A comparison of two computational models for simulations of red blood cells in flow

Iveta Jančigová[1], iveta.jancigova@fri.uniza.sk

[1] Cell-in-fluid research group, KST, FRI, University of Žilina

Keywords: numerical model, simulation, red blood cell, fluid flow
Abstract: Modelling blood flow on micro-scale for various biomedical applications is currently a very active field of study. However, those modelled processes are so complex, that only rarely do the research groups implement more than one model. Usually, the models are calibrated using sets of experiments with biological cells. It is then fairly difficult to compare the computational models among themselves or assess the quality of a new model. In this work, we propose a few comparison tests for this kind of models in general and use them for comparison of two specific models. Both of them use Lattice-Boltzmann Method for the fluid - blood plasma. The fluid-object coupling is done using the Immersed Boundary Method with indirect velocity computations in one of them and using dissipative drag force approach in the other. The elastic objects are modelled using Finite Elements Method in the first one and using a spring network model in the second one.

1 Introduction

Computational models are a very useful tool for gaining insight into blood flow on micro-scale - on the level of individual elastic cells. Some of them are spectrin models [1], various spring network models (SNM), e.g. [2] or [3], finite elements methods (FEM) in continuous models [4], etc. Each of these has its advantages and disadvantages, but what they all share is a significant complexity, especially if one wants to resolve a full 3D dynamics with two-way coupling to the fluid, while keeping the model scalable so that it can be used for running large and/or dense simulations.

As a consequence, only rarely do the research groups implement more than one of these models and so it is difficult to directly compare them. In our previous work [5], we have described the implementation of our model (basic overview also follows in the next Section) and we have been interested in how it compares to the current state-of-the-art. Except for our implementation, which is available as part of

the scientific package ESPResSo [6], we are not aware of any other free open source implementation. Therefore, we have contacted Prof. Timm Krüger, computational physicist at Edinburgh University, who kindly agreed to share information and data about his model [7] for comparison purposes.

2 Description of models

In this section we briefly describe the two models: Prof. Krüger's model (Model 1) and our model (Model 2). Both of them compute the fluid using lattice-Boltzmann method (LBM) with D3Q19 lattice.

2.1 Model 1: LBM + IBM + FEM

Model 1 is capable of 2, 3 and 4-point stencil discretisation for Immersed Boundary Method (IBM), however, only the 2-point was used in the comparisons, since the linear 2-point interpolation is the closest match for the coupling method in Model 2. The coupling in Model 1 enforces the no slip boundary condition and uses indirect position updates, which make the model sensitive to velocity interpolation errors. This model uses the finite elements method for the elastic objects, e.g. red blood cells.

2.2 Model 2: LBM + DF + SNM

In Model 2, we use a spring network model (SNM) for the elastic objects. It means that we apply elastic forces to the triangulation points on the surface of the object. These forces provide a mechanism for elastic deformations: stretching, bending, local and global area conservation and volume conservation. The SNM is coupled to LBM by dissipative drag force (DF): $F_d = \xi(\mathbf{v} - \mathbf{u})$, where ξ is a drag coefficient, $\mathbf{v}$ is the velocity of the object's boundary point and $\mathbf{u}$ is the fluid velocity interpolated to this point. The disadvantage of this method is that it requires a non-zero difference of velocities on the object's boundary in order to have movement, i.e. the no-slip condition is not satisfied. On the other hand, it offers stability in cases when fluid lattice and object triangulation are coarse.

3 Proposed comparison tests for the models

While the two models are capable of modelling the same processes, their comparison is not quite straightforward. For example, the elasticity is implemented differently and thus a straightforward exchange of elastic parameters is not helpful. Therefore, we propose the following comparison simulations that reveal the model behavior of interest.

Test A: Efficiency of simulator In order to compare the overall performance of the implemented models, we propose a simulation of rectangular channel with some cells that do not undergo collisions and simulate the same amount of time. We compare the computational time needed for this simulation.

Test B: Comparison of elasticity models To compare the elasticity of RBCs, we propose a simulation of single cell in steady flow in a periodic rectangular chamber. The cell assumes a parachute shape due to the parabolic profile of the fluid flow. Quantitative investigation of such shapes of biological red blood cells in microcapillary flow is described in [8]. In the model comparison, we look at the deformation and peak velocity under the same conditions.

Test C: Comparison of dynamics To compare the dynamics of the models, we propose to look at single cell under shear flow of given rate. Depending on the rate, the cell will undergo either tumbling or transition or tanktreading motion and one can compare the deformation of shape, inclination angle and frequency of rotation. Furthermore, after the shear flow is turned off, one can observe the return of the deformed cell to biconcave discoid shape and measure the time needed for such relaxation.

Test D: Object-object interactions Blood is a dense suspension of cells (red blood cells comprise $\sim 45\%$ of blood volume) and in dense simulations, the cell-cell collisions are very frequent. To observe those, we propose a simulation of a channel with a barrier that only has a small opening - such that one cell can pass through if it deforms significantly. In the simulation, two cells start at the beginning of the channel, are brought together by the flow and collide as they try to enter the opening. One can observe the fluid velocity field and the deformations of cells.

4 Results of comparisons

As of now, the first two of the described comparison tests have been performed for models introduced in Section 2. On the other hand, we include an additional Test 0, since the comparison tests could not have been done on the same machines and we had to start with a very basic comparison of hardware.

Test 0: Very basic comparison of hardware The benchmark test was the same C code (compiled with O3 optimisation) run on all machines for a brute force computation of all prime numbers smaller than 10^8. The times needed for this computation are summarised in Table 1, where laptop 1 is Prof. Krüger's machine (2.7 GHz Intel i7-2620M) and laptop 2 (2 GHz Intel Core i7) and Tesla (Nvidia M2090, 2.4 GHz

Intel Xeon E5-2609 CPUs) are ours. Language C was chosen because the computational cores of both implementations of described models are in C/C++.

Table 1: Time needed for execution of the same C code on different machines.

	L1, Intel i7-2620M 2.7 GHz	L2, Intel Core i7 2.0 GHz	Tesla, Intel Xeon E5-2609 1 CPU: 2.4 GHz
time [s]	217	319	567

Test A: Efficiency of simulator We have performed a simulation of rectangular channel with dimensions $100 \times 40 \times 40\mu m^3$ and grid size $1\mu m$ (160000 lattice Boltzmann nodes). We have placed 16 red blood cells (with roughly the same number of triangulation nodes) at the beginning of the channel, Figure 1, and measured the time needed to simulate 1000 times steps. The breakdown of computational time is shown in Table 2.

Table 2: Breakdown of computational time needed for Test A, in seconds. Note: We have measured membrane computations and IBM together.

machine	L1	L2	Tesla (LBM on GPU)
initialisation	0.08	7.6	7.7
LBM	31.46	64.3	0.85
membrane + IBM	13.05 + 2.60	92.4	102.8
total	47.19	164.3	111.35

While the overall performance is worse in our case, the LBM is running extremely fast on the GPU. This means that the focus needs to be on how to speed up the membrane calculations. While not shown in this table, in our case, the cell-cell interactions in particular appear to be the most serious bottleneck.

Test B: Comparison of elasticity models For this test, we have performed a simulation of single cell in a periodic rectangular chamber with dimensions $20 \times 10 \times 10\mu m^3$, Figure 2 left. The cell triangulation had 1002 nodes. The applied force density on the fluid was $f = 1 \cdot 10^5 N/m^3$, the inner and outer kinematic viscosities were the same, $\nu = 1 \cdot 10^{-6} m^2/s$, and fluid density was $\rho = 1000 kg/m^3$ (water).

The elastic coefficients in both models were set to correspond to physical values of $k_s \sim 5 \cdot 10^{-6} Nm^{-1}$ and $k_b \sim 2 \cdot 10^{-19} Nm$ [9]. The local and global area constraints were set in such a way that the area would not differ by more than 5 and 1% respectively form the values in relaxed state. The volume was also preserved.

The peak cell velocity was $6.3 \cdot 10^4 ms^{-1}$ in Model 1 and $5.0 \cdot 10^4 ms^{-1}$ in Model 2. The small difference most likely due to the different mass of the cells.

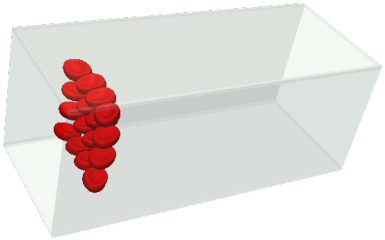

Figure 1: 16 regularly placed red blood cells in a rectangular channel. Fluid flowing from left to right

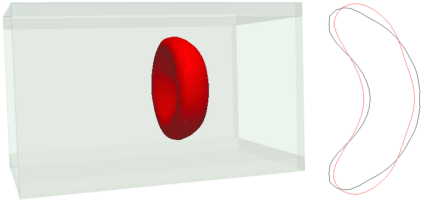

Figure 2: Left: Parachute shape of the cell in steady periodic flow. Right: Two parabolic profiles of red blood cell: Model 1 in black, Model 2 in red

These velocities indicate shear rate $\sim 100s^{-1}$, which is the rate under which red blood cells certainly undergo elastic deformations. The resulting parachute profiles of the cell are depicted in Figure 2 right. While there is a small difference between them, they are qualitatively very similar and this shows that our model is capable of capturing elastic deformations similarly to Model 1.

5 Conclusion

In this work, we have proposed four simulation tests for comparisons of models of red blood cells in flow. We have used two of these tests to compare the efficiency of two implementations and behavior of the simulated cell that undergoes deformations. While our spring network model and its implementation is somewhat slower than the FEM we have compared it to, it offers great speed-up of simulations with large domains when fluid is computed on GPU and it remains the only free open source tool available for simulations of red blood cells in flow.

6 Acknowledgements

I would like to thank Prof. Krüger, computational physicist and Chancellor's Fellow at University of Edinburgh, who kindly provided information and data about the performance of his model. This work was supported by the Slovak Research and Development Agency under the contract No. APVV-0441-11.

References

[1] D. A. Fedosov, B. Caswell, and G. E. Karniadakis, "A multiscale red blood cell model with accurate mechanics, rheology, and dynamics," *Biophysical Journal*, vol. 98, no. 10, pp. 2215–2225, 2010.

[2] I. V. Pivkin and G. E. Karniadakis, "Accurate coarse-grained modeling of red blood cells," *Physical Review Letters*, vol. 101, no. 11, p. 118105, 2008.

[3] M. Nakamura, S. Bessho, and S. Wada, "Spring network based model of a red blood cell for simulating mesoscopic blood flow," *Int J numer method biomed eng*, vol. 29, no. 1, pp. 114–128, 2013.

[4] M. Dao, J. Li, and S. Suresh, "Mechanics of the human red blood cell deformed by optical tweezers," *Journal of the Mechanics and Physics of Solids*, vol. 51, pp. 2259–2280, 2003.

[5] I. Cimrák, M. Gusenbauer, and I. Jančigová, "An ESPResSo implementation of elastic objects immersed in a fluid," *Computer Physics Communications*, vol. 185, no. 3, pp. 900 – 907, 2014.

[6] A. Arnold, O. Lenz, S. Kesselheim, R. Weeber, F. Fahrenberger, D. Roehm, P. Košovan, and C. Holm, "ESPResSo 3.1 - molecular dynamics software for coarse–grained models," in *Meshfree Methods for Partial Differential Equations VI, Lecture Notes in Computational Science and Engineering*, M. Griebel and M. Schweitzer, Eds., vol. 89, 2013, pp. 1–23.

[7] T. Krüger, D. Holmes, and P. V. Coveney, "Deformability-based red blood cell separation in deterministic lateral displacement devices - A simulation study," *Biomicrofluidics*, vol. 8, 2014, 054114.

[8] G. Tomaiuolo and S. Guido, "Start-up shape dynamics of red blood cells in microcapillary flow," *Microvascular Research*, vol. 82, no. 1, pp. 35–41, 2011.

[9] Y. Park, C. A. Best, T. Kuriabova, M. L. Henle, M. S. Feld, A. J. Levine, and G. Popescu, "Measurement of the nonlinear elasticity of red blood cell membranes," *Phys Rev E Stat Nonlin Soft Matter Phys.*, vol. 83, no. 5, 2011.