Templated CUDA Lattice Boltzmann Method: generic CFD solver for single and multi-phase problems

Michał Dzikowski[a], Grzegorz Gruszczyński[a,b]

23rd March, 2019

[a]University of Warsaw, Interdisciplinary Centre for Mathematical and Computational Modelling
[b]Warsaw University of Technology, Faculty of Power and Aeronautical Engineering

Table of contents

Introduction

Goal

Simulate an incompressible flow
coupled with advection - diffusion equation of some scalar field.

The continuity and momentum equations are:

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = 0 \\ \rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot (\mu[\nabla \mathbf{u} + (\nabla \mathbf{u})^{\top}]) + \mathbf{F} \end{cases}$$

And the advection - diffusion equation of some scalar field ϕ:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \mathrm{u}) = \nabla \cdot (M \nabla \phi)$$

Lattice Boltzmann Method

Probability of finding a particle in the phase space:

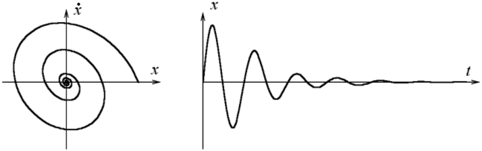


Figure 1: $\Psi = \Psi(t, \mathbf{x}, \dot{\mathbf{x}})$

In an infinitesimally small volume of the phase space $d\mathbf{x}d\mathbf{u}$:

$$\Psi_{no\ collisions}(t + dt, \mathbf{x} + d\mathbf{x}, \mathbf{u} + d\mathbf{u})d\mathbf{x}d\mathbf{u} = \Psi_{no\ collisions}(t, \mathbf{x}, \mathbf{u})d\mathbf{x}d\mathbf{u}$$

Now, include the collision term $\mathbb{C}(\Psi)$:

$$\Psi(t + dt, \mathbf{x} + d\mathbf{x}, \mathbf{u} + d\mathbf{u})d\mathbf{x}d\mathbf{u} = \Psi(t, \mathbf{x}, \mathbf{u})d\mathbf{x}d\mathbf{u} + \mathbb{C}(\Psi)d\mathbf{x}d\mathbf{u}dt$$

Taylor series expansion:

$$\Psi(t+dt, \mathbf{x}+d\mathbf{x}, \mathbf{u}+d\mathbf{u}) = \Psi(t, \mathbf{x}, \mathbf{u}) + \frac{\partial \Psi}{\partial t} dt + \nabla_{\mathbf{x}} \Psi d\mathbf{x} + \nabla_{\mathbf{u}} \Psi d\mathbf{u}$$

Plug in:

$$\left[\Psi(t, \mathbf{x}, \mathbf{u}) + \frac{\partial \Psi}{\partial t} dt + \nabla_{\mathbf{x}} \Psi d\mathbf{x} + \nabla_{\mathbf{u}} \Psi d\mathbf{u}\right] d\mathbf{x} d\mathbf{u} = \left[\Psi(t, \mathbf{x}, \mathbf{u}) + \mathbb{C}(\Psi) dt\right] d\mathbf{x} d\mathbf{u}$$

Reformulate velocity $\mathbf{u} = \frac{d\mathbf{x}}{dt}$ and acceleration $\frac{d\mathbf{u}}{dt} = \frac{\mathbf{F}}{\rho}$:

$$\frac{\partial \Psi}{\partial t} + (\mathbf{u} \cdot \nabla_{\mathbf{x}}) \Psi + (\frac{\mathbf{F}}{\rho} \cdot \nabla_{\mathbf{u}}) \Psi = \mathbb{C}(\Psi)$$

Streaming and Collision:

$$\underbrace{\Psi(t+dt,\mathsf{x}+d\mathsf{x},\mathsf{u}+d\mathsf{u})d\mathsf{x}d\mathsf{u}}_{\textit{Streaming}} = \underbrace{\Psi(t,\mathsf{x},\mathsf{u})d\mathsf{x}d\mathsf{u}+\mathbb{C}(\Psi)d\mathsf{x}d\mathsf{u}dt}_{\textit{Collision}}$$

The Boltzmann equation can be viewed as a substantial derivative (of an intensive quantity Ψ) which is equal to the collision term $\mathbb{C}$ applied to the distribution function of Ψ:

$$\frac{\partial\Psi}{\partial t}+(\mathsf{u}\cdot\nabla_{\mathsf{x}})\Psi+(\frac{\mathsf{F}}{\rho}\cdot\nabla_{\mathsf{u}})\Psi=\mathbb{C}(\Psi)$$

$$\underbrace{f_i(\mathrm{x} + \mathrm{e_i}\Delta \mathrm{x}, t + \Delta t)}_{\textit{Streaming}} = \underbrace{f_i(\mathrm{x}, t) - \frac{1}{\tau}(f_i - f_i^{eq}) + F_i(\mathrm{x}, t)}_{\textit{Collision}}$$

- $\tau = \tau(\nu)$ relaxation parameter, ν is the kinematic viscosity
- f_i - discrete probability distribution function
- F_i - source term (ex. gravity force)

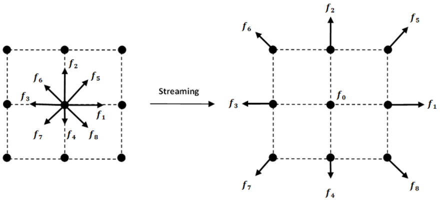


Figure 2: D2Q9: Streaming

Algorithm: Fluid

1 Initialize f_i^{in}

2 Compute
$\rho = \sum_{i=0}^{8} f_i^{in}(\mathrm{x}, t)$ *and* $\mathrm{u}(\mathrm{x}, t) = \frac{1}{\rho} \sum_{i=0}^{8} f_i^{in}(\mathrm{x}, t)\mathrm{e}_i + \frac{\mathbf{F}}{2\rho}\delta t$

3 Compute $f_i^{eq}(\mathrm{x}, t) = w_i \rho \left[1 + \frac{\mathrm{e}_i \mathrm{u}}{c_s^2 e^2} + \frac{(\mathrm{e}_i \mathrm{u})^2}{2 c_s^4 e^4} - \frac{\mathrm{u}^2}{2 c_s^2 e^2}\right]$ where $c_s^2 = \frac{1}{3}$

4 Collision $f_i^{out}(\mathrm{x}, t) = f_i^{in}(\mathrm{x}, t) - \frac{1}{\tau_f}\left[f_i^{in}(\mathrm{x}, t) - f_i^{eq}(\mathrm{x}, t)\right] + F_i(\mathrm{x}, t)$

5 Streaming $f_i^{in}(\mathrm{x} + \mathrm{e}_i, t + 1) = f_i^{out}(\mathrm{x}, t)$

Multiphysics models

How to re-use the same algorithm to track temperature field or interface between phases?

Consider Advection-Diffusion Equation of scalar field ϕ.

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \mathrm{u}) = \nabla \cdot (M \nabla \phi)$$

The ϕ field can be viewed as a temperature ... or a phase indicator.

Imagine a separation flux, $\mathbf{j}_S$, supposed to counteract the diffusion and reach a predefined interface profile in the equilibrium state:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \mathbf{u}) = \nabla \cdot (\underbrace{M \nabla \phi}_{\mathbf{j}_D} - \mathbf{j}_S)$$

Let us use a *tanh* to smooth the step interface:

$$\phi^{eq} = \frac{1}{2} tanh \left(\frac{2(\mathbf{x} - \mathbf{x}_0)}{\gamma} \right)$$

Evaluate diffusive flux in the equilibrium:

$$\mathbf{j}_D^{eq} = M\nabla \overbrace{\left[\frac{1}{2} tanh\left(\frac{2(\mathbf{x}-\mathbf{x}_0)}{\gamma}\right)\right]}^{\phi^{eq}} = \frac{M}{2}\mathbf{n}\frac{\partial}{\partial \mathbf{x}_n} tanh\left[\left(\frac{2(\mathbf{x}-\mathbf{x}_0)}{\gamma}\right)\right]$$

$$= \frac{M}{\gamma}\mathbf{n}\underbrace{\left[1 - tanh^2\left(\frac{2(\mathbf{x}-\mathbf{x}_0)}{\gamma}\right)\right]}_{1-4(\phi^{eq})^2}$$

Therefore:

$$\mathbf{j}_S = M\mathbf{n}\frac{1-4\phi^2}{\gamma} \qquad where \quad \mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}$$

Conclusion

Modifying the advection-diffusion equation:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \mathrm{u}) = \nabla \cdot (M \nabla \phi)$$

The interface tracking equation can been obtained:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \phi \mathbf{u} = \nabla \cdot M \left(\nabla \phi - \frac{\nabla \phi}{|\nabla \phi|} \frac{[1 - 4(\phi - \phi_0)^2]}{\gamma} \right)$$

Double distribution function approach

Algorithm: Advection - Diffusion

'Advection - Diffusion' of ϕ is solved on a separate D2Q9 lattice

1 Initialize $h_i^{in}(\mathrm{x}, t)$

2 Compute $\phi = \sum_{i=0}^{9} h_i^{in}(\mathrm{x}, t)$

3 Compute $h_i^{eq}(\mathrm{x}, t) = \phi w_i \left[1 + \frac{\mathrm{e}_i \mathrm{u}}{c_s^2 e^2} + \frac{(\mathrm{e}_i \mathrm{u})^2}{2 c_s^2 e^4} - \frac{\mathrm{u}^2}{2 c_s^2 e^2}\right]$

4 Collision $h_i^{out}(\mathrm{x}, t) = h_i^{in}(\mathrm{x}, t) - \frac{1}{\tau_\phi}\left[h_i^{in}(\mathrm{x}, t) - h_i^{eq}(\mathrm{x}, t)\right] + F_i^{\phi}(\mathrm{x}, t)$

5 Streaming $h_i^{in}(\mathrm{x} + \mathrm{e}_i, t + 1) = h_i^{out}(\mathrm{x}, t)$

Sample simulations

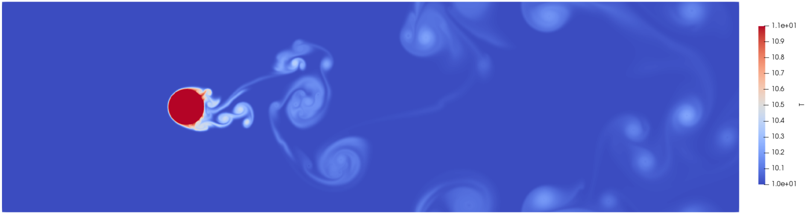


Figure 3: Re = 1000, Pr = 0.71, D = 128 [lu]

Scenarios

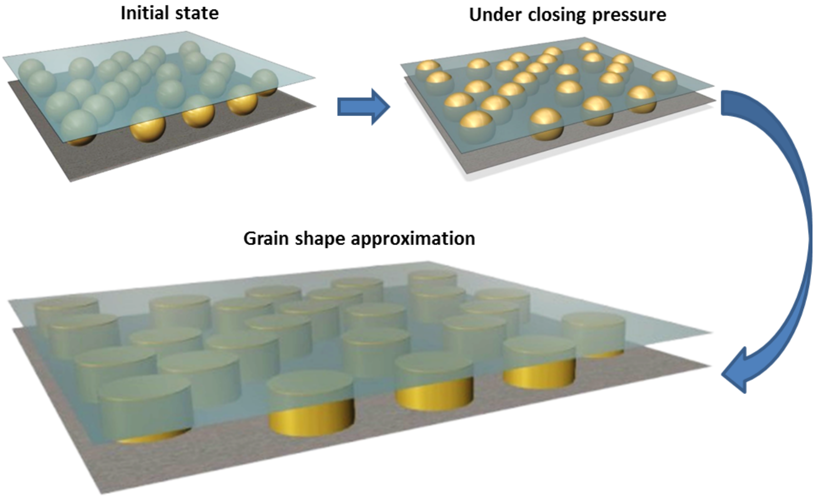
Case-ID	Lattice Size	Velocity set	Blockage Ratio	D	U	Pr	Re	ν	k
$Pr10_{small}$	1000x150x3	D3Q27Q27	1/5	30	0.01	10	10	3E-02	3E-03
$Pr10_{medium}$	2000x300x3	D3Q27Q27	1/5	60	0.005	10	10	3E-02	3E-03
$Pr10_{large}$	4000x600x3	D3Q27Q27	1/5	120	0.0025	10	10	3E-02	3E-03
$Pr100_{small}$	1000x150x3	D3Q27Q27	1/5	30	0.01	100	10	3E-02	3E-04
$Pr100_{medium}$	2000x300x3	D3Q27Q27	1/5	60	0.005	100	10	3E-02	3E-04
$Pr100_{large}$	4000x600x3	D3Q27Q27	1/5	120	0.0025	100	10	3E-02	3E-04
$Pr1000_{small}$	1000x150x3	D3Q27Q27	1/5	30	0.01	1000	10	3E-02	3E-05
$Pr1000_{medium}$	2000x300x3	D3Q27Q27	1/5	60	0.005	1000	10	3E-02	3E-05
$Pr1000_{large}$	4000x600x3	D3Q27Q27	1/5	120	0.0025	1000	10	3E-02	3E-05

Table 1: Case-ID: lookup table.

Case-ID	$Nu^{1st\ order\ bc}_{CM\ HIGHER}$	$Nu^{2nd\ order\ bc}_{CM\ HIGHER}$	$Nu^{1st\ order\ bc}_{Cumulants}$	$Nu^{2nd\ order\ bc}_{Cumulants}$	Nu_{FEM}
$Pr10_{small}$	4.91	4.81	5.04	4.83	4.82
$Pr10_{medium}$	4.86	4.81	4.92	4.82	4.82
$Pr10_{large}$	4.83	4.81	4.87	4.81	4.82
$Pr100_{small}$	10.66	10.27	14.50	11.52	10.1
$Pr100_{medium}$	10.32	10.13	11.82	10.36	10.1
$Pr100_{large}$	10.19	10.08	10.83	10.13	10.1
$Pr1000_{small}$	27.09	24.58	94.31	58.33	21.43
$Pr1000_{medium}$	22.73	21.84	53.97	34.19	21.43
$Pr1000_{large}$	21.84	21.37	34.04	24.56	21.43

Table 2: Influence of kernel and BC on Nu number.
1st order bc = BB (hydrodynamics) + EQ (thermodynamics)
2nd order bc = IBB (hydrodynamics) + IABB (thermodynamics)

Multiphase flow in fractures with proppant

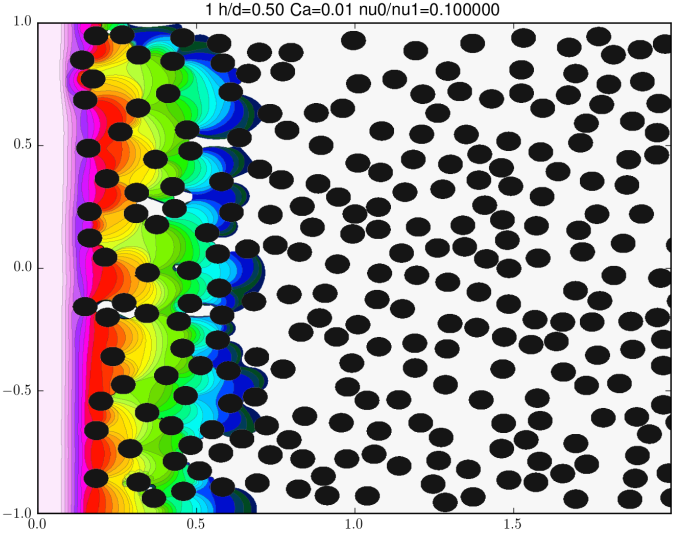


Movie:

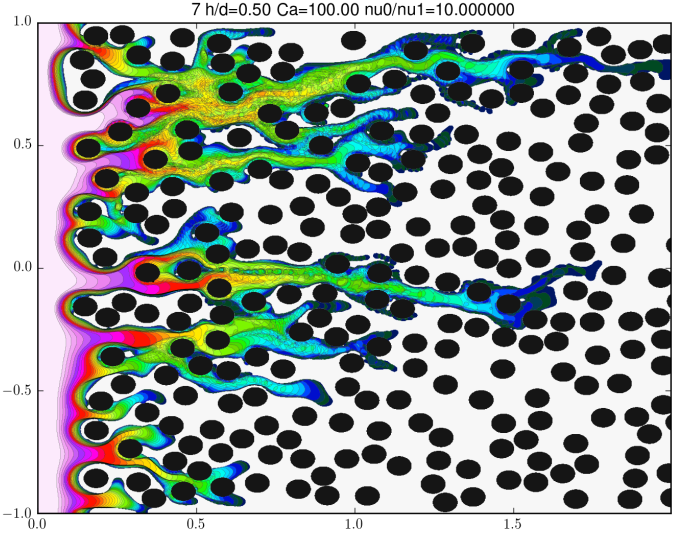
https://www.youtube.com/watch?v=JoMFy2M_RDI

Capillary invasion

And multiphase flow (strong Capilary intraction, >10mln LBM cells):



And multiphase flow (weak Capilary intraction, >10mln LBM cells):



Small system - detailed mesh

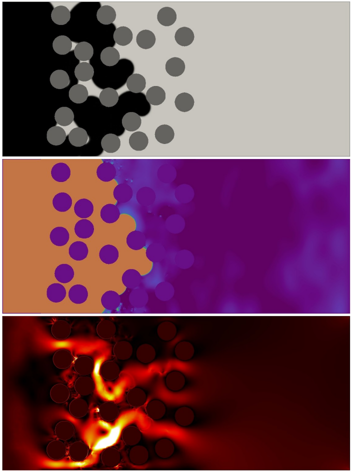


Figure 4: Viscous invasion.

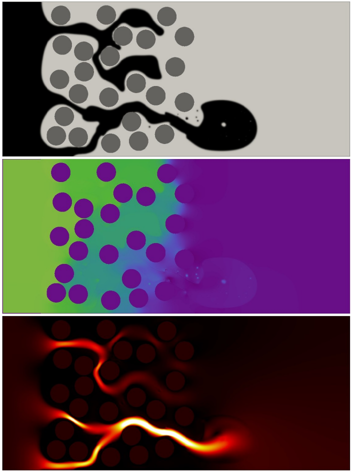
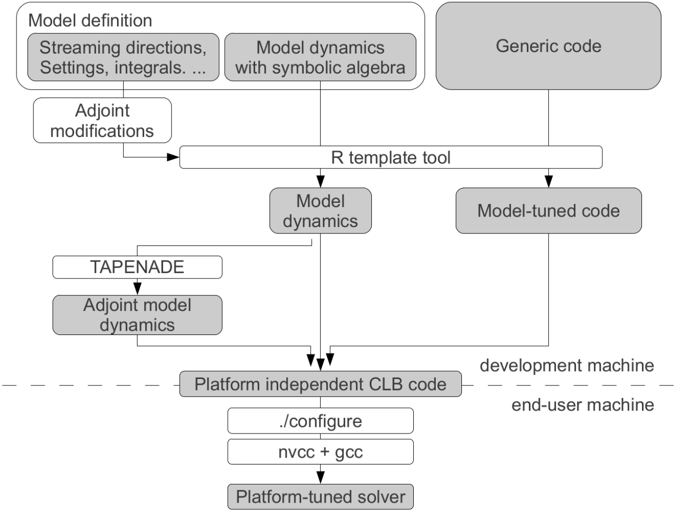


Figure 5: Capillary invasion.

Solver Structure

Solver Structure



Memory requirements

Memory requirements

	No [-]	[double]	[Bytes]
DF	2x27	54	432
DF temp	2x27	54	432
q	2x27	13.5	108
flag	1	0.5	4
memory per node		122	976

Table 3: Theoretical memory requirements for:
3D, DDF model with interpolated BC.

Memory requirements

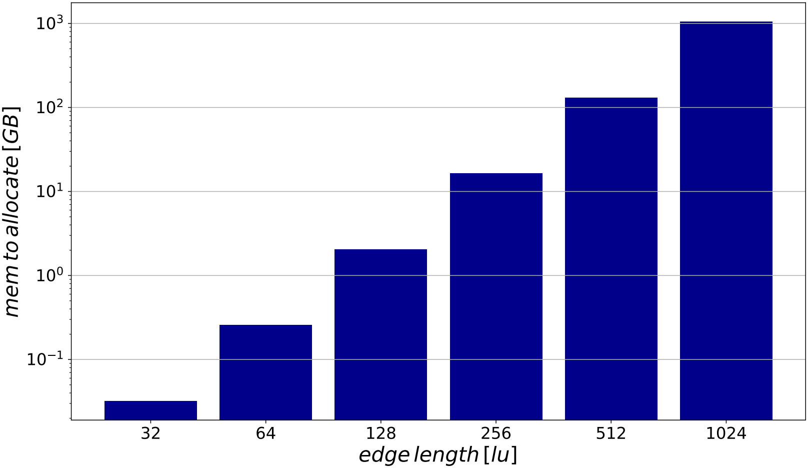


Figure 6: Theoretical memory requirements for a 3D cube domain, D3Q27Q27 lattice with interpolated BC.

Q: Does the high memory requirement limits applicability of LBM?

Not necassary. Similar problems occur in other CFD methods like VOF, FEM, FD.

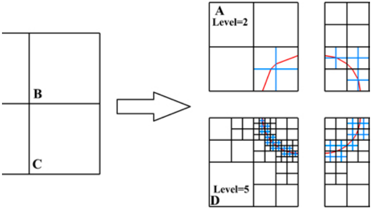


Figure 7: Mesh refinement.

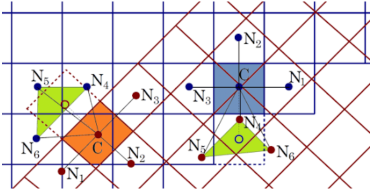


Figure 8: Overlapping mesh.

Questions?

?

https://github.com/CFD-GO/TCLB