

HemeWeb: Blood flow simulation in the cloud using docker

Steven Steven



Master of Science
School of Informatics
University of Edinburgh
2016

Acknowledgements

Many thanks to X Y Z

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(Steven Steven)

Table of Contents

1	Introduction	1
1.1	Motivation	1
1.2	Objectives	2
1.3	Outline	2
2	Background	4
2.1	Current HemeLB workflow	4
2.2	Cloud computing	5
2.3	Containerization Technology	5
2.4	Other High Performance Computing	5
	Bibliography	6

List of Figures

2.1	Current HemeLB workflow taken from Steven [2016]	5
-----	--	---

Chapter 1

Introduction

**** Software are increasingly complex. Our everyday software are crammed with features that makes its usage difficult. To people without familiarity with the product, it will be a barrier of entry to use it even when it is really good for them.

This also ties in to the complexity of the research. Many of this softwares are developed as part of researches. Open science dictates that research should be reproducible or replicable for it to better validate the research. However, recent findings have shown that not many research in psychology or even computation are replicable easily. ****

1.1 Motivation

To study how blood flow in a given vessel, Mazzeo and Coveney [2008] developed a fluid dynamic simulation software named HemeLB. Currently, it is actively developed and used by researchers to help their study. For example, Itani et al. [2015] used HemeLB for automated ensemble simulation of blood flow for a range of exercise intensities, Bernabeu et al. [2015] used it for detecting difference of retinal hemodynamics with regards to diabetic retinopathy, and recently Franco et al. [2015, 2016] used it to understand branching pattern of blood vessel networks.

As I have written in the proposal for this dissertation [Steven, 2016], HemeLB works by calculating fluid flow in parallel by using lattice-Boltzmann method [Mazzeo and Coveney, 2008]. This calculation allows HemeLB to simulate blood flow within a given blood vessel structure. Unfortunately, the calculation part is only a small part of the workflow to run the simulation. There are multiple pre-processing and post-processing steps needed to run the simulation from start to the end. These includes of preparing the input so HemeLB can work on it, and also processing the output so it is

ready to view.

These long pipelines of steps needed to run simulation, coupled with complexity of configuration of the software created a high barrier of entry for scientists and doctors to use HemeLB. Furthermore, as observed previously [Steven, 2016], interesting simulation will require parallel computing resources like ARCHER supercomputer which might be difficult to get access to by interested parties. While smaller simulation instances can run on typical laptop, most of the problems will require more powerful machines. These facts might prevent usage of the software by interested parties. More importantly, it shows there are still improvement that can be done to lower the barrier of entry for users to use HemeLB. This is important for HemeLB, especially when it is envisioned to be an integral part of future medical decision [Green, 2014].

Another aspect that HemeLB workflow can be improved is with regards to its reproducibility aspect. Researches that used HemeLB embrace reproducibility as one of its concern. As observed before [Steven, 2016], There are steps that are in place to make sure HemeLB and its simulation result are reproducible. First, the entire code base are publicly available on Github. Second, in running simulation with HemeLB, version of the software is automatically recorded. Lastly, in addition to the version used, input files and configurations are also recorded automatically. These facts can be seen from the publications mentioned above [Bernabeu et al., 2015; Itani et al., 2015; Franco et al., 2015, 2016] that include all these information. These information allow researchers interested to replicate the simulation to do it manually. Automation of these steps could further improve HemeLB's reproducibility and allow peers to replicate, duplicate, and audit its simulation results quickly and easily. This automation will be important, in addition to being usable, for HemeLB to become integral part of medical decision in the future.

1.2 Objectives

Based upon the needs to improve the usability, reproducibility, and auditability aspect of HemeLB project, I will develop a prototype web interface for HemeLB. This prototype web interface will lower barrier of entry in using HemeLB software compared to the current approach of using command line interface. In addition to that, using web interface will also allow features to be added to the simulation workflow which might not be essential to the HemeLB core itself. For example, automating packaging, sharing, and reproducing simulation result. These features are not essential for the

HemeLB core, but definitely help the overall workflow of blood flow simulation.

Using the dynamic capabilities of cloud computing vendor, the web backend should be able to dynamically scale without much efforts. On top of that, these infrastructures are available to everyone with a costs, allowing its user to access it without having to get access to supercomputers. Its user should be able to run a blood flow simulation without manually configuring the infrastructure.

1.3 Outline

I provide a brief introduction to the topic of this dissertation in this chapter. The rest of the chapters will be organized as follow:

- **Chapter 2.** I will provide background information that are necessary for readers to understand the concepts, technology and implementation that are done in this dissertation. HemeLB, containerization technology, cloud computing, High-Performance computing infrastructure, and other topics will be discussed in details in this chapter.
- **Chapter 3.** I will discuss the bulk of the work in this chapter. Implementation details and design of the proposed solutions will be provided and discussed in details.
- **Chapter 4.** Evaluation
- **Chapter 5.** Analysis
- **Chapter 6.** Future work

Chapter 2

Background

2.1 Current HemeLB workflow

Running a blood flow simulation using HemeLB currently consists of multiple steps. To understand how the proposed project can improve the current conditions, I will elaborate on how HemeLB workflow currently work.

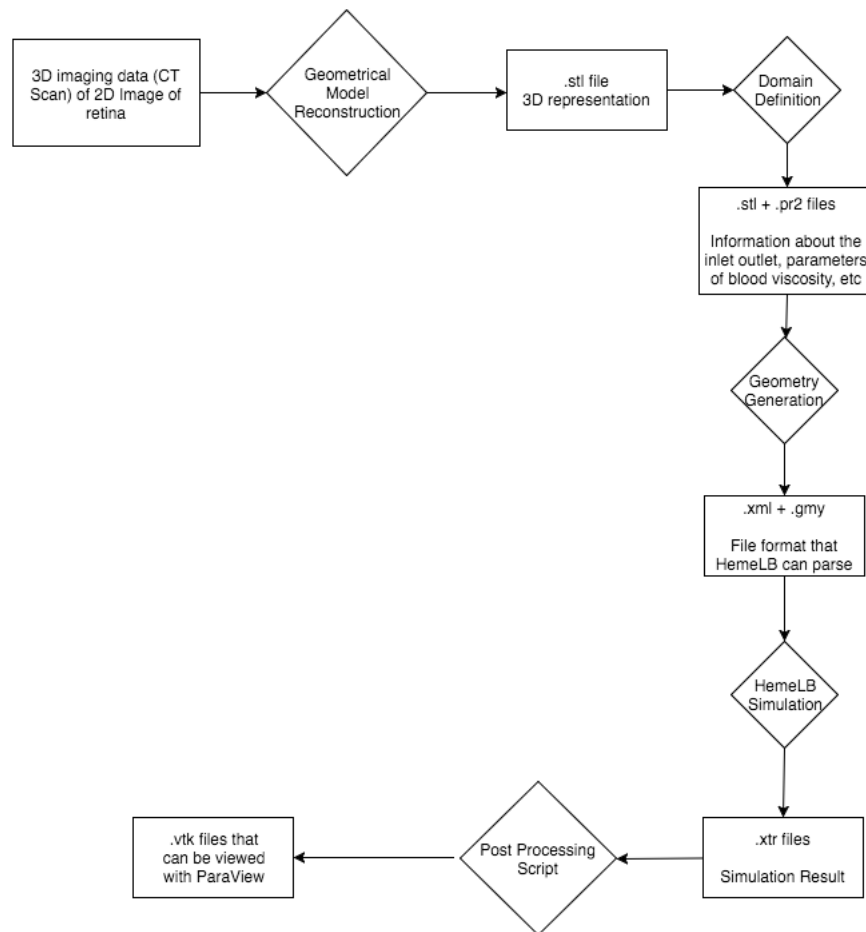


Figure 2.1: Current HemeLB workflow taken from Steven [2016]

2.2 Cloud computing

2.3 Containerization Technology

2.4 Other High Performance Computing

Bibliography

- Bernabeu, M. O., Lu, Y., Lammer, J., Aiello, L. P., Coveney, P. V., and Sun, J. K. (2015). Characterization of parafoveal hemodynamics associated with diabetic retinopathy with adaptive optics scanning laser ophthalmoscopy and computational fluid dynamics. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 8070–8073. IEEE.
- Franco, C. A., Jones, M. L., Bernabeu, M. O., Geudens, I., Mathivet, T., Rosa, A., Lopes, F. M., Lima, A. P., Ragab, A., Collins, R. T., et al. (2015). Dynamic endothelial cell rearrangements drive developmental vessel regression. *PLoS Biol*, 13(4):e1002125.
- Franco, C. A., Jones, M. L., Bernabeu, M. O., Vion, A.-C., Barbacena, P., Fan, J., Mathivet, T., Fonseca, C. G., Ragab, A., Yamaguchi, T. P., et al. (2016). Non-canonical wnt signalling modulates the endothelial shear stress flow sensor in vascular remodelling. *Elife*, 5:e07727.
- Green, C. (2014). Computer simulation could become 'integral' in the diagnosis of.
- Itani, M. A., Schiller, U. D., Schmieschek, S., Hetherington, J., Bernabeu, M. O., Chandrashekar, H., Robertson, F., Coveney, P. V., and Groen, D. (2015). An automated multiscale ensemble simulation approach for vascular blood flow. *Journal of Computational Science*, 9:150–155.
- Mazzeo, M. D. and Coveney, P. V. (2008). Hemelb: A high performance parallel lattice-boltzmann code for large scale fluid flow in complex geometries. *Computer Physics Communications*, 178(12):894–914.
- Steven, S. (2016). Hemeweb: Container based high performance computing scenario in cloud infrastructure for hemelb. Informatics research proposal, University of Edinburgh.