SoftwareX 18 (2022) 101097

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

MPLBM-UT: Multiphase LBM library for permeable media analysis

Javier E. Santos [a], Alex Gigliotti [b], Abhishek Bihani [b], Christopher Landry [b], Marc A. Hesse [c,d], Michael J. Pyrcz [b], Maša Prodanović [b,*]

[a] Center for Nonlinear Studies and Computational Earth Science Group, Earth and Environmental Sciences Division, Los Alamos National Laboratory, United States of America
[b] Hildebrand Department of Petroleum and Geosystems Engineering, The University of Texas at Austin, United States of America
[c] Jackson School of Geosciences, The University of Texas at Austin, United States of America
[d] Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, United States of America

ARTICLE INFO

Article history:
Received 1 October 2021
Received in revised form 4 April 2022
Accepted 26 April 2022

Keywords:
Multiphase
Singlephase
Relative permeability
Permeability
Lattice-Boltzmann
Porous media
Digital rock physics

ABSTRACT

MPLBM-UT is a specialized lattice-Boltzmann library that makes running single- and two-phase flow simulations in porous media accessible to everyone. We provide a suite of tools to pre-process computational domains for simulation, to set up custom boundary conditions, to run simulations, to post-process simulation outputs, and to visualize simulation results and data. All of these tools are easily accessible to users through the mplbm_utils Python package included in and automatically installed with MPLBM-UT. The high-performance, highly parallel library Palabos is used as the solver backend. MPLBM-UT is easily deployed in a variety of systems, from laptops to supercomputer clusters. MPLBM-UT also features multiple examples and benchmark templates that allow for fast prototyping of different porous media problems. We also provide an interface for reading in different file types and downloading domains from the Digital Rocks Portal to perform simulations.

Code metadata

Current code version	v1.0
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-21-00184
Legal Code License	GNU GENERAL PUBLIC LICENSE Version 3
Code versioning system used	git
Software code languages, tools, and services used	C++, Bash, Python
Compilation requirements, operating environments & dependencies	Unix/Linux system, Python 3.6+, MPI, GCC (or other C++ compiler)
Support email for questions	alex.gigliotti@utexas.edu

1. Motivation and significance

Problems involving fluids flowing through porous media are ubiquitous in nature and industrial applications. There are subsurface applications such as hydrocarbon recovery [1], CO_2 sequestration [2], and groundwater aquifer production [3]. Further, there are naturally occurring examples that include percolation of brine through rock salt [4], snow and ice melt percolation through glaciers [5–7], methane migration in marine sediments [8], and melt migration during planetary core formation [9–12]. For industrial applications, batteries [13], fibers [14], filters, and even coffee-making [15] all present challenges involving fluid flow through porous materials. In order to forecast, design, describe, understand, or upscale problems involving fluid flow through porous media, it is important to describe and quantify how different fluids flow through and interact within these complex materials.

Different experimental and numerical methods can be used to estimate the transport properties of a given porous material sample.

First, laboratory measurements can provide the bulk flow properties, such as permeability and relative permeability, of a typically centimeter-scale sample through special core analysis measurements [16]. Micromodel experiments provide dynamic visualization of microscopic phenomena in quasi two-dimensional (2D) transparent porous media on the pore scale (on the order of micrometers) [17]. Three-dimensional (3D) observation of pore scale phenomena is possible, for instance, using X-ray microtomography, though with somewhat limited ability to

* Corresponding author.
E-mail address: masha@utexas.edu (M. Prodanović).

https://doi.org/10.1016/j.softx.2022.101097

observe dynamic phenomena or reactive flow [18]. Another avenue for describing the flow properties of a sample is by employing published functional relationships that estimate these based on fitting parameters that account for geometrical descriptors (i.e., porosity, tortuosity, pore size distribution). While these methods are fast to implement, they do not provide accurate results in the presence of complex domains.

Next, there are several numerical simulation methods that can reproduce the fluid flow physics at the pore-scale. Among these, direct simulation methods are attractive since they solve the desired partial differential equations in realistic domains, which give the user a snapshot of how the fluids interact with themselves and the medium. Among the direct simulation methods, there are several options often used to obtain flow properties from 3D images: the finite volume method [19,20], smoothed particle hydrodynamics [21,22], the finite element method [23], and the lattice Boltzmann Method (LBM) [19,20,24,25]. These simulation methods provide an accurate picture of how fluid flows through complex geometries with a resolution on the scale of micrometers, and even smaller; and with the advances in computational performance, larger domains can be simulated practically. Out of the listed options, LBM is often chosen due to its well-tested capabilities of simulating flow through complex geometries, such as porous materials [26]. LBMs can be constructed to operate based on minimal assumptions and are well-suited toward modeling fluids within complex materials without requiring simplified geometries.

For direct flow simulations in porous media, it is common to use a 3D image of a real sample as the input geometry. *Digital rock physics* refers to a collection of techniques that utilize the microscopic description of a porous material geometry for simulation of different physical phenomena of interest. Through 3D imaging techniques and understanding digital rock physics, it is possible to create simulation input geometries from real samples. The conventional workflow for creating a digital sample starts with using an x-ray scanner to produce a gray-scale volume of the real sample. Then, the gray-scale volume is segmented to produce a binary image. A review of these image processing methods is provided in [18]. This binary image, which indicates where pore and solid voxels are located, can then be used as an input for simulation. Further processing may be required for different types of simulations such as finite volume or finite element methods. It is also worth noting that the above workflow can be applied or modified to study many different types of porous materials, including those described in the aforementioned application areas. A wide variety of digital rock samples are also available at the Digital Rocks Portal [27].

For MPLBM-UT [28], we chose to focus on implementing a LBM simulation workflow due to its proven success with porous media, its flexibility, its ability to be parallelized, and its ability to easily use digital rock images as input [3,29]. LBM was first proposed [30] as an alternative to the lattice-gas automaton [31] to study hydrodynamic properties with the Boltzmann equation. In short, LBMs solve a modified version of the Boltzmann equation on a uniform grid, and the LBM formulation approximates the Navier-Stokes (NS) equation [32]. LBM particularly stands out in problems where solving the NS equation becomes troublesome, like during flow through complex geometries, turbulent flow, large Knudsen number flows, etc. Since LBM does not require solving complex systems and is discretized on a uniform grid, the implementation is relatively simple and is easily parallelized.

The mathematical description of how fluid flows is given by the NS equation, and approximating NS practically in complex geometries is a challenging task. Nevertheless, LBM is able to approximate a solution by means of its kinetic theory formulation. From kinetic theory, the Boltzmann equation can be viewed as a molecular-scale analog of the NS equation. Below is the Boltzmann equation:

$$\underbrace{\frac{\partial f}{\partial t}}_{\text{Change in PDF}} + \underbrace{\underline{v}\frac{\partial f}{\partial \underline{x}}}_{\text{Diffusion}} + \underbrace{\underline{F}\frac{\partial f}{\partial \underline{v}}}_{\text{External Forces}} = \underbrace{\left(\frac{\partial f}{\partial t}\right)_{coll}}_{\text{Collision Term}} \tag{1}$$

where f is the probability density function of the bulk molecules, $\underline{x}$ is the position vector, $\underline{v}$ is the molecule velocity, $\underline{F}$ is an external force vector, t is the time, and $\left(\frac{\partial f}{\partial t}\right)_{coll}$ is the molecular collision term. Conceptually, Eq. (1) models the probability of molecules having a certain position and velocity (or momentum) over time subject to molecular diffusion, external forces, and molecular collisions.

LBM relies on the probability density function (PDF) of the Boltzmann equation to represent fluids in the simulation domain; each node of the domain hosts fluid particles represented by a PDF that propagate and interact in their neighborhood based on the type of LBM implementation. From the PDF, macroscopic characteristics, such as density and velocity, can be calculated. For single-phase flow simulations, users can choose either the Bhatnagar-Gross-Krook (BGK) collision term [33] or the Multi-Relaxation-Time (MRT) collision term [34,35].

For two-phase flow simulations, MPLBM-UT uses the Shan–Chen LBM framework. The Shan-Chen model [36,37] starts with the Boltzmann equation (initially without external forces). Next, the Boltzmann equation is discretized with the BGK collision term, and each fluid phase is represented by two separate probability density functions:

$$f_{\sigma,i}(\underline{x}, t) - f_{\sigma,i}(\underline{x} + e_i\Delta t, t + \Delta t) = \frac{1}{\tau}\left(f_{\sigma,i}(x, t) - f^{o}_{\sigma,i}(x, t)\right) \tag{2}$$

where σ represents which fluid phase, i represents the lattice direction, x represents the location on the lattice, t represents the time, e represents the velocity, and τ represents the characteristic relaxation time. Furthermore, MPLBM-UT adds to the Shan-Chen model by including fluid–fluid and fluid–solid interactions with additional terms representing molecular forces. With these additional parameters, it is possible to set interfacial tension and contact angle values [38]. With the Shan-Chen model, it is possible to obtain the flow properties along with the capillary pressure and relative permeability curves.

We use the Palabos platform [39] as our LBM backend. Palabos is written in C++, is based on MPI for parallel executions, and uses C++ templates and forms of object-oriented polymorphism to support a broad range of LBM models while exhibiting high computational performance.

Currently, our library supports three main simulation setups:

- Single-phase flow: permeability and preferential path assessment
- Unsteady-state multiphase-phase flow: drainage and imbibition, capillary pressure curves, unsteady relative permeability
- Steady-state multiphase-phase flow: relative permeability, contact angle studies

Our multiphase models have been extensively validated [40] using the analytical solutions of the Young–Laplace Equation [41,42], the Washburn equation [43], and the Brookes–Corey relative permeability model [44]. These three capture the most important micro-scale physics in multi-phase porous media flows.

The main objectives of the MPLBM-UT library are the following:

- provide user-friendly methods for LBM setup and simulation
- provide simulation methods to help understand the role of surface wetting phenomena
- provide plotting, visualization, and animation scripts for qualitative and quantitative interpretation

- help set-up parameter sweeps to develop better constitutive relationships (i.e., relative permeability) to upscale pore-scale processes
- help create massive amounts of data to train machine learning models [45,46].

2. Software description

The MPLBM-UT library is divided into four main components: input parsing and domain construction, simulation, processing of the simulation results, and visualization. Each of the components will be thoroughly explained in the subsection below and are detailed in Fig. 1. The MPLBM-UT library and all its core components are installed with one line of code (bash script) independent of the user's hardware. We have tested MPLBM-UT on laptops (Unix, Mac, and Linux subsystem for Windows), workstations, and institutional supercomputer clusters. In order to provide quick and easy access to the core componenets, the MPLBM-UT library is united under the mplbm_utils Python package, which is included with the library and automatically installed.

2.1. Software functionalities

MPLBM-UT can perform steady- and unsteady-state flow simulations. The BGK and MRT models are used for single-phase flow, and the Shan–Chen model is used for two-phase flow. For the purpose of flow simulations, MPLBM-UT contains the following functionalities:

- Geometry reading, cleaning and pre-processing
- Single- and multi-phase simulation setups with a variety of boundary conditions used in porous media problems
- Output reading and post-processing

In addition, MPLBM-UT also contains tools that perform the following functions:

- Capillary pressure-saturation behavior calculations
- Absolute and relative permeability calculations and plotting
- Percolation path analysis
- Contact angle studies
- Visualization and animation of flow simulation outputs

The illustrative example in Section 3 shows a commonly used workflow featuring the above tools and functionalities.

2.2. Software architecture

Inputs. The first step of the flow simulation process is to import a binary image and optimize it for simulation. As detailed in Table Table 1, user-provided geometries in tiff, raw, png, hdf5, csv, and txt formats can be read as inputs. Additionally, geometries stored in the Digital Rock Portal can be downloaded through the interface [47]. After the domain is loaded, the disconnected regions in the image are removed, and percolation in the flow direction of interest is ensured. It is also possible to create initial two-phase fluid configurations using a PoreSpy [48] drainage simulation, which has been implemented as a tool in the mplbm_utils Python package. The drainage simulation from PoreSpy is particularly useful for initializing steady state simulations and generally to decrease simulation run times. Finally, the binary image is converted into an image with three labels: pore-space (or wetting fluid), non-wetting fluid, bounce-back boundaries, and inner solids. Since many porous materials range 10% to 35% porosity, pre-processing with these labels allows for a more computationally efficient simulation domain.

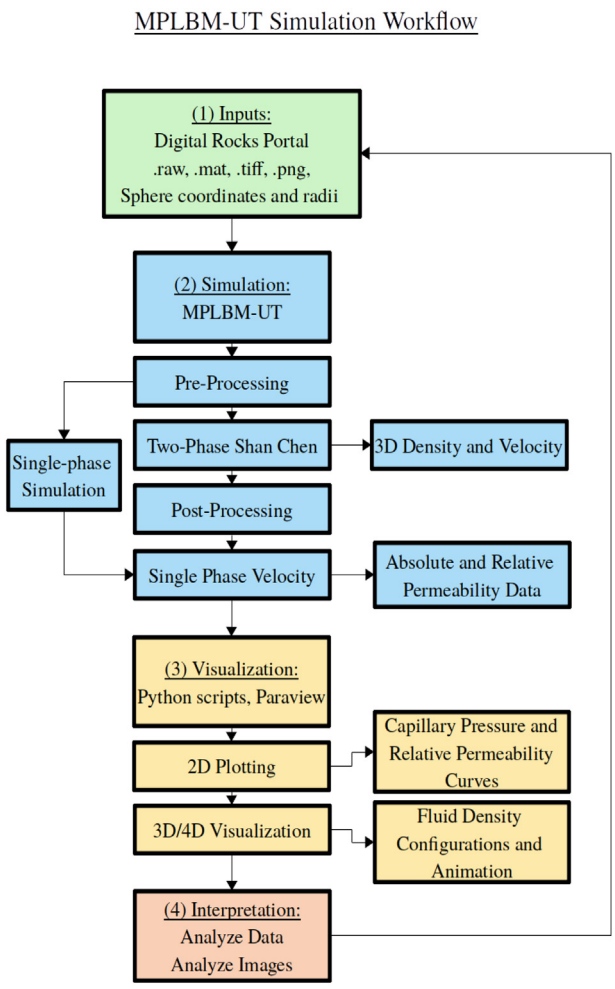

Fig. 1. MPLBM-UT simulation workflow.

Table 1
Allowed input file-formats.

Input Format	File Contents
.txt/.csv	Sphere coordinates
.tiff/.png stack	2D cross-sections
.raw/.h5	3D image
Digital Rock Portal url	Downloaded files

Simulation. Once the input geometries have been processed, the input files for either a single-phase and/or a two-phase simulation must be created. A variety of example input files are provided in the MPLBM-UT examples directory. For both single-phase and two-phase simulations, the domain size, output directories, fluid boundary conditions, convergence criteria, and output data all need to be specified. For single-phase simulations, the applied fluid pressure can also be chosen. Specifically for two-phase flow the following must be indicated in the input file: initial positions of each fluid phase, interfacial tension, wetting forces, fluid densities, and specified pressure or constant force boundary conditions. If pressure boundary conditions are used, the smallest inscribed sphere radius to invade and the number of pressure steps (points) to simulate are needed; this data is converted to pressure and the equivalent density difference for

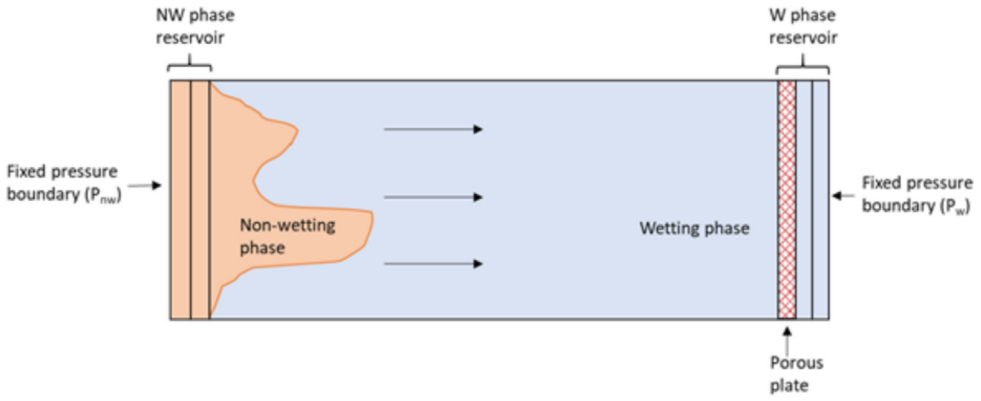

Fig. 2. Unsteady-state simulation schematic.

the code. If constant forces are used, the applied forces for each fluid are needed.

After the input files have been generated, the user can run a single-phase simulation or two-phase Shan–Chen simulation. When running single-phase simulations, this is the last step to obtain absolute permeability and velocity data. For two-phase simulations, an additional post-processing step is needed after the Shan–Chen simulation is complete in order to obtain relative permeability data. This step involves accomplishing two tasks: the first task is to take the resulting fluid density configuration files from the Shan–Chen simulation and separate each fluid phase into two different files, one for each fluid phase; and the second task is to convert each separated fluid configuration file into an efficient geometry as described in the Inputs section. Once the individual fluid phase configuration files have been processed, single-phase simulations can be completed on each file to obtain relative permeability. First, a single phase permeability simulation with the original geometry is used to obtain absolute permeability. Then, single phase permeability simulations are carried out using each separated fluid configuration file to obtain effective permeabilities. Relative permeability can then be calculated as the ratio of effective to absolute permeability. These two tasks and the calculations have been automated with scripts that are included in MPLBM-UT.

Visualization. 3D visualization is a challenging task on its own, and it is even more challenging in porous media. In order to aid in this challenge, visualization tools were developed for MPLBM-UT. By default, Palabos outputs VTK files that contain the 3D density and velocity data. These VTK files can be visualized using a variety of software packages (such as Paraview or Visit), but in order to maximize flexibility, Python scripts utilizing the PyVista [49] and Vedo [50] modules were created for isosurface visualization of the density configuration files. This same visualization script is also used to create animations from screenshots. In addition, a plotting script was also created to help with quick visualization of capillary pressure and relative permeability curves.

3. Illustrative examples

3.1. Unsteady-state two-phase flow through a sphere pack

The purpose of this example is to illustrate the workflow for a two-phase Shan–Chen simulation using MPLBM-UT. Specifically, an unsteady-state flow simulation is used to simulate a drainage test using a sphere pack.

First, the Finney Packing of Spheres from Digital Rocks Portal [51] was selected as the input geometry. This geometry was chosen since sphere packs are well studied and the results can be verified easily. A drainage test is setup by completely saturating the pore space with the wetting fluid, and then non-wetting fluid is injected to measure the capillary pressure and relative permeabilities (this setup is shown in Fig. 2). It was also assumed that the wetting phase completely wets the grains in this simulation. As the second step, the input files for MPLBM-UT were configured to reflect the aforementioned setup.

Once the simulation results were completed, the plotting utilities included in MPLBM-UT were used to plot the capillary pressure and relative permeability data. Additionally, an animation of the two-phase flow simulation was created using the visualization scripts. The animation is included in the supplementary materials. The results of the simulation show expected behavior [52,53]. As seen in Fig. 3, a mostly uniform front of non-wetting fluid displaces the wetting fluid until residual wetting phase saturation is reached. This is reflected in the capillary pressure and relative permeability curves shown in Fig. 3 as well.

Further examples and numerical validations are available in the MPLBM-UT examples directory.

4. Impact and conclusion

The MPLBM-UT library provides an easily accessible interface that specializes in porous media research while also utilizing Palabos' state-of-the-art solver. Building on top of Palabos also allows for efficient parallelization, and simulations can be run anywhere from single-core systems up to peta-scale clusters [54]. The main feature of MPLBM-UT is that the user does not have to be familiar with the Palabos classes or data descriptors in C++ to install the code or run a simulation. Installing MPLBM-UT is simple: it requires one line of code. This is more user-friendly compared to other open-source direct simulation and LBM softwares. Furthermore, the most common simulation cases have been provided as examples, so the main modifications will come from modifying the values in the example input files. MPLBM-UT also offers additional 2D, 3D, and 4D visualization tools to help perform additional simulation analysis. Overall, MPLBM-UT adds a flexible LBM simulation tool that can easily be used and customized without a deep knowledge of advanced computational tools and coding.

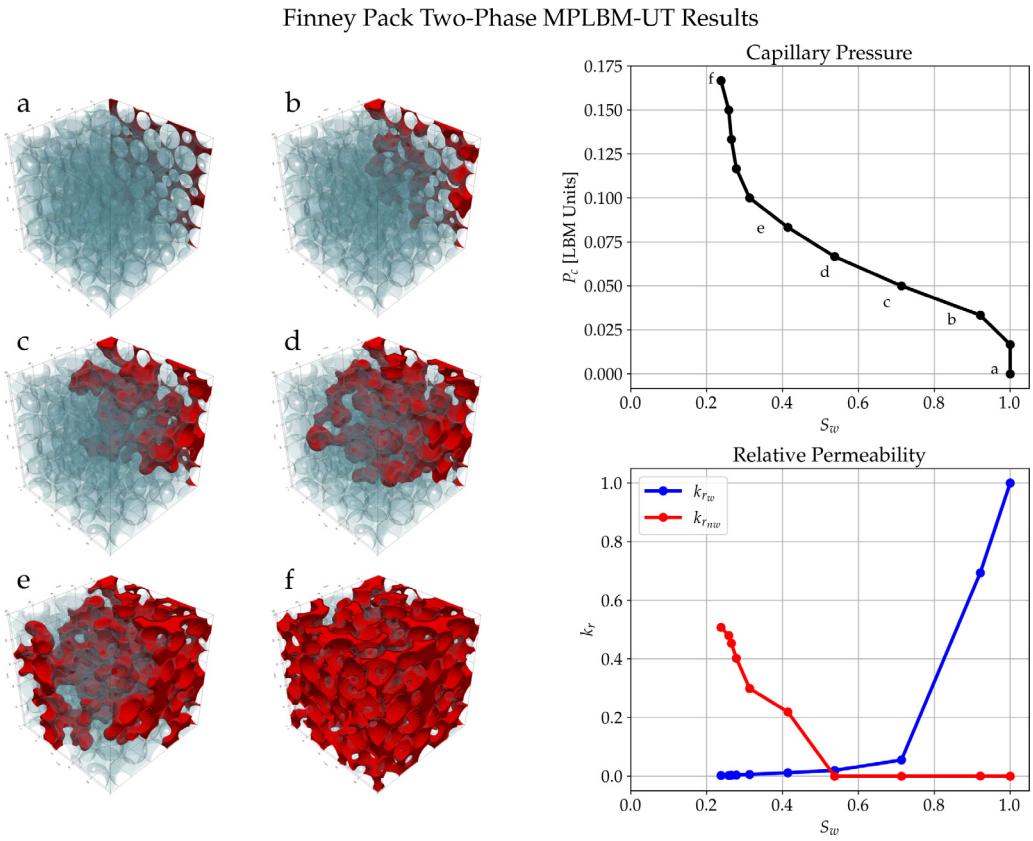

Fig. 3. Images (a)–(f) are frames from the sphere pack LBM simulation. The wetting-phase is represented in blue and the non-wetting phase in red. The top left image (a) is the simulation initial condition, and the bottom right image (f) is the simulation end point. The labels are also displayed on the capillary pressure curve at the corresponding saturation values.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

J.E.S gratefully acknowledges the support of the U.S. Department of Energy through the LANL/LDRD Program and the Center for Non-Linear Studies for this work. The authors acknowledge the Oden Institute for Computational Engineering and Sciences along with the Texas Advanced Computing Center (TACC) at The University of Texas at Austin for providing high performance computing resources that have contributed to the research results reported within this paper. A.G., M.P. and M.A.H would also like to acknowledge funding from NASA-EW grant 80NSSC19K0505.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.softx.2022.101097.

References

[1] Xu R, Prodanovic M, Landry C. Pore scale study of water adsorption and subsequent methane transport in clay in the presence of wettability heterogeneity. 2020, Digital Rocks Portal http://dx.doi.org/10.17612/EV4G-KX65. URL https://www.digitalrocksportal.org/projects/276/.

[2] Guiltinan E, Estrada Santos J, Kang Q, Cardenas B, Espinoza DN. Fractures with variable roughness and wettability. 2020, Digital Rocks Portal. http://dx.doi.org/10.17612/P522-CC94. URL https://www.digitalrocksportal.org/projects/314.

[3] Pan C, Hilpert M, Miller CT. Lattice-Boltzmann simulation of two-phase flow in porous media. Water Resour Res 2004;40(1). http://dx.doi.org/10.1029/2003WR002120, URL http://agupubs.onlinelibrary.wiley.com/doi/10.1029/2003WR002120.

[4] Ghanbarzadeh S, Hesse MA, Prodanović M, Gardner JE. Deformation-assisted fluid percolation in rock salt. Science 2015;350(6264):1069–72. http://dx.doi.org/10.1126/science.aac8747, URL http://science.sciencemag.org/content/350/6264/1069.

[5] Ambach W, Blumthaler M, Kirchlechner P. Application of the gravity flow theory to the percolation of melt water through firn. J Glaciol 1981;27(95):67–75. http://dx.doi.org/10.3189/S0022143000011230, Publisher: Cambridge University Press. URL https://www.cambridge.org/core/journals/journal-of-glaciology/article/application-of-the-gravity-flow-theory-to-the-percolation-of-melt-water-through-firn/CA0101904760A33F629EFFFC04D7AF63.

[6] Fountain AG, Walder JS. Water flow through temperate glaciers. Rev Geophys 1998;36(3):299–328. http://dx.doi.org/10.1029/97RG03579, URL http://doi.wiley.com/10.1029/97RG03579.

[7] Pringle DJ, Miner JE, Eicken H, Golden KM. Pore space percolation in sea ice single crystals. J Geophys Res Oceans 2009;114(C12). http://dx.doi.org/10.1029/2008JC005145, _eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2008JC005145. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2008JC005145.

[8] Daigle H, Cook A, Fang Y, Bihani A, Song W, Flemings PB. Gas-driven tensile fracturing in shallow marine sediments. J Geophys Res Solid Earth 2020;125(12):1–19. http://dx.doi.org/10.1029/2020JB020835.

[9] Wark DA, Watson EB. Grain-scale permeabilities of texturally equilibrated, monomineralic rocks. Earth Planet Sci Lett 1998;164(3):591–605. http://dx.doi.org/10.1016/S0012-821X(98)00252-0, URL https://www.sciencedirect.com/science/article/pii/S0012821X98002520.

[10] Ghanbarzadeh S, Hesse MA, Prodanović M. Percolative core formation in planetesimals enabled by hysteresis in metal connectivity. Proc Natl Acad

Sci 2017;114(51):13406–11. http://dx.doi.org/10.1073/pnas.1707580114, URL https://www.pnas.org/content/114/51/13406.

[11] Gigliotti A, Hesse MA, Prodanović M. Pore-scale simulation of two-phase melt percolation during core formation in planetesimals. 2021, LPSC. URL https://www.hou.usra.edu/meetings/lpsc2021/pdf/2328.pdf.

[12] Gigliotti AE. Two-phase percolation in texturally equilibrated porous media (Ph.D. thesis), 2021, http://dx.doi.org/10.26153/tsw/21533. URL https://repositories.lib.utexas.edu/handle/2152/94614. [Accepted: 2022-01-12T01:40:36Z].

[13] Weber AZ, Mench MM, Meyers JP, Ross PN, Gostick JT, Liu Q. Redox flow batteries: A review. J Appl Electrochem 2011;41(10):1137–64. http://dx.doi.org/10.1007/s10800-011-0348-2.

[14] Gostick JT, Fowler MW, Pritzker MD, Ioannidis MA, Behra LM. In-plane and through-plane gas permeability of carbon fiber electrode backing layers. J Power Sources 2006;162(1):228–38. http://dx.doi.org/10.1016/j.jpowsour.2006.06.096.

[15] Wadsworth FB, Vossen CEJ, Heap MJ, Kushnir A, Farquharson JI, Schmid D, et al. The force required to operate the plunger on a french press. Amer J Phys 2021;89(8):769–75. http://dx.doi.org/10.1119/10.0004224, Publisher: American Association of Physics Teachers.

[16] Blunt MJ. Multiphase flow in permeable media: a pore-scale perspective. Cambridge: Cambridge University Press; 2017, http://dx.doi.org/10.1017/9781316145098, URL http://ebooks.cambridge.org/ref/id/CBO9781316145098.

[17] Anbari A, Chien H-T, Datta SS, Deng W, Weitz DA, Fan J. Microfluidic model porous media: Fabrication and applications. Small 2018;14(18):1703575. http://dx.doi.org/10.1002/smll.201703575, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/smll.201703575. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/smll.201703575.

[18] Wildenschild D, Sheppard AP. X-ray imaging and analysis techniques for quantifying pore-scale structure and processes in subsurface porous medium systems. Adv Water Resour 2013;51:217–46. http://dx.doi.org/10.1016/j.advwatres.2012.07.018, URL https://www.sciencedirect.com/science/article/pii/S0309170812002060.

[19] Blunt MJ, Bijeljic B, Dong H, Gharbi O, Iglauer S, Mostaghimi P, et al. Pore-scale imaging and modelling. Adv Water Resour 2013;51:197–216. http://dx.doi.org/10.1016/j.advwatres.2012.03.003, URL https://www.sciencedirect.com/science/article/pii/S0309170812000528.

[20] Mehmani A, Verma R, Prodanović M. Pore-scale modeling of carbonates. Mar Pet Geol 2020;114:104141. http://dx.doi.org/10.1016/j.marpetgeo.2019.104141, URL https://www.sciencedirect.com/science/article/pii/S0264817219305938.

[21] Zhu Y, Fox PJ. Smoothed particle hydrodynamics model for diffusion through porous media. Transp Porous Media 2001;43(3):441–71. http://dx.doi.org/10.1023/A:1010769915901.

[22] Tartakovsky AM, Meakin P, Scheibe TD, Wood BD. A smoothed particle hydrodynamics model for reactive transport and mineral precipitation in porous and fractured porous media: SPH REACTIVE TRANSPORT MODEL. Water Resour Res 2007;43(5). http://dx.doi.org/10.1029/2005WR004770, URL http://doi.wiley.com/10.1029/2005WR004770.

[23] Hou TY, Wu X-H. A multiscale finite element method for elliptic problems in composite materials and porous media. J Comput Phys 1997;134(1):169–89. http://dx.doi.org/10.1006/jcph.1997.5682, URL https://linkinghub.elsevier.com/retrieve/pii/S0021999197956825.

[24] Huang H, Sukop M, Lu X. Multiphase lattice boltzmann methods: theory and application. John Wiley & Sons; 2015, Google-Books-ID. gsbvCQAAQBAJ.

[25] Krüger T, Kusumaatmaja H, Kuzmin A, Shardt O, Silva G, Viggen EM. The lattice Boltzmann method: Principles and practice. In: Graduate texts in physics, Cham: Springer International Publishing; 2017, http://dx.doi.org/10.1007/978-3-319-44649-3, URL http://link.springer.com/10.1007/978-3-319-44649-3.

[26] Chen S, Doolen GD. Lattice Boltzmann method. Annu Rev Fluid Mech 1998;30(Kadanoff 1986):329–64. http://dx.doi.org/10.1007/978-3-540-27982-2, ISBN: 978-3-540-27981-5.

[27] Prodanovic M, Esteva M, Hanlon M, Nanda G, Agarwal P. Digital rocks portal: A repository for porous media images. 2015, http://dx.doi.org/10.17612/P7CC7K. URL https://www.digitalrocksportal.org/.

[28] Santos JE, Bihani A, nv4dll, Gigliotti A, Jack. MPLBM-UT: Multiphase LBM toolbox for permeable media analysis. 2021, Zenodo. http://dx.doi.org/10.5281/zenodo.4665905. URL https://zenodo.org/record/4665905.

[29] Armstrong RT, McClure JE, Berrill MA, Rücker M, Schlüter S, Berg S. Beyond Darcy's law: The role of phase topology and ganglion dynamics for two-fluid flow. Phys Rev E 2016;94(4):043113. http://dx.doi.org/10.1103/PhysRevE.94.043113, https://link.aps.org/doi/10.1103/PhysRevE.94.043113.

[30] McNamara GR, Zanetti G. Use of the boltzmann equation to simulate lattice-gas automata. Phys Rev Lett 1988;61(20):2332–5. http://dx.doi.org/10.1103/PhysRevLett.61.2332, ISBN: 1079-7114.

[31] Wolfram S. Statistical mechanics of cellular automata. Rev Modern Phys 1983;55(3):601–44. http://dx.doi.org/10.1103/RevModPhys.55.601, http://linkinghub.elsevier.com/retrieve/pii/S0092867413008957. https://link.aps.org/doi/10.1103/RevModPhys.55.601.

[32] He X, Luo L-S. Lattice boltzmann model for the incompressible navier–stokes equation. J Stat Phys 1997;88(3):927–44. http://dx.doi.org/10.1023/B:JOSS.0000015179.12689.e4, https://doi.org/10.1023/B:JOSS.0000015179.12689.e4.

[33] Bhatnagar PL, Gross EP, Krook M. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. Phys Rev 1954;94(3):511–25. http://dx.doi.org/10.1103/PhysRev.94.511, https://link.aps.org/doi/10.1103/PhysRev.94.511, Publisher: American Physical Society.

[34] Higuera FJ, Jiménez J. Boltzmann approach to lattice gas simulations. Europhys Lett (EPL) 1989;9(7):663–8. http://dx.doi.org/10.1209/0295-5075/9/7/009, https://doi.org/10.1209/0295-5075/9/7/009, Publisher: IOP Publishing.

[35] d'Humiéres D. Multiple–relaxationtime lattice boltzmann models in three dimensions. In: Coveney PV, Succi S, Ginzburg I, Krafczyk M, Lallemand P, Luo L-S, editors. Philos Trans R Soc Lond 2002;360(1792):437–51. http://dx.doi.org/10.1098/rsta.2001.0955, https://royalsocietypublishing.org/doi/10.1098/rsta.2001.0955.

[36] Shan X, Chen H. Lattice Boltzmann model for simulating flows with multiple phases and components. Phys Rev E 1993;47(3):1815–9. http://dx.doi.org/10.1103/PhysRevE.47.1815, URL https://link.aps.org/doi/10.1103/PhysRevE.47.1815.

[37] Shan X, Chen H. Simulation of nonideal gases and liquid-gas phase transitions by the lattice Boltzmann equation. Phys Rev E 1994;49(4):2941–8. http://dx.doi.org/10.1103/PhysRevE.49.2941.

[38] Huang H, Thorne DT, Schaap MG, Sukop MC. Proposed approximation for contact angles in Shan-and-Chen-type multicomponent multiphase lattice Boltzmann models. Phys Rev E Stat Nonlinear Soft Matter Phys 2007;76(6):1–6. http://dx.doi.org/10.1103/PhysRevE.76.066701, ISBN: 1550-2376.

[39] Latt J, Malaspinas O, Kontaxakis D, Parmigiani A, Lagrava D, Brogi F, et al. Palabos: Parallel lattice Boltzmann solver. Comput Math Appl 2021;81:334–50. http://dx.doi.org/10.1016/j.camwa.2020.03.022, URL https://www.sciencedirect.com/science/article/pii/S0898122120301267.

[40] Santos JE, Prodanović M, Landry CJ, Jo H. Determining the impact of mineralogy composition for multiphase flow through hydraulically induced fractures. In: Proceedings of the 6th unconventional resources technology conference. Society of Petroleum Engineers; 2018, p. 1–15. http://dx.doi.org/10.15530/urtec-2018-2902986.

[41] Young T. An essay on the cohesion of fluids. Philos Trans R Soc Lond 1805;95:65–87, Publisher: The Royal Society. URL http://www.jstor.org/stable/107159.

[42] Laplace PS, Gordon H, Gordon J. Traitee de mecanique celeste. A Paris: Chez J.B.M. Duprat; 1798, http://dx.doi.org/10.5479/sil.338664.39088005644752, De L'Imprimerie de Crapelet. URL https://library.si.edu/digital-library/book/traite-de-mecanique-celeste.

[43] Washburn EW. The dynamics of capillary flow. Phys Rev 1921;17(3):273–83. http://dx.doi.org/10.1103/PhysRev.17.273, Publisher: American Physical Society. URL https://link.aps.org/doi/10.1103/PhysRev.17.273.

[44] Brooks RH, Corey A. Hydraulic properties of porous media and their relation to drainage design. Trans ASABE 1964;7:26–8.

[45] Santos JE, Xu D, Jo H, Landry CJ, Prodanović M, Pyrcz MJ. PoreFlow-net: a 3D convolutional neural network to predict fluid flow through porous media. Adv Water Resour 2020;103539. http://dx.doi.org/10.1016/j.advwatres.2020.103539, URL http://www.sciencedirect.com/science/article/pii/S0309170819311145.

[46] Santos JE, Yin Y, Jo H, Pan W, Kang Q, Viswanathan HS, et al. Computationally efficient multiscale neural networks applied to fluid flow in complex 3D porous media. No. 0123456789. Springer Netherlands; 2021, http://dx.doi.org/10.1007/s11242-021-01617-y, ISSN: 15731634 Publication Title: Transport in Porous Media.

[47] Santos JE, Pyrcz MJ, Prodanović M. 3D Dataset of binary images: A collection of synthetically created digital rock images of complex media. Data Brief 2022;40. http://dx.doi.org/10.1016/j.dib.2022.107797.

[48] Gostick J, Khan Z, Tranter T, Kok M, Agnaou M, Sadeghi M, Jervis R. PoreSpy: A Python Toolkit For Quantitative Analysis of Porous Media Images. J Open Source Softw 2019;4(37):1296. http://dx.doi.org/10.21105/joss.01296.

[49] Sullivan B, Kaszynski A, Favelier G, Koyama T, Deak A, Jones J, et al. Pyvista. 2021, Zenodo. http://dx.doi.org/10.5281/zenodo.5499095. URL https://zenodo.org/record/5499095.

[50] Musy M, Jacquenot G, Dalmasso G, Neoglez, Bruin Rd, Pollack A, et al. marcomusy/vedo: 2022.1.0. 2022, Zenodo. http://dx.doi.org/10.5281/zenodo.6247803. URL https://zenodo.org/record/6247803.

[51] Finney JL. Finney packing of spheres. 2016, type: dataset. Digital Rocks Portal. http://dx.doi.org/10.17612/P78G69. URL https://www.digitalrocksportal.org/projects/47/.

[52] Bryant S, Blunt M. Prediction of relative permeability in simple porous media. Phys Rev A 1992;46(4):2004–11. http://dx.doi.org/10.1103/PhysRevA.46.2004, URL https://link.aps.org/doi/10.1103/PhysRevA.46.2004.

[53] Mousavi MA. Pore-scale characterization and modeling of two-phase flow in tight gas sandstones (Ph.D. thesis), 2010, URL https://repositories.lib.utexas.edu/handle/2152/ETD-UT-2010-05-1452. [Accepted: 2011-01-07T21:56:29Z].

[54] Tian M, Gu W, Pan J, Guo M. Performance analysis and optimization of PalaBos on petascale sunway BlueLight MPP supercomputer. Procedia Eng 2013;61:241–5. http://dx.doi.org/10.1016/j.proeng.2013.08.010, Publisher: Elsevier B.V. URL http://dx.doi.org/10.1016/j.proeng.2013.08.010.