Accepted Manuscript

Finite-difference method Stokes solver (FDMSS) for 3D pore geometries: Software development, validation and case studies

Kirill M. Gerke, Roman V. Vasilyev, Siarhei Khirevich, Daniel Collins, Marina V. Karsanina, Timofey O. Sizonenko, Dmitry V. Korost, Sébastien Lamontagne, Dirk Mallants

PII: S0098-3004(17)30623-4

DOI: [10.1016/j.cageo.2018.01.005](https://doi.org/10.1016/j.cageo.2018.01.005)

Reference: CAGEO 4078

To appear in: *Computers and Geosciences*

Received Date: 1 June 2017

Revised Date: 13 October 2017

Accepted Date: 14 January 2018

Please cite this article as: Gerke, K.M., Vasilyev, R.V., Khirevich, S., Collins, D., Karsanina, M.V., Sizonenko, T.O., Korost, D.V., Lamontagne, Sé., Mallants, D., Finite-difference method Stokes solver (FDMSS) for 3D pore geometries: Software development, validation and case studies, *Computers and Geosciences* (2018), doi: 10.1016/j.cageo.2018.01.005.

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Finite-difference method Stokes solver (FDMSS) for 3D pore geometries: software development, validation and case studies

Kirill M. Gerke^{a,b,c}, Roman V. Vasilyev^d, Siarhei Khirevich^e, Daniel Collins^f, Marina V. Karsanina^{a,b,c}, Timofey O. Sizonenko^b, Dmitry V. Korost^d, Sébastien Lamontagne^g, Dirk Mallants^g

^aInstitute of Geosphere's Dynamics of Russian Academy of Sciences, Moscow, Russia

^bInstitute of Physics of the Earth of Russian Academy of Sciences, Moscow, Russia

^cDokuchaev Soil Science Institute of Russian Academy of Sciences, Moscow, Russia

^dGeology Faculty, Lomonosov Moscow State University, Moscow, Russia

^eAli I. Al-Naimi Petroleum Engineering Research Center, KAUST, Thuwal, Saudi Arabia

^fCSIRO IM&T, Perth, Australia

^gCSIRO Land and Water, PB2 Glen Osmond, Australia

Corresponding author: Kirill M. Gerke, Tel.: +79661877715, E-mail: kirill.gerke@gmail.com,

Address: Institute of Geosphere's Dynamics of RAS, Leninskiy prosp., 38/1, 119334, Moscow, Russia

Abstract:

Permeability is one of the fundamental properties of porous media and is required for large-scale Darcian fluid flow and mass transport models. Whilst permeability can be measured directly at a range of scales, there are increasing opportunities to evaluate permeability from pore-scale fluid flow simulations. We introduce the free software Finite-Difference Method Stokes Solver (FDMSS) that solves Stokes equation using a finite-difference method (FDM) directly on voxelized 3D pore geometries (i.e. without meshing). Based on explicit convergence studies, validation on sphere packings with analytically known permeabilities, and comparison against lattice-Boltzmann and

other published FDM studies, we conclude that FDMSS provides a computationally efficient and accurate basis for single-phase pore-scale flow simulations. By implementing an efficient parallelization and code optimization scheme, permeability inferences can now be made from 3D images of up to 10^9 voxels using modern desktop computers. Case studies demonstrate the broad applicability of the FDMSS software for both natural and artificial porous media.

Highlights:

- 1) Free software to simulate pore-scale single-phase flow
- 2) Software is simple to use and contains both batch and GUI versions
- 3) Parallelization and code optimization make simulations on modern laptops possible
- 4) Software was verified using known sphere packing and by comparison to published data
- 5) Five different case studies illustrate the main capabilities and functionalities of the software

Keywords: pore-scale modeling; Stokes flow solver; X-ray micro-tomography (XMT); velocity field; pressure field; REV analysis

1. Introduction

Permeability is one of the fundamental properties of any porous media and is needed for modelling a variety of processes, such as groundwater flow, multi-phase oil and gas flow, and any industrial processes using the Darcian continuum theory. Whilst permeability can be evaluated at a range of scales, in some instances measurements cannot be performed due to size or shape limitations (e.g. drilling chips) or samples being damaged by flow measurements, in which case it is preferable to evaluate permeability at the pore scale (Blunt et al., 2013). Pore-scale modelling requires solution of flow and transport problems directly within 3D pore structures and may also require knowledge on pore wall constituents to properly address fluid-solid interactions. It is widely acknowledged that the structure of porous media defines all its properties, i.e. the spatial

arrangement of its originating phases including quartz grains, minerals, organic matter, clays, water, oil, air, etc. When a defined structure is spatially discretized to define all relevant features, it can be used to solve the governing equations to infer the desired flow and transport properties at the pore scale.

Pore scale evaluation of permeability overcomes some of the limitations typical of laboratory measurements, such as: 1) measurement errors (Diamond, 2000; Capek et al., 2014); 2) changes to the physical properties of samples by cracking, self-filtration, and other processes (Dikinya et al., 2008; Zeinijahromi et al., 2016); 3) distortion by boundary effects (Duwig et al., 2008; Scheibe et al., 2015); 4) core sizes being smaller than the Representative Elemental Volume (REV) (Mostaghimi et al., 2013); 5) difficulties to measure on irregular shapes or small sizes (Gerke et al., 2015a), 6) inability to determine dynamic structure influences on permeability (Coppola et al., 2012), and 7) limitations on measuring directional or tensorial dependencies (Renard et al., 2001; Guibert et al., 2015). Recent advances in measurement and modelling techniques have resulted in significant improvements in the ability to reconstruct pore networks in 3D (Adler et al., 1990; Yeong and Torquato, 1998; Moctezuma-Berthier et al., 2002; Wildenschild et al., 2002; Øren and Bakke, 2003; Arns et al., 2003; Cnudde et al., 2006; Thovert and Adler, 2011; Gerke et al., 2012b; Jiao and Chawla, 2014; Gerke et al., 2015c; Mariethoz et al., 2015; Li et al., 2016; Havelka et al., 2016), which is a prerequisite to evaluate permeability. For simplicity and without loss of applicability, we focus in the following on permeability for single phase flow for Newtonian fluids.

Single-phase permeabilities considered to represent fully saturated flow may vary, for the same pore structure, depending on scale effects (e.g., gas flow in nanoporous rocks (Shabro et al., 2009; Mehmani et al., 2013)) or boundary conditions (e.g., high speed filtration or compressed fluid filtration under large pressure gradients, Langmuir desorption during gas production (Shabro et al., 2001)). Permeabilities that do not obey Darcy's law and do not depend linearly on pressure are usually called non-Darcian. These are beyond the scope of the current paper. From now the discussion will focus on Darcian flow, i.e. laminar single-phase fluid flow with no-flow boundary

conditions on the pore walls. One approach to use pore structure information for permeability estimation is based on the Kozeny-Carman relationships (Xu and Yu, 2008) or the effective medium approximations (Lock et al., 2002). Such approaches work well only for simple geometries such as spheroid arrays but fail if applied to real rock and soil 3D pore images (Narsilio et al., 2009; Mostaghimi et al., 2013; Ghanbarian and Daigle, 2016). Because of the limitations of these approaches, various pore-scale modelling methods have been developed. Our work uses existing pore-scale theories as a basis for developing a simulator for fluid flow in 3D pore geometries.

Several computational methods can be used to model laminar fluid flow within heterogeneous pore spaces. Among the most popular are the: 1) Lattice Boltzmann method (LBM) (Bhatnagar et al., 1954; Ginzbourg and Adler, 1994; Khirevich et al., 2015); 2) finite element (FEM), finite volume (FVM) and volume of fluid (VoF) methods (Garcia et al., 2009; Akanji and Matthai, 2010; Zaretskiy et al., 2010; Raeini et al., 2012; Bird et al., 2014); 3) smoothed particle hydrodynamics (SPH) (Tartakovsky et al., 2015; Holmes et al., 2016); 4) finite-difference solution of Stokes equation (FDM) (Vladimirova et al., 1966; Chorin, 1967; Martys et al., 1994; Shabro et al., 2012; Noiriél et al., 2015); and 5) pore-network models (PNM) (Silin and Patzek, 2006; Jiang et al., 2007; Dong and Blunt, 2009; Miao et al., 2017). The FDM approach was used extensively in the 1980-90s to solve flow in 3D geometries (Adler et al., 1990; Martys et al., 1994). However, all computational techniques to model laminar fluid flow are computationally intensive, which limits the size of the pore network domain that can be evaluated. Because of the need to work on as large as possible 3D pore geometries in order to capture the REV of a porous medium, there is a need to significantly improve the efficiency for all laminar fluid flow modelling techniques.

The objective of this paper is to present a computationally-efficient, artificial compressibility FDM scheme to solve single fluid flow within heterogeneous pore space represented by voxelized 3D geometries. This scheme was encapsulated in a free software that can be employed on most modern laptop and desktop computer systems. Moreover, the software was made user-friendly by including a graphical interface for users less familiar with computational fluid dynamics. We

verified and compared our code extensively against existing published data, a double relaxation LBM scheme, and various analytical solutions of idealised porous media made of spheres of different sizes and packing.

2. FDMSS software description

The software solves Stokes equation (see Appendix A.1) to obtain a 3D velocity field, which can be used to calculate permeability (Appendix A.2). The artificial compressibility FDM scheme utilized is described in Appendix B. The FDMSS software is written in C++ (solver) and C#, C++/CLI with usage of OpenGL (GUI program), and consists of two main parts: 1) the computational executables with 2nd and 4th order accuracy numerical schemes; 2) the pre and post-processing tools with a graphical interface which provides a facility to setup computations with minimal effort. Appendix C describes all necessary details on the solver's structure, computation error criterion, input data and parameters, visualization and post-processing of modelling results, parallelization efficiency, as well as overall computational performance and benchmarking against published FDM solvers.

3. Validation against analytical solutions, comparison to LBM and other FDM

A number of validation studies have been performed for FDM Stokes solvers. Bentz and Martys (2007) and Manwart et al. (2002) tested their FDM solvers against analytical solutions for circular and square pipes and found the error for hydraulic conductance to be within 0.01-2%. Mostaghimi et al. (2013) tested against analytical velocity profiles for flow between two parallel plates with varying apertures and reported a perfect agreement. They also modelled flow around solid cubes and compared the velocity field against the OpenFOAM solution and found that the average velocity was within 0.1% and local velocities were within 0.5% of the reference solution. Moreover, Mostaghimi et al. (2013) compared their solutions for real porous medium samples with LBM solutions, again with almost perfect agreement. However, no LBM type or parameters were

reported. Whilst validations on FDM and other models can be made on simple single pore geometries (Bentz and Martys (2007); Manwart et al. (2002); Mostaghimi et al. (2013)), these lack the pore-structure complexity exhibited in real porous media. In this section, we rigorously test our solver against analytical solutions for sphere packs and also compare it to a two-relaxation-times LBM scheme with three different ‘magic number’ parameters controlling the location of zero-velocity between adjacent solid and fluid voxels (Khirevich et al., 2015). In addition, we simulate flow in the Mostaghimi et al. (2013) samples and compare the resulting permeabilities.

3.1 General validation on sphere packs and comparison against LBM

Results of computations on sphere packings using LBM with different ‘magic numbers’ (see Appendix D for sphere packing and LBM computational approach description details) and for FDM with the 2nd and 4th order spatial accuracy scheme are presented on Figure 1 in terms of % error against analytical drag force. Generally speaking, the accuracy converges with increasing sphere discretization (d_{sp}) for all methods used, but passes a local maximum for some of the geometries. With properly adjusted ‘magic numbers’ (i.e. $\Lambda = 3/16$ and $3/8$), LBM shows better accuracy than FDM for lower d_{sp} values. Still, the curve patterns for LBM and FDM are quite similar and may indicate generally similar causes that led to the observed errors. At low discretization resolutions ($d_{sp} < 20$) and non-simple cubic (SC) geometries the FDM error behaviour is quite close to the LBM with smaller Λ values, namely $\Lambda = 1/8$ in Khirevich et al. (2015) where the location of zero velocities approaches the fluid voxel center adjacent to the wall voxel.

Pore-solids geometry and porosity plays a significant role, as the error for a given discretization drops significantly with increasing porosity with FDM reaching minimum values for an SC packing with porosity of 0.476. This may be directly related to the size of significant flow pathways (pore throats), as they grow in size with porosity and discretization. There is also one additional significant difference between LBM and FDM in treating diagonal connections between voxels. LBM has flow along diagonal directions, whilst FDM maintains only orthogonal directions. For a

very low degree of discretizations, diagonal connections grow in their importance for flow and LBM simulations continue to show good agreement with analytical drags. On the other hand, FDM-derived permeabilities had a very high error due to only orthogonal flow and even provided zero permeability for only diagonally connected packings (FCC 0.26 with d_{sp} of 3.37 and 4.16). Tuning LBM parameters may result in drastic changes in the accuracy of the permeability inferences, whilst improving the FDM spatial accuracy scheme provided only minor increase in simulated permeability accuracy.

The staggered marker-and-cell (MAC) grid is usually praised for its ability to put zero velocity boundaries exactly at the pore-solid interface (Payret and Taylor, 1983; Mostaghimi et al., 2013) which is believed to produce accurate solutions. In practice, as seen from Figure 1, with decreasing discretization this approach produced large deviations from analytical values. On the other hand, when tuning the position of the zero velocity with ‘magic number’ parameters, LBM can be efficiently tuned to produce more accurate results when using lower discretizations for sphere packings. However, these parameters are not known *a priori*. Unlike for LBM, the FDM method does not possess tuning parameters that could be used in a similar manner as ‘magic numbers’.

Sphere packings provide a refined model of heterogeneous porous media and, as was mentioned earlier, analysis of LBM tuning for more complex geometries still remains an active research topic. Based on our previous simulations for soil samples (Gerke et al., 2017, see example of the similar 3D image in Section 4.2), we now briefly compare computed permeabilities for 3D pore geometries obtained by LBM with $\Lambda = 3/16$, $3/8$ and a 4th accuracy scheme FDM. Pore geometries were obtained using XMT scanning. The difference between the two LBM Λ results is within 15-20% for four different soil samples. The difference between FDM and LBM with $\Lambda = 3/16$ and $\Lambda = 3/8$ varies between 23-34% and 34-43%, respectively. Similar to results obtained on sphere packings, FDM-inferred permeability values were lower than LBM, but the difference was within 43%. However, relative to expectations (Figure 1), less variability than expected was found for the more complex geometries of real porous media. In this context, the conclusion of Manwart et al. (2002)

that under similar boundary conditions LBM and FDM methods produce similar results is a valid statement for real pore geometries. However, this conclusion does not hold for low discretized sphere packings, where LBM clearly outperforms FDM in terms of accuracy. This should not be considered as a significant drawback of the FDM approach if applied to real porous media images, as poorly discretized systems (e.g. low-resolution imaging resulting in single-voxel dominated pore-throats) are by definition not appropriate input data. In addition, there is always uncertainty whether or not diagonally connected voxels represent real porous media connections. Partial volume effects contribute to this uncertainty (Wildenschild and Sheppard, 2013).

Fig.1. Comparison of numerical errors for drag force of the sphere packings obtained using the LBM method with magic number $\Lambda = 1/8$, $3/16$ and $3/8$ and FDMSS with 2nd and 4th order accuracy

scheme. FCC = face-centered cubic; BCC = body-centered cubic; SC = simple cubic, and ϕ = porosity.

The segmentation procedure to interpret XMT images is another important aspect to consider for FDM and LBM Stokes flow simulations. Allocation of zero velocity boundaries by binarization of grey scale images inevitably fluctuates around real pore-solids interfaces due to image resolution effects, i.e. partial volume effects (Gerke et al., 2012a; Wildenschild and Sheppard, 2013). Thus, different segmentation approaches will act as discrete magic numbers and will affect FDM and LBM simulations differently. Many conventional segmentation and threshold selection techniques generally oversegment grey scale images significantly (Gerke et al., 2017). Because the accuracy of FDM and LBM solutions for a given sample will strongly depend on the approach used, the segmentation procedure should be selected carefully and compared whenever possible against known geometries.

Our implementation of the FDM method is more computationally efficient than the FDM code of Mostaghimi et al. (2013) and, also the LBM. However, direct comparison of computational efficiency is complicated in the latter case. First, memory and computational time requirements for the LBM method can depend significantly on the preferred scheme, e.g. D3Q13 or D3Q19. Manwart et al. (2002) reported that their implementation of FDM required 2.5 times less RAM than their realization of D3Q19 LBM. Their pressure correction FDM scheme is quite similar to that of Mostaghimi et al. (2013) and our method is around 2.5-3 times more efficient in terms of RAM, giving 6-7 times less RAM consumption compared to LBM. Note that the LBM RAM requirements can be optimized significantly depending on the porosity of the 3D image (Ma et al., 2010). Generally, any LBM approach will require a float number for each velocity direction to be stored for each fluid voxel, plus some connectivity information (Ma et al., 2010). In our realization of FDM, we store only one pressure for each voxel plus the six velocities shared with neighbouring voxels. This very simple approximation improves our FDM RAM efficiency by a factor of 4 to 6. If the FDM approach of Manwart et al. (2002) is indeed similar to that of Mostaghimi et al. (2013),

our FDM is approximately an order of magnitude faster than LBM. Manwart et al. (2002) noticed that computational time for their FDM and LBM were similar. Comparison of computational time requirements for the 2nd and 4th order spatial schemes of our FDM approach revealed only a 10-20% difference between the two (see next section for details). As a 4th order scheme requires 10-20% more computational operations than a 2nd order one, it may indicate that RAM operations have more impact on computation speed than CPU time. In this case, as FDM requires less RAM, this produces additional speedup.

3.2 Comparison against other available data

To further elucidate the capabilities of FDMSS, we evaluated absolute permeabilities for rock samples that were previously investigated using other FDM Stokes solvers (Mostaghimi et al., 2012; Mostaghimi et al., 2013) and the LBM simulations of Dong and Blunt (2009). Similar to these previous studies, our permeability estimates are arithmetical averages over computations in the three principal directions. Relative to the other studies, our permeability estimates should be slightly more accurate according to the criterion that the normalized error $E < 0.01$ (see Appendix C.1 and SI.Section 1 for details), because we used a 4th order accuracy scheme. The inferred permeabilities and other general information for the samples are given in Table 1.

Table 1. Comparison of permeabilities computed using FDMSS, Mostaghimi et al. (2012) and Mostaghimi et al. (2013) (FDM_M), and LBM simulations reported by Dong and Blunt (2009) (LBM_DB).

Sample name	Imaging resolution (μm)	Volume (voxels)	Porosity (%)	Permeability (D)		
				LBM_DB	FDM_M	FDMSS
Berea	5.3	400 ³	19.64	1.3	1.4	1.3
Carbonate 1	2.9	400 ³	23.26	1.1	1.1	1.4

Carbonate 2	5.3	400 ³	16.83	0.072	0.14	0.20
Sandpack F42A	10	450 ³	32.2	59	62	49
Sandpack F42B	10	450 ³	33.25	52	47	49
Sandpack F42C	10	450 ³	32.88	50	54	45
Sandpack LV60A	10	450 ³	36.3	35	39	27
Sandpack LV60C	10	450 ³	36.84	19	22	25
Sandstone 1	8.7	300 ³	14.13	1.7	1.9	1.8
Sandstone 2	5	300 ³	24.63	3.9	2.9	3.6
Sandstone 3	9.1	300 ³	16.86	0.22	0.19	0.70
Sandstone 4	9	300 ³	17.13	0.26	0.31	0.39

Generally, FDM_M permeabilities were similar to or somewhat higher than the ones from the LBM simulations of Dong and Blunt (2009). Our FDM results fluctuate around FDM_M values and are either higher or lower depending on the sample and deviate by 4-73% (samples Sandpack F42B and Sandstone 3 being extremes) and on average by 21%. These differences in the computed permeabilities cannot be exclusively attributed to differences in finite-difference schemes, as Mostaghimi et al. (2013) reported calculations for 300³ voxels samples, while the 3D images we could obtain from the Imperial College collection had somewhat higher dimensions up to 450³ voxels (Table 1). At least for Sandstone samples 1-4 with a volume of 300³ voxels, our results were generally higher. The comparison with the LBM simulations from Dong and Blunt (2009) were hampered by the lack of parameterization details of their model (relaxation model, magic numbers, etc.). In general, observed differences strongly suggest that FDM_M and LBM_DB permeability estimates for the sphere packs discussed in previous subsections would be underestimated in a similar manner as observed for FDM. Overall, this analysis highlighted some of the difficulties involved in comparing different codes for flow simulations. Issues include: 1) need for robust surveys on similar 3D images, and 2) requirement to report all details and parameters for the model simulations. In general, our FDMSS code provides comparable results as the FDM_M method and can produce both higher and lower values depending on the 3D pore geometry at hand.

4. Case studies and application examples

This section describes a number of applications where the software was utilized to solve particular problems. We cover a broad set of porous materials and show the usefulness of the software for a multitude of research areas. All case studies make use of 3D images obtained using a Skyscan-Bruker 1172 desktop XMT scanner. Analysis of XMT images involved subcropping using the Fiji/ImageJ software, denoising (using median or non-local means filters) and segmentation using an in-house software. The segmentation is applied based on indicator kriging with two manually adjusted confidence thresholds or converging active contours (see Gerke et al. (2017) for more details on the segmentation approach).

4.1 Oil-bearing sandstone permeability

This example involves calculating permeability values for oil bearing sandstone rocks by simulating Stokes flow. A 4-cm diameter core sample of polymictic sandstone representing Tyumen Formation, Western Siberia, was subdrilled to produce a 10-mm sample for XMT scanning. The 4-cm core was also utilized to measure permeability using a standard laboratory method based on fluid filtration under known pressure gradient and outflow measurements.

A subset of manually binarized XMT images was cropped from the original stack for simulations resulting in a 3D modelling domain containing 700^3 voxels with a resolution of $5\text{ }\mu\text{m}$. The total flow domain covers a rock volume of 3.5^3 mm^3 . Flow velocity fields were modelled by applying pressure gradients across three principal directions while treating all other sides of the cube as solid walls (see Figure 2a,b). Based on the velocity fields calculated with Stokes' equation, the software calculated the core-averaged permeability in each principal direction. To compare calculated permeability against laboratory measurements performed on a larger rock volume, it is recommended to arithmetically average permeability over all directions (Dong and Blunt, 2009). This is done because the exact resulting pressure gradient and flow lines may be different from

those modelled in the small subdomain while all principal directions may contribute to the flow if this subvolume is submerged into the larger sample (e.g., 3.5 mm^3 subvolume within $\sim 100 \text{ cm}^3$ of 4-cm diameter rock core).

Fig. 2. Schematic of cubic domain geometry and boundary conditions for REV analysis: a) representation of the sub-cubes sampled from the original full size domain nested within each other, b) a position of the cubic modelling domain for flow simulations in one of the principal orthogonal directions (other directions are assessed by a rotation of the cube), c) 2D slice through the cube showing the implementation of solid walls boundary conditions on the faces parallel to the main flow directions.

The 3D pore structure of the sandstone sample together with calculated pressure and velocity fields are shown in Figure 3. The directional permeabilities were calculated to be 192.1, 140.5 and 133.2 mD along X, Y and Z axes, respectively. The arithmetically averaged value of 155.3 mD agrees quite well the laboratory measured permeability of 136 mD.

Fig.3. 3D rock pore structure (left), calculated pressure field (middle) and velocity field (right) for a simulation along the sample X axis (full velocities are shown as scalars for each voxel).

4.2 Soil permeability

Using XMT to assess soil hydraulic properties is a topic of significant interest (Narsilio et al., 2009; Khan et al., 2012; Hyväluoma et al., 2012; Dal Ferro et al., 2015). Here we use a 3D XMT image taken from the Ah horizon of a Haplic Greyzem soil representing a 700^3 voxel cube with resolution of $20.96 \mu\text{m}$. The original grey-scale image was re-segmented using the converging contours approach (Sheppard et al., 2004) with manually selected thresholds slightly lower than in Gerke et al. (2017). In the latter study it was shown that general segmentation approaches usually result in significant oversegmentation. The results of the simulations are show in Figure 4.

Fig.4. Visualization of simulation results on soil sample (in X direction), including pore structure (left), pressure field (middle) and major velocity flow paths.

The directional hydraulic conductivities (assuming hydrostatic pressure and standard room temperature of 20 °C conditions for water) were calculated to be 4.25, 5.63 and 4.11 m/day along X, Y and Z axes, respectively. Both the arithmetic average value of 4.66 m/day and the Z-axis value of 4.11 m/day agree quite well with the laboratory measurements on undisturbed columns of 5.20 m/day (measurement direction corresponds with X-direction from simulations) and with field infiltrometer measurements of 4.32 m/day (measurement direction corresponds with Z-direction from simulations).

4.3 Ceramics (homogeneous porous media) permeability

Unlike complex and hierarchical natural porous media like soils and rocks, artificial porous media such as ceramics provide a great verification tool for XMT-based permeability predictions because they exhibit a homogeneous pore geometry with controlled pore sizes. The latter is achieved by sintering the material from constituents such as corundum grains of known size distribution (Gerke et al., 2015b). This also ensured that the entire pore space can be resolved with a prescribed XMT scanning setup. A 500^3 voxel size subcube scanned with the resolution of $2.24\text{ }\mu\text{m}$ was filtered using a non-local means filter and segmented using indicator kriging with manually chosen thresholds.

The directional permeabilities were calculated to be 2.74 , 2.27 and $2.38\text{ }\mu\text{m}^2$ along X, Y and Z axes, respectively (see simulation results in SI. Figure 9). An averaged value of $2.46\text{ }\mu\text{m}^2$ agrees quite well the laboratory measured permeability of $1.96\text{ }\mu\text{m}^2$.

4.4 REV determination

One of the potential applications for pore-scale modelled permeability estimates is to increase data availability on the heterogeneous subsurface and parameterize large-scale flow models. However, this implies that smaller subsamples used for pore-scale modelling can be treated as

Representative Elementary Volume (REV) to evaluate Darcy fluxes. If the matrix dominates the hydraulic properties of the porous medium at the scale of the problem of interest, this assumption is reasonable (that is, faults, fractures or other features do not control flow at the scale of interest). According to the conventional definition of permeability REV (Bear, 1972; Bayeve and Sposito, 1984), the REV is obtained when the measured or calculated permeability plotted versus increasing sample size reaches a plateau value. Porosity and surface area may be indicators of the permeability REV and, while such an approach has worked for simple sphere packings (Costanza-Robinson et al., 2011), a recent study presented strong evidence for a separate convergence of the permeability, porosity and surface area REVs for heterogeneous natural porous media (Mostaghimi et al., 2013).

Fig.5. Permeability/porosity REV analysis for a) sandstone, b) ceramics and c) carbonate samples. Permeability (in three principal orthogonal directions x, y and z) and porosity values are reported for increasing sizes of the subvolume used for simulations.

We demonstrate the applicability of the FDMSS software for permeability REV analysis using three materials: i) sandstone (this sample is different from the one analysed in Section 4.1), ii) ceramic and iii) carbonate rock. The image resolutions are 5.00, 2.24 and 5.16 μm , respectively. A

Russian nesting doll approach was used, where image subsamples of increasing size were taken starting from a 50^3 voxel sample in the very middle of the 700^3 modelling domain (Figure 2c). Subsequent subsamples were incremented by 50 pixels, yielding eight measurement points for each cube (Figure 5). Directional permeabilities were evaluated for each subcube.

Figure 5 shows the results of the REV analysis for the aforementioned samples. For the ceramic and sandstone samples, the computed permeability REV was reached at 300-700 voxels subsamples. However, no REV was found for the carbonate sample. This was not unexpected because some natural porous media do not have a readily definable REV (Mostaghimi et al., 2013), or the REV may be larger than the measurement scale investigated here.

4.5 Stochastic reconstruction and design of porous media structures

Obtaining a detailed 3D pore geometry structure to simulate fluid flow and determine hydraulic properties is not always possible. In part, 3D imaging methods may not always be available or do not provide the necessary sample-to-resolution ratio (Gerke et al., 2015c). A number of different stochastic reconstruction techniques were developed to obtain 3D information from 2D image(s) and other limited information. To verify a given stochastic reconstruction methodology, inferred morphological and connectedness metrics must be compared to reference structures and the reconstructed images themselves should resemble the original ones closely (Gerke et al., 2012b; Karsanina et al., 2015). For example, depending on the physical property of interest, one could verify the results using mechanical properties simulations (Kumar et al., 2006; Chen et al., 2016), NMR response modelling (Vesely et al., 2015), permeability (Manwart et al., 2002; Gerke and Karsanina, 2015) or relative permeability (Tahmasebi and Sahimi, 2012) simulations. Here, we demonstrate the verification of stochastic reconstructions based on directional correlation functions (Gerke et al., 2014) and the modified Yeong-Torquato method (Yeong and Torquato, 1998a) using FDMSS. The latter method utilizes simulated annealing optimization to match calculated 2D

correlation functions of evolving structure to a reference set of correlation functions by voxels permutations (Figure 6).

Fig.6. General scheme of stochastic reconstruction based on directional correlation functions (for reconstructing a 3D structure from a single 2D image we assumed isotropic properties).

Permeability comparison between original and reconstructed images is especially sensitive to the connectedness of the pore space, which is frequently underestimated (Manwart et al., 2002) in reconstructions produced using the original Yeong-Torquato technique (Yeong and Torquato, 1998b). Figure 6 also shows the similarity between two velocity fields computed using FDMSS for an original 3D image of ceramics (Gerke et al., 2015b) and for its subsequent stochastic reconstruction (Gerke and Karsanina, 2015). The resulting permeabilities differ by around 10%, consistent with a more realistic representation of connected pore spaces when using the modified Yeong-Torquato approach.

In addition to pore network reconstructions, one can also generate pore geometries with predetermined pore sizes and permeabilities (Karsanina et al., 2015b) using analytically constructed

correlation functions (Jiao et al., 2007). Potentially numerous other properties of porous media can be optimized and constructed by utilizing different sets of correlations functions and solving Stokes' equation with the help of FDMSS. Such an approach opens a whole new way to engineer porous media for specific applications. This includes devising schemes for carbon storage (Herring et al., 2015) or vadose zone soil filter mechanisms (Zhao et al., 2005).

7. Summary and outlook

We have created the computationally efficient FDMSS software to solve Stokes flow problems on pore-scale images for different porous media. Simulations for relatively large subvolumes (e.g., 300^3 - 1000^3 voxels) are shown to be practical on regular modern laptops and desktop computers. In comparison with other popular approaches such as other types of FDM, LBM or FEM-based models, our solver is quite fast, and does not require meshing (that is, simulations are performed directly on the voxelized 3D images). The accuracy of the inferred permeabilities is satisfactory but will vary as a function of the complexity of the pore structure evaluated.

There are many opportunities to further improve the software and apply it to other problems requiring a solution to Stokes equation. For example, it should be possible to further improve spatial accuracy schemes or even to rework FDM stencils by implementing diagonal flow through voxels. However, further research is needed to uncover if this is a valid approach because depending on the local pore geometry and the resolution of the image, pore geometry can vary significantly in natural porous media. Although additional efforts could be expended to increase the parallel efficiency, this is only effective when realistic initial optimization targets are provided. The authors consider the current performance a good return for the relatively small effort of selecting appropriate loops for parallelisation with OpenMP. If the software sees extensive use, we will consider scaling the existing code to larger problem sizes through the use of MPI or CUDA technologies. In terms of other applications, the velocity fields obtained using FDMSS can be utilized to compute dispersion (Mostaghmi et al., 2012; Nunes et al., 2015) and fluid stretching (Dentz et al., 2016). Our highly

memory-CPU optimized code can be easily modified to incorporate dynamics of the pore structure geometry to model reactive flow (Kang et al., 2006; Willingham et al., 2008) from mineral dissolution and precipitation (Godinho et al., 2016; Liu et al., 2017) or biofilm growth.

One possibility to improve pore-scale single fluid flow simulations is to improve the spatial discretization of 3D pore geometry images using stochastic image fusion (Gerke et al., 2015c) based on the stochastic reconstruction approach (as demonstrated in some of the case studies). According to our results (Figure 1), this will not only improve the accuracy of the velocity field and calculated permeability values, but also significantly accelerate convergence (SI.Fig.6). Such further developments are warranted because utilising larger images improves the likelihood to study a porous medium at its REV.

Five different case studies were presented to demonstrate the wide applicability of the FDMSS software: we computed permeabilities of sandstone, soil and artificial homogeneous ceramic samples based on their XMT scans and compared the inferences against laboratory measurements. We performed REV evaluations on sandstone, carbonate and ceramic samples to determine if current practical imaging volumes are large enough to provide realistic estimates of porous medium matrix properties. We provided a framework to validate stochastic reconstruction techniques using permeability calculations. Together with exhaustive descriptions of the numerical scheme and its accuracy, these examples demonstrated how FDMSS can be used in applied and fundamental research studies across different disciplines requiring an understanding of processes at the pore-scale.

Acknowledgements

We are grateful to Brendan Speet (CSIRO Land and Water) for his help with the computational resources. We also thank Prof. Vladimir Voevodin for providing computational resources of Moscow State University Computational Centre. Collaborative effort of the authors is within FaT iMP (Flow and Transport in Media with Pores) research group (www.porenetwork.com). K.M.G.,

R.V.V, M.V.K., D.V.K., T.O.S. were supported by Russian Science Foundation grant 14-17-00658 (X-ray microtomography), while K.M.G., M.V.K., T.O.S. were also supported by Russian Science Foundation grant 17-17-01310 (pore-scale simulations).

The software, its manual and all sphere packing 3D images are available at www.porenetwork.com/download/fdmss.

Author contributions: K.M.G. designed the study, software functionality and managed the development project; R.V.V. coded the solver and visualizer; M.V.K., T.O.S. and K.M.G. provided programming support and tested the software; D.C. performed code profiling and implemented OpenMP parallelization; K.M.G. and T.O.S. processed and segmented 3D images of porous media samples; K.D.V. performed all XMT scans and measured permeabilities in the lab; K.M.G and S.K. performed all FDM and LBM computations; K.M.G and S.K. analyzed the data; M.V.K., K.M.G. and S.K. prepared graphical materials; K.M.G. wrote the majority of the paper and the manual; S.K. and D.C. wrote some subsections of the text; D.M. and S.L. helped to interpret the results and edited the manuscript; all authors reviewed the final version of the paper.

Appendix A. Pore-scale modelling of single phase flow

This appendix briefly explains critical theoretical background for incompressible fluid flow under low Reynolds' number conditions and how to compute macroscopic properties (permeability) from velocity field solutions obtained using FDMSS.

A.1 Creeping flow and Stokes equation

The most basic equation of hydrodynamics is the Euler equation which describes the flow of an ideal fluid, i.e. without internal energy dissipation, such as by viscosity or heat transfer. By subsequently adding inner friction to the Euler equation and introducing the viscous stress tensor, and then combining it with the continuity equation results in the classical Navier-Stokes formulation for viscous incompressible flow (Landau and Lifshitz, 1987):

$$\begin{cases} \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} - \frac{\mu}{\rho} \Delta \mathbf{v} + \frac{\nabla p}{\rho} = 0 \\ \text{div } \mathbf{v} = 0, \end{cases} \quad (\text{A.1})$$

where $\mathbf{v} = (v_x, v_y, v_z)$ is velocity field, μ is fluid viscosity, ρ is fluid density, and p is pressure field. Reynolds number (Re) is assumed to be small:

$$Re = \frac{\rho v l}{\mu}, Re \ll 1, \quad (\text{A.2})$$

where v is absolute flow velocity along direction of interest, and l is the pore/channel characteristic size. Small Re values are considered to be true for most problems of flow in porous media (Bear, 1972), e.g., filtration in oil and gas reservoirs, aquifers, soils, porous filters and packed beds. By neglecting the convective term of $(\mathbf{v} \cdot \nabla) \mathbf{v}$ in Eq. A.1 as $(\mathbf{v} \cdot \nabla) \mathbf{v} \propto \frac{v^2}{l}$, while $\frac{\mu}{\rho} \Delta \mathbf{v} \propto \frac{\mu v}{\rho l^2}$

(Temam, 1977) and by also assuming steady-state flow, i.e. $\frac{\partial \mathbf{v}}{\partial t} = 0$, Eq. A.1 is rearranged to produce the Stokes equation:

$$\begin{cases} \mu \Delta \mathbf{v} - \nabla p = 0 \\ \text{div } \mathbf{v} = 0. \end{cases} \quad (\text{A.3})$$

Eq. A.3 is also called creeping or Stokes flow problem, and can be solved numerically to obtain a velocity field under prescribed pressure boundary conditions.

A.2 Obtaining permeability from a velocity field using Darcy's equation

If the velocity field and pressure boundary conditions are known, one can easily determine the macroscopic permeability according to Darcy's law:

$$K = \frac{\mu L Q}{\Delta p S}, \quad (\text{A.4})$$

where K is permeability, μ is dynamic viscosity, L is the distance for which a pressure difference Δp is applied, and Q is flow rate through the cross-sectional area S . The flow rate through a cross-sectional area can be determined according to:

$$Q = \langle v \rangle S_{eff}, \quad (A.5)$$

where $\langle v \rangle$ is the velocity in the porous space averaged over the whole cross-sectional area, and S_{eff} is the effective area equal to the cross-sectional area of pores (a part of the total cross section S). Note that for an incompressible fluid the flow rate should ideally be the same in all cross-sections. If boundary conditions for pressure are given as $\Delta p / L = \text{grad } p = 1$ (as in our case for a finite-difference solution) or the same body force is applied through the volume (as for the lattice-Boltzmann calculations), the permeability can be calculated as:

$$K = \frac{\mu \langle v \rangle S_{eff}}{S}. \quad (A.6)$$

In practice averaging over the volume V and (effective) porosity V_{pore} instead of over the areas S and S_{eff} is more accurate, as it smoothens any computational noise due to numerical fluctuations and slow convergence for complex geometries:

$$K = \frac{\mu \langle v \rangle V_{pore}}{V}. \quad (A.7)$$

Appendix B. Finite-difference scheme

To obtain the velocity field and permeability for a given pore structure one needs to solve Stokes equation (Eq. A.3). The main problem lies in decoupling velocity and pressure. To solve it, we applied an artificial compressibility method (Vladimirova et al., 1966; Chorin, 1967), essentially creating an assemblage of perturbed systems described by equations for a slightly compressible fluid. This was done by solving the unsteady Navier-Stokes equation (Eq. A.1) without the convective term and by assuming that the steady-state solution at $t \rightarrow \infty$ is the correct solution

satisfying Eq. A.3. In this way, time t has no physical meaning and represents a numerical solution strategy to resolve divergence constraints. Payret and Taylor (1983) called this approach the pseudo-unsteady method. It was shown by Fortin et al. (1971) that the exact solution of the unsteady Stokes problem converges to the corresponding steady-state problem.

More specifically, our approach includes finding a vector-function for flow velocity $\mathbf{v} : \Omega \rightarrow R^3$ and scalar function for pressure $p : \Omega \rightarrow R$ in a restricted pore volume $\Omega \subseteq [0, N]^3$ such that (the motion part B.1a was obtained from Eq. A.1 by multiplying it by ρ):

$$\rho \frac{\partial \mathbf{v}}{\partial t} - \mu \Delta \mathbf{v} + \nabla p = 0 \text{ in } \Omega, \quad (\text{B.1a})$$

$$\text{div } \mathbf{v} = 0 \text{ in } \Omega, \quad (\text{B.1b})$$

$$\mathbf{v}(\mathbf{x}, t) = \mathbf{0} \text{ on } \partial\Omega, \quad (\text{B.1c})$$

$$\mathbf{v}(\mathbf{x}, 0) = \mathbf{v}_0(\mathbf{x}). \quad (\text{B.1d})$$

Here the modelling domain Ω is a subset of the pore space within the cubic modelling domain $C \subset R^3$. This domain is divided into N^3 equal volumetric elementary cells – voxels. Each voxel has a cubic shape and linear width (voxel resolution) and is solely attributed to either the solid phase or porous space, while $\partial\Omega$ is the surface dividing these two subdomains. Domain Ω defined in this manner is now properly discretized for numerical calculations. On all boundary surfaces of the cube C , we either apply periodic boundary conditions or treat four surfaces parallel to the pressure gradient as solid walls. Flow is calculated only within pore voxels. To do so, pressures are fixed at the inlet and outlet faces to create a positive pressure gradient in the same direction for which the permeability is calculation. The third condition in Eq. B.1 provides the no-slip boundary conditions on all solid walls. An initial guess for velocity $\mathbf{v}_0$ is set for all fluid voxels (assuming saturated flow, i.e. all pore voxels are filled with fluid). After running sufficient iterative steps with recalculation of the pressure and velocity field, $\mathbf{v}(\mathbf{x}, t)$ converges to the solution satisfying Eq. B.1.

Spatial discretization of velocity $\mathbf{v}(\mathbf{x})$ and pressure $p(\mathbf{x})$ is performed on a staggered marker-and-cell (MAC) grid (Patankar, 1980), where pressure is defined in the middle of the cell and velocities on the cell edges (see SI.Fig.1 for details) using finite difference approximations of the 2nd and 4th order. The equation of state for a slightly compressible fluid, where fluid density ρ depends linearly on pressure p and a small proportionality constant $\varepsilon > 0$, is given as:

$$\rho = \rho_0 + \varepsilon p. \quad (\text{B.2})$$

If the following simplifying conditions are applied to Eq. B.2, i.e. the density $\rho_0 = 1$ under zero pressure conditions, and ε is $\ll 1$, Eq. B.2 becomes $\rho = \rho_0 = 1$. This leads to a significant simplification of the motion equation as $\rho \frac{\partial \mathbf{v}}{\partial t} = (\rho_0 + \varepsilon p) \frac{\partial \mathbf{v}}{\partial t} \rightarrow \frac{\partial \mathbf{v}}{\partial t}$, i.e. the density term can be simplified in Eq. B.1. Now consider the general form of the continuity equation for compressible flow with dependence on time t (Landau and Lifshitz, 1987):

$$\frac{\partial \rho}{\partial t} + \rho \operatorname{div} \mathbf{v} + \mathbf{v} \operatorname{grad} \rho = 0. \quad (\text{B.3})$$

Differentiating Eq. B.2 with time and reorganizing the terms keeping in mind that $\rho_0 = 1$ produces:

$$\frac{\partial \rho}{\partial t} = \varepsilon \frac{\partial p}{\partial t}, \quad (\text{B.4a})$$

$$\operatorname{grad} \rho = \frac{\partial \rho}{\partial x} \mathbf{i} + \frac{\partial \rho}{\partial y} \mathbf{j} + \frac{\partial \rho}{\partial z} \mathbf{k} = \varepsilon \left(\frac{\partial p}{\partial x} \mathbf{i} + \frac{\partial p}{\partial y} \mathbf{j} + \frac{\partial p}{\partial z} \mathbf{k} \right) = \varepsilon \operatorname{grad} p, \text{ thus} \quad (\text{B.4b})$$

$$\mathbf{v} \operatorname{grad} \rho = \varepsilon (\mathbf{v} \operatorname{grad} p). \quad (\text{B.4c})$$

Finally by substituting expressions from Equations B.4a and B.4c into B.3, the latter becomes:

$$\varepsilon \frac{\partial p}{\partial t} + \rho \operatorname{div} \mathbf{v} + \varepsilon (\mathbf{v} \operatorname{grad} p) = 0. \quad (\text{B.5})$$

As ε is $\ll 1$ and $(\rho_0 + \varepsilon p) \frac{\partial \mathbf{v}}{\partial t} \rightarrow \frac{\partial \mathbf{v}}{\partial t}$, the term $\varepsilon (\mathbf{v} \operatorname{grad} p) \rightarrow 0$ and $\rho \operatorname{div} \mathbf{v} \rightarrow \operatorname{div} \mathbf{v}$. Therefore, our calculations are based on the system of equations of the following form:

$$\begin{cases} \frac{\partial \mathbf{v}}{\partial t} - \mu \Delta \mathbf{v} + \text{grad } p = 0 \\ \varepsilon \frac{\partial p}{\partial t} + \text{div } \mathbf{v} = 0. \end{cases} \quad (\text{B.6})$$

The solution method implies sequential recalculation of the velocity and pressure fields at each time step. The equation of motion in Eq. B.6 can be rewritten as an assemblage of three equations, each describing the flow along corresponding axes x , y and z :

$$\begin{cases} \frac{\partial v_x}{\partial t} - \mu \Delta v_x + \frac{\partial p}{\partial x} = 0 \\ \frac{\partial v_y}{\partial t} - \mu \Delta v_y + \frac{\partial p}{\partial y} = 0 \\ \frac{\partial v_z}{\partial t} - \mu \Delta v_z + \frac{\partial p}{\partial z} = 0, \end{cases} \quad (\text{B.7})$$

where v_x, v_y, v_z are velocity vector components and x, y, z the spatial Cartesian coordinates.

Recalculation of each component of the velocity field is performed independently. Rewriting Eq.

B.7 in finite difference form gives:

$$\frac{\partial v_i}{\partial t}(t_0) \approx \frac{v_i(t_0 + \delta t) - v_i(t_0)}{\delta t}, \quad (\text{B.8})$$

$$\frac{\partial p}{\partial t}(t_0) \approx \frac{p(t_0 + \delta t) - p(t_0)}{\delta t}, \quad (\text{B.9})$$

$$\frac{\partial p}{\partial x}(x_0) \approx \frac{p(x_0 + \delta x) - p(x_0)}{\delta x}, \text{ similarly for } \frac{\partial p}{\partial y} \text{ and } \frac{\partial p}{\partial z}. \quad (\text{B.10})$$

The same finite difference from applies to the continuum equation in B.4:

$$\begin{aligned} \text{div } \mathbf{v}(\mathbf{x}_0) &= \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} \approx \\ &\approx \frac{v_x(x_0 + \delta x) - v_x(x_0)}{\delta x} + \frac{v_y(y_0 + \delta y) - v_y(y_0)}{\delta y} + \frac{v_z(z_0 + \delta z) - v_z(z_0)}{\delta z}. \end{aligned} \quad (\text{B.11})$$

Here, $\delta x, \delta y, \delta z$ are infinitesimal increments in the spatial coordinates and v_i represents the velocity

field components, where $i = x, y, z$. Computing the term $\mu \Delta v$ requires some additional effort

because of the need to independently calculate three values $\mu\Delta v_x$, $\mu\Delta v_y$ and $\mu\Delta v_z$. As an example, let us write Δv_x in finite differences by first considering the Laplace operator:

$$\Delta v_x = \frac{\partial^2 v_x}{\partial x^2} + \frac{\partial^2 v_x}{\partial y^2} + \frac{\partial^2 v_x}{\partial z^2} \quad (\text{B.12})$$

The velocity derivative along the x axis will be similar for all voxels independent of the local calculation geometry: $\frac{\partial^2 v_x}{\partial x^2} \approx \frac{v_x(x_0 - \delta x) - 2v_x(x_0) + v_x(x_0 + \delta x)}{2(\delta x)^2}$ for a 2nd order accuracy and

$$\frac{\partial^2 v_x}{\partial x^2} \approx \frac{-v_x(x_0 - 2\delta x) + 16v_x(x_0 - \delta x) - 30v_x(x_0) + 16v_x(x_0 + \delta x) - v_x(x_0 + 2\delta x)}{12(\delta x)^2} \quad \text{for a 4th order}$$

accuracy. Derivatives along directions perpendicular to the main flow direction (as defined by the applied pressure gradient) will significantly affect energy dissipation. Calculations for these directions are therefore performed considering local geometry of Ω , by tracing the locations of the nearest solid voxels representing $\partial\Omega$. For a 2nd order accuracy scheme, the vicinity of two voxels is considered, while for a 4th order scheme this is four voxels. In both perpendicular directions y and z, respectively 6 and 15 cases are possible for computing the derivative as shown on SI.Fig.2.

For example, consider case number 12 on SI.Fig.2a. As $\delta x = \delta y = \delta z = 1$ for the MAC cells, we need to rewrite derivative $\frac{\partial^2 v_x}{\partial y^2}$ in finite differences form in the vicinity of four voxels (all relevant velocities are shown in SI.Fig.2c). All velocities, except v_2 at point A, will be expanded in a Taylor series to the 4th order, thereby neglecting all higher order terms:

$$v_1 = v_2 - \frac{1}{2} \frac{\partial v_2}{\partial y} + \frac{1}{8} \frac{\partial^2 v_2}{\partial y^2} - \frac{1}{48} \frac{\partial^3 v_2}{\partial y^3} + \frac{1}{384} \frac{\partial^4 v_2}{\partial y^4},$$

$$v_3 = v_2 + \frac{\partial v_2}{\partial y} + \frac{1}{2} \frac{\partial^2 v_2}{\partial y^2} + \frac{1}{6} \frac{\partial^3 v_2}{\partial y^3} + \frac{1}{24} \frac{\partial^4 v_2}{\partial y^4},$$

$$v_4 = v_2 + 2 \frac{\partial v_2}{\partial y} + 2 \frac{\partial^2 v_2}{\partial y^2} + \frac{4}{3} \frac{\partial^3 v_2}{\partial y^3} + \frac{2}{3} \frac{\partial^4 v_2}{\partial y^4}, \quad (\text{B.13})$$

$$v_5 = v_2 + 3 \frac{\partial v_2}{\partial y} + \frac{9}{2} \frac{\partial^2 v_2}{\partial y^2} + \frac{9}{6} \frac{\partial^3 v_2}{\partial y^3} + \frac{27}{8} \frac{\partial^4 v_2}{\partial y^4},$$

$$v_6 = v_2 + 4 \frac{\partial v_2}{\partial y} + 8 \frac{\partial^2 v_2}{\partial y^2} + \frac{32}{3} \frac{\partial^3 v_2}{\partial y^3} + \frac{32}{3} \frac{\partial^4 v_2}{\partial y^4}.$$

According to the boundary conditions in Eq. B.1, $v_1 = 0$, the system of five equations in B.13 are solved for the unknown derivatives:

$$\frac{\partial^2 v_2}{\partial y^2} = -\frac{65}{12} v_2 + \frac{22}{9} v_3 - \frac{1}{2} v_4 + \frac{2}{21} v_5 - \frac{1}{108} v_6, \quad (\text{B.14})$$

which provides the derivative $\frac{\partial^2 v_x}{\partial y^2}$ in a finite difference form:

$$\frac{\partial^2 v_x(x_0)}{\partial y^2} \approx -\frac{65}{12} v_x(x_0) + \frac{22}{9} v_x(x_0 + \delta y) - \frac{1}{2} v_x(x_0 + 2\delta y) + \frac{2}{21} v_x(x_0 + 3\delta y) - \frac{1}{108} v_x(x_0 + 4\delta y) \quad (\text{B.15})$$

All other derivatives/directions are performed in a similar manner. After obtaining all equations in finite difference form we can write the numerical scheme:

$$\begin{aligned} \tilde{v}_x &= v_x + \left(\mu \left(\frac{\partial^2 v_x}{\partial x^2} + \frac{\partial^2 v_x}{\partial y^2} + \frac{\partial^2 v_x}{\partial z^2} \right) - \frac{p(x_0 + \delta x) - p(x_0)}{\delta x} \right) \tau, \\ \tilde{v}_y &= v_y + \left(\mu \left(\frac{\partial^2 v_y}{\partial x^2} + \frac{\partial^2 v_y}{\partial y^2} + \frac{\partial^2 v_y}{\partial z^2} \right) - \frac{p(y_0 + \delta y) - p(y_0)}{\delta y} \right) \tau, \\ \tilde{v}_z &= v_z + \left(\mu \left(\frac{\partial^2 v_z}{\partial x^2} + \frac{\partial^2 v_z}{\partial y^2} + \frac{\partial^2 v_z}{\partial z^2} \right) - \frac{p(z_0 + \delta z) - p(z_0)}{\delta z} \right) \tau, \\ \tilde{p} &= p - \frac{1}{\varepsilon} \left(\frac{v_x(x_0 + \delta x) - v_x(x_0)}{\delta x} + \frac{v_y(y_0 + \delta y) - v_y(y_0)}{\delta y} + \frac{v_z(z_0 + \delta z) - v_z(z_0)}{\delta z} \right) \tau, \end{aligned} \quad (\text{B.16})$$

where $\tilde{v}_x, \tilde{v}_y, \tilde{v}_z, \tilde{p}$ are velocity component and pressure values at the next time step τ . A numerical scheme for a 2nd order accuracy is simpler and can be easily obtained in a similar manner.

So far our focus has been on improving the accuracy of the spatial scheme, while little effort has devoted to improving the temporal scheme. This approach is justified when a pseudo-unsteady numerical scheme is used. A detailed comparison of the flow velocity fields revealed some notable differences between the 2nd and 4th order accuracies, especially closer to the pore walls (Vasilyev et al., 2016). While not immediately relevant to permeability calculations, a higher order spatial accuracy scheme may have significant impact for problems where local velocity field differences matter, e.g., in fluid stretching calculations (Dentz et al., 2016) or dispersion modelling (Mostaghimi et al., 2012). The difference between accuracy schemes is explored in SI.Section 2.

Appendix C. Software structure and usage

C.1 Solver's structure and error criterion

The main algorithm consists of two main stages – pre-processing of the 3D pore geometry followed by flow modelling. Pre-processing is necessary to establish all necessary relationships within each MAC cell and to optimize ulterior calculations. The resulting data architecture contains the information on which lattice points require pressure and orthogonal velocity recalculations. Pressure recalculations are only performed in pore (fluid) voxels and, thus, all solid voxels are omitted. The local formula used for recalculations depends on the form of the derivative of the velocity component, which in turn depends on the number of fluid voxels in two orthogonal directions. For applying periodic boundary conditions, the size of the cubic modelling domain C is enlarged on cube sides with the number of voxel layers equal to the accuracy scheme (e.g., for an input geometry of 300^3 voxels the edge of the actual modeling cube will be 308^3 voxels with a 4th accuracy scheme). The geometry of these layers is simply copied onto the opposite face.

All calculations start by assigning pressure values at each cell according to a linear relationship with a gradient $\Delta p / L = \text{grad } p = 1 \text{ kg}/(\text{vx}^2 \times \text{s}^2)$ across the entire domain (i.e., pressure at the outlet equals 1, while at the inlet it is equal to the width of the modelling domain L plus 1), where vx is the voxel size. The pressure gradient is always prescribed along one direction, typically the x -

direction. Next, all y and z components of the velocity field are set to zero, while the x -component is prescribed with some guess values (input parameters are described later). After this initialization, periodic or no flow boundaries are applied followed by the start of the iteration process. For each iteration the following sequence is performed: 1) application of boundary conditions, 2) recalculation of pressure field, and 3) recalculation of velocity field. Recalculation of the velocity components can be performed independently from one another as the data structure is organized in such a fashion that v_x , v_y and v_z are independent.

Iterations are stopped if either a threshold value or a prescribed solution accuracy is reached. Both looping exit conditions are input parameters. Accuracy of the numerical scheme is evaluated by calculating the error in the Stokes equation (Eq. A.3); this error is minimized to ensure convergence of both velocity and pressure fields. For the continuity equation, the error E_C is assessed as an anomalous flux for each fluid voxel:

$$E_C = |x_{in} - x_{out} + y_{in} - y_{out} + z_{in} - z_{out}|, \quad (C.1)$$

where the *in* and *out* subscripts for the velocity components x , y and z refer to inflow and outflow for a given MAC cell. For the motion equation, the error E_M (with their components E_{Mx} , E_{My} and E_{Mz}) is calculated for each velocity component:

$$\begin{aligned} E_{Mx} &= |FD(\Delta x) \mu - \Delta p_x|, \\ E_{My} &= |FD(\Delta y) \mu - \Delta p_y|, \\ E_{Mz} &= |FD(\Delta z) \mu - \Delta p_z|. \end{aligned} \quad (C.2)$$

Here the FD notation refers to a finite difference representation, which is different for each fluid voxel and depends on the local pore geometry. For example, when the x component is calculated on the boundary of two voxels perpendicular to the x axis, the pressure difference across this boundary is $\Delta p_x = p_2 - p_1$, where p_2 and p_1 are pressure values at opposite sides of this boundary where p_2 has the largest x coordinate. A similar finite difference scheme can be written for Δp_y and Δp_z . For

the purpose of total error normalization, the sum of all velocities used for the calculation of the normalized error, E , is used as the denominator:

$$E = \frac{\sum_{a.p.v.}(E_c)}{\sum_{a.p.v.}(F)} + \frac{\sum_{a.p.v.}(E_{Mx})}{\sum_{a.p.v.}|\Delta p_x|} + \frac{\sum_{a.p.v.}(E_{My})}{\sum_{a.p.v.}|\Delta p_y|} + \frac{\sum_{a.p.v.}(E_{Mz})}{\sum_{a.p.v.}|\Delta p_z|}, \quad (C.3)$$

where F is the average within-voxel velocity calculated as in Eq. C.4 and a.p.v. stands for all pore voxels:

$$F = \frac{|x_{in}|+|x_{out}|}{2} + \frac{|y_{in}|+|y_{out}|}{2} + \frac{|z_{in}|+|z_{out}|}{2}. \quad (C.4)$$

The need to introduce such a criterion different from the one described by Mostaghimi et al. (2013), see Eq. C.5, is justified in SI.Section 1. There are two ways to execute the numerical solution scheme – either via the GUI pre- and post-processing program, or as executable with input parameters defined in xml files (which is efficient for batch simulations).

C.2 Input data and parameters

Because the above-defined solution is valid only for low Reynolds numbers, the following parameters are fixed to ensure $Re \ll 1$: the pressure gradient across the entire domain is always $1 \text{ kg}/(\text{vx}^2 \times \text{s}^2)$ in the direction of the flow modelling, $\mu=100 \text{ kg}/(\text{vx} \times \text{s})$ and $\varepsilon = 1.5 \times 10^6$. The compressibility parameter ε significantly affects convergence time, although based on our extensive computations the computed permeabilities showed no dependence of ε after reaching steady-state (results not shown). The fixed value of $\varepsilon = 1.5 \times 10^6$ suggested by Bentz and Martys (2007) was found to be a compromise between convergence speed and numerical accuracy.

Adjustable parameters include the time step (τ in s), initial velocity guess ($\mu\text{m/s}$), desired numerical accuracy according to one of three criteria (see SI.Section 1 for a detailed description of all three criteria implemented in FDMSS), voxel resolution (μm ; can be fixed to 1 if a dimensionless permeability in voxels is needed), number of iterations after which the output will be produced (intermediate results are saved into a file which includes permeability values in $\text{mD}/\mu\text{m}^2$ and the current accuracy E), number of consecutive iteration steps before the iteration is stopped

and final results are stored (if the prescribed error accuracy E has not been reached), boundary conditions (periodic or no flow on modelling domain C faces parallel to main flow direction), and number of threads (depending on available computational resources) to be used during computations.

All parameters can be changed in the pre-processing GUI program or via a XML based configuration file config.xml (SI.Fig.3) when executables are run. Examples of the GUI program in defining the problem and running the solver are provided in the Manual (Supplementary Information).

The main input data to calculate permeability and simulate velocity fields is the 3D pore geometry, which can be extracted from binarized (segmented into two phases, i.e. pores and solids) stacks of greyscale images. The GUI program can import and analyze such binarized stack of images. X-ray micro-tomography (XMT) or other types of 3D imaging methods provide stacks of grey scale images. In order to obtain 3D pore structures we need to binarize them, i.e., divide them into pore and solid voxels. This can be done using a multitude of segmentation techniques, among which indicator kriging (Oh and Lindquist, 1999; Houston et al., 2013), gradient masks (Schlüter et al., 2013) and Markov random field (Kulkarni et al., 2012) are probably the most popular. None of these methods was proven to be universal and many researchers still use binarization by manual threshold selection (Baveye et al., 2010; Gerke et al., 2017). Regardless of the segmentation method, the user needs to provide a stack of cubic size (i.e., $n_x=n_y=n_z$, where n is the width of the image) made of *.bmp images with pores represented by 0 and solids by 255 (for example, this can be easily done in open-source software ImageJ/Fiji (Schneider et al., 2012)). Using the Converter option in the GUI visualizer provides three options to save the resulting 3D geometry .raw file (image3d) based on a single stack of images (see SI.Fig.4 and the Manual), so that flow can be calculated in different directions.

C.3 Visualization and post-processing

Independent of the numerical solution method, the modelling results such as permeability (according to Eq. A.7 and expressed in mD and μm^2) and average flow velocity (expressed in $\mu\text{m/s}$) and its accuracy (according to the chosen criterion, see SI Section 1 for detailed description) will be saved in the output.xml file and interim results log, and reported in a separate window when the GUI is used to run the model. In addition to the aforementioned parameters, the solver will store pressure and velocity fields in binary files for visualization and further processing. This includes one file for pressure and four files for velocities (one for each principal direction plus one for the absolute modulus values in the fullvels file). While in the FDMSS structures MAC grid pressures are calculated within voxels, velocities are defined on voxel edges. Thus, for visualization and interpretation purposes, velocities are recalculated as averages at two opposite edges in the direction of interest (i.e., for a given velocity component, as shown in SI.Fig.5), giving a single value for each voxel for ease of representation. More advanced approaches to create streamlines useful for pore-scale dispersion modelling can also be developed for this purpose (Mostaghimi et al., 2012; Nunes et al., 2015). In addition, average velocities in all slices perpendicular to the pressure gradient are reported separately in the compvels files. Velocity and pressure binaries can be visualized using our GUI software or other free packages such as ParaView. Examples of structure and velocity/pressure field visualizations are shown in the Manual and in the case study section below. Note that velocities can be represented as absolute or signed values to highlight local flow direction.

In order to avoid any confusion with axes and directions, we used lower case x , y and z in equations and figures devoted to Stokes' equation while for the numerical scheme i , j and k are adopted for the internal coordinate system as part of the visualization (as show in SI.Fig.4). The pressure gradient is always applied along the i -direction. Finally, X , Y and Z are used to represent flow directions within porous medium samples. The latter coordinate system is unique and defined for any stack of images.

C.4 Parallelization efficiency

Improving computational efficiency was the main objective for developing FDMSS and this was primarily achieved by parallelization. Hot path profiling was carried out with the Visual Studio 10 profiling tools, which showed that just three functions accounted for 99.5% of CPU time. These three functions were responsible for updating velocities (83%), updating pressures (16.5%), and application of periodic boundary conditions (13.3%). All three functions contained multiple loops over the three spatial dimensions, and appeared to be good candidates for parallelism with OpenMP parallel for directives. Various scheduling options were tested, with static scheduling of threads giving the best results due to the reduced scheduling overheads and the constant workload for each loop iteration. The same parallelized code was ported to Linux.

Using FDMSS compiled for both the Windows and Linux platforms, a series of tests were conducted to assess speedup, parallel efficiency and the relative performance between the Windows and Linux targets. The CSIRO Bragg cluster was used for all tests, as nodes with identical hardware specifications are available running with either Linux or Windows operating systems. Each node contained dual Intel Xeon E5-2650 CPUs, with 128MB RAM, providing 16 cores per node. A single dataset of 8 million voxels was tested with 1, 2, 4, 8, and 16 threads (which is the number of cores on a Bragg node). The solver was hardcoded to conduct exactly 8000 iterations before exiting to avoid early convergence distorting the results. For each combination of platform (Linux, Windows) and thread count (1, 2, 4, 8, 16), the test was conducted three times with the average run-time being used. The Windows code was compiled with Microsoft Visual Studio 10, using the -O2 optimization defaults. The Linux code was compiled with the Intel C++ 14.1 compiler, again using -O2 optimization defaults. The single-threaded code was generated with the -openmp-stubs compiler option (and its equivalent in Visual Studio).

The Windows code was consistently slower than the Linux port by around 40% (Fig.C.1a). Due to the identical test hardware, code, and data, this effect is attributed to differences in compiler optimization. However, tests using the Intel compiler for Windows could not be conducted. The

speedup plot indicates the improvement in execution time obtained by increased thread counts relative to the single-threaded execution time (Fig.C.1a). The ideal linear speedup is also shown (straight line). Both compilers and platforms generated a code with nearly identical speedup results, despite the fact that the Microsoft compiled code was consistently slower overall. This is to be expected because the proportion of serial to parallel code is the same on both platforms and this ratio is the primary driver of speedup and parallel efficiency. The strong scaling efficiency plot quantifies the reduction in efficiency as the thread-count is increased (Fig.C.1b). At 8 threads, efficiency is approximately 70% of ideal (speedup of $\sim 6x$), while doubling the thread count (and thus the hardware resources) achieves just an additional 2x speedup, with efficiency dropping to $\sim 50\%$.

Fig.C.1. Parallelization efficiency for 4th order accuracy scheme under different platforms: a) the speedup plot indicating the improvements in execution time obtained by increased thread counts relative to the single-threaded execution time, and the strong scaling efficiency plot quantifying the reduction in efficiency as the thread-count is increased; b) the wall time plot shows total runtime in seconds under Windows and Linux systems.

To study the overall computational performance of our solver, we compared it to another published FDM solver (Mostaghimi et al., 2013). In Mostaghimi et al. (2013) the sandpack sample LV60 was used as a test case for single-CPU computations. Two LV60 subsamples are available online (<http://www3.imperial.ac.uk/earthscienceandengineering/research/perm/porescalemodelling/micro-ct%20images%20and%20networks>, accessibility verified on 16.01.2015), both with size of 450^3 voxels. We used these 3D images and their smaller size sub-croppings from the center to estimate the time needed to converge to the same accuracy using a single-threaded solution with our solver. To ensure that the same accuracies are reached we implemented similar convergence criteria in the form of (see Eq. 10 in Mostaghimi et al. (2013)):

$$E_{CC} = \frac{\sum_{all\ pore\ voxels} (b_p)}{QN}, \quad (C.5)$$

where b_p is the continuity imbalance in each fluid voxel, Q is the total flux in the direction of the pressure gradient, and N is the number of voxels in this direction. This accuracy measure showed significant fluctuations (see SI.Fig.6 and explanations in SI.Section 1) and, thus, our stopping criterion in the FDMSS software are based on Eq. C.3 while Eq. C.5 is used only for validation against previously published studies. In Mostaghimi et al. (2013), the maximum sample volume was larger than 450^3 voxels; no specifications were given if the LV60A or C subvolumes were used. We therefore performed simulations on both original samples. Zero-flow (solid wall) boundary conditions on all four facies parallel to the main direction of flow were used. To evaluate if the computational effort depended on both the domain size and its pore geometry and to assess the effect of larger volumes, we also evaluated an additional 700^3 voxels sandstone sample in this analysis. Differences between convergence criteria expressed by Eq. C.3 and Eq. C.5 will be discussed in the following section. All calculations reported in this subsection were performed on a Windows workstation with Intel Xeon 2.27 GHz CPU using a single thread and the 4th order accuracy scheme solver.

All subsamples with volumes of 50^3 - 700^3 voxels converged to $E_{CC} < 10^{-4}$ (Table C.1). Because a larger porosity means a larger amount of voxels taking part in computations, the geometry of the pore space also influenced the computational time, as was observed for the Sandstone sample. Use of the sample with a smaller porosity and finer pores in this sample results in a longer convergence time and a larger imbalance. Finally, in comparing computational performance against the code of Mostaghimi et al. (2013), the much faster convergence of FDMSS becomes apparent (Fig.C.2). Note, that FDMSS utilized a 4th order spatial accuracy scheme; in case a 2nd order accuracy scheme was used, reported convergence times would be 20-30% faster (see detailed analysis and comparison between two schemes in SI, SI.Fig.8 and SI.Section 2). These differences could be due in part to the differences in CPU performance used in both studies; however, differences in run time of an order of magnitude are less likely dominated by differences in CPU. An overall similarity in pore structure between LV60A and LV60C resulted in a similar convergence time.

Overall, FDMSS does not underperform other recently published codes. Moreover, the memory requirements for FDMSS are lower than that of Mostaghimi et al. (2013). For example, FDMSS required approximately 1.2 and 16 Gb for 300^3 and 700^3 voxels image of the Sandstone sample, respectively, whereas Mostaghimi et al. (2013) required 3.5 and 40 Gb for the same images, respectively. RAM estimates are given for the samples studied here and may be slightly different for other samples with higher porosity values.

Mostaghimi et al. (2013) quoted the work of Tamamidis et al. (1996) in which artificial compressibility and pressure-based (SIMPLE) schemes for solving the Stokes equation are compared. Tamamidis et al. (1996) conclude that the pressure-based method requires 16 times less memory than the artificial compressibility approach. They further argue that the faster pressure-based computations were specific to vector machines and that an artificial compressibility algorithm converges much faster on scalar machines. Most likely, this is the main reason why our computations are much faster than that of Mostaghimi et al. (2013).

Table C.1. Subsample data and convergence times for FDMSS.

Sample size, voxels	50 ³	100 ³	150 ³	200 ³	250 ³	300 ³	350 ³	400 ³	450 ³	700 ³
Sand pack LV60A										
Porosity	31.54	35.24	35.86	36.31	36.64	36.88	36.67	36.43	36.31	N/a*
Time, s	6	99	468	540	3373	6992	15169	32424	49822	N/a
E _{CC}	8.69E-05	9.60E-05	9.99E-05	9.94E-05	9.95E-05	9.73E-05	9.94E-05	9.91E-05	9.98E-05	N/a
Sand pack LV60C										
Porosity	36.9	36.84	37.77	38.1	37.81	37.49	37.42	37.2	36.84	N/a
Time, s	4	41	741	1713	3520	6760	14066	43681	58877	N/a
E _{CC}	9.68E-05	9.80E-05	9.82E-05	9.44E-05	9.73E-05	9.87E-05	9.78E-05	9.82E-05	9.82E-05	N/a
Sandstone										
Porosity	17.53	19.04	19.42	19.19	19.57	19.69	N/a	N/a	20.16	20.33
Time, s	43	799	1145	6548	10873	12726	N/a	N/a	67694	306436
E _{CC}	9.36E-05	8.98E-05	9.69E-05	9.06E-05	9.86E-05	9.61E-05	N/a	N/a	9.95E-05	9.75E-05

* not calculated due to the smaller size of the sample

Fig.C.2. CPU time to converge to $E_{CC} < 10^{-4}$ using FDMSS for samples LV60A, LV60C and Sandstone, and digitized CPU values for LV60 sample reported by Mostaghimi et al. (2013).

Appendix D. Sphere packings (analytical solutions) and LBM simulations

Regular (ordered) sphere packings can be seen as an optimal geometry for the validation of pore-scale simulations. On the one hand, they are more complex than simple, well studied analytical geometries (like open channel), on the other hand, regular packings are still simple enough to have several semi-analytical solutions available with different independent approaches (Zick and Homsy, 1982; Sangani and Acrivos, 1982; Larson and Higdon, 1989). While discretization of a sphere packing may appear to be a straightforward procedure, its naive implementation may introduce significant discrete scatter in simulated transport coefficients, especially for the case of regular packings and low discretization resolutions. This is mainly due to the low amount of fluid voxels sampling the void spaces between spheres. A possible way to increase the amount of fluid voxels (and to reduce data scatter) is to replicate unit cells containing an ordered packing by using the periodicity property, followed by discretizing the resulting geometry and giving special attention to

the selection of the integral dimensions for the output grid. As shown by Khirevich et al. (2015), such an approach dramatically reduces data scatter and allows studying the performance of the numerical scheme itself without artifacts of a discrete geometry. We note that the aforementioned "discrete noise" is mainly pronounced in one-unit-cell geometries (like regular packings), while random packings of sufficient size are virtually free from this artifact (Khirevich et al., 2015).

In this study we used three different types of sphere packings: simple cubic (SC), body-centered cubic (BCC) and face-centered cubic (FCC). For SC we utilize only the minimum porosity realization (for the case of the non-overlapping spheres), while for both BCC and FCC minimum porosity packings were supplemented with two more loose packs of higher porosity. For each packing type and porosity we created up to 15 3D images with a range of prescribed discretization resolutions according to Khirevich et al. (2015) (see examples of visualized sphere packings on Fig. D.1). The detailed information on all sphere packings used for numerical experiments is shown in Table D.1. Permeability can be converted into drag force and vice versa according to:

$$F_d = \frac{d_{sp}^2}{18(1 - \phi)k}, \quad (30)$$

where F_d is the drag force, d_{sp} is sphere diameter in voxel units, ϕ is porosity, and k is the permeability in voxel units squared.

Table D.1. Sphere packs volumes, porosities and discretizations

Array	Porosity (ϕ)	Analytical drag force (F_d)	Discretization (sphere diameter in voxels (d_{sp})/sample's volume in voxels)
SC	$1-\pi/6 \approx$	42.1	$4.76/100^3$; $5.88/100^3$; $7.69/100^3$; $8.36/301^3$; $9.09/100^3$; $12.04/301^3$;
(touching spheres)	0.476		$13.69/301^3$; $14.28/100^3$; $17.71/301^3$; $23.16/301^3$; $27.37/301^3$; $33.34/100^3$; $37.63/301^3$; $60.22/301^3$; $100.36/301^3$

BCC	1-	162.8	4.12/100 ³ ; 5.09/100 ³ ; 6.66/100 ³ ; 7.24/301 ³ ; 7.87/100 ³ ; 10.43/301 ³ ;
(touching spheres)	$\sqrt{3}\pi/8 \approx 0.32$		11.85/301 ³ ; 12.38/100 ³ ; 15.33/301 ³ ; 20.05/301 ³ ; 23.7/301 ³ ; 28.87/100 ³ ; 32.58/301 ³ ; 52.13/301 ³ ; 86.88/301 ³
BCC	0.366	114.4	4.03/100 ³ ; 4.98/100 ³ ; 6.5/100 ³ ; 7.07/301 ³ ; 7.69/100 ³ ; 10.19/301 ³ ;
(loose packing)			11.57/301 ³ ; 12.09/100 ³ ; 14.98/301 ³ ; 19.59/301 ³ ; 23.15/301 ³ ; 28.2/100 ³ ; 31.83/301 ³ ; 50.93/301 ³ ; 84.88/301 ³
FCC	1-	432	3.7/100 ³ ; 4.16/100 ³ ; 5.4/100 ³ ; 5.91/301 ³ ; 6.43/100 ³ ; 8.51/301 ³ ; 9.67/301 ³ ;
(touching spheres)	$\sqrt{2}\pi/6 \approx 0.26$		10.1/100 ³ ; 12.52/301 ³ ; 16.37/301 ³ ; 19.34/301 ³ ; 23.57/100 ³ ; 26.6/301 ³ ; 42.56/301 ³ ; 70.93/301 ³
FCC	0.366	146.5	3.2/100 ³ ; 3.95/100 ³ ; 5.17/100 ³ ; 5.61/301 ³ ; 6.1/100 ³ ; 8.08/301 ³ ; 9.19/301 ³ ;
(loose packing)			9.59/100 ³ ; 11.89/301 ³ ; 15.55/301 ³ ; 18.37/301 ³ ; 22.38/100 ³ ; 25.26/301 ³ ; 40.42/301 ³ ; 67.37/301 ³

We employed the Lattice-Boltzmann method (LBM) as a complement to the FDM approach for calculation of flow in regular packings. LBM simulates the evolution of the lattice gas comprised of distribution functions of gas molecule fluxes along a prescribed discrete set of lattice links (19 in this study) connecting each lattice node with its neighbors. Each LBM iteration included two steps: i) movement of fluxes between lattice nodes along lattice links (streaming step), and ii) collision of fluxes at each lattice node (collision step). The LBM is constructed in such a way that with the course of time the distribution function (of fluxes) iteratively approaches its equilibrium. Simulations are stopped when the variation of the distribution function becomes sufficiently small or, in other words, when equilibrium is reached up to a given accuracy. While the streaming step is universal for various LBM models, the implementation of the collision differs among them.

A simple and commonly used version of a collision operator is the so-called "BGK" (Bhatnagar et al., 1954), which contains only one relaxation time τ defining the decay rate of the distribution function towards its equilibrium as well as viscosity of the simulated fluid. When using a bounce-back boundary condition (a standard approach for simulations based on XMT images as input geometry), a well-known drawback of BGK LBM is the dependency of the simulated permeabilities on τ . Further developments of LBM models resulted in collision operators with collisions occurring

in the space of hydrodynamic and kinetic moments (like density, momentum, energy, and energy flux), with corresponding conversion of the distribution function to/from the momentum space. Such a formulation of the collision operator allows introduction of multiple (up to the amount of links per lattice node, or 19 in this study, $\tau_{0..18}$) relaxation times — so called MRT collision operator — for each individual mode, which also poses a question on a particular selection of those parameters remaining unanswered till now. However, it is suggested that for Stokes simulations, individual adjustment of all MRT relaxation rates is not necessary. Especially in the case of proper grouping relaxation rates, only two prescribed values for $\tau_{v,f}$ are sufficient to provide viscosity independent simulations of permeability. In this case the multiple-relaxation-times collision operator is reduced to two-relaxation-times (TRT) where, from the point of simulations, τ_v controls viscosity of the simulated fluid while τ_f remains an adjustable parameter. It was shown (Ginzburg and Adler, 1994) that when the combination of these two relaxation rates (also known as 'magic' parameter'), $\Lambda = (\tau_v - 0.5) (\tau_f - 0.5)$ stays fixed, LBM provides viscosity-independent permeability up to machine accuracy in any geometry. Mathematical analysis of LBM behavior for Poiseuille flow previously revealed that Λ controls the location of a zero-velocity boundary between adjacent solid and fluid voxels, and there are several specific Λ values for this simple geometry. Namely, $\Lambda = 3/16$ provides correct location for the horizontally-oriented open channel while $\Lambda = 3/8$ — for the diagonal one, and $\Lambda = 1/8$ gives accurate (up to machine accuracy) average flow velocity. Analysis of more complex geometries still remains an open research topic (Khirevich et al., 2015), and therefore in this study we took the minimum and maximum values mentioned above ($\Lambda = 1/8, 3/16$ and $3/8$) to get an idea about the possible variation of LBM results with Λ .

Fig.D.1. Examples of studied 3D geometries: FCC sphere packs (porosity = 0.366) with different discretization (values are given under each pack) for six samples of a) 100^3 and b) 301^3 voxels in volume, and c) real samples studied with XMT: sandstones S1, S3, carbonate C1, C2 and C71.

References

1. Adler, P.M., Jacquin, C.G., Quiblier, J.A., 1990. Flow in simulated porous media. *Int. J. Multiph. Flow* 16, 691–712. doi:10.1016/0301-9322(90)90025-E
2. Akanji, L. T., Matthai, S. K., 2010. Finite element-based characterization of pore-scale geometry and its impact on fluid flow. *Transp. Porous Media* 81(2), 241-259. doi: 10.1007/s11242-009-9400-7

3. Arns, C.H., Knackstedt, M.A., Mecke, K.R., 2003. Reconstructing complex materials via effective grain shapes. *Phys. Rev. Lett.* 91, 215506. doi:10.1103/PhysRevLett.91.215506
4. Baveye, P.C., Laba, M., Otten, W., Bouckaert, L., Dello Sterpaio, P., Goswami, R.R., Grinev, D., Houston, A., Hu, Y., Liu, J., Mooney, S., Pajor, R., Sleutel, S., Tarquis, A., Wang, W., Wei, Q., Sezgin, M., 2010. Observer-dependent variability of the thresholding step in the quantitative analysis of soil images and X-ray microtomography data. *Geoderma* 157, 51–63. doi:10.1016/j.geoderma.2010.03.015
5. Baveye, P., Sposito, G., 1984. The Operational Significance of the Continuum Hypothesis in the Theory of Water Movement Through Soils and Aquifers. *Water Resour. Res.* 20, 521–530. doi:10.1029/WR020i005p00521
6. Bear, J., 1972. *Dynamics of fluids in porous media*, Dover Publications.
7. Bentz, D.P., Martys, N.S., 2007. A Stokes permeability solver for three-dimensional porous media. Tech. Rep. No. 7416.
8. Bhatnagar, P.L., Gross, E.P., Krook, M., 1954. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. *Phys. Rev.* 94, 511–525. doi:10.1103/PhysRev.94.511
9. Bird, M.B., Butler, S.L., Hawkes, C.D., Kotzer, T., 2014. Numerical modeling of fluid and electrical currents through geometries based on synchrotron X-ray tomographic images of reservoir rocks using Avizo and COMSOL. *Comput. Geosci.* 73, 6–16. doi:10.1016/j.cageo.2014.08.009
10. Blunt, M.J., Bijeljic, B., Dong, H., Gharbi, O., Iglauer, S., Mostaghimi, P., Paluszny, A., Pentland, C., 2013. Pore-scale imaging and modelling. *Adv. Water Resour.* 51, 197–216. doi:10.1016/j.advwatres.2012.03.003
11. Čapek, P., Veselý, M., Hejtmánek, V., 2014. On the measurement of transport parameters of porous solids in permeation and Wicke-Kallenbach cells. *Chem. Eng. Sci.* 118, 192–207. doi:10.1016/j.ces.2014.07.039

12. Chen, H., Meng, L., Chen, S., Jiao, Y., Liu, Y., 2016. Numerical investigation of microstructure effect on mechanical properties of bi-continuous and particulate reinforced composite materials. *Comput. Mater. Sci.* 122, 288–294. doi:10.1016/j.commatsci.2016.05.037
13. Chorin, A.J., 1967. A numerical method for solving incompressible viscous flow problems. *J. Comput. Phys.* 2, 12–26. doi:10.1016/0021-9991(67)90037-X
14. Cnudde, V., Masschaele, B., Dierick, M., Vlassenbroeck, J., Hoorebeke, L. Van, Jacobs, P., 2006. Recent progress in X-ray CT as a geosciences tool. *Appl. Geochemistry* 21, 826–832. doi:10.1016/j.apgeochem.2006.02.010
15. Coppola, A., Gerke, H.H., Comegna, A., Basile, A., Comegna, V., 2012. Dual-permeability model for flow in shrinking soil with dominant horizontal deformation. *Water Resour. Res.* 48, W08527. doi:10.1029/2011WR011376
16. Costanza-Robinson, M.S., Estabrook, B.D., Fouhey, D.F., 2011. Representative elementary volume estimation for porosity, moisture saturation, and air-water interfacial areas in unsaturated porous media: Data quality implications. *Water Resour. Res.* 47, W07513. doi:10.1029/2010WR009655
17. Dal Ferro, N., Strozzi, A.G., Duwig, C., Delmas, P., Charrier, P., Morari, F., 2015. Application of smoothed particle hydrodynamics (SPH) and pore morphologic model to predict saturated water conductivity from X-ray CT imaging in a silty loam Cambisol. *Geoderma* 255–256, 27–34. doi:10.1016/j.geoderma.2015.04.019
18. Dentz, M., Lester, D. R., Borgne, T. L., de Barros, F.P., 2016. Fluid stretching as a Lévy process. arXiv 1602.04904.
19. Diamond, S., 2000. Mercury porosimetry: an inappropriate method for the measurement of pore size distributions in cement-based materials. *Cem. Concr. Res.* 30, 1517–1525. doi:10.1016/S0008-8846(00)00370-7

20. Dikinya, O., Hinz, C., Aylmore, G., 2008. Decrease in hydraulic conductivity and particle release associated with self-filtration in saturated soil columns. *Geoderma* 146, 192–200. doi:10.1016/j.geoderma.2008.05.014
21. Dong, H., Blunt, M.J., 2009. Pore-network extraction from micro-computerized-tomography images. *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.* 80, 036307. doi:10.1103/PhysRevE.80.036307
22. Duwig, C., Delmas, P., Muller, K., Prado, B., Ren, K., Morin, H., Woodward, A., 2008. Quantifying fluorescent tracer distribution in allophanic soils to image solute transport. *Eur. J. Soil Sci.* 59, 94–102. doi:10.1111/j.1365-2389.2007.00970.x
23. Fortin, M., Peyret, R., and Temam, R., 1971. Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics, in: Holt, M. (Ed.), *Lecture Notes in Physics*, Vol.8. Springer-Verlag, pp. 336–342.
24. Garcia, X., Akanji, L. T., Blunt, M. J., Matthai, S. K., Latham, J. P., 2009. Numerical study of the effects of particle shape and polydispersity on permeability. *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.* 80(2), 021304. doi: 10.1103/PhysRevE.80.021304.
25. Gerke K.M., Sizonenko T.O., Khirevich S., Karsanina M.V., Umarova A.B., Korost D.V., Matthai S.K., Lamontaigne S., 2017. Obtaining soil saturated hydraulic conductivity using X-ray microtomography: comparison of LBM, FDM and PNM approaches (in revision, *Journal of Hydrology*).
26. Gerke, K.M., Korost, D. V., Vasilyev, R. V., Karsanina, M. V., Tarasovskii, V.P., 2015b. Studying structure and determining permeability of materials based on X-Ray microtomography data (using porous ceramics as an example). *Inorg. Mater.* 51, 951–957. doi:10.1134/S002016851509006X
27. Gerke, K.M., Skvortsova, E.B., Korost, D. V., 2012a. Tomographic method of studying soil pore space: Current perspectives and results for some Russian soils. *Eurasian Soil Sci.* 45, 700–709. doi:10.1134/S1064229312070034

28. Gerke, K. M., Karsanina, M. V., Skvortsova, E. B., 2012b. Description and reconstruction of the soil pore space using correlation functions. *Eurasian Soil Sci.* 45(9), 861-872. doi: 10.1134/S1064229312090049
29. Gerke, K.M., Karsanina, M. V., 2015. Improving stochastic reconstructions by weighting correlation functions in an objective function. *EPL (Europhysics Lett.* 111, 56002. doi:10.1209/0295-5075/111/56002
30. Gerke, K.M., Karsanina, M. V., Vasilyev, R. V., Mallants, D., 2014. Improving pattern reconstruction using directional correlation functions. *EPL (Europhysics Lett.* 106, 66002. doi:10.1209/0295-5075/106/66002
31. Gerke, K.M., Sidle, R.C., Mallants, D., 2015a. Preferential flow mechanisms identified from staining experiments in forested hillslopes. *Hydrol. Process.* 29, 4562–4578. doi:10.1002/hyp.10468
32. Gerke, K.M., Karsanina, M. V, Mallants, D., 2015c. Universal Stochastic Multiscale Image Fusion: An Example Application for Shale Rock. *Sci. Rep.* 5, 15880. doi:10.1038/srep15880
33. Ghanbarian, B., Daigle, H., 2016. Permeability in Two-Component Porous Media: Effective-Medium Approximation Compared with Lattice-Boltzmann Simulations. *Vadose Zone J.* 15. doi:10.2136/vzj2015.05.0071
34. Ginzbourg, I., Adler, P.M., 1994. Boundary flow condition analysis for the three-dimensional lattice Boltzmann model. *J. Phys.* 4, 191–214. doi:10.1051/jp2:1994123
35. Godinho, J.R.A., Gerke, K.M., Stack, A.G., Lee, P.D., 2016. The dynamic nature of crystal growth in pores. *Sci. Rep.* 6, 33086. doi:10.1038/srep33086
36. Guibert, R., Horgue, P., Debenest, G., Quintard, M., 2016. A Comparison of Various Methods for the Numerical Evaluation of Porous Media Permeability Tensors from Pore-Scale Geometry. *Math. Geosci.* 48, 329–347. doi:10.1007/s11004-015-9587-9

- 1020 37. Havelka, J., Kučerová, A., Sýkora, J., 2016. Compression and reconstruction of random
 1021 microstructures using accelerated lineal path function. *Comput. Mater. Sci.* 122, 102–117.
 1022 doi:10.1016/j.commatsci.2016.04.044
- 1023 38. Helmig, R., 1997. Multiphase flow and transport processes in the subsurface. A
 1024 contribution to the modeling of hydrosystems, *Environmental Engineering*. Springer-
 1025 Verlag.
- 1026 39. Herring, A.L., Andersson, L., Schlüter, S., Sheppard, A., Wildenschild, D., 2015.
 1027 Efficiently engineering pore-scale processes: The role of force dominance and topology
 1028 during nonwetting phase trapping in porous media. *Adv. Water Resour.* 79, 91–102.
 1029 doi:10.1016/j.advwatres.2015.02.005
- 1030 40. Holmes, D.W., Williams, J.R., Tilke, P., Leonardi, C.R., 2016. Characterizing flow in oil
 1031 reservoir rock using SPH: Absolute permeability. *Comput. Part. Mech.* 3, 141–154.
 1032 doi:10.1007/s40571-015-0038-7
- 1033 41. Houston, A.N., Otten, W., Baveye, P.C., Hapca, S., 2013. Adaptive-window indicator
 1034 kriging: A thresholding method for computed tomography images of porous media.
 1035 *Comput. Geosci.* 54, 239–248. doi:10.1016/j.cageo.2012.11.016
- 1036 42. Hyväluoma, J., Thapaliya, M., Alaraudanjoki, J., Sirén, T., Mattila, K., Timonen, J.,
 1037 Turtola, E., 2012. Using microtomography, image analysis and flow simulations to
 1038 characterize soil surface seals. *Comput. Geosci.* 48, 93–101.
 1039 doi:10.1016/j.cageo.2012.05.009
- 1040 43. Jiang, Z., Wu, K., Couples, G., van Dijke, M.I.J., Sorbie, K.S., Ma, J., 2007. Efficient
 1041 extraction of networks from three-dimensional porous media. *Water Resour. Res.* 43,
 1042 W12S03. doi:10.1029/2006WR005780
- 1043 44. Jiao, Y., Stillinger, F.H., Torquato, S., 2007. Modeling heterogeneous materials via two-
 1044 point correlation functions: Basic principles. *Phys. Rev. E - Stat. Nonlinear, Soft Matter*
 1045 *Phys.* 76, 031110. doi:10.1103/PhysRevE.76.031110

- 1046 45. Jiao, Y., Chawla, N., 2014. Modeling and characterizing anisotropic inclusion orientation in
 1047 heterogeneous material via directional cluster functions and stochastic microstructure
 1048 reconstruction. *J. Appl. Phys.* 115, 093511. doi:10.1063/1.4867611
- 1049 46. Kang, Q., Lichtner, P.C., Zhang, D., 2006. Lattice Boltzmann pore-scale model for
 1050 multicomponent reactive transport in porous media. *J. Geophys. Res. Solid Earth* 111,
 1051 B05203. doi:10.1029/2005JB003951
- 1052 47. Karsanina M.V., Gerke K.M., Vasilyev R.V., Korost, D.V., 2015. Using correlation
 1053 functions to model material's structure with desired physical properties. *Math. Model.*
 1054 *Comput. Simulations* 27, 50–63.
- 1055 48. Karsanina, M. V., Gerke, K.M., Skvortsova, E.B., Mallants, D., 2015. Universal spatial
 1056 correlation functions for describing and reconstructing soil microstructure. *PLoS One* 10,
 1057 e0126515. doi:10.1371/journal.pone.0126515
- 1058 49. Khan, F., Enzmann, F., Kersten, M., Wiegmann, A., Steiner, K., 2012. 3D simulation of the
 1059 permeability tensor in a soil aggregate on basis of nanotomographic imaging and LBE
 1060 solver. *J. Soils Sediments* 12, 86–96. doi:10.1007/s11368-011-0435-3
- 1061 50. Khirevich, S., Ginzburg, I., Tallarek, U., 2015. Coarse-and fine-grid numerical behavior of
 1062 MRT/TRT lattice-Boltzmann schemes in regular and random sphere packings. *J. Comput.*
 1063 *Phys.* 281, 708–742. doi:10.1016/j.jcp.2014.10.038
- 1064 51. Kulkarni, R., Tuller, M., Fink, W., Wildenschild, D., 2012. Three-Dimensional Multiphase
 1065 Segmentation of X-Ray CT Data of Porous Materials Using a Bayesian Markov Random
 1066 Field Framework. *Vadose Zo. J.* 11. doi:10.2136/vzj2011.0082
- 1067 52. Kumar, H., Briant, C.L., Curtin, W.A., 2006. Using microstructure reconstruction to model
 1068 mechanical behavior in complex microstructures. *Mech. Mater.* 38, 818–832.
 1069 doi:10.1016/j.mechmat.2005.06.030
- 1070 53. Landau, L.D., Liflishtz, E.M., 1987. Theoretical Physics, in 10 volumes, Vol. 6, Fluid
 1071 Mechanics. Oxford: Butterworth-Heinemann.

54. Larson, R.E., Higdon, J.J.L., 1989. A periodic grain consolidation model of porous media. *Phys. Fluids A Fluid Dyn.* 1, 38–46. doi:10.1063/1.857545
55. Li, H., Singh, S., Shashank Kaira, C., Mertens, J.C.E., Williams, J.J., Chawla, N., Jiao, Y., 2016. Microstructural Quantification and Property Prediction Using Limited X-ray Tomography Data. *JOM* 68, 2288–2295. doi:10.1007/s11837-016-2024-9
56. Liu, M., Shabaninejad, M., Mostaghimi, P., 2017. Impact of mineralogical heterogeneity on reactive transport modelling. *Computers & Geosciences* 104, 12–19. doi:10.1016/j.cageo.2017.03.020
57. Lock, P.A., Jing, X., Zimmerman, R.W., Schlueter, E.M., 2002. Predicting the permeability of sandstone from image analysis of pore structure. *J. Appl. Phys.* 92, 6311–6319. doi:10.1063/1.1516271
58. Ma, J., Wu, K., Jiang, Z., Couples, G.D., 2010. SHIFT: An implementation for lattice Boltzmann simulation in low-porosity porous media. *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.* 81, 056702. doi:10.1103/PhysRevE.81.056702
59. Manwart, C., Aaltosalmi, U., Koponen, A., Hilfer, R., Timonen, J., 2002. Lattice-Boltzmann and finite-difference simulations for the permeability for three-dimensional porous media. *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.* 66, 016702. doi:10.1103/PhysRevE.66.016702
60. Mariethoz, G., Straubhaar, J., Renard, P., Chuginova, T., Biver, P., 2015. Constraining distance-based multipoint simulations to proportions and trends. *Environ. Model. Softw.* 72, 184–197. doi:10.1016/j.envsoft.2015.07.007
61. Martys, N.S., Torquato, S., Bentz, D.P., 1994. Universal scaling of fluid permeability for sphere packings. *Phys. Rev. E* 50, 403–408. doi:10.1103/PhysRevE.50.403
62. Mehmani, A., Prodanović, M., Javadpour, F., 2013. Multiscale, Multiphysics Network Modeling of Shale Matrix Gas Flows. *Transp. Porous Media* 99, 377–390. doi:10.1007/s11242-013-0191-5

63. Miao X., Gerke K.M., Sizonenko T.O., 2017. A new way to parameterize hydraulic conductances of pore elements: A step forward to create pore-networks without pore shape simplifications. *Adv. Water Resour.* 105, 162–172. doi:10.1016/j.advwatres.2017.04.021
64. Moctezuma-Berthier, A., Vizika, O., Adler, P.M., 2002. Macroscopic conductivity of vugular porous media. *Transp. Porous Media* 49, 313–332. doi:10.1023/A:1016297220013
65. Mostaghimi, P., Bijeljic, B., Blunt, M., 2012. Simulation of flow and dispersion on pore-space images. *SPE J.* 17, 1131–1141. doi:10.2118/135261-PA
66. Mostaghimi, P., Blunt, M.J., Bijeljic, B., 2013. Computations of Absolute Permeability on Micro-CT Images. *Math. Geosci.* 45, 103–125. doi:10.1007/s11004-012-9431-4
67. Narsilio, G.A., Buzzi, O., Fityus, S., Yun, T.S., Smith, D.W., 2009. Upscaling of Navier-Stokes equations in porous media: Theoretical, numerical and experimental approach. *Comput. Geotech.* 36, 1200–1206. doi:10.1016/j.compgeo.2009.05.006
68. Noiriél, C., Steefel, C.I., Yang, L., Bernard, D., 2015. Effects of pore-scale precipitation on permeability and flow. *Adv. Water Resour.* 95, 125–137. doi:10.1016/j.advwatres.2015.11.013
69. Oh, W., Lindquist, B., W., 1999. Image thresholding by indicator kriging. *IEEE Trans. Pattern Anal. Mach. Intell.* 21, 590–602. doi:10.1109/34.777370
70. Øren, P.E., Bakke, S., 2003. Reconstruction of Berea sandstone and pore-scale modelling of wettability effects. *J. Pet. Sci. Eng.* 39, 177–199. doi:10.1016/S0920-4105(03)00062-7
71. Patankar, S., 1980. Numerical heat transfer and fluid flow, Series in computational methods in mechanics and thermal sciences.
72. Pereira Nunes, J.P., Bijeljic, B., Blunt, M.J., 2015. Time-of-Flight Distributions and Breakthrough Curves in Heterogeneous Porous Media Using a Pore-Scale Streamline Tracing Algorithm. *Transp. Porous Media* 109, 317–336. doi:10.1007/s11242-015-0520-y
73. Peyret, R., Taylor, T.D., 1983. Computational methods for fluid flow. Springer Science & Business Media.

74. Raeini, A.Q., Blunt, M.J., Bijeljic, B., 2012. Modelling two-phase flow in porous media at the pore scale using the volume-of-fluid method. *J. Comput. Phys.* 231, 5653–5668. doi:10.1016/j.jcp.2012.04.011
75. Renard, P., Genty, A., Stauffer, F., 2001. Laboratory determination of the full permeability tensor. *J. Geophys. Res. Solid Earth* 106, 26443–26452. doi:10.1029/2001JB000243
76. Sangani, A.S., Acrivos, A., 1982. Slow flow through a periodic array of spheres. *Int. J. Multiph. Flow* 8, 343–360. doi:10.1016/0301-9322(82)90047-7
77. Scheibe, T.D., Perkins, W.A., Richmond, M.C., McKinley, M.I., Romero-Gomez, P.D.J., Oostrom, M., Wietsma, T.W., Serkowski, J.A., Zachara, J.M., 2015. Pore-scale and multiscale numerical simulation of flow and transport in a laboratory-scale column. *Water Resour. Res.* 51, 1023–1035. doi:10.1002/2014WR015959
78. Schlüter, S., Weller, U., Vogel, H.J., 2010. Segmentation of X-ray microtomography images of soil using gradient masks. *Comput. Geosci.* 36, 1246–1251. doi:10.1016/j.cageo.2010.02.007
79. Schneider, C. A., Rasband, W.S., Eliceiri, K.W., 2012. NIH Image to ImageJ: 25 years of image analysis. *Nat. Methods* 9, 671–675. doi:10.1038/nmeth.2089
80. Shabro, V., Javadpour, F., Torres-Verdín, C., 2009. A generalized finite-difference diffusive-advective (FDDA) model for gas flow in micro-and nano-porous media. *World Journal of Engineering* 6(3), 7-15.
81. Shabro, V., Torres-Verdin, C., Javadpour, F., 2011. Numerical simulation of shale-gas production: from pore-scale modeling of slip-flow, Knudsen diffusion, and Langmuir desorption to reservoir modeling of compressible fluid. SPE 144355 Technical paper, presented at SPE Unconventional Resources Conference and Exhibition In North American Unconventional Gas Conference and Exhibition. doi: 10.2118/144355-MS

82. Shabro, V., Torres-Verdín, C., Javadpour, F., Sepehrnoori, K., 2012. Finite-Difference Approximation for Fluid-Flow Simulation and Calculation of Permeability in Porous Media. *Transp. Porous Media* 94, 775–793. doi:10.1007/s11242-012-0024-y
83. Sheppard, A.P., Sok, R.M., Averdunk, H., 2004. Techniques for image enhancement and segmentation of tomographic images of porous materials. *Physica A: Statistical Mechanics and Its Applications*. pp. 145–151. doi:10.1016/j.physa.2004.03.057
84. Silin, D., Patzek, T., 2006. Pore space morphology analysis using maximal inscribed spheres. *Phys. A Stat. Mech. its Appl.* 371, 336–360. doi:10.1016/j.physa.2006.04.048
85. Tahmasebi, P., Sahimi, M., 2012. Reconstruction of three-dimensional porous media using a single thin section. *Phys. Rev. E* 85, 06670913. doi:10.1103/PhysRevE.85.066709
86. Tamamidis, P., Zhang, G., Assanis, D., 1996. Comparison of pressure-based and artificial compressibility methods for solving 3D steady incompressible viscous flows. *J. Comput. Phys.* 13, 1–13. doi:10.1006/jcph.1996.0041
87. Tartakovsky, A.M., Trask, N., Pan, K., Jones, B., Pan, W., Williams, J.R., 2015. Smoothed particle hydrodynamics and its applications for multiphase flow and reactive transport in porous media. *Comput. Geosci.* 20, 807–834. doi:10.1007/s10596-015-9468-9
88. Temam, R., 1977. *Navier-Stokes equations. Theory and numerical analysis*. Amsterdam, New York, Oxford: North-Holland publishing company.
89. Thovert, J.F., Adler, P.M., 2011. Grain reconstruction of porous media: Application to a Bentheim sandstone. *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.* 83, 056116. doi:10.1103/PhysRevE.83.056116
90. Vasilyev, R. V., Gerke, K.M., Karsanina, M. V., Korost, D. V., 2016. Solution of the stokes equation in three-dimensional geometry by the finite-difference method. *Math. Model. Comput. Simulations* 8, 63–72. doi:10.1134/S2070048216010105
91. Veselý, M., Bultreys, T., Peksa, M., Lang, J., Cnudde, V., Van Hoorebeke, L., Kočířík, M., Hejtmánek, V., Šolcová, O., Soukup, K., Gerke, K., Stallmach, F., Čapek, P., 2015.

- Prediction and Evaluation of Time-Dependent Effective Self-diffusivity of Water and Other Effective Transport Properties Associated with Reconstructed Porous Solids. *Transp. Porous Media* 110, 81–111. doi:10.1007/s11242-015-0557-y
92. Vladimirova N.N., Kuznetsov B.G., Yanenko N.N., 1956. Numerical simulations of viscous incompressible fluid symmetrical flow over the plate. In *Some aspects of computational and applied mathematics* (Ed. Marchuk G.I.), Novosibirsk, Nauka, 186-192 (in Russian).
93. Wildenschild, D., Sheppard, A.P., 2013. X-ray imaging and analysis techniques for quantifying pore-scale structure and processes in subsurface porous medium systems. *Adv. Water Resour.* 51, 217–246. doi:10.1016/j.advwatres.2012.07.018
94. Willingham, T.W., Werth, C.J., Valocchi, A.J., 2008. Evaluation of the effects of porous media structure on mixing-controlled reactions using pore-scale modeling and micromodel experiments. *Environ. Sci. Technol.* 42, 3185–3193. doi:10.1021/es7022835
95. Xu, P., Yu, B., 2008. Developing a new form of permeability and Kozeny-Carman constant for homogeneous porous media by means of fractal geometry. *Adv. Water Resour.* 31, 74–81. doi:10.1016/j.advwatres.2007.06.003
96. Yeong, C., Torquato, S., 1998. Reconstructing random media. II. Three-dimensional media from two-dimensional cuts. *Phys. Rev. E* 58, 224–233. doi:10.1103/PhysRevE.58.224
97. Zaretskiy, Y., Geiger, S., Sorbie, K., Förster, M., 2010. Efficient flow and transport simulations in reconstructed 3D pore geometries. *Adv. Water Resour.* 33, 1508–1516. doi:10.1016/j.advwatres.2010.08.008
98. Zeinijahromi, A., Farajzadeh, R., Bruining, J., Bedrikovetsky, P., 2016. Effect of fines migration on oil-water relative permeability during two-phase flow in porous media. *Fuel* 176, 222–236. doi:10.1016/j.fuel.2016.02.066

- 1198 99. Zhao, F., Landis, H.R., Skerlos, S.J., 2005. Modeling of porous filter permeability via
1199 image-based stochastic reconstruction of spatial porosity correlations. *Environ. Sci.*
1200 *Technol.* 39, 239–247. doi:10.1021/es035228b
- 1201 100. Zick, A.A., Homsy, G.M., 1982. Stokes flow through periodic arrays of spheres. *J.*
1202 *Fluid Mech.* 115, 13–26. doi:10.1017/S0022112082000627

Highlights:

- 1) Free software to simulate pore-scale single-phase flow
- 2) Software is simple to use and contains both batch and GUI versions
- 3) Parallelization and code optimization make simulations on modern laptops possible
- 4) Software was verified using known sphere packing and by comparison to published data
- 5) Five different case studies illustrate the main capabilities and functionalities of the software