and Tan, Y. and Shu, S. and Sheng Y. PIBM: particulate immersed boundary method for fluid-particle interaction problems. *Powder Technol.* (2015) **272**:1–13.

[7] Habte, M.A. and Wu, C. Particle sedimentation using hybrid lattice Boltzmann immersed boundary method scheme. *Powder Technol.* (2017) **315**:486–498.

[8] Maier, M-L. and Henn, T. and Thäter, G. and Nirschl, H. and Krause, M.J. Multiscale simulation with a two-way coupled lattice Boltzmann method and discrete element method. *Chem. Eng. Technol.* (2017) **40**:1591–1598.

[9] Rettinger, C. and Rüde, U. A coupled lattice Boltzmann method and discrete element method for discrete particle simulations of particulate flows. *Comput. Fluids* (2018) **172**:706–719.

[10] Zhang, J. and Wang, L. and Ouyang, J. Lattice Boltzmann model for the volume-averaged Navier-Stokes equations. *Europhys Lett.* (2014) **107**:1–9.

[11] Henn, T. and Thäter, G. and Dörfler, W. and Nirschl, H. and Krause, M.J. Parallel Dilute Particulate Flow Simulations in the Human Nasal Cavity. *Comput. Fluids* (2016) **124**:197–207.

[12] Trunk, R. and Henn, T. and Dörfler, W. and Nirschl, H. and Krause, M.J. Inertial Dilute Particulate Fluid Flow Simulations with an Euler-Euler Lattice Boltzmann Method. *J. Comput. Sci.* (2016) **17**(2):438–445.

[13] d'Humières, D. Multiple-relaxation-time lattice Boltzmann models in three dimensions. *Philos. Trans. A Math. Phys. Eng. Sci.* (2002) **360**:437–451.

[14] Succi, S. *The lattice Boltzmann equation for fluid dynamics and beyond*. OXFORD university press, 2001.

[15] Schiller, L. and Naumann, A. Uber die grundlegende Berechnung bei der Schwerkraftaufbereitung. *Ver. Deut. Ing.* (1933) **44**:318–320.

[16] Chen, H. and Zhang, R. and Staroselsky, I. and Jhon, M. Recovery of full rotational invariance in lattice Boltzmann formulations for high Knudsen number flows. *Physica A* (2006) **362**:125–131.