UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Hemocell: a high-performance microscopic cellular library

Závodszky, G.; van Rooij, B.; Azizi, V.; Alowayyed, S.; Hoekstra, A.

DOI
10.1016/j.procs.2017.05.084
Publication date
2017
Document Version
Final published version
Published in
Procedia Computer Science
License
CC BY-NC-ND

Link to publication

Citation for published version (APA):
Závodszky, G., van Rooij, B., Azizi, V., Alowayyed, S., & Hoekstra, A. (2017). Hemocell: a high-performance microscopic cellular library. *Procedia Computer Science*, *108*, 159-165. https://doi.org/10.1016/j.procs.2017.05.084

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:21 Apr 2025

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 108C (2017) 159–165

Procedia Computer Science

International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland

Hemocell: a high-performance microscopic cellular library

Gábor Závodszky[1,2], Britt van Rooij[1], Victor Azizi[1], Saad Alowayyed[1,3], and Alfons Hoekstra[1,4]

[1] Computational Science Lab, Institute for Informatics, Faculty of Science, University of Amsterdam, Amsterdam, the Netherlands
g.zavodszky@uva.nl
[2] Department of Hydrodynamic Systems, Budapest University of Technology, Hungary
[3] King Abdulaziz City for Science and Technology (KACST), Riyadh, Saudi Arabia
[4] High Performance Computing Department, ITMO University, Saint Petersburg, Russia

Abstract

We present a high-performance computational framework (Hemocell) with validated cell-material models, which provides the necessary tool to target challenging biophysical questions in relation to blood flows, e.g. the influence of transport characteristics on platelet bonding and aggregation. The dynamics of blood plasma are resolved by using the lattice Boltzmann method (LBM), while the cellular membranes are implemented using a discrete element method (DEM) coupled to the fluid as immersed boundary method (IBM) surfaces. In the current work a selected set of viable technical solutions are introduced and discussed, whose application translates to significant performance benefits. These solutions extend the applicability of our framework to up to two orders of magnitude larger, physiologically relevant settings.

© 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the International Conference on Computational Science

Keywords: High-performance computing, Cellular flow, Initial conditions, Particle dense packing, Adaptive time-steps

1 Introduction

On the cellular level, blood is a complex suspension constituted of a continuous fluid phase (the plasma) and several types of suspended cells. The accurate modelling of the emerging transport phenomena of such a system is of utmost importance to progress our understanding of several in-vivo processes, e.g. thrombus formation, appearance of non-Newtonian viscosity, margination of platelets, the Fåhræus effect, appearance of a cell-free layer, or the scaling properties of shear-induced diffusion of red blood cells (RBCs) [1]. Such complex systems dealing with large amount of cells ($> 10^4 - 10^6$ cells) provide several computational challenges, such as the set up of the initial conditions for the cells or the storage of the resulting data, or simply the raw processing power required by the simulations. In Hemocell, the plasma is represented as a continuous fluid simulated with the use of Palabos [2], an open-source LBM solver. The cells are represented as surfaces modelled by DEM membranes coupled to the plasma flow through

1877-0509 © 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the International Conference on Computational Science
10.1016/j.procs.2017.05.084

a tested in-house immersed-boundary implementation [3, 4], where we demonstrated that the simulation can be scaled up to 10^6 cells executing on 8192 cores without significant loss of parallel efficiency. The implementation itself is designed to be very flexible and applicable in various scenarios. In the following we discuss how we have further improved the computational performance of Hemocell by separating the time-scale of the material model integration from that of the fluid dynamics and by pre-computing a randomised dense packing of red blood cells to provide improved initial conditions.

2 Methods

Initial conditions for cellular flows are usually not trivial. An uniform packing of cells can be easily calculated using their bounding-boxes as a basis for packing space requirement. RBCs, however, have a unique biconcave shape that fills the bounding box with a low volume ratio. The usually applied term that describes the surface [5, 6] is as follows:

$$y = R\sqrt{1 - \frac{x^2+z^2}{R^2}}\left[c_0 + c_1\frac{x^2+z^2}{R^2} + c_2\frac{(x^2+z^2)^2}{R^4}\right],$$

where $(R, c_0, c_1, c_2) = (3.91\mu m, 0.1358, 1.001, -0.5614)$. If we aim to fill a rectangular domain with rectangular bounding boxes, in fortuitous cases the volume fill ratio (Φ_{BB}) might go up to a 100%. On the level of RBCs within the bounding boxes, this yields $\Phi_{RBC} \approx 32\%$ which might be far from the desired value since physiologic blood has a hematocrit (Φ_{RBC}) of approximately 45%. If non-rectangular domains are considered, e.g. in the case of flows in smaller vessels, this ratio might fall even lower. In our approach, we present a possible solution using a kinematic simulation of encompassing ellipsoids to effectively generate dense cell distributions with random positions and alignments. The diameters of the ellipsoid for the RBCs are $(D_x, D_y, D_x) = (2.5\mu m, 1.2\mu m, 2.5\mu m)$. The overlap of the RBC shape and the encompassing ellipsoid is shown in Fig. 1.

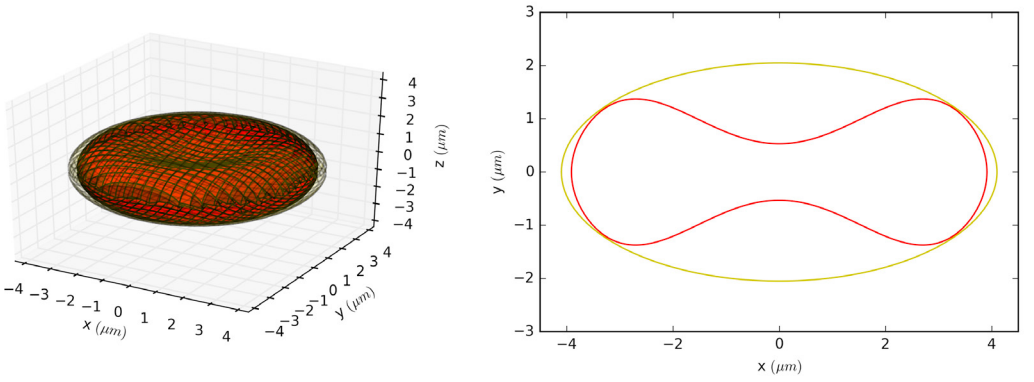

Figure 1: Left: 3D view of enclosing ellipsoid (yellow) overlayed on an RBC (red). Right: cut-plane visualising the volume fill ratios using the same colours for the contours.

Naturally, the enclosing ellipsoid has a larger volume than the RBC itself, therefore the above mentioned optimal rectangular bounding-box packing in case of $\Phi_{BB} = 100\%$ translates

to an ellipsoid volume ratio of $\Phi_{Ell} = 53\%$ (with the corresponding RBC volume ratio of $\Phi_{RBC} = 32\%$). We can improve the situation by calculating the packing of ellipsoids instead of bounding boxes. The optimal dense packing of ellipsoidal shapes is not known, nevertheless, several recent studies[7, 8] imply that the highest achievable ratio for ellipsoids of the diameters used here should be around $\Phi_{Ell} = 80\%$, which in turn would represent $\Phi_{RBC} = 49\%$. This value is high enough for physiological blood-like suspensions.

For this computation, a simple kinetic model of hard ellipsoid dense packing, called the force-bias model [9, 10, 11], was implemented. It distributes the locations randomly and then defines two radii scaling for every cell type (e.g. RBC, platelet): d^{in} represents the possible largest radius in the system without any overlap and d^{out} is initially set so that the merged volume of all the ellipsoids scaled with it yields the required volume ratio. Then we apply a repulsive force between overlapping ellipsoids proportional to the volume of the overlapping regions:

$$F_{ij} = \boldsymbol{\delta_{ij}} p_{ij} \frac{\vec{r_j} - \vec{r_i}}{|\vec{r_j} - \vec{r_i}|},$$

where $\boldsymbol{\delta_{ij}}$ equals 1 if there is an overlap between particle i and j and 0 otherwise, while p_{ij} is a potential function. The positions are updated following the Newtonian mechanics where mass is proportional to the particle scaling radius. This ensures that larger particles will move slower than smaller ones. The potential function was selected to be proportional to the overlapping volume of the d^{out} scaled particles. As a final step the size of d^{out} is reduced every iteration according to a chosen τ contraction rate. The computation stops when $d^{out} \leq d^{in}$. Using this method, we were able to push up to $\Phi_{Ell} = 76\%$, which yields the required $\Phi_{RBC} = 46\%$ for the RBCs.

Additionally, we can integrate the particle motions by only allowing translation of their centre of mass, thus predefining the alignment of the particles. This might be beneficial for higher velocity flows where the cells are expected to be lined up with the streamlines. Figure 2 presents two sample initial conditions generated with this method.

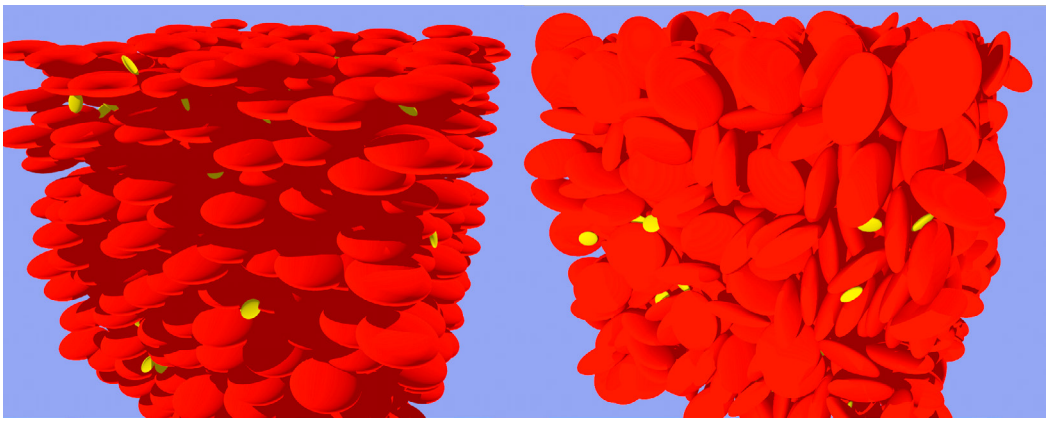

Figure 2: Two cubic domains presenting different methods of initialising cell positions and rotations with the hematocrit of 30% and 35%, respectively. The red coloured ellipsoids are the encompassing ellipsoids of the RBCs and the yellow coloured ones are for the platelets. Left: randomly distributed positions with fixed alignments. Right: both positions and alignments are randomly distributed. Please note: high volume-fill ratios ($> 40\%$) are difficult to achieve if alignments are fixed.

Another issue for our framework is the computational cost which is steeply increased by the application of the lattice Boltzmann method. LBM is a highly parallel and computationally efficient method known to work accurately in vascular geometries [2, 12, 13]. However, it is an explicit method which translates to very small time-steps for microfluid flows, usually in the order of $10^{-8}s$. The usual physiologically relevant time-range scales from $0.1s$ to several seconds which translates to several millions of required iterations. Naturally, this does not pose any serious computational challenge for resolving the Newtonian and non-turbulent plasma flow. The material model of the cellular membranes on the other hand is rather demanding in this regard and it constitutes over 95% of the total computational time [14]. In our implementation it can rely on either Euler or Adams-Bashforth integration schemes. Even though these are explicit schemes as well, this model performs accurately with up to two orders of magnitude larger time-steps than the LBM used for plasma flow computation. For this reason, the integration time-steps of the background CFD simulation are decoupled from the membrane material model integration using a simple adaptivity rule: the largest appearing force of the material system is probed at every time-step. When it is below a certain threshold, the integration step size is increased up to a limit, while if it surpasses a maximal value, we decrease its size gradually. This threshold is fine-tuned for each geometry to ensure a balance between performance and numerical stability. The separation of time-steps introduces an additional small numerical error to the system. Its effect was investigated by comparing the cellular positions and velocities after 10^5 iterations in the case of a straight vessel segment with the separation ratios of $1:1$ and $1:100$. The difference is negligible (below 1%).

3 Results

In the current section we demonstrate the applicability of the ideas described above by a use-case of a small vessel flow with high hematocrit. It is a periodic flow driven by external body force. The average flow velocity is $5mm/s$. The spatial resolution is $0.5\mu m$ and the LBM time-step size is $\delta_{LBM} = 5 \times 10^{-8}s$. For cellular flows starting from uniform initial particle distribution, we employ the common convention of requiring a particle with average flow velocity to circulate the geometry ten times before the actual simulation takes place. This is to ensure that the system does not contain any artificial transients. Since the geometry is $l = 256\mu m$ long, this translates to approximately 10 million iterations for warm-up. For domains with randomised initial positions we found that approximately one full cycle results in a similar cell distribution. Hence, our random high-hematocrit initial condition results in a very substantial factor 10 reduction in warm-up time. For measuring transport properties we consider a $0.1s$ time range after the warm-up. One state of this flow system initialised by our method is visualised in Fig. 3.

Depending on the actual case at hand (mostly on the geometry and on the hematocrit), the material model might take up more than 95% of the computational time. The reason for it is the high computational cost of the constitutive model per membrane node relative to the computational cost of a fluid node. Therefore, decoupling the two integration time-steps can yield a significant increase in computational efficiency. Furthermore, a dense system can be sensitive to spontaneous cluster formations which might lead to increased temporary deformation of the cells. Thus, managing the time-step size adaptively becomes similarly important.

Figure 4 shows the appearing largest forces in lattice units during a typical computation of a cellular flow inside the small vessel. The small peaks in the graph usually denote cell-cell collisions and smaller cluster-formation events, while the large peak around 800k iterations corresponds to a momentary large structure formation, where a dense cluster appears close to

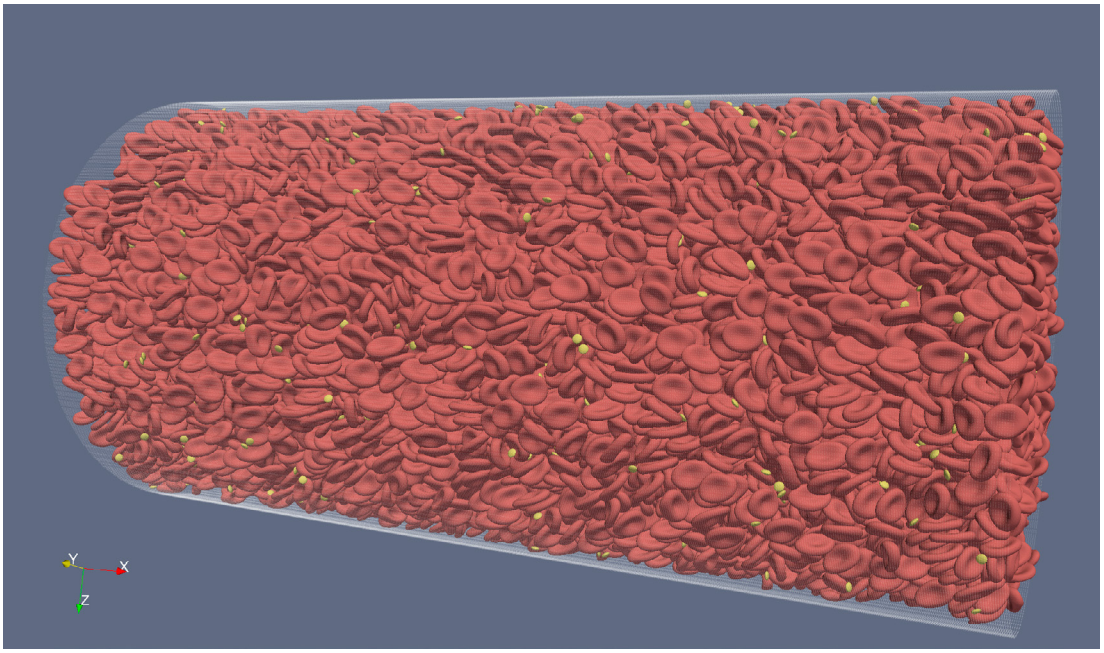

Figure 3: Simulation of cellular flow inside a small vessel with the diameter of $128\mu m$ at a hematocrit level of 46%.

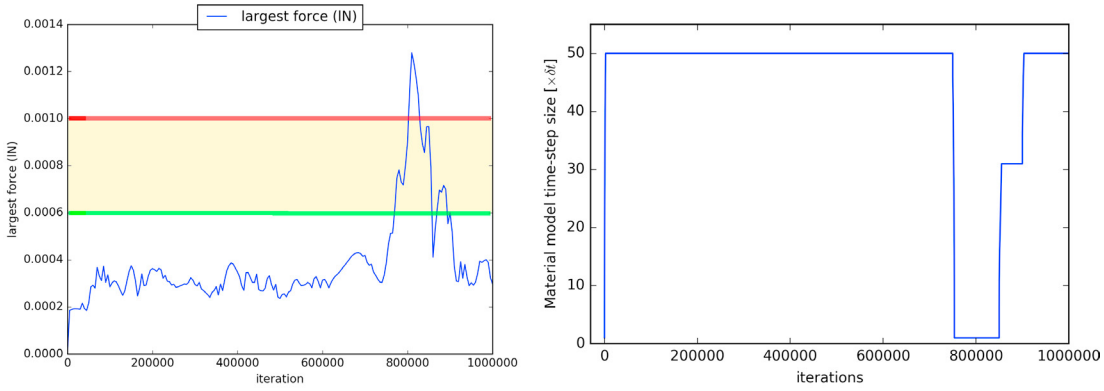

Figure 4: Left: The largest appearing force in the simulated system at every iteration. The zone of adaptivity is marked by a yellow coloured region. Below the green line the time-step size for integration is increased gradually (up to a pre-set maximum value), while above the red line the step-size is decreased. Right: the behaviour of the time-step adaptivity during the simulation. The largest allowed integration step-size for the material model is $50 \times \delta t$, where δt is the step-size of the plasma flow computation.

the edge of the flow and presses some cells against the wall. This event is associated with higher levels of deformation, thus with larger corresponding forces. In this case, a smaller iteration step-size is necessary for the integration of the constitutive equations to retain numerical stability. As the force peak progresses above the limit of $0.001lN$, the integrations steps are set to a smaller value. The typical bounds for the material model integration steps in our simulations is $\delta t_{material} \in [1; 100] \times \delta t_{LBM}$.

The overview of the computational time requirements for this high-hematocrit vessel flow case is shown in Table 1. These computations were carried out in the strong scaling limit, where further subdivision of the domain is not possible. The timings are based on performance results obtained using 12-core Intel Xeon E5-2690 v3 CPUs. The total computational times are denoted as *expected total time* since the first two configurations were not fully executed. These total time values are computed from the average iteration time of the first 5×10^5 iterations and the total number of required iterations.

	time / iteration (average)	**warmup iterations**	**simulation iterations**	**expected total time**
original	2.3 s	10M	2M	319.5 days
+ init	2.3 s	1M	2M	80 days
+ init + adaptivity	0.06 s	1M	2M	2.1 days

Table 1: Performance results from the case-study of a small vessel flow (see Fig. 3) executed on 1024 cores.

Overall the joint application of these two techniques for the current vessel flow case results in a substantial decrease of the total computation time. The randomised initialisation saves the majority of the warm-up cycles, while the adaptivity allow for stable time-step ratios up to $1 : 100$ bringing the computational time down from more than 300 days to around 2 days in the investigated case.

4 Conclusions

Hemocell, a cellular biofluid flow simulation library, is capable of computing the flow of blood-like suspensions accurately. The presented methods extend the range of its applicability to vessel diameters of above $100\mu m$, as well as to simulated time-scales in the range of a few seconds. These scales reach into the domain of macroscopic flows. Therefore, with the application of this framework it is possible to close the gap between the cellular and the macroscopic scales for several flow types of biological relevance, such as flows around micro-medical devices, investigations of margination, cell trafficking, and thrombus formation.

5 Acknowledgement

This work was sponsored by NWO Exacte Wetenschappen (Physical Sciences) for the use of supercomputer facilities, with financial support from the Nederlandse Organisatie voor Wetenschappelijk Onderzoek (Netherlands Organisation for Scientific Research, NWO). We also acknowledge partial funding from the European Union Horizon 2020 research and innovation programme under grant agreement no. 671564, the ComPat project.

References

[1] L Mountrakis, E Lorenz, and AG Hoekstra. Scaling of shear-induced diffusion and clustering in a blood-like suspension. *EPL (Europhysics Letters)*, 114(1):14002, 2016.

[2] Bastien Chopard, Daniel Lagrava, Orestis Malaspinas, Rafik Ouared, Jonas Latt, Karl-Olof Lovblad, and Vitor Pereira-Mendes. A lattice boltzmann modeling of bloodflow in cerebral aneurysms. In *V: European Conference on Computational Fluid Dynamics, ECCOMAS CFD*, volume 453, page 12, 2010.

[3] Lampros Mountrakis, Eric Lorenz, Orestis Malaspinas, Saad Alowayyed, Bastien Chopard, and Alfons G Hoekstra. Parallel performance of an ib-lbm suspension simulation framework. *Journal of Computational Science*, 9:45–50, 2015.

[4] Lampros Mountrakis, Eric Lorenz, and Alfons G Hoekstra. Validation of an efficient two-dimensional model for dense suspensions of red blood cells. *International Journal of Modern Physics C*, 25(12):1441005, 2014.

[5] Evan A Evans, Richard Skalak, and S Weinbaum. Mechanics and thermodynamics of biomembranes, 1980.

[6] Igor V. Pivkin and George Em Karniadakis. Accurate coarse-grained modeling of red blood cells. *Physical Review Letters*, 101(11):1–4, 2008.

[7] Fabian M. Schaller, Robert F. B. Weigel, and Sebastian C. Kapfer. Densest Local Structures of Uniaxial Ellipsoids. *Physical Review X*, 6(4):041032, nov 2016.

[8] Giorgio Cinacchi and Salvatore Torquato. Hard convex lens-shaped particles: Densest-known packings and phase behavior. *Journal of Chemical Physics*, 143(22), 2015.

[9] J Mościński, M Bargieł, ZA Rycerz, and PWM Jacobs. The force-biased algorithm for the irregular close packing of equal hard spheres. *Molecular Simulation*, 3(4):201–212, 1989.

[10] Monika Bargieł and Jacek Mościński. C-language program for the irregular close packing of hard spheres. *Computer Physics Communications*, 64(1):183–192, 1991.

[11] Alexander Bezrukov, Monika Bargiel, Dietrich Stoyan, et al. Statistical analysis of simulated random packings of spheres. *Particle & Particle Systems Characterization*, 19(2):111, 2002.

[12] Gábor Závodszky and György Paál. Validation of a lattice Boltzmann method implementation for a 3D transient fluid flow in an intracranial aneurysm geometry. *International Journal of Heat and Fluid Flow*, 44:276–283, dec 2013.

[13] Miki Hirabayashi, Makoto Ohta, Daniel A Rüfenacht, and Bastien Chopard. Characterization of flow reduction properties in an aneurysm due to a stent. *Physical Review E*, 68(2):021918, 2003.

[14] Lampros Mountrakis. *Transport of blood cells studied with fully resolved models.* PhD thesis, University of Amsterdam, September 2015.