State of the Practice for Lattice Boltzmann Solvers

Spencer Smith, Zahra Motamed, Peter Michalski

Faculty of Engineering, McMaster University

June 25, 2021

McMaster
University

Overview

- Goals
  - ▶ Understand the state of the software development practices for LBM Software
  - ▶ Make recommendations for improvements
  - ▶ A publication that is useful to the community
- We have developed a standard methdology for assessing SOP for Domain X
- The methodology requires a **domain expert** to:
  - ▶ Vet the preliminary results
  - ▶ Assess the feasibility of the recommendations
  - ▶ Navigate the publication process
  - ▶ Answer **developer** interview questions on pain points
- Today's meeting
  - ▶ Informal
  - ▶ Questions do not have to be answered in real time, or by domain expert

Overall Process

1. *Domain Expert*: Create a top ten list
2. Brief *Domain Expert*
3. Initial list of candidate software packages
4. *Domain Expert*: Vet domain software list
5. Domain Analysis
6. *Domain Expert*: Vet domain analysis
7. Collect empirical measures
8. Measure using measurement template
9. Interview developers
10. Use AHP process to rank the software packages
11. *Domain Expert*: Vet AHP ranking
12. *Domain Expert*: Review recommendations

Vet Software List

- How does our list compare to the domain expert's list?
- Is any software missing?
- Is there software that should be included?
- Any other questions/comments or concerns?

SOP Software List

- Ludwig
- ESPResSo
- Palabos
- OpenLB
- LUMA
- pyLBM
- DL_MESO (LBE)
- waLBerla
- Sailfish
- laboetie
- TCLB
- MechSys

- lettuce
- ESPResSo++
- MP-LABS
- SunlightLB
- LB3D - paper
- LIMBES
- LB2D-Prime
- HemeLB
- lbmpy
- LB3D-Prime
- LatBo.jl

Common With Domain Expert List

- Palabos
- OpenLB
- LUMA
- waLBerla
- Sailfish
- HemeLB
- LB3D - paper

- MP-LABS
- SunlightLB
- pyLBM
- DL_MESO (LBE)
- LIMBES
- TCLB (also called CudneLB)

Only on Domain Expert List

- ASL (general purpose tool for PDEs, LBM a small part)
- ch4-project (no docs, no installation guide)
- Open FSI (uses Palabos for LBM, new project)
- LIFE (small and new project)
- LBSIM (dead project)
- Taxila LBM (dead project, could not install, broken links in documentation)

Only on SOP List

- Ludwig
- ESPResSo
- laboetie
- MechSys
- lettuce

- ESPResSo++
- LB2D-Prime
- lbmpy
- LB3D-Prime
- LatBo.jl

Why is pylbm on the domain expert list twice, should one be lbmpy?

Domain Analysis

- So users can pick the right software for their task
- Think of the software as a family of related software products, like:
  - ▶ Family of automobiles
  - ▶ Computer hardware
  - ▶ etc.
- Commonalities
- Variabilities
- Parameters of variation
- Do the following slides capture the commonalities and variabilities?
- We can follow up over e-mail, if that is easier

Commonalities

- Lattice - discretized domain within a boundary
- Collision operator (The Bhatnagar-Gross-Krook Collision Operator is common)
- Probability density function
- Equilibrium distribution function
- Boltzmann transport equation

SOP Variabilities

- May use parallel processing (May use varied tools for this - MPI, CUDA..)
- Different equilibrium distribution functions (compressible and incompressible)
- Coefficients for equilibrium distribution function (based on velocity directions - 2, 3, 5, 9, 13, 15, 19, 27)
- Lattice dimensions (1D, 2D, 3D)
- Varied velocity directions, partially based on dimensions (2, 3, 5, 9, 13, 15, 19, 27)
- Collision operators (SRT, TRT, MRT, BGK)
- Collision vs collision-free transport equations
- Number of fluids that can be modeled simultaneously
- Fluid parameters (Reynolds num., density, viscosity, etc.)
- Lattice boundary (reflective or non-reflective)

Domain Expert Variabilities

- May use parallel processing (MPI, CUDA, OpenMP, ...)
- Lattice dimensions (1D, 2D, 3D)
- Turbulent (Yes, No)
- FSI (Fluid Structure Interaction)
- Complex Geometry
- Non-Newtonian Fluids
- Programming Language (C++, Python, etc)

Summary of Measures

- Empirical measures (number of stars, forks, pull requests, lines of code, etc)
- Measurement template
  - ▶ Installability
  - ▶ Correctness and verifiability
  - ▶ Surface reliability
  - ▶ Surface robustness
  - ▶ Surface usability
  - ▶ Maintainability
  - ▶ Surface Understandability
  - ▶ Visibility and transparency
- Developer interviews (looking for pain points)
- AHP Process with pairwise comparisons

Thoughts on Overall Ranking?
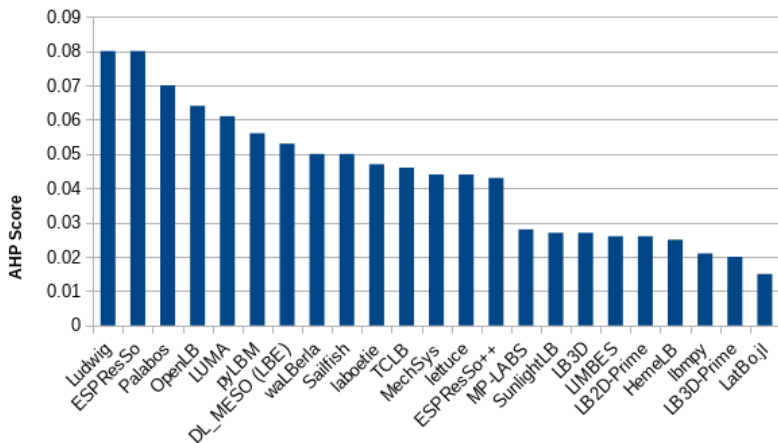
Ranking Follow-Up

- We would like feedback on the ranking for each of the qualities
- Is it feasible to review all 10 graphs, and the associated write-up?
- Maybe there is a grad student that can review the rankings?

Pain Points from Developer Interviews

Do these fit with your experience?

- Lack of time to implement new features
- Team members lack software development experience
- Lack of funding for software development
- No organizations to help with developing high quality software
- Documentation could be improved, only has shallow overview of the theory
- Setting up parallelization is challenging
- Setting up continuous integration is challenging
- Difficulties in ensuring correctness because of size of test cases
- Users sometimes use tools inappropriately
- Technical debt due to coupling between components

Recommendations

Do these seem feasible? What other ideas do we have?

- Consult with software development organization
  - ▶ Better Scientific Software (BSSw)
  - ▶ Software Sustainability Institute
  - ▶ Software Carpentry
- Citations for software (Katz project)
- Redefine productivity to include time working on tasks like testing, continuous integration and documentation

Publication

- Who do you see as the targeted readers?
- Where should we publish this paper?