

# Simulation of the Alcohol-Oil Mixture in a Micromixer Using the lattice Boltzmann Method on a GPU Device

**Luiz Otávio Saraiva Ferreira, lotavio@fem.unicamp.br**

**Fabiola Martins Campos de Oliveira, fabiola.bass@gmail.com**

UNIVERSITY OF CAMPINAS, Faculty of Mechanical Engineering, Department of Computational Mechanics, Rua Mendeleev 200, CEP: 13083-860, Campinas - SP, Brazil

**Edgar Leonardo Martínez Arias, elmartinez@feq.unicamp.br**

**Rubens Maciel Filho, maciel@feq.unicamp.br**

UNIVERSITY OF CAMPINAS, Faculty of Chemical Engineering, Department of Chemical Process Development, Av. Albert Einstein 500, CEP: 13083-852, Campinas - SP, Brazil

**Abstract.** *This work presents the 2D simulation of two geometries of micromixer using the lattice Boltzmann method and the sailfish simulator, that runs on GPUs. The simulator was validated by simulating Poiseuille flows of Castor-oil and alcohol on straight channels, and the binary flow of Castor-oil and alcohol was simulated on Omega and Tesla shaped micromixers. The flow patterns presented reasonable agreement with published data, and the use of GPUs as computational platform shown an upper than ten times performance gain over CPU equivalent computer code. **Keywords:** Micromixer, Two Phase Flow, Microfluidics, Lattice Boltzmann Method, GPU*

## 1. INTRODUCTION

There is a strong trend on application of microfluidics to analysis (Berthier and Silberzan, 2006) and synthesis (Hessel and Löwe, 2003a,c,b) of chemicals. Biodiesel, a renewable fuel of wide economic and environmental importance today, had its synthesis by enzymatic transesterification on microreactors studied in recent years by (Arias and Maciel, 2010), by mixing Castor oil to ethanol and a catalyst on a micromixer device. Detailed knowledge of the flow inside the Biodiesel reactor is essential to its optimization such that it may a real alternative to present paths of Biodiesel synthesis. But flows in microscale devices present different performance from macroscopic flows, mainly because a much larger surface to volume ratio, which makes negligible surface phenomena on macroscopic flows to be of fundamental importance on microfluidics.

Theoretical analysis of microflows is limited to very simple problems, and often the numerical analysis is used to gain knowledge of the system behavior, with emphasis on aspects not available from experiments (Zhang, 2011).

Traditional computational fluid dynamics (CFD) methods have difficulties on incorporating phenomena such as the dynamics of wetting and interfacial slip (Zhang, 2011). The Lattice Boltzmann method (LBM) was introduced in 1988, by McNamara and Zanetti, and since then it is being studied as a powerful method for solving fluid dynamics problems (Mohamad, 2011). LBM models the fluid as fictitious particles that collide and propagate over a lattice. It handles easily complex boundaries, multicomponent multiphase flows, reactive flows, and presents efficient parallelization (Succi, 2001). LBM-like algorithms have been proposed to solve other phenomena, like heat transfer, electric fields, magnetic fields, diffusion processes, flows in porous materials, and shallow flows (Zhang, 2011).

A limiting factor on the numerical analysis of flows on microfluidic systems is the huge amount of calculations required, that results on several days long runs on a workstation even for relatively simple problems.

In December 2006, NVIDIA<sup>TM</sup> presented a new generation of graphics processing units (GPUs) based on its Compute Unified Device Architecture (CUDA<sup>TM</sup>) disruptive technology for massively parallel processing using graphics hardware, that is cutting down the computational cost of such simulations by using graphics processors (GPUs) as massively parallel mathematical co-processors (Owens *et al.*, 2007; Nguyen, 2007). This new high-performance computer technology shown to be valuable on accelerating LBM simulations (Bao, 2010), with typical acceleration factors higher than 10X when

compared to CPUs. Several different free LBM simulators are available as, for example, the GPU code *sailfish* and the CPU code *LB2D\_prime* from Sukop and Thorne Jr. (2005).

This work presents the 2D simulation of the ethanol-Castor oil mixture in Tesla and Omega shaped micromixers using the lattice Boltzmann method simulator *sailfish* running on a GPU equipped computer.

## 2. NUMERICAL MODELS AND METHODS

Lattice Boltzmann method was a spin-off of the Lattice Gas Cellular Automata (*LGCA*) method of fluid-flow simulation (Wolf-Gladrow, 2005). It recover Navier-Stokes equations in the continuum limit (Succi, 2001) based on a discrete momentum space, from Boltzmann kinetic theory. The LB model adopted on this simulation was extracted from the book of Sukop and Thorne Jr. (2005). The most significant aspects of the model are repeated bellow, and more details could be found on the original text.

### 2.1 Discretized 2D Boltzmann equation

The equation of the lattice Boltzmann method is obtained by discretizing Boltzmann equation on time, space and momentum (Aidun and Clausen, 2010; Zhang, 2011) and projecting it onto a discrete spatial lattice, Eq.(1), and the simulation is separated in two steps: streaming and collision.

$$f_a(\mathbf{x} + \mathbf{e}_a \Delta t, t + \Delta t) = f_a(\mathbf{x}, t) - \frac{[f_a(\mathbf{x}, t) - f_a^{eq}(\mathbf{x}, t)]}{\tau} \quad (1)$$

Where  $\mathbf{x}$  is the position of the particle,  $\mathbf{e}_a$  is its microscopic velocity,  $t$  is the time,  $\Delta t$  is the time-step of the simulation,  $f_a(\mathbf{x} + \mathbf{e}_a \Delta t, t + \Delta t) = f_a(\mathbf{x}, t)$  is the streaming part and  $[f_a(\mathbf{x}, t) - f_a^{eq}(\mathbf{x}, t)]/\tau$  is the collision term.

The collision term of Eq.1 came from a simplified model (BGK) introduced in 1954 by Bhatnagar, Gross and Krook.

An advantage of the discretized Boltzmann equation is that it may be used for dense fluids, while the original Boltzmann equation has application limited to short-range interactions in low-density gas flow (Aidun and Clausen, 2010). One of the most used LBE models is the D2Q9 (2 dimensions and 9 velocities), where the microscopic velocity  $\mathbf{e}_a$  ( $a = 0, \dots, 8$ ) is restricted to 8 directions plus a rest particle, 3 magnitudes, and there is a single particle mass, as shown on the left side of Fig.(1). It was assumed on Eq.(1) that particle mass= 1, so that microscopic velocities and momenta are equivalent.

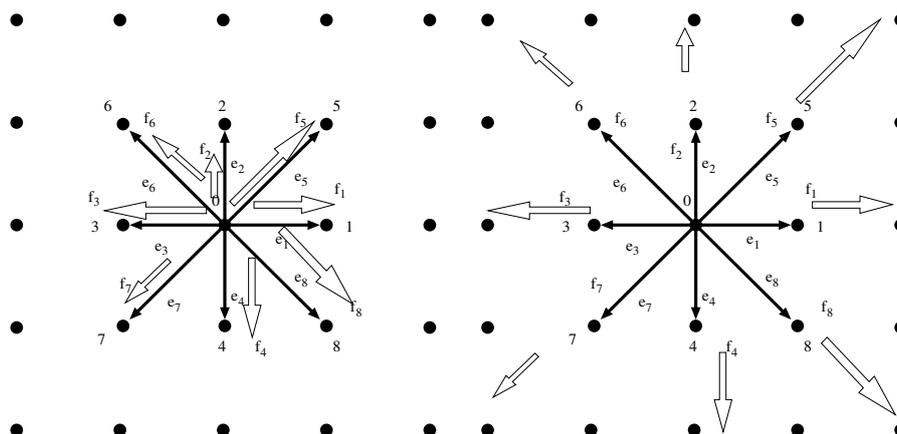


Figure 1. Left: D2Q9 lattice and microscopic velocities. Each velocity  $\mathbf{e}_a$  (black arrows) has an associated frequency  $f_a$  (white arrows). Right: Streaming step of D2Q9 lattice.

The units of length and time measurement are the *lattice unit* ( $lu$ ) and the *time step* ( $ts$ ), respectively. Velocity magnitudes of  $\mathbf{e}_1$  through  $\mathbf{e}_4$  is  $1lu/ts$ , and velocity magnitudes of  $\mathbf{e}_5$  through  $\mathbf{e}_8$  is  $\sqrt{2}lu/ts$ . If we think the distribution

function is a frequency of occurrence, its sum results on the macroscopic fluid density Eq.(2):

$$\rho = \sum_{a=0}^8 f_a \quad (2)$$

And the macroscopic velocity  $u$  is calculated as on Eq.(3):

$$\mathbf{u} = \frac{1}{\rho} \sum_{a=0}^8 f_a \mathbf{e}_a \quad (3)$$

On the streaming step, each density  $f_a$  is moved to the nearest neighbor lattice node pointed by the corresponding arrow, as shown on the right side of Fig.(1), resulting on updated values of  $f$  for each node of the lattice. Next step is the calculation of the value of the equilibrium distribution function  $f^{eq}$  for each node of the lattice, using Eq.(4).

$$f_a^{eq}(\mathbf{x}) = w_a \rho(\mathbf{x}) \left[ 1 + 3 \frac{\mathbf{e}_a \cdot \mathbf{u}}{c^2} + \frac{9}{2} \frac{(\mathbf{e}_a \cdot \mathbf{u})^2}{c^4} - \frac{3}{2} \frac{\mathbf{u}^2}{c^2} \right] \quad (4)$$

Where  $w_a$  is the weight for each particle:  $4/9$  for  $a = 0$ ,  $1/9$  for  $a = 1, 2, 3, 4$ , and  $1/36$  for  $a = 5, 6, 7, 8$ . And  $c$  is the lattice sound speed.

On the last step, the BGK approximation for the collision term of Eq.(1) is calculated for each node of the lattice.

Multicomponent flows are modeled with a state transition equation like Eq.(1) for each individual component  $\sigma$  (Sukop and Thorne Jr., 2005). The long range interparticle forces are incorporated in the model by summing the resulting velocities change  $\Delta \mathbf{u}$  Eq.(5) to the macroscopic velocity  $\mathbf{u}^{eq}$  Eq.(6) used to calculate  $f^{eq}$ , like on the case of gravity.

$$\Delta \mathbf{u} = \frac{\tau \mathbf{F}}{\rho} \quad (5)$$

$$\mathbf{u}^{eq} = \mathbf{u} + \frac{\tau \mathbf{F}}{\rho} \quad (6)$$

The equilibrium distribution  $f^{eq}$  must be computed from a composite macroscopic velocity  $\mathbf{u}'$  Eq.(7).

$$\mathbf{u}' = \frac{\sum_{\sigma} \frac{1}{\tau_{\sigma}} \sum_a f_a^{\sigma} \mathbf{e}_a}{\sum_{\sigma} \frac{1}{\tau} \rho_{\sigma}} \quad (7)$$

The interaction force on each one of a two component fluid is given by Eq.(8):

$$\mathbf{F}_{\sigma}(\mathbf{x}) = -G \Psi_{\sigma}(\mathbf{x}, t) \sum_a w_a \Psi_{\bar{\sigma}}(\mathbf{x} + \mathbf{e}_a \Delta t, t) \mathbf{e}_a \quad (8)$$

Where  $\bar{\sigma}$  is the other component and  $\Psi_{\sigma} = \rho_{\sigma}$  and  $\Psi_{\bar{\sigma}} = \rho_{\bar{\sigma}}$ . Then  $F$  is included in the computation of the equilibrium macroscopic velocity for each fluid is given by Eq.(9).

$$\mathbf{u}_{\sigma}^{eq} = \mathbf{u}' + \frac{\tau_{\sigma} \mathbf{F}_{\sigma}}{\rho_{\sigma}} \quad (9)$$

That is used for the computation of  $f^{eq}$ .

The forces from the interaction of fluids with solid surfaces are incorporated to the LBM the same way as the forces between fluid components described above (Sukop and Thorne Jr., 2005), and the equation of the force is given by Eq.(10):

$$\mathbf{F}_{ads}(\mathbf{x}, t) = -G_{ads}\Psi(\mathbf{x}, t) \sum_a w_a s(\mathbf{x} + \mathbf{e}_a \Delta t) \mathbf{e}_a \quad (10)$$

Where  $G_{ads}$  is the adsorption coefficient that specifies the strength of the force contribution, and  $s$  is one if the site at  $\mathbf{x} + \mathbf{e}_a \Delta t$  is a solid and is zero if the site is not a solid,  $\Psi(\mathbf{x}, t) = \Psi(\rho) = \Psi_0 \exp(-\rho_0/\rho)$ , and  $\Psi_0$  and  $\rho_0$  are arbitrary constants.

The boundary conditions of the numerical experiments are described in the following section.

## 2.2 Boundary conditions

Bounce-Back condition was assumed on the walls of the mixers, inlet boundary conditions for Castor-oil and alcohol were set as constant velocity, and the outlet boundary conditions were set as

## 2.3 Simulator setup

Each *sailfish* simulation is defined by a script file code written on the computer language *PYTHON*. The script specifies the boundary conditions, the geometry of the walls, the initial conditions and the model used on the simulation. Several different command line arguments may be passed when running the script as, for example, the duration of the simulation (in *lus*), the data output to a text file, or the on-line visualization of the simulation while it is running. A Poiseuille 2D flow was used to validate the simulator, before the simulation of the mixers.

## 3. RESULTS AND DISCUSSION

The physical properties and parameters of the simulations are shown in Tab.1:

Table 1. Physical properties of the Castor-oil and alcohol

Property	Castor-oil	alcohol
Density ( $kg/m^3$ )	957.3	789
Viscosity ( $Pa.s$ )	0.689	0.0012

Table 2. Numerical parameters of the simulation

Property	Castor-oil	alcohol
Density	1	0.824
Viscosity	0.16	0.0402
Relaxation time	0.98	0.62
Interaction parameter	-1.2	
Lattice spacing	$1.1910^{-5}$	
Time step	$3.74^{-}$	

Figure 4 shows the omega (left) and the Tesla (right) micromixers, where on both the input channel width is  $500 \mu m$ . The simulation grid size was of  $320 \times 320 lu$  for Omega mixer and  $320 \times 160 lu$  for Tesla mixer.

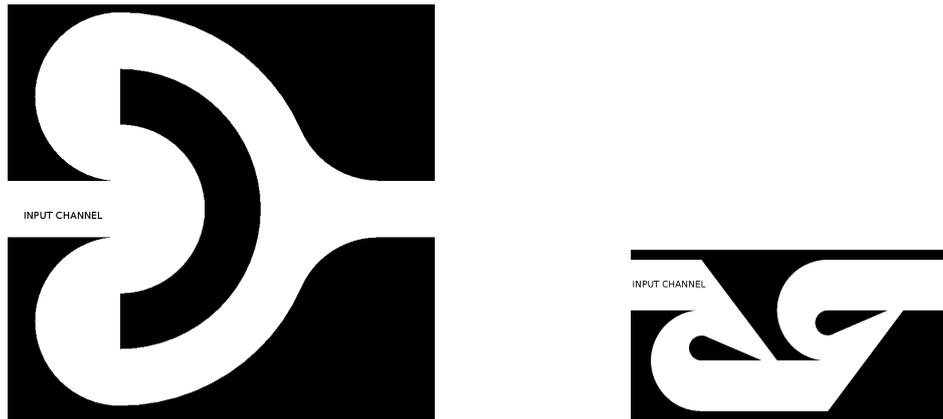


Figure 2. Left: Omega shaped mixer. Right: Tesla shaped mixer. Both mixers have  $500 \mu m$  wide input channels.

### 3.1 Poiseuille flow validation

A 2D simulation with Poiseuille flow on a  $40 \times 1000 lu$  channel with bounce-back walls and numerical viscosity equal to 0.02 resulted on the expected velocity profiles shown on Fig (3).

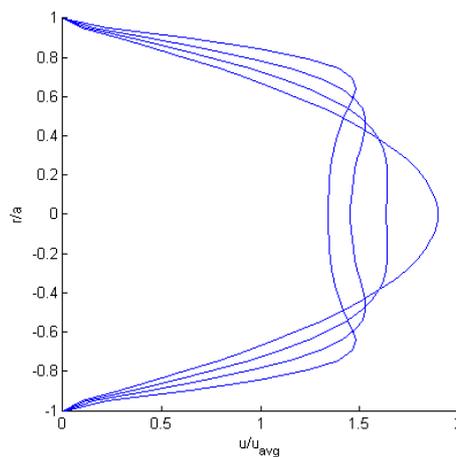


Figure 3. Poiseuille flow profile to validate the sailfish simulator. The input constant velocity profile evolutes to a parabolic profile.

Then the velocity profile for Poiseuille flow of Castor-oil and alcohol were simulated, as show on Fig. ():

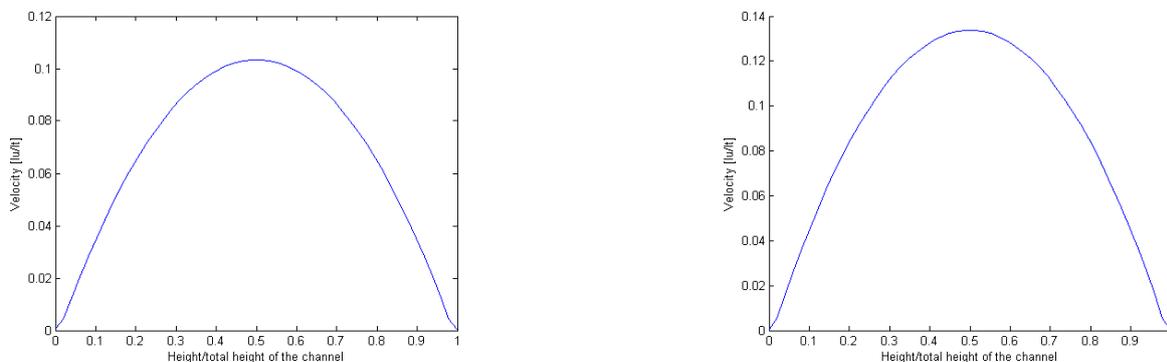


Figure 4. Left: Poiseuille flow velocity profile for alcohol on a straight channel. Right: Poiseuille flow velocity profile for Castor-oil on a straight channel.

### 3.2 Two-components flow velocity profile

Castor-oil and alcohol flowing side to side on a straight channel, with the simulation parameters of the 2, resulted on the velocity profile show on 5:

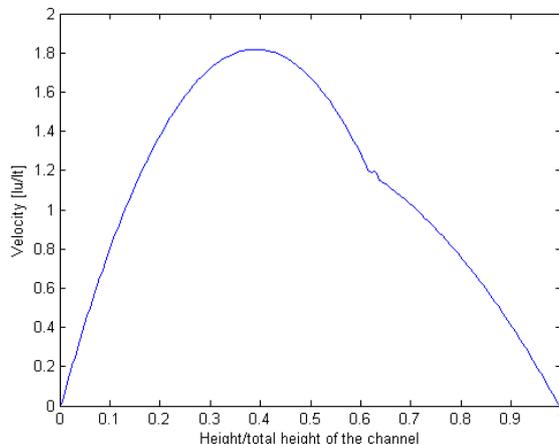


Figure 5. Flow profile of Castor-oil and alcohol flowing side-to-side on a straight channel.

### 3.3 Two-components flow patterns

Simulations of Castor-oil and alcohol flowing on Omega and Tesla shaped micromixers produced the flow patterns shown on Fig. (6)

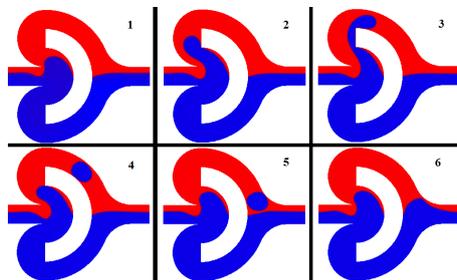


Figure 6. Temporal evolution of flow patterns of Castor-oil and alcohol on Omega mixers.

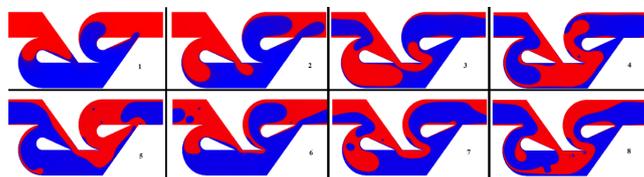


Figure 7. Temporal evolution of flow patterns of Castor-oil and alcohol on Tesla mixers.

## 4. CONCLUSIONS

The lattice Boltzmann Method was used to simulate two different micromixer geometries, Omega and Tesla, using the *sailfish* simulator. This simulator was validated, reproducing the well known Poiseuille flow profile of Castor-oil and alcohol on a straight 2D channel. Flows of Castor-oil and alcohol on complex geometry channels were simulated for the Omega micromixer and for the Tesla micromixer. The interaction parameters of the model were adjusted to reproduce the patterns reported by YuMei *et al.* (2011), that is an approximate case for kerosene-water flow. On continuation, an

experimental setup is being developed for experimental validation of the LBM simulation of this exact kind of binary mixture on microchannels. The experience on using the *sailfish* simulator, that runs on GPUs and is more than ten times faster than the *LB2D\_prime*, that runs on CPU, encouraged us to start the development of a custom simulator to run on a cluster equipped with GPUs, to reach high performance on 3D LBM simulations.

## 5. ACKNOWLEDGEMENTS

This work was supported by CNPq with a PIBIC scholarship, and by FAPESP grant process number 2010/09717-7. The authors also like to acknowledge Michal Januszewski, head of the *sailfish* development team, for his support on questions about his simul

## 6. REFERENCES

- Aidun, C.K. and Clausen, J.R., 2010. "Lattice-Boltzmann Method for Complex Flows". *ANNUAL REVIEW OF FLUID MECHANICS*, Vol. 42, pp. 439–472. ISSN 0066-4189. doi:{10.1146/annurev-fluid-121108-145519}.
- Arias, E.L.M. and Maciel, R., 2010. *Desenvolvimento e Avaliação de Microreatores: Aplicação para Produção de Biodiesel*. Master's thesis, Universidade Estadual de Campinas Faculdade de Engenharia Química.
- Bao, T., 2010. *High density ratio multi-component lattice Boltzmann flo model for fluid dynamics and CUDA parallel computation*. Ph.D. thesis, University of Pittsburgh - Swanson School of Engineering.
- Berthier, J. and Silberzan, P., 2006. *Microfluidics for Biotechnology*. Microelectromechanical Systems Series. Artech House, Inc.
- Hessel, V. and Löwe, H., 2003a. "Microchemical engineering: Components, plants concepts, user acceptance - part 1". *Chemical Engineering and Technology*, Vol. 26, pp. 13–24.
- Hessel, V. and Löwe, H., 2003b. "Microchemical engineering: Components, plants concepts, user acceptance - part 1". *Chemical Engineering and Technology*, Vol. 26, pp. 531–544.
- Hessel, V. and Löwe, H., 2003c. "Microchemical engineering: Components, plants concepts, user acceptance - part 2". *Chemical Engineering and Technology*, Vol. 26, pp. 391–408.
- Mohamad, A.A., 2011. *Lattice Boltzmann Method: Fundamentals and Engineering Applications with Computer Codes*. Springer.
- Nguyen, H., 2007. *GPU Gems 3*. Addison-Wesley Professional. ISBN 9780321515261.
- Owens, J.D., Luebke, D., Govindaraju, N., Harris, M., Kruger, J., Lefohn, A.E. and Purcell, T.J., 2007. "A survey of general-purpose computation on graphics hardware". *Computer Graphics Forum*, Vol. 26, No. 1, pp. 80–113. ISSN 0167-7055. doi:10.1111/j.1467-8659.2007.01012.x. URL <http://dx.doi.org/10.1111/j.1467-8659.2007.01012.x>.
- Succi, S., 2001. *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*. Numerical Mathematics and Scientific Computation. Oxford Science Publications.
- Sukop, M.C. and Thorne Jr., D.T., 2005. *Lattice Boltzmann Modeling: An Introduction to Geoscientists and Engineers*. Springer.
- Wolf-Gladrow, D.A., 2005. *Lattice-Gas Cellular Automata and Lattice Boltzmann Models - An Introduction*, Vol. 1725. Springer.
- YuMei, Y., Chao, Y., Yi, J., Ameya, J., YouChun, S. and SiaoLong, Y., 2011. "Numerial simulatio of immiscible liquid-liquid flow in microchannels using lattice boltzmann method". *Science China Chemistry*, Vol. 54, pp. 244–256.
- Zhang, J., 2011. "Lattice boltzmann method for microfluidics: models and applications". *Microfluid Nanofluid*, Vol. 10.

## 7. RESPONSIBILITY NOTICE

The authors are the only responsables for the printed material included in this paper.