METALLURGY AND FOUNDRY ENGINEERING – Vol. 38, 2012, No. 1
http://dx.doi.org/10.7494/mafe.2012.38.1.33

Michał Szucki*, Józef S. Suchy**

A METHOD FOR TAKING INTO ACCOUNT LOCAL VISCOSITY CHANGES IN SINGLE RELAXATION TIME THE LATTICE BOLTZMANN MODEL

1. INTRODUCTION

Casting production is accompanied by various physical phenomenon such as heat and mass exchange, crystallization, phase transformations, stress formation and etc. One of the most important factors which influence the quality of a final product is the nature and the kinetics of a liquid metal motion during mould filling.

High temperature which occurs in this process, makes it difficult to observe these type of flows in real conditions. Thus, computer simulation software is more often used in order to analyze temperature changes, microstructure formation and obviously liquid metal motion inside the mould [1, 2]. The usage of numerical flow models, which are usually based on the Navier-Stokes equation, in those programs is associated with the need to perform complicated and time consuming calculations. For this reason, there is a clear tendency to search for alternative approach in computer modeling of foundry processes. One of the most interesting solutions is so called lattice Boltzmann method (LBM) which, as it was shown in [3, 4] can be successfully used for numerical simulation of liquid metal flows. A big advantage of LBM based algorithms is simplicity in numerical implementation as well as very effective use of modern CPU/GPU architecture, including parallel computation.

One of the most important property of liquids, which have significant influence on nature and kinetics of incompressible flows (i.e. liquid metals as well) is kinematic viscosity. In case of foundry alloys, this parameter depends highly on temperature. In numerical models used in casting simulation software, the influence of crystallization process on motion of liquid metal is also taken into account through changes in the local viscosity [5].

* M.Sc.,** Prof., PhD., D.Sc.: AGH University of Science and Technology, Faculty of Foundry Engineering, Krakow, Poland; e-mail: mszucki@agh.edu.pl

In typical lattice Boltzmann models, kinematic viscosity of fluid is constant for all cells in the mesh. This assumption makes it difficult to use LBM algorithms for the simulation of real processes which appear in casting productions. For this reason, there is a reasonable need to search for solutions which allows to take into account local viscosity changes in the lattice Boltzmann method.

The work is an extension of the abstract presented at the scientific conference on the occasion of 60th anniversary of the Faculty of Foundry Engineering at AGH University of Science and Technology [6].

2. THE LATTICE BOLTZMANN METHOD

In algorithms based on the Lattice Boltzmann method, model evolution is described by the following equation:

$$f_i\left(x+e_i\delta t,t+\delta t\right)=f_i\left(x,t\right)+\frac{f_i^{eq}\left(x,t\right)-f_i\left(x,t\right)}{\tau} \tag{1}$$

where f_i is a particle distribution function in i direction, δt time step, f^{eq} an equilibrium distribution function characteristic for the specific type of LBM model [7], τ relaxation time, e_i a set of discrete dimensionless velocities.

The equilibrium distribution function f^{eq} may stay in a general form as fallows [8]:

$$f_i^{eq}=\omega_i\rho\left[1+\frac{\left(e_iu\right)}{c_s^2}+\frac{\left(e_iu\right)^2}{2c_s^4}-\frac{u^2}{2c_s^2}\right] \tag{2}$$

where: ρ and u are fluid density and velocity respectively, c_s sound speed for a model.

The value of the parameters appearing in equation (2) for the discussed here two-dimensional D2Q9 model (2 dimensions, 9 velocities) are presented in Table 1.

Table 1. Parameters for D2Q9 model

	ω_i	e_i	c_s
***i* = 0**	4/9	(0, 0)	$\frac{c}{\sqrt{3}}$
***i* = 1, 2, 3, 4**	1/9	$\left(\cos\left[(i-1)\pi/2\right],\sin\left[(i-1)\pi/2\right]\right)c$	
***i* = 5, 6, 7, 8**	1/36	$\left(\cos\left[(i-1)\pi/2+\pi/4\right],\sin\left[(i-1)\pi/2+\pi/4\right]\right)\sqrt{2}c$	

Parameter c from Table 1 is defined as $c = \delta x/\delta t$, where δx is a cell size. Usually it is assumed that $c = 1$, hence $\delta x = \delta t$.

In the lattice Boltzmann method the problem is usually solved on a regular, square (two- dimensional case) grid (Fig. 1).

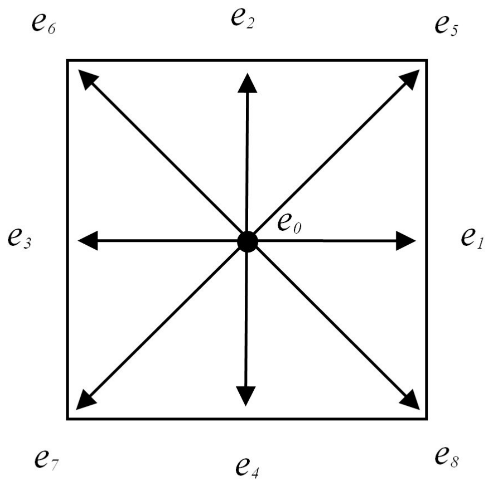

Fig. 1. *Discrete velocity set for the D2Q9 model on a two-dimensional square grid*

Local density ρ and fluid velocity *u* for each time step can be calculated as the sum of a particle distribution function, according to equations:

$$\rho = \sum_{i=0}^{8} f_i, \quad \rho u = \sum_{i=0}^{8} f_i e_i \tag{3}$$

Equations (1) is the most common approximation of a lattice Boltzmann equation i.e. approximation BGK (Bhatnagar-Gross-Krook). In this case for the whole computational domain a single relaxation time τ is usually used (mentioned single relaxation time model), which can be computed from:

$$\upsilon \frac{\delta t}{\delta x^2} = (\tau - 0.5) c_s^2 \tag{4}$$

where υ stands for fluid kinematic viscosity.

3. VISCOSITY AND NUMERICAL STABILITY OF LBM MODELS

By analyzing the Eq. (4), it can be seen that in the lattice Boltzmann method a direct relationship exists between relaxation time and fluid kinematic viscosity. Hence, for typical algorithms based on LBM, viscosity is also a constant parameter for all cells in the mesh.

In some works [9] direct attempts of using a separate relaxation time τ (i.e. a different kinematic viscosity as well) for each cells, can be found. However, it is often a source of instability in the model.

It is also worth mentioning that problems with numerical stability are one of the major drawback of algorithms based on LBM. An influence of particular parameters on the stability of the lattice Boltzmann method can be visualized through a presentation of relaxation time τ in the form [10]:

$$\tau = \frac{MaN}{c_s \operatorname{Re}} + 0.5 \tag{5}$$

where *Ma*, *Re*, *N* are Mach number, Reynolds number and number of cells in characteristic direction, respectively.

To perform a computation it is necessary to satisfy the condition*:* τ > 0.5. From Eq. (5) it can be seen clearly that together with an increase of Reynolds number (decrease of kinematic viscosity) model stability decreases (τ → 0.5). This problem is crucial in the case of simulation of liquid metals flow, which, due to their high density (compared to other fluids), are cha-racterized by low kinematic viscosity.

One of the most commonly used method for the stabilization of LBM algorithms is so called *Smagorinsky subgrid model* [11], in which model instability is compensated through increase of local viscosity of fluid. From a physics point of view, such an approach is questionable, because this kind of interference (which does not result from the physics of the process) may change the nature and the kinetic of the flow.

4. METHOD FOR TAKING INTO ACCOUNT LOCAL KINEMATIC VISCOSITY CHANGES

In this work, the authors' used so called Fractional Step (FS) method, originally presented by Shu *et al.* [10], in order to stabilize numerical solutions for high Reynolds number flows. This involves the need to divide calculations in each time step into two parts:

- Predicator: Solving of the equation (1) for predetermined relaxation time τ = 1 (and for predetermined viscosity) which provides numerical stability:

$$f_i\left(x + e_i\delta t, t + \delta t\right) = f_i^{eq}\left(x, t\right) \tag{6}$$

- Corrector: Solving the equation below in order to correct velocity field to values corresponding to real fluid viscosity:

$$\frac{\partial u}{\partial t} = b\nabla^2 u \tag{7}$$

The main idea of the presented solution is an application of the Fractional Step method not only to stabilize the model, but also to take into account local changes of kinematic visco-sity. Thus b parameter is defined as:

$$b=\left(\upsilon-\upsilon_{FS}\right) \tag{8}$$

where υ is a value of local viscosity for each mesh cell $\upsilon = \upsilon(x)$ (in the case when viscosity depends on temperature $\upsilon = \upsilon(T)$), υ_{FS} is fictitious viscosity calculated from the equation (4) with parameter $\tau = 1$:

$$\upsilon\frac{\delta t}{\delta x^2}=0.5c_s^2 \tag{9}$$

Thanks to this approach, relaxation time is still constant for all mesh cells, while the real kinematic viscosity is taken into account in the correction step. The equation (7) can be effectively solved with the use of an explicit difference scheme.

5. MODEL VALIDATION

A validation procedure of presented solutions was performed both for flows with constant kinematic viscosity and for the case where viscosity depends on the horizontal position of the cell in the computational domain ($\upsilon = \upsilon(x)$).

In the literature there is a lack of analytical solutions for the fluid flows with locally variable viscosity. That is why, the obtained results were confronted with data received from commercial simulation environment COMSOL Multiphysics. This application allows for the simultaneous modeling wide range of physical phenomenon and allows for their cross-correlation.

Validation of the proposed method was made based on a flow in square domain (Fig. 2), in which the top wall moves with constant velocity u_0 (*Lid driven cavity problem*).

Detailed parameters of performed simulations are presented in Table 2.

The analysis has been conducted for three different cases A, B and C. In first one (A) viscosity is constant in the whole system $\upsilon = \upsilon_1$. For B case, viscosity change linearly from υ_1 value (in point $x = 0$) to value υ_2 (for $x = L$) according to equation:

$$\upsilon=\left(\upsilon_2-\upsilon_1\right)x+\upsilon_1 \tag{10}$$

In the third case (C) viscosity is also constant for all mesh cells: $\upsilon = \upsilon_{2.}$

Case B reflects situation more closely related to problems occurring during mould filling, where areas of liquid metal with lower temperature (closer to mould walls) are characterized by much higher viscosity than places with higher temperature.

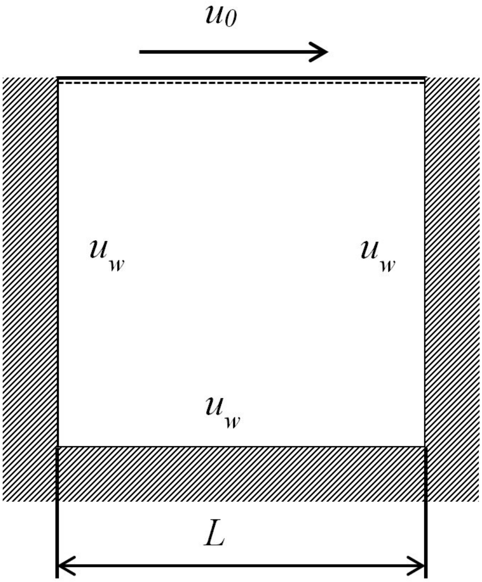

Fig. 2. *The computational domain used in this work*

Table 2. Parameters used in simulations

Parameter	Value
Length of computational domain L, m	1
Velocity on the top wall u_0, m/s	0.01
Velocity on the other walls u_w, m/s	0 (No-Slip boundary condition)
Kinematic viscosity υ_1, m^2/s	0.0004
Kinematic viscosity υ_2, m^2/s	0.000004

6. BOUNDARY CONDITIONS

In this work two different boundary conditions in the analyzed domain i.e. *No-Slip* and *Moving Wall,* were used. In the first case the fluid velocity at the wall is set to 0 [m/s] (left, right and bottom wall). The second boundary condition was implemented on the top wall which moves (driving fluid) with constant velocity of 0.01 [m/s].

In the lattice Boltzmann method (predicator step) No-Slip boundary is, in the simplest form defined as follows:

$$f_i\left(x,t+\delta t\right)=f_{inv}\left(x,t\right) \tag{11}$$

where *inv* means the direction inverse to *i*.

The distribution function is computed form equation (11) instead of equation (6) in the case when in *inv* direction there is a neighboring *No-Slip* wall. For the rest of the boundary cells (*Moving Wall* boundary) it is necessary to transfer the momentum of a moving wall into fluid. This can be done by boundary conditions usually used for taking into account fluid – obstacle interaction [12]:

$$f_i\left(x,t+\delta t\right)=f_{inv}\left(x,t\right)+6\omega_i e_i\cdot u_0 \tag{12}$$

The fractional Step Method requires also a definition of boundary conditions for corrector step. In this work, explicit difference scheme was used in order to solve the equation (7). Hence, the implementation of boundary conditions for corrector step is relatively simple and leads to defining constant values of velocity on the domain walls, accordingly $u_x = 0.01$, $u_y = 0$ for the top wall and $u_x = 0$, $u_y = 0$ for other walls, where x and y stand respectively for horizontal and vertical coordinates in the Cartesian system.

7. RESULTS AND DISCUSSION

Figure 3 shows a direct comparison of velocity fields, for three chosen cases (A, B, C), obtained with a computer program based on a presented model (left column) and as a result of simulation in COSMOL Multiphysics software (right column).

In all three analyzed cases, the obtained results are characterized by fairly high compatibility. For issues A and C the results from computer program based on the presented model as well as data from the COMSOL system, are almost identical. It is also confirmed by an analysis of velocity profile (case A) in the horizontal axis located in the middle of the computational domain (Fig. 4).

Some differences are visible for case B (Fig. 5). It can be caused by using an irregular computation grid in the COMSOL software. In this case, the distribution of grid nodes is also irregular, which combined with viscosity dependence on *x* coordinate can cause some minor errors.

8. CONCLUSIONS

The proposed method allows for taking into account local kinematic viscosity changes while maintaining the stability of a numerical solution. The obtained results are characterized by high compatibility with data received from commercial simulating environment COMSOL Multiphysics. It concerns both for flows, in which kinematic viscosity is constant for the whole computation domain, as well as for the case, when it is different for particular lattice cells. By the expansion of the presented method with the heat transfer process (so called *Thermal Lattice Boltzmann Method*) it is possible to simulate the motion of a liquid metal in the case where viscosity depends on temperature.

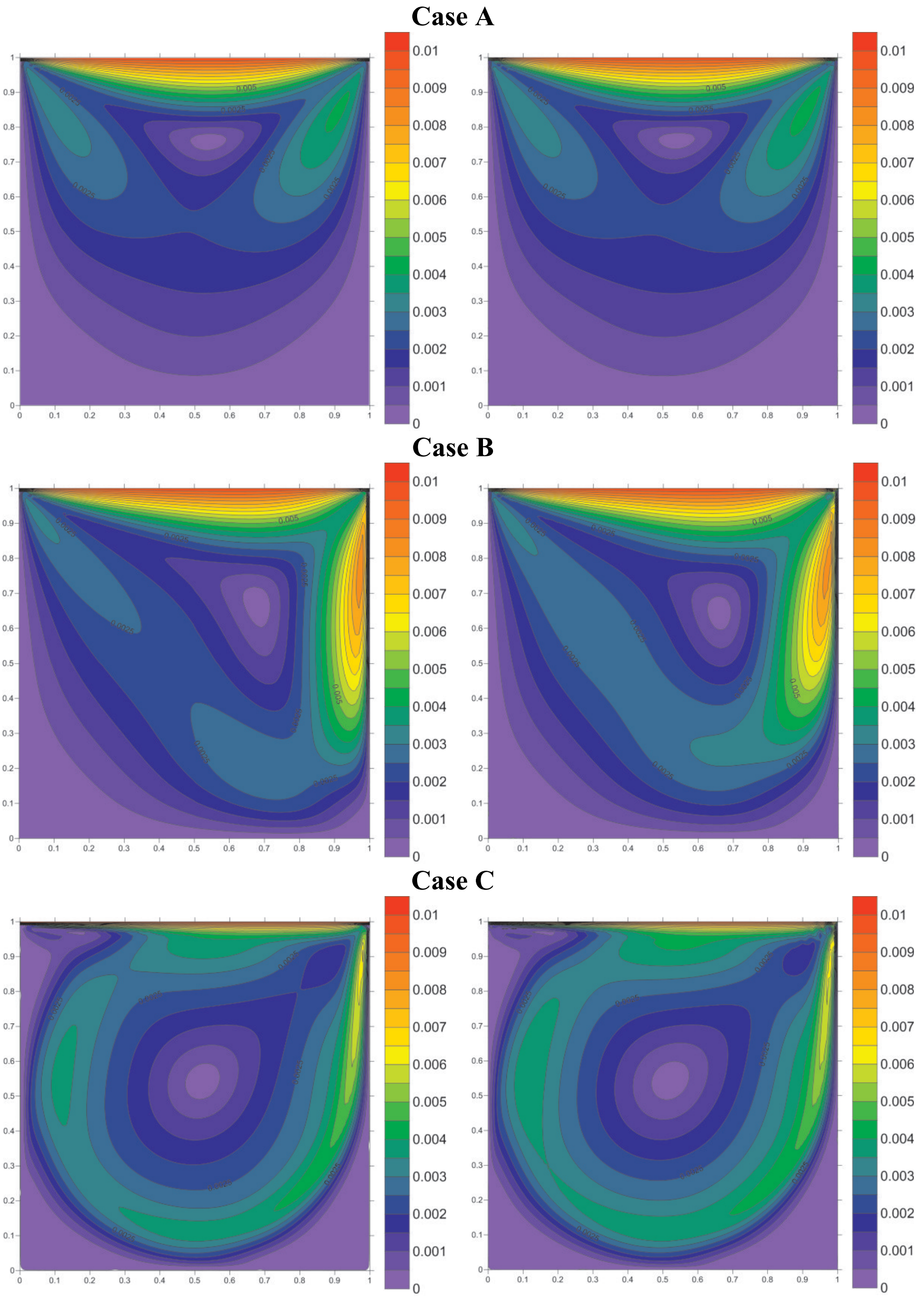

Fig. 3. Velocity field [m/s] for three chosen cases (A, B, C), derived from the presented model (left column) and as a result of simulations in COSMOL Multiphysics software (right column)

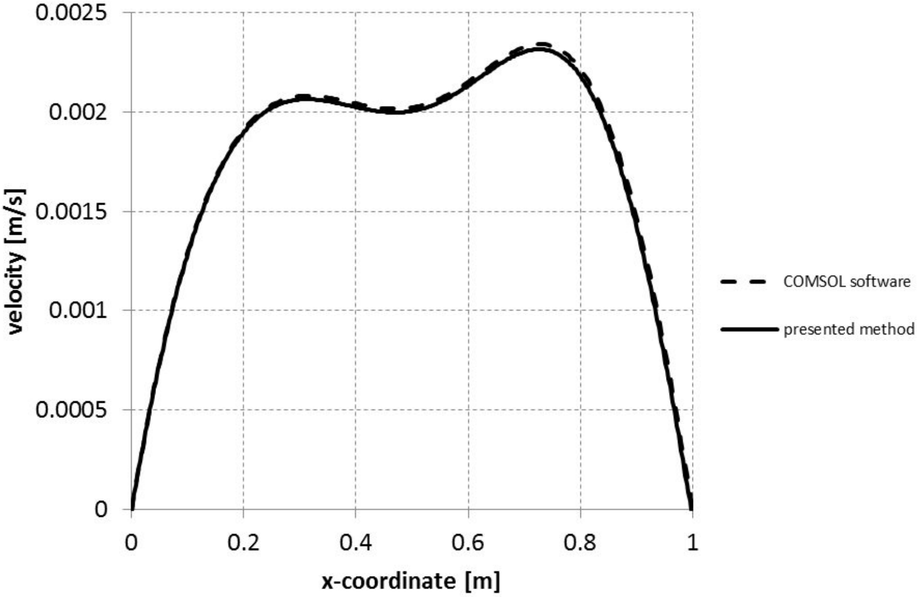

Fig. 4. *Comparison of velocity profile in the horizontal axis which goes through the center of the analyzed system (Case A)*

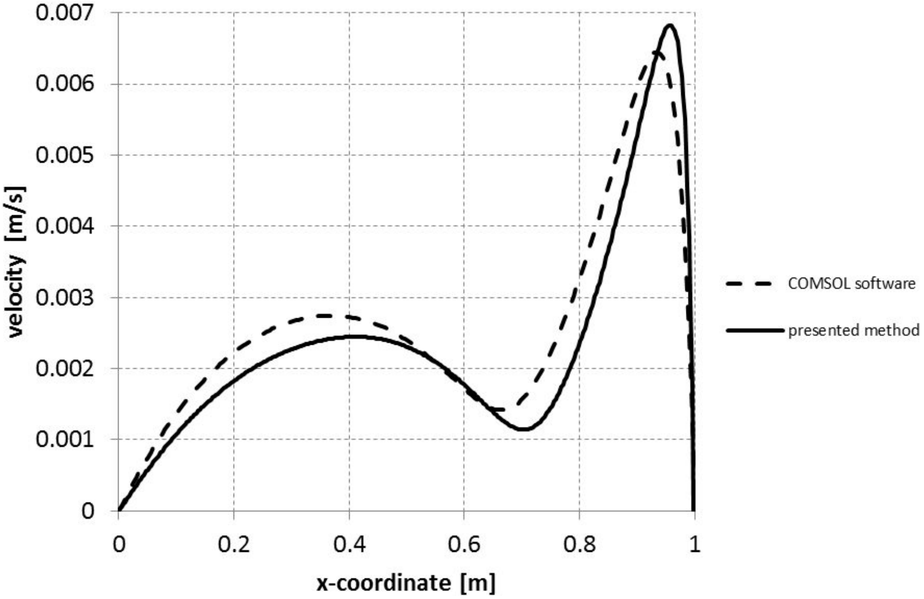

Fig. 5. *Comparison of velocity profile in the horizontal axis which goes through the center of the analyzed system (Case B)*

Acknowledgements

The authors acknowledge financial support from The Polish Ministry of Science and Higher Education through grant No. N N508 480638 (AGH No. 18.18.170.383).

The authors also would like to thank Niu Xiaodong and Shu Chang for help with the implementation of the FS algorithm.

REFERENCES

[1] *Piwowarski G., Krajewski W.K., Lelito J.*: Optimization of casting technology of the pressure die cast AZ91D Mg-based alloy, MaFE Metallurgy and Foundry Engineering, 36 (2) (2010), 105–111

[2] *Żak P., Lelito J., Krajewski W.K., Suchy J.S., Gracz B., Szucki M.*: Model of dendrite growth in metallic alloys, MaFE Metallurgy and Foundry Engineering, 36 (2) (2010), 131–136

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

[4] *Szucki M., Suchy J.S., Żak P., Lelito J., Gracz B.*: Extended free surface flow model based on the lattice boltzmann approach, MaFE Metallurgy and Foundry Engineering, 36 (2) (2010), 113–121

[5] *Kowalewski T.A., Cybulski A., Michałek T., Kowalczyk M.*: Laboratory benchmarks for validating numerical simulation of casting problems, Institute of Fundamental Technological Research Polish Academy of Science (2005)

[6] *Szucki M., Suchy J.S., Żak P., Lelito J., Gracz B.*: Numerical model based on the lattice Boltzmann method for flows with locally variable viscosity, 60th anniversary of the Faculty of Foundry Engineering at AGH University of Science and Technology, conference proceedings (in Polish) (2011)

[7] *Chen S., Doolen G.D.*: Lattice Boltzmann method for fluid flows, Annual Reviews Fluid Mec., 30 (1998), 329–364

[8] *He X. & Luo L.*: Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation, Physical Review E, 56 (6) (1997), 6811–6817

[9] *Guo Z., Zhao T.S.*: Lattice Boltzmann simulation of natural convection with temperature-dependent viscosity in a porous cavity, Progress in Computational Fluid Dynamics, 5 (2005), 110–117

[10] *Shu C., Niu X.D., Chew Y.T., Cai Q.D.*: A fractional step lattice Boltzmann method for simulating high Reynolds number flows, Mathematics and Computers in Simulation, 72 (2006), 201–205

[11] *Hou S., Sterling J., Chen S., Doolen G.D.*: A Lattice Boltzmann Subgrid Model for High Reynolds Number Flows, Amer. Mathematical Society, 6 (1994), 151–166

[12] *Thurey N., Iglberger K., Rude U.*: Free Surface Flows with Moving and Deforming Objects for LBM, Vision, Modeling and Visualization 2006, conference proceedings, (2006), 193–200

Received
February 2012