LA-UR-14-28350
Approved for public release; distribution is unlimited.

Title: FY14 Report: Lattice Boltzmann Modeling of Microreactor Systems

Author(s): Roberts, Randy Mark

Intended for: Report

Issued: 2014-10-27

FY14 Report: Lattice Boltzmann Modeling of Microreactor Systems

Randy M. Roberts

October 15, 2014

1 Summary of Last Year's Results

Last year's work saw the implementation of the Flekkøy advective-diffusion model[Fle93] into the OpenLB parallel C++ lattice Boltzmann simulation framework[HK10].

Unfortunately this model does not adequately describe multicomponent diffusion. To correctly model multicomponent diffusion we require a model that more accurately approaches Maxwell-Stefan diffusion[Asi09].

In addition to the advection-diffusion model, we implemented an immiscible flow model based on the Swift-Osborn model[SOOY96]. At the time it seemed that the Swift-Osborn model was the appropriate model for immiscible multicomponent systems. It has since been demonstrated that this model, along with several other established lattice Boltzmann immiscible flow models, suffer from unphysical parasitic currents at the boundaries between the components[Lee09, CL12].

An example of the the spurious currents calculated around a bubble is shown in Figure 1.

2 Lattice Boltzmann Method for Advective-Diffusive Multicomponent Systems

I began researching the alternatives to the Flekkøy advective-diffusion model, in order to more accurately model the mass transfer in our microfluidics system.

At first I looked into Luo's model[LG02, LG03] for binary mixtures. This model model adds extra force terms to the lattice Boltzmann equation to account for the coupling between mixture components. I sent Professor Luo emails asking for implementation details for his model. I was informed by him that he did not feel his model was ready to use in actual applications, since it is only a first-order accurate model. I proceeded to look into other models as alternatives.

The next model I researched was Arcidiacono's model[AKMF07]. Arcidiacono's model recovers the Maxwell-Stefan diffusion equations only in the mixture-averaged diffusion approximation (MADA)[Asi09]. The

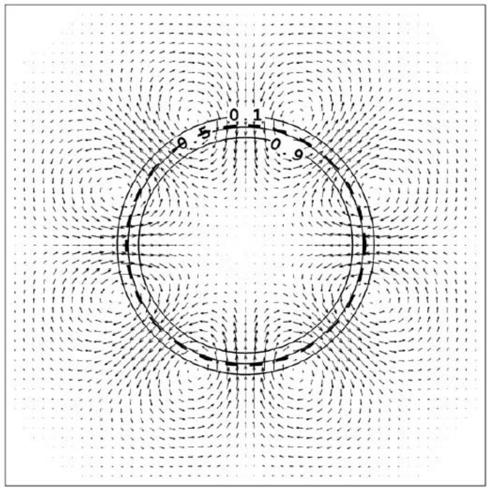

Figure 1: An example of spurious currents. An artificial velocity field, with the same underlying symmetry as the lattice, persists indefinitely, preventing the achievement of an equilibrium state. The numbers indicate contours of the order parameter distinguishing phases. The image is courtesy of Refs. [Lee09, CL12].

MADA approximation assumes that species flow speeds are small deviations from the barycentric velocity of the mixture.

I determined that Asinari's model[Asi09] is most promising for this project. It recovers the Maxwell-Stefan diffusion equation in the continuum limit for larger deviations from the barycentric velocity of the mixture.

It may be that the MADA approximation is good enough for this project. This needs to be further investigated.

3 Lattice Boltzmann Method for Immiscible Multicomponent Systems

After it became apparent that the Swift-Osborn model produced unphysical spurious currents that could adversely affect mass transfer rates for the microreactor system, I started investigating alternative immiscible multicomponent models. The most successful models were based on the performing a transformation of the density variables into pressure variables. This work was pioneered by He[HCZ99].

Lee improved He's approach by developing a more stable and accurate discretization of the resulting equations[LL05]. I had decided to implement this method to model the simulation of the microreactor system.

Since Lee's improved method (He-Lee) results in non-local finite-element stencils for the lattice Boltzmann calculation, OpenLB would have to be drastically modified in order to incorporate this method into its framework. Fortunately, I discovered the Palabos lattice Boltzmann simulation framework[Pala].

Palabos is another parallel C++ lattice Boltzmann framework based on the OpenLB C++ code base that provides the necessary API to implement non-local lattice Boltzmann models. Neither OpenLB nor Palabos were architected with non-local stencils in mind, but Palabos has at least provided a usable API to implement non-local stencils without having to modify its internal code base.

The importance of supporting non-local stencils is not just from an ease of programming perspective. The real importance of an API supporting non-local stencils comes from the parallel implementation of the lattice Boltzmann method.

A lattice Boltzmann calculation usually consists of two main phases, calculating the collision terms on a cell, and then streaming the results of the collisions to neighboring cells. To parallelize a lattice Boltzmann calculation the domain is decomposed into sub-domains, the sub-domains are distributed across the parallel processors.

When the lattice Boltzmann algorithm is implemented with local stencils all of the data required by the collision operator are calculated from data that are local to that processor. The only time that data are required from other processors is during the streaming phase of the calculation, which is efficiently handled for you by the lattice Boltzmann framework.

When non-local stencils are required for the collision term, a carefully choreographed interweaving of data calculation and communication is required to ensure that the correct data are available on the local processor from calculations performed on other processors. Palabos' API allows the developer to specify that choreographed interweaving of data calculation and communication.

As an extra bonus, Palabos provided an example implementation of the He-Lee model. Unfortunately this model was written exclusively as a 3D model, and did not include any implementation of wall boundary conditions. It is important to run the microreactor simulations in a 2D mode in order to provide insightful results within a reasonable timeframe, without having to run the simulations on large parallel machines.

Abstracting the He-Lee sample code from 3D to a single code that supported both 2D and 3D also allowed me to clean up the sample code to be a more flexible implementation. I expect this refactoring to earn its dividends when I begin to implement wall boundary conditions in the model.

Results from the 2D code, run on initially elliptical shaped bubbles with a density ratio and viscosity ratio of 10.0, are shown in Figures 3, 4, and 5. Results from the 3D code, run on initially ellipsoidally shaped bubbles with a density ratio and viscosity ratio of 10.0, are shown in Figures 6, 7, 8, and 9. The two bubbles were set in a velocity field leading each bubble eventually to collide with the other. The initial velocities and bubble radii of the two simulations are the same. The differences in character of the two runs result from the different geometries of the two simulations, 2D vs. 3D.

Additionally, one can notice some non-physical bubble shrinkage, especially in the 3D simulation. This phenomena has been examined by Zheng[ZLGR14]. It seems at this time that bubble shrinkage is an unavoidable consequence of free-energy, phase-field based models, though their effects can be mitigated. I will need to incorporate the results of this paper into further work on the microreactor system.

I had begun running 2D and 3D test problems, but it soon became apparent that without adequate wall boundary conditions it would be impossible to simulate the geometries being employed by the experimental part of the project.

3.1 Wall boundary conditions

I've just begun investigating methods to implement wall boundary conditions within the He-Lee model. Lee has proposed a non-local model for wall boundaries based on contact angles[LL08, LL09], θ_C in Figure 2.

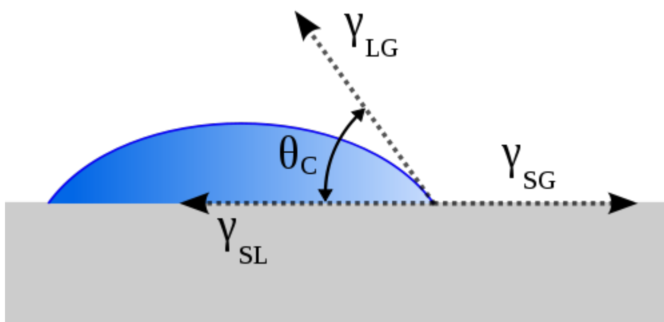

Figure 2: Schematic of a liquid drop showing the quantities in Young's equation. From [Wik].

It took me a while to figure out how to incorporate non-local boundary conditions into Palabos. The normal way to incorporate boundary conditions into Palabos requires local stencils for the boundary condition calculations.

Treating the boundary conditions as you would the non-local collision terms requires processing of bulk data for calculations that should only require iterating over boundary cells. This would be very inefficient. Additionally, this would lead to difficulties determining, during the boundary condition calculations, which cells were boundary cells, which cells were interior cells, and what were the boundary meta-information (see below).

I finally came up with a strategy that I think will work, though I have not yet tried to implement it. One would create a non-local data processor in much the same way that one is created for non-local collision term calculations. It seems that Palabos allows you to restrict the execution of a data processor to a developer-specified sub-domain of the problem. In this case I would restrict the data processor to execute only along the boundaries of the problem. See the Palabos documentation for data processors[Palb] for more information.

This is not as easy as it sounds, since boundary conditions require meta-information for each boundary. The boundary type (planar, interior edge, exterior edge, interior corner, or exterior corner) and boundary orientation (e.g. $\pm\hat{x}$, $\pm\hat{y}$, or $\pm\hat{z}$ direction for planar type) need to be specified for each segment of the boundary. This information would have to somehow be encoded into the boundary data processors.

4 Challenges

The first work that needs to be accomplished is the implementation of wall boundary conditions for the He-Lee model. This now seems straight forward, after overcoming the issues of non-local boundary conditions within the Palabos framework, but the devil is in the details.

The major challenge that I foresee is how to implement the Asinari diffusion model along with the He-Lee immiscible flow model. Each one was not designed with the other in mind. I have some ideas, but they are in their infancy.

References

[AKMF07] S. Arcidiacono, I. V. Karlin, J. Mantzaras, and C. E. Frouzakis. Lattice Boltzmann model for the simulation of multicomponent mixtures. *Physical Review E*, 76:046703, 2007.

[Asi09] P. Asinari. Lattice boltzmann scheme for mixture modeling: Analysis of the continuum diffusion regimes recovering Maxwell-Stefan model and incompressible Navier-Stokes equations. *Physical Review E*, 80(5):056701, 2009.

[CL12] K. Connington and T Lee. A review of spurious currents in the lattice Boltzmann method for multiphase flows. *Journal of Mechanical Science and Technology*, 26(12):3857–3863, 2012.

[Fle93] E. G. Flekkøy. Lattice Bhatnagar-Gross-Krook models for miscible fluids. *Physical Review E*, 47(6):4247, 1993.

[HCZ99] X. He, S. Chen, and R. Zhang. A lattice Boltzmann scheme for incompressible multiphase flow and its application in simulation of Rayleigh–Taylor instability. *J. Comput. Phys.*, 152:642–663, 1999.

[HK10] V. Heuveline and M. J. Krause. OpenLB: towards an efficient parallel open source library for lattice Boltzmann fluid flow simulations. In PARA, editor, *International Workshop on State-of-the-Art in Scientific and Parallel Computing*, volume 9, 2010.

[Lee09] T. Lee. Effects of incompressibility on the elimination of parasitic currents in the lattice Boltzmann equation method for binary fluids. *Computers and Mathematics with Applications*, 58(5):987–994, 2009.

[LG02] L.-S. Luo and S.S. Girimaji. Lattice Bolzmann model for binary mixtures. *Physical Review E*, 66:035301(R), 2002.

[LG03] L.-S. Luo and S.S. Girimaji. Theory of the lattice Boltzmann method: two-fluid model for binary mixtures. *Physical Review E*, 67:036302, 2003.

[LL05] T. Lee and C.-L. Lin. A stable discretization of the lattice Boltzmann equation for simulation of incompressible two-phase flows at high density ratio. *Journal of Computational Physics*, 206:16–47, 2005.

[LL08] T. Lee and L. Liu. Wall boundary conditions in the lattice Boltzmann equation method for non-ideal gases. *Physical Review E*, 78:017702, 2008.

[LL09] L. Liu and T. Lee. Wall free energy based polynomial boundary conditions for non-ideal gas lattice Boltzmann equation. *International Journal of Modern Physics C*, 20(11):1749–1768, 2009.

[Pala] Palabos. http://www.palabos.org.

[Palb] Palabos. http://www.palabos.org/documentation/userguide/data-processors.html.

[SOOY96] M. R. Swift, E. Orlandini, W. R. Osborn, and J. M. Yeomans. Lattice Boltzmann simulations of liquid-gas and binary fluid systems. *Physical Review E*, 54(5):5041, 1996.

[Wik] Wikipedia. http://en.wikipedia.org/wiki/contact_angle.

[ZLGR14] L. Zheng, T. Lee, Z. Guo, and D. Rumschitzki. Shrinkage of bubbles and drops in the lattice Boltzmann equation method for nonideal gases. *Physical Review E*, 89:033302, 2014.

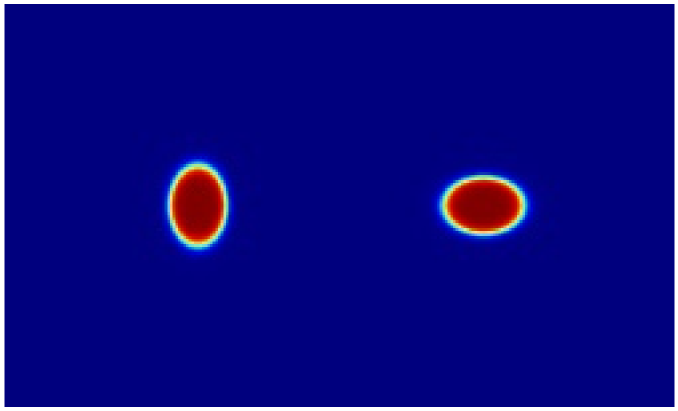

Figure 3: 2D Results t= 0

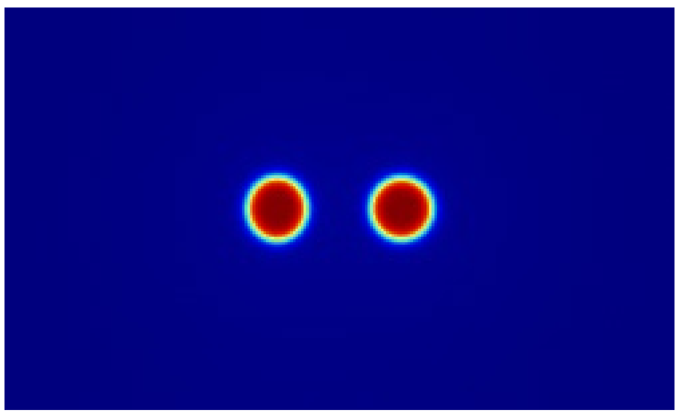

Figure 4: 2D Results t=20040

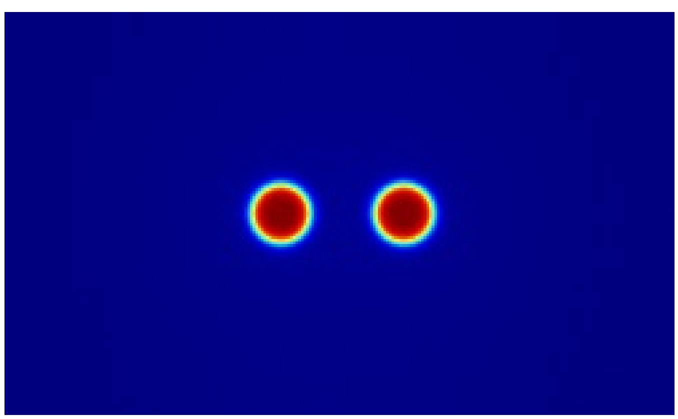

Figure 5: 2D Results t=39960

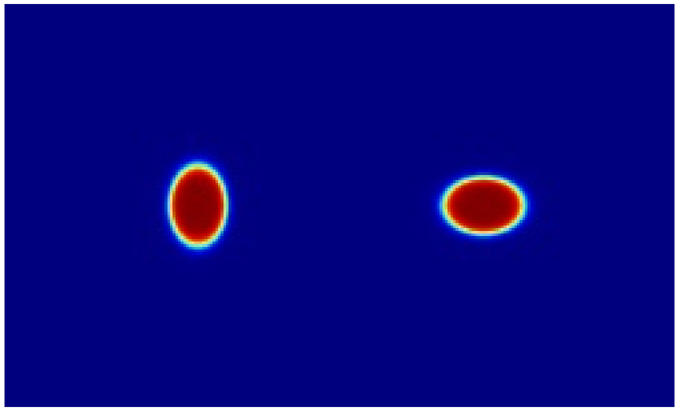

Figure 6: 3D Results t=0

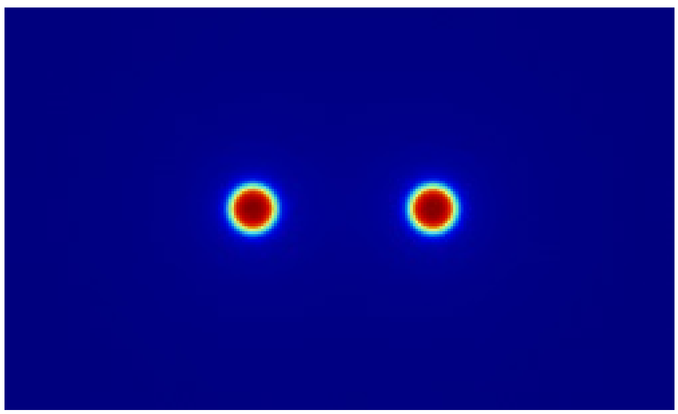

Figure 7: 3D Results t=20040

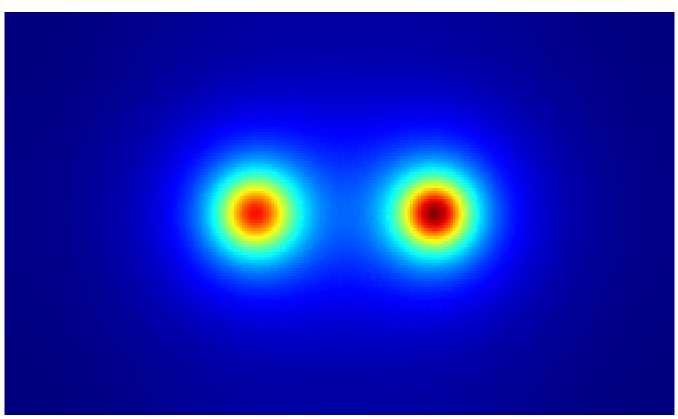

Figure 8: 3D Results t=30000

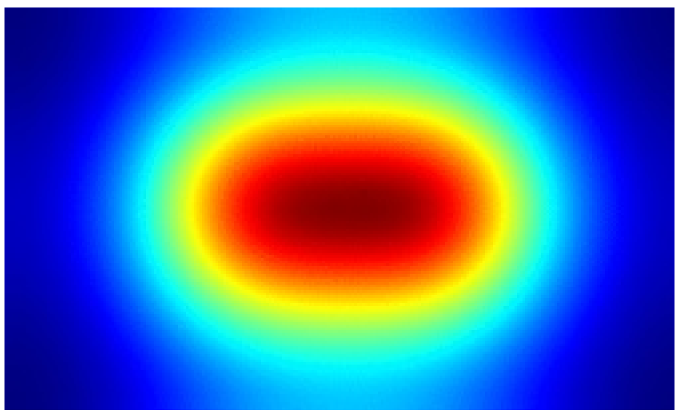

Figure 9: 3D Results t=39960