Journal of Computational Physics 229 (2010) 7456–7476

Contents lists available at ScienceDirect

Journal of Computational Physics

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

Direct pore-level modeling of incompressible fluid flow in porous media

Saeed Ovaysi *, Mohammad Piri

Department of Chemical and Petroleum Engineering, University of Wyoming, Laramie, WY 82071-2000, USA

ARTICLE INFO

Article history:
Received 6 September 2009
Received in revised form 15 June 2010
Accepted 15 June 2010
Available online 23 June 2010

Keywords:
Moving particle semi-implicit
Particle-based methods
Porous media
Incompressible fluid flow
Micro-CT X-ray imaging

ABSTRACT

We present a dynamic particle-based model for direct pore-level modeling of incompressible viscous fluid flow in disordered porous media. The model is capable of simulating flow directly in three-dimensional high-resolution micro-CT images of rock samples. It is based on moving particle semi-implicit (MPS) method. We modify this technique in order to improve its stability for flow in porous media problems. Using the micro-CT image of a rock sample, the entire medium, i.e., solid and fluid, is discretized into particles. The incompressible Navier–Stokes equations are then solved for each particle using the MPS summations. The model handles highly irregular fluid–solid boundaries effectively. An algorithm to split and merge fluid particles is also introduced. To handle the computational load, we present a parallel version of the model that runs on distributed memory computer clusters. The accuracy of the model is validated against the analytical, numerical, and experimental data available in the literature. The validated model is then used to simulate both unsteady- and steady-state flow of an incompressible fluid directly in a representative elementary volume (REV) size micro-CT image of a naturally-occurring sandstone with 3.398 μm resolution. We analyze the quality and consistency of the predicted flow behavior and calculate absolute permeability using the steady-state flow rate.

1. Introduction

Transport phenomena in porous media is of great interest to scientists in many areas of science and engineering, e.g., petroleum reservoir and environmental engineering. This is mainly due to the need for an improved understanding of the phenomena that control fluid flow behavior in various applications, e.g., enhanced hydrocarbon recovery and remediation strategies. It is therefore imperative to develop experimental and modeling techniques that allow detailed investigation of fundamental phenomena seen in these applications.

Physically-based pore-level models are among the most appealing techniques that are used to shed light on the impact of various pore-level phenomena on the fluid flow behavior and macroscopic flow properties, e.g., relative permeabilities and capillary pressures used by Darcy scale models. The predictive capabilities of these techniques, however, has been dependent on three major components: (1) detailed understanding of multiphase displacement mechanisms at the pore level, (2) availability of accurate three-dimensional maps of pore space in the porous medium, and (3) availability of inexpensive and powerful computational resources.

During the last two decades, our knowledge of the physics of multiphase flow at the pore level has significantly improved through experimental investigation of displacements in micromodels and core samples [1,2]. But, it was only during the last 10–20 years that more realistic representations of pore space in porous medium have become available using various techniques such as serial sectioning [3], confocal laser scanning microscopy [4], stochastic reconstruction using statistical

* Corresponding author. Tel.: +1 307 766 4923; fax: +1 307 766 6777.
E-mail addresses: sovaysi@uwyo.edu (S. Ovaysi), mpiri@uwyo.edu (M. Piri).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.06.028

methods [5], process-based reconstruction [6] and X-ray microtomography [7,8]. Among them, X-ray microtomography is perhaps the most general method that can be used to image relatively large samples at very high resolutions and at a reasonably low cost.

Pore-level models are generally divided into two main categories: (1) pore network models and (2) direct numerical models. In the first category, the model, whether dynamic or quasi-static, reads as input a bundle of capillary tubes or a two- or three-dimensional network as a representative of the porous medium [9]. During the last 5–10 years random three-dimensional networks have become available, which are built mostly for sandstone rocks using their X-ray images or thin sections. These models have been successfully used to predict two- and three-phase flow properties in sandstones [10–14]. In these networks, the pores of the rock sample are represented by idealized geometries and transformation of capillary channels in, for instance, X-ray images to idealized geometries inevitably leads to loss of some topological features that could introduce uncertainties into the predicted flow properties. Furthermore, the existence of very large pores in, for instance, carbonates can lead to formation of dynamic effects that are not reliably addressed by dynamic pore network models. Therefore, the challenging flow problems, e.g., flow in carbonates and near wellbore high capillary number flows, require a different modeling approach. This brings us to the direct pore-level modeling approach, the second category. This group of pore-level models includes those that aim at solving the governing PDEs directly using numerical techniques. In other words, they aim at using a realistic representation of the pore space in the porous medium, e.g., a three-dimensional voxel X-ray image, and solving the Navier–Stokes equations directly in them. The progress toward this challenging goal, however, has been hampered over the years due to various reasons, e.g., computational cost and difficulties in method development. It is worth mentioning that the majority of methods in this group are Lagrangian particle-based (mesh free) methods, e.g., Lattice-Boltzmann, smoothed particle hydrodynamic (SPH), and moving particle semi-implicit (MPS). Mesh-based methods, e.g., finite element, have also been employed to compute, for instance, hydraulic conductivity in three-dimensional X-ray images of porous media, see, for example, [15]. And also, to study the effects of pore space on the permeability of statistically generated random porous media, Smolarkiewicz and Winter in [16] utilized an implicit immersed-boundary method. But, the random nature of porous materials and high deformability of fluid interfaces make the mesh free methods easier to design [17].

The Lattice-Boltzmann method provides a platform to look at some interesting fluid flow phenomena [18,19]. But, the method simulates virtual fluids and exhibits poor stability. The particle-based method SPH has been adopted to solve the governing equations of flow in porous media [20,21], but despite its success in handling compressible fluids, it is not a suitable technique to model flow of an incompressible fluid [22]. It uses the Navier–Stokes equations and an equation of state to generate the pressure field. Application of an equation of state makes it difficult to handle an incompressible fluid. Moving particle semi-implicit, on the other hand, has proved to be a potent particle-based method to model flow of incompressible fluids [23,24], although there are still some issues regarding its stability that have to be resolved. Semi-Lagrangian particle-based methods such as particle in cell (PIC) [25] and fluid-implicit-particle (FLIP) [26,27] present an alternative approach in modeling fluid flow which exhibit better stability via utilizing auxiliary grids to enforce incompressibility and boundary conditions.

Over the years, the above-mentioned techniques have been used to model flow at the pore level with different degrees of success in predicting transport properties. Salles et al. [28] used finite difference along with the artificial compressibility method [29] to simulate fluid flow in a reconstructed random porous medium that represents Fontainebleau sandstone. The results however differ by a maximum factor of five from the experimental data. Two-dimensional Lagrangian simulations using SPH, cf. [30], have been carried out in randomly generated porous systems with no comparison with experimental data. To the best of our knowledge, SPH has not been used to model flow directly in 3D microtomography images of porous media. Arns et al. [31,32] used finite element and Lattice-Boltzmann to predict elastic properties and absolute permeability from microtomography images. The predicted properties agreed well with those obtained from Gassmann's equations (for elastic properties) and experimental data (for absolute permeability). Øren et al. [33] have also used finite element, finite difference, Lattice-Boltzmann and random walk methods to obtain elastic moduli, electrical resistivity, absolute permeability, and magnetic resonance of two different sandstone lithofacies from North Sea using numerically reconstructed rocks and X-ray microtomography images. The computed properties for the reconstructed rocks and X-ray microtomography images compared favorably. Also, Fredrich et al. [34] used Lattice-Boltzmann method to calculate the absolute permeability of a naturally-occurring sandstone. The calculated permeability was in the range of the experimental data for the same rock sample.

The discussion above suggests that unavailability of a stable and general particle-based method for direct pore-level modeling of flow may have hindered the progress toward the goal of directly modeling various processes in three-dimensional high-resolution images to predict different properties. Furthermore, the computational cost inherent to the particle-based methods and unavailability of efficient parallel algorithms have only added to the complexity of the work. Without scalable parallel algorithms it would be impractical to model flow in large three-dimensional images, e.g., representative elementary volume (REV). For instance, the computational cost and scalability problems have been among the reasons for the authors in [21,35] not to use more realistic systems in their SPH modeling work. Noteworthy, the currently available parallel algorithms for particle-based methods have been developed for problems in astrophysics, see the work of Hernquist et al. [36].

In this work, we present a modified moving particle semi-implicit (MMPS) technique to model flow of incompressible fluids directly in three-dimensional high-resolution X-ray images of random man-made or naturally-occurring porous media. This paper is structured as follows. We start with a brief discussion on pore space representation and medium used in this work. This is followed by a detailed description of the model and the techniques used to handle specific complexities

associated with modeling pore-level flow in random porous systems, e.g., different boundary conditions. The model is then validated against analytical, numerical and experimental data available in the literature. We continue with presenting the results of a flow simulation in an REV size naturally-occurring sandstone. Finally, we discuss the highlights of parallelization of MMPS followed by a comprehensive analysis of the scalability of our implementation.

2. Pore space representation

The use of physically-based pore-level models of fluid flow, e.g., MMPS, requires faithful representations of pore space in the porous medium. The X-ray microtomography techniques provide high-resolution voxel X-ray images of the porous medium. These images include coordinates of the solid and void voxels in three dimensions. We use these information to build the particle version of the image by assigning a particle to every voxel. Therefore, each particle, depending on its coordinates, is marked as solid or fluid. Also, resolution of the original X-ray image is used as the initial particle size.

In Fig. 1 we show the rock sample studied in this work. It is a reservoir sandstone, called S_9, that is taken from [8]. Fig. 1(a) illustrates the original X-ray image of this sandstone. The particle version of this image is shown in Fig. 1(b) where we, for illustrative purposes, visualize only the fluid particles.

Various properties, e.g., resolution, of these images can have direct implications for accuracy of the predicted flow behavior. Fig. 2 shows the cross section of an imaginary pore, where the solid line represents the real boundary of the pore. The corresponding X-ray image of this pore would be the gray and white grids, which represent the solid and void areas, respectively. Regardless of where we put the particles in each grid, the final pore represented by the fluid particles (dashed line) is larger than the pore represented by the X-ray data, which in turn is different from the real pore. As a consequence, the overall conductivity of the porous sample computed using a particle-based method could be greater than its actual conductivity. However, if the voxel size, i.e., resolution, of the X-ray image is much smaller than the size of the average pore in the porous sample, the aforementioned error will be negligible which results in more accurate fluid flow simulations.

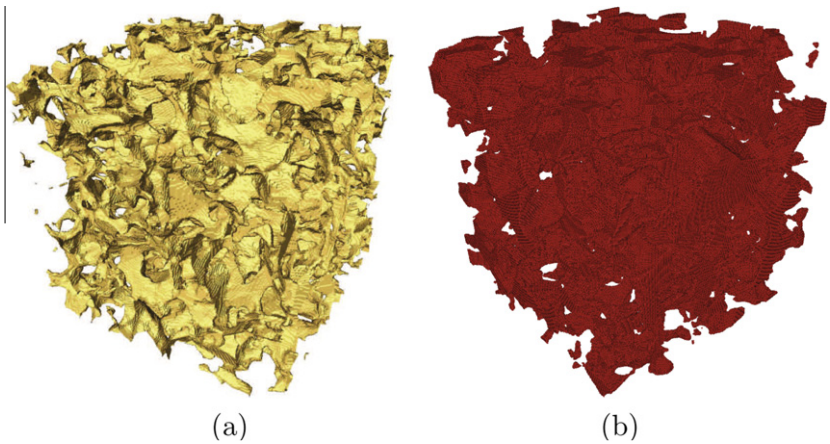


Fig. 1. A 1.02 × 1.02 × 1.02 mm 3D voxel image of a sandstone generated by X-ray microtomography with the resolution of 3.398 μm. (a) Iso-surface. (b) Particle representation of the image (only fluid particles are shown).

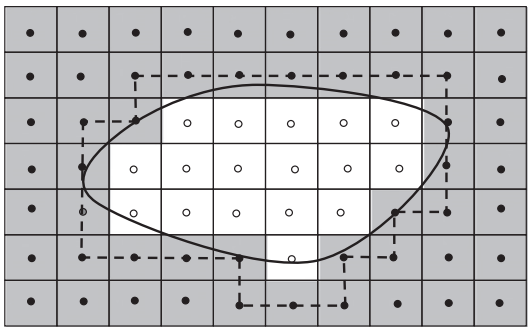


Fig. 2. An imaginary pore and its X-ray image and particle representation. The actual pore is shown with a solid line, the gray and white grids represent the void and solid spaces detected by X-ray, respectively, and the dashed line is the pore represented by the particles.

This fully realistic representation of the porous medium, however, introduces challenges regarding the treatment of irregular fluid/solid boundaries and preservation of the connectivity between the fluid particles without compromising the accuracy. In following section, we first introduce the model and then include detailed discussions regarding these challenges and the techniques we use to handle them.

3. Pore scale model

3.1. Particle-based summations

In a particle-based method such as MPS, a continuous fluid is represented by a set of $i = 1, \ldots, N$ particles where the position of each particle is shown by the vector $\mathbf{r}_i$. The computational units in this method are particles that are tracked in time and have several physical properties, e.g., velocity and pressure, associated with them. The movement of these computational units simulates fluid flow. All the dependent variables of the flow problem are found for each particle using an interpolation scheme as we will explain shortly.

Consider a continuous function $A(\mathbf{r})$ at some given point in the medium with the distance vector $\mathbf{r}$ from the origin. An integral representation of this function can be written as

$$A(\mathbf{r}) = \int A(\mathbf{r}')W(\mathbf{r} - \mathbf{r}', h)d\mathbf{r}', \tag{1}$$

where W is the kernel. The parameter h is the kernel size that defines the domain of influence of the kernel for each particle. The kernel is an even function and may, in special cases, become the Dirac delta function when h approaches zero, i.e.,

$$\lim_{h\to 0} W(\mathbf{r} - \mathbf{r}', h) = \delta(\mathbf{r} - \mathbf{r}'). \tag{2}$$

But, one should note that unlike the kernels used in SPH, the kernels in MPS *do not* necessarily satisfy the normalization condition given by

$$\int W(\mathbf{r} - \mathbf{r}', h)d\mathbf{r} = 1. \tag{3}$$

Finally, the kernel is chosen in a way that it adopts zero beyond the boundaries of the domain of influence for particle i (compact condition), i.e.,

$$W(\mathbf{r} - \mathbf{r}', h) = 0, \quad \text{when} \quad |\mathbf{r} - \mathbf{r}'| > h, \tag{4}$$

which allows the computational cost of the simulations to be controlled particularly in large flow problems [23]. Properties of particle i can be calculated using a normalized approximation of the integral representation, i.e.,

$$\overline{A}(\mathbf{r}_i) = \frac{\sum_j^{\mathbb{N}} A_j W(|\mathbf{r}_i - \mathbf{r}_j|)}{\sum_j^{\mathbb{N}} W(|\mathbf{r}_i - \mathbf{r}_j|)}, \tag{5}$$

where $\mathbb{N}$ is the total number of particles in the neighborhood "h" of i, $\overline{A}(\mathbf{r}_i)$ is an approximation of $A(\mathbf{r})$ for particle i and $A_i = A(\mathbf{r}_i)$.

Eq. (6) presents the kernel that we use in this work. It is a second-order polynomial that was proposed by Koshizuka et al. [23]:

$$W_{ij} = \begin{cases} -\left(\frac{2r}{h}\right)^2 + 2 & 0 \leqslant r < \frac{h}{2} \\ \left(\frac{2r}{h} - 2\right)^2 & \frac{h}{2} \leqslant r < h \\ 0 & r \geqslant h \end{cases}, \tag{6}$$

where $W_{ij} = W(r_{ij})$ and $r = r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$. Fig. 3 illustrates the variation of kernel in the neighborhood of any particle i.

In order to convert partial differential equations, arising from physical conservation laws, to their particle-based equivalents (ordinary differential equations), one would need to use the particle-based form of various differential vector operators. In [23] authors propose Eqs. (7), (8), and (9) for the gradient, divergence, and laplacian operators, respectively, acting on any particle i:

$$\nabla_i A = \frac{d}{\mathcal{N}_i} \sum_j^{\mathbb{N}} \frac{A_j - A_i}{r_{ij}^2} (\mathbf{r}_j - \mathbf{r}_i) W_{ij}, \tag{7}$$

$$\nabla_i \cdot \mathbf{v} = \frac{d}{\mathcal{N}_i} \sum_j^{\mathbb{N}} \frac{(\mathbf{v}_j - \mathbf{v}_i) \cdot (\mathbf{r}_j - \mathbf{r}_i)}{r_{ij}^2} W_{ij}, \tag{8}$$

$$\nabla_i^2 A = \frac{2d}{\lambda \mathcal{N}_i} \sum_j^{\mathbb{N}} (A_j - A_i) W_{ij}, \tag{9}$$

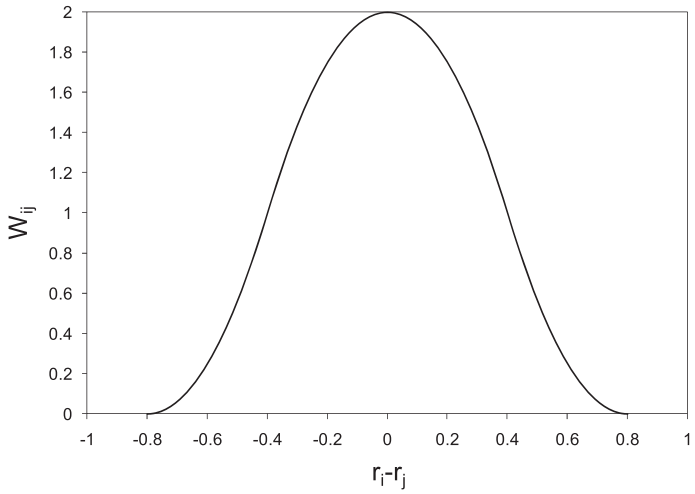


Fig. 3. A second-order polynomial kernel with $h = 8$ μm, see Eq. (6).

where d is the number of spatial dimensions and the parameter λ is

$$\lambda = \frac{\int_V W(r) r^2 dv}{\int_V W(r) dv}, \tag{10}$$

also, $\mathcal{N}_i$, particle number density, is defined as

$$\mathcal{N}_i = \sum_j^{\mathbb{N}} W_{ij}. \tag{11}$$

3.2. MMPS formulation

The Lagrangian version of the governing equations, i.e., Navier–Stokes equations, for flow of incompressible fluids with uniform density are:

$$\frac{D\mathbf{v}}{Dt} = -\frac{1}{\rho}\nabla P + \frac{\mu}{\rho}\nabla^2 \mathbf{v} + \mathbf{g}, \tag{12}$$

$$\nabla \cdot \mathbf{v} = 0, \tag{13}$$

where $\mathbf{v}$ is velocity, ρ is density, P is the total pressure, μ is viscosity, and $\mathbf{g}$ is gravity. Applying the projection method to the governing equations, and splitting the velocity into the explicit and implicit parts:

$$\mathbf{v} = \mathbf{v}_E + \mathbf{v}_I, \tag{14}$$

leads to:

$$\frac{D\mathbf{v}_E}{Dt} = -\frac{1}{\rho}\nabla P_s + \frac{\mu}{\rho}\nabla^2 \mathbf{v} + \mathbf{g}, \tag{15}$$

and

$$\frac{D\mathbf{v}_I}{Dt} = -\frac{1}{\rho}\nabla P_d, \tag{16}$$

Here, P_s and P_d represent the static and dynamic pressures, respectively, and $P = P_s + P_d$. Splitting of pressure enables us to differentiate between the static pressure that is solely due to the external sources, i.e., upstream and downstream pressures, and the pressure that is caused by the movement of fluid (dynamic pressure). P_d is obtained from the solution of:

$$\nabla^2 P_d = \frac{\rho}{\Delta t}\nabla . \mathbf{v}_E \tag{17}$$

and P_s is calculated by

$$\nabla^2 P_s = 0. \tag{18}$$

Morris et al. [37] have also discussed splitting the total pressure into static and dynamic parts in the context of SPH. In that work, the (hydro-) static pressure gradient refers to an external body force that is applied to all the particles equally. In our work, however, that approach is not appropriate due to the fact that we deal with complicated random porous systems where false connectivity (see Section 3.3) can affect the distribution of P_s. Therefore, we cannot apply the same (hydro-) static pressure gradient to all the particles. Instead, we use the static pressure that satisfies Eq. (18) and its profile across the system is *fixed*. But, as a fluid particle changes location with time, it adopts new static pressure consistent with its new location. Static pressure of a particle depends on its connectivity and location and plays the same role as body force with this difference that connectivity of the fluid is taken into account. This, along with the boundary conditions on the above-mentioned equations, will be elaborated upon after a discussion on false connectivity.

Also, to make up for the cumulative numerical error arising from the solution of Eq. (17), Eq. (13) for each particle i can be rewritten as

$$\frac{1}{\Delta t}\sum_{k=1}^{n}\nabla\cdot\mathbf{v}_i^k\Delta t^k+\nabla\cdot\mathbf{v}_i=0, \tag{19}$$

and since static pressure is not a function of velocity, only Eq. (17) will change, i.e.,

$$\nabla^2 P_{d,i}=\frac{\rho}{\Delta t}\left(\nabla\cdot\mathbf{v}_{E,i}+\frac{1}{\Delta t}\sum_{k=1}^{n}\nabla\cdot\mathbf{v}_i^k\Delta t^k\right), \tag{20}$$

where the summation brings in the contribution of the continuity equation from the previous time steps, i.e., 1 to n, and n is the number of time steps. Since this summation is built up over time and there is no need to calculate the whole summation at every time step, the extra computational load is negligible.

The particles are advanced in time using [30]:

$$\mathbf{r}_i^{n+1}=\mathbf{r}_i^n+\mathbf{v}_i^n\Delta t+\frac{D\mathbf{v}_i^n}{dt}\frac{\Delta t^2}{2}, \tag{21}$$

where Δt is calculated based on the CFL criterion using:

$$\Delta t=\frac{CD_p}{\mathcal{V}_{\max}} \tag{22}$$

where D_p is the initial particle spacing and determines the size of each particle, $\mathcal{V}_{\max}$ is the maximum velocity of particles at any given time and C is the Courant number. We use $0.01\leqslant C\leqslant 0.1$ in this work.

It has to be mentioned that the original MPS formulation [23,24] calculates an intermediate number density from the explicit part. Using this intermediate number density in the continuity equation, one can restore the initial number density and hence the incompressibility by calculating pressure for each particle. However, based on our experience, that formulation seems to be unstable, due to the existence of Δt^2 in the denominator of the pressure equation (see Eq. (14) in [24]). Therefore, smaller time steps would not alleviate the instability. Whereas, using MMPS, we were able to perform much more stable simulations allowing us to take larger time steps.

3.3. False connectivity

In a real random porous material, it is often the case that two neighboring channels are separated by a thin solid wall, see Fig. 4(a) for a schematic diagram. The presence of such solid walls should prevent any direct impact of fluid particles on its opposite sides on each other. In other words, flow streams on the opposite sides should be "independent", if the flow field necessitates. However, as shown in Fig. 4(a), the fluid particles in channel A can be affected directly by the unrelated fluid

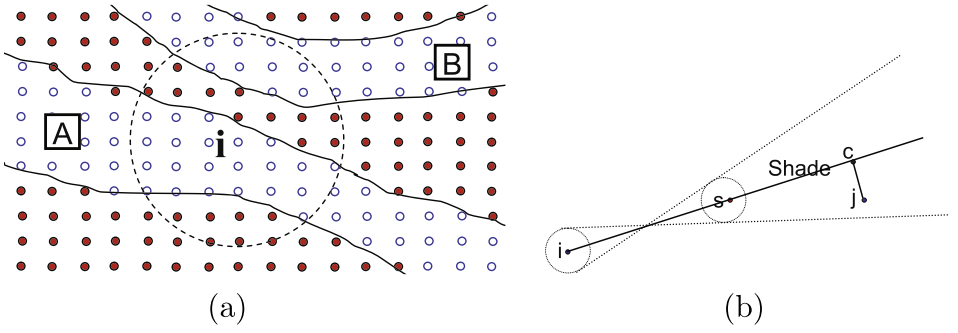


Fig. 4. (a) A false connectivity with fluid particles in a neighboring fluid channel. The fluid and solid particles are shown with open and solid circles, respectively. (b) Particle j in the shaded area behind solid particle s is excluded from the calculations for particle i.

particles in the neighboring channel B, or vise versa. We call this phenomena *false connectivity*, which is an artifact of using particle-based methods. Also, bodies of solid material, i.e., solid branches, in the flow domain should not allow direct interactions between fluid particles on its opposite sides. Due to the random nature of porous materials, the presence and the structure of solid material within the neighborhood of a fluid particle changes with time as it moves, and therefore it is impractical to detect the possible false connectivities before the simulation starts. Thus, false connectivities must be detected automatically during the simulation.

To eliminate the impact of unconnected neighbors on particle i, an imaginary beam of light is radiated from particle i in every direction, see Fig. 4(b). Considering the solid particles as the only barriers to this light, the particles in the shadow area (unconnected particles) are excluded from the calculations for particle i. One should note that, the shaded fluid particles may still impact particle i indirectly, i.e., through other fluid particles, which is completely legitimate. In this approach, each solid particle is considered as an obstacle. Using Eq. (23), the shaded fluid particles can be detected for each solid particle:

$$\begin{cases} |r_{jc}| \leqslant D_p \frac{|r_{ic}|}{|r_{is}|} \\ \vec{is} \cdot \vec{cs} < 0 \end{cases}, \tag{23}$$

where:

$$\mathbf{r}_c = \mathbf{r}_i + (\mathbf{r}_s - \mathbf{r}_i) \frac{\vec{is} \cdot \vec{ji}}{\vec{is} \cdot \vec{is}}. \tag{24}$$

As it is shown in Fig. 4(b), point c is the intersection of the $\vec{is}$ and $\vec{jc}$ lines which are orthogonal and s is the solid particle.

Detecting the unconnected fluid particles for particle i could be computationally demanding. In order to alleviate this, we use the position of the solid particles in the neighborhood of i from the previous time step to detect the unconnected particles in the neighbor search step using Eq. (23). Note that we can make use of the position of the solid particles from the previous time step only because the distance traveled by each fluid particle in each time step is very small compared to the size of the particle.

3.4. Boundaries

In the context of fluid flow, we may categorize the boundaries to (I) open and (II) phase boundaries. Open boundaries are those that create the driving force for the flow, i.e., the *upstream* and *downstream* sections. The second category refers to the fluid/fluid and fluid/solid interfaces. In this section, we discuss how we handle these two groups of boundaries for single-phase flow in porous media.

3.4.1. Open boundaries

In the context of particle-based methods, open boundaries have been dealt with in two different ways. In the first approach, it is tried to modify the gradient of kernel [38,39] in the boundaries. Since in MMPS we do not utilize the gradient of kernel, this method cannot be applied here. The second approach is periodic boundary condition and a good example of it is presented in [37]. But, due to the random nature of the heterogeneous porous systems studied here, periodic boundary conditions is not a suitable solution.

In the present work, we introduce upstream (US) and downstream (DS) sections to the inlet and outlet of the system, see Fig. 5(a). Since the US and DS sections create the driving force for flow, their particles do not evolve with time, i.e., we do not calculate acceleration for these particles. Conditions of the upstream and downstream of the flow problem under study determine the conditions of the particles in the US and DS sections. The length of each of the US and DS sections should

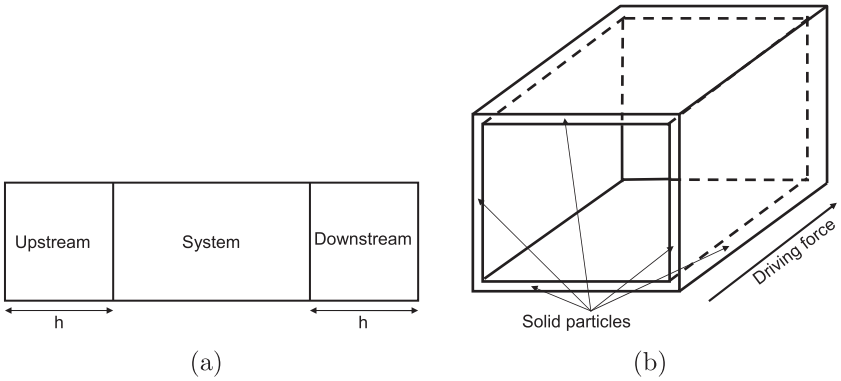


Fig. 5. (a) Upstream and downstream sections are added to the system to mimic the conditions of the actual flow system. (b) The porous sample is put inside a shell of solid particles with the thickness of kernel size (h) to limit the flow only to the pores inside the sample.

be greater than or equal to the kernel size, h, in order to have a complete sphere of neighborhood for the particles at the inlet and outlet. Other dimensions of the US and DS sections are determined based on the dimensions of the system. Since no calculation is done for the US and DS particles, no extra computational load is created. This configuration allows the system to get its net pressure gradient driving force from the US and DS sections, instead of using a body force as driving force.

For each particle in the US and DS sections, a matching particle in the *main domain* of system is found in order to copy its velocity. The matching particle for each US or DS particle is its *closest* particle in the main domain. This is done because velocity is not calculated for the particles in these sections. In other words, the US and DS particles move with the same velocity as their matching particles. This configuration satisfies $\nabla \cdot \mathbf{v} = 0$ for the particles of the main domain that are located in the proximity of the US and DS sections. Note that the US and DS particles that fall in the sphere of neighborhood of their matching particles have identical velocities.

Upon crossing the boundary between the US section and the main domain, the US particles join the particles in the main domain and their properties must evolve based on the governing equations. Also, to make up for the reduced volume in the US section, a new US particle must be added to the beginning of this section. Similarly, the main domain particles that enter the DS section must copy the velocity of a matching particle in the main domain. Furthermore, since the volume of the DS section is constant, when a system particle enters this section, a downstream particle is removed permanently. To calculate the inflow/outflow of the system, one can readily measure the number (volume) of the US/DS particles injected/removed as a function of time.

Both the US and DS regions should not be influenced by the disturbances caused by the flow irregularities in the main domain. This is to maintain the assumption of zero dynamic pressure for these regions. However, if flow disturbances do influence the US and DS regions, a stabilizer section is added between the US section and the inlet of the main domain and between the outlet of the main domain and DS section to ensure a recirculation-free pattern in the US and DS sections. In the stabilizer sections the flow disturbances are attenuated to zero at the time $\tau = \frac{V_{\text{stab}}}{Q}$, where V_{stab} is the volume of the stabilizer section and Q is the flow rate.

Also, to create a no-flow boundary condition on the sides of sample parallel to the main flow direction, we form a shell of solid particles, see Fig. 5(b), around the sample. This confines the flow only to the space inside the shell and prevents any artificial fluid channels forming outside the main domain. The shell has the thickness of the kernel size, h, and covers the flow system in all sides except where the upstream and downstream pressures are applied. The fluid/solid boundary condition discussed in the next section is applied to the shell solid particles as well as the internal solid particles.

3.4.2. Phase boundaries

Here we observe the fluid/solid boundary conditions by defining the right domain for each of the equations described in Section 3.2. But first we define $\mathcal{M}_i$ as the number of solid and unconnected fluid particles in the neighborhood of particle i, and introduce an alternative form of Eq. (11), i.e.,

$$\widehat{\mathcal{N}}_i = \sum_{j}^{\mathbb{N}-\mathcal{M}_i} W_{ij}, \tag{25}$$

which will prove useful in the following discussion.

Noting that static pressure is only a function of connectivity and the position of the fluid particles, for every fluid particle i Eq. (18) is valid only for its domain of connected fluid particles. Therefore, after expansion over $\mathbb{N} - \mathcal{M}_i$ using Eq. (9), Eq. (18) can be rewritten as

$$\widehat{\mathcal{N}}_i P_{si} - \sum_{j}^{\mathbb{N}-\mathcal{M}_i} P_{sj} W_{ij} = 0, \tag{26}$$

which, using US and DS pressure as boundary conditions, presents a system of linear equations for P_s. The same approach is applied when P_s is used in the expanded version of Eq. (15), i.e.,

$$\left(\frac{D\mathbf{v}_E}{Dt}\right)_i = -\frac{1}{\rho}\frac{d}{\widehat{\mathcal{N}}_i}\sum_{j}^{\mathbb{N}-\mathcal{M}_i}\frac{(P_{sj}-P_{si})\cdot(\mathbf{r}_j-\mathbf{r}_i)}{r_{ij}^2}W_{ij} + \frac{\mu}{\rho}\frac{2d}{\lambda\mathcal{N}_i}\sum_{j}^{\mathbb{N}}(\mathbf{v}_j-\mathbf{v}_i)W_{ij} + \mathbf{g}, \tag{27}$$

where $\mathbf{v}_j = 0$ for $j = 0, \ldots, \mathcal{M}_i$.

It is important to note that we do not assign zero velocity to the fluid particles close to the solid boundaries, neither do we use any repulsive force to create a no-flow fluid/solid interface. Instead, by including the solid particles in Eq. (20), the dynamic pressure is calculated for both the solid and fluid particles. This evolves the velocities of fluid particles in accordance to incompressible Navier–Stokes equations and naturally generates boundary velocities, i.e.,

$$\mathcal{N}_i P_{di} - \sum_{j}^{\mathbb{N}} P_{dj} W_{ij} = -\frac{\rho\lambda}{2\Delta t}\left(\sum_{j}^{\mathbb{N}}\frac{(\mathbf{v}_{E,j}-\mathbf{v}_{E,i})\cdot(\mathbf{r}_j-\mathbf{r}_i)}{r_{ij}^2}W_{ij} + \sum_{j}^{\mathbb{N}}\frac{\left(\mathbf{v}_j^n-\mathbf{v}_i^n\right)\cdot(\mathbf{r}_j-\mathbf{r}_i)}{r_{ij}^2}W_{ij}\right) - \frac{\rho\lambda\mathcal{N}_i}{2d\Delta t^2}\sum_{k=1}^{n-1}\nabla\cdot\mathbf{v}_i^k\Delta t^k, \tag{28}$$

where $\mathbf{v}_j = \mathbf{v}_{E,j} = 0$ for $j = 0, \ldots, \mathcal{M}_i$. Eq. (28) is solved using US and DS dynamic pressures of zero as boundary conditions. The calculated dynamic pressure, for both the fluid and solid particles, is then used in Eq. (16), i.e.,

$$\left(\frac{D\mathbf{v}_I}{Dt}\right)_i = -\frac{1}{\rho}\frac{d}{\mathcal{N}_i}\sum_j^{\mathbb{N}} \frac{(P_{dj} - P_{di})\cdot(\mathbf{r}_j - \mathbf{r}_i)}{r_{ij}^2} W_{ij} \tag{29}$$

where the summation is over all the fluid and solid particles. Having available $\mathbf{v}_E$ and $\mathbf{v}_I$ concludes the time step via Eq. (14).

The dynamic pressure calculated from Eq. (28) and used in Eq. (29) will ensure no penetration of fluid particles into the solid domain, as the possible penetrating fluid particles will adopt a dynamic pressure, according to the governing equations, that will automatically repel them back to the fluid domain.

3.5. Splitting and merging of fluid particles

Direct flow simulations in naturally-occurring porous rocks often means that fluid particles have to pass through very narrow conduits. If the fluid particles are not small enough, – i.e., model resolution is inadequate – then hydraulic conductivity of these narrow conduits can be lost leading to a permanent loss of fluid mass in them. This can, depending on the pore space topology, take place in numerous locations in a porous system, which could translate into inaccuracies in the predicted flow behavior and properties. One solution would be to reduce the size of the fluid particles in the entire system to avoid such problems. But, this can dramatically increase the computational load making it impractical to simulate flow through large enough, e.g., REV size, samples. To avoid the extra computational load, an alternative approach is to refine fluid particles *during* the flow simulations *only* in narrow conduits with very high fluid velocities. However, considering the complexity of the pore space in random porous materials, this has to be done automatically. Hong et al. [27] have proposed a particle refinement technique implemented with fluid-implicit-particle (FLIP) method. The authors have been able to successfully simulate surface flows using this technique. Reynolds number and the depth from the free surface were the criteria used by the authors to split and merge particles. These criteria, however, are specific to free surface flow problems. Here we introduce an alternative criterion to split (or merge) fluid particles. The goal is to split the fast moving particles in narrow flow channels and merge the slow moving particles in wide flow channels. The latter is done to control the computational cost by preventing the spread of the split particles to low velocity areas where high resolution is not needed. This leads to accuracies comparable to high-resolution simulations at much lower computational cost. It has to be stressed that, it is impractical to determine the narrow conduits as potential particle splitting sites *before* the flow simulation starts. This is because pressure gradient and complexities of the pore space and flow regime impact the flow through these conduits and their relative activity, i.e., by making them passive or active in conducting the flow, and hence they may or may not qualify as splitting sites. Therefore, they must be detected *during* the flow simulations.

From energy balance in a closed system we get:

$$v_{\max} = \sqrt{\frac{2\Delta P}{\rho} + 2g\Delta L}, \tag{30}$$

where ΔP is the pressure difference along the system, ΔL is the length of the system and g is the acceleration of gravity. We use $v_{\max}$, the maximum achievable velocity, to locally refine/coarsen the fluid particle resolution in a given system, i.e.,

$$\begin{cases} v > f_1 v_{\max} \;\rightarrow\; \text{Split}, \\ v < f_2 v_{\max} \;\rightarrow\; \text{Merge}, \end{cases} \tag{31}$$

where f_1 and f_2 are less than one and $f_2 < f_1$. A small f_1 leads to a high rate of splitting whereas for $f_1 = 1$ the rate of splitting is zero. On the other hand, the rate of merging is zero if we choose $f_2 = 0$. For the case of $f_2 = 1$, all the particles have to merge. Therefore, depending on the desired rate of splitting and merging, the proper values are chosen. In this work, we use 0.01 and 0.0001 for f_1 and f_2, respectively. When splitting/merging, particles must follow the same general rule for the intensive and extensive properties of a fluid. This rule is observed in Eq. (32) when i is split to a and b and also when a and b merge to form i. Note that the linear momentum is also conserved as $m_i\mathbf{v}_i = m_a\mathbf{v}_a + m_b\mathbf{v}_b$ holds true for both cases.

$$\text{Splitting}: \begin{cases} w_a = w_b = \frac{w_i}{2} \\ m_a = m_b = \frac{m_i}{2} \\ V_a = V_b = \frac{V_i}{2} \\ P_a = P_b = P_i \\ \mathbf{v}_a = \mathbf{v}_b = \mathbf{v}_i \\ \mathbf{r}_a = \mathbf{r}_i + 0.2D_p \\ \mathbf{r}_b = \mathbf{r}_i - 0.2D_p \end{cases}, \quad \text{Merging}: \begin{cases} w_i = w_a + w_b \\ m_i = m_a + m_b \\ V_i = V_a + V_b \\ P_i = \frac{(w_aP_a + w_bP_b)}{w_a + w_b} \\ \mathbf{v}_i = \frac{(w_a\mathbf{v}_a + w_b\mathbf{v}_b)}{w_a + w_b} \\ \mathbf{r}_i = \frac{(w_a\mathbf{r}_a + w_b\mathbf{r}_b)}{w_a + w_b} \end{cases}, \tag{32}$$

where V denotes volume and w is the weighting factor of a particle, which is 1 for the particles with the original resolution and indicates whether the particle still has the original size or it has been split/merged. It should be mentioned that position

of the split particles does not follow a certain rule, except that they are positioned within the close neighborhood of the original particle.

The fluid particles can split several times, as long as $N_w > f_w \mathbb{N}$ where $N_w = \sum_j^{\mathbb{N}} w_j$ and parameter f_w determines the extent of splitting for any particle that is eligible for splitting, which is always less than 1. In other words, it controls the fraction of the split particles in the neighborhood of particle i to restrict the rate of splitting and its associated computational cost. In this work, we use $f_w = 0.7$. This means that if particle i meets the splitting condition in Eq. (31), it can split only if more than 70% of the particles in its neighborhood are still untouched (have the original weighting factor).

Since after each splitting/merging event particles with different weighting factors are introduced, the particle summations have to be capable of incorporating them. We use a modified kernel that is found by multiplying Eq. (6) by the weighting factors of the contributing particles, i.e., $W_{ij} = W_{ij} w_i w_j$, which must be used in all the particle summations. The kernel size should also be modified to prevent unnecessary computational cost, i.e., $h_i = \frac{N_w}{N_n} h$. This will ensure the same numerical precision throughout the system.

3.6. Pressure solver

We use Bi-Conjugate Gradient STABilized (Bi-CGSTAB) [40] with diagonal preconditioning to solve the system of linear equations given by Eqs. (26) and (28). Bi-CGSTAB is an iterative method that approximates the solution of a large non-symmetric linear system by a vector in the Krylov subspace and therefore is appropriate for our system of equations.

The parallel Bi-CGSTAB, implemented in this work, utilizes the already available sparse matrix. Every point on the main diagonal of the matrix, resulting from Eq. (28), corresponds to a particle. Considering that only the neighbors of particle i contribute to this equation, the only nonzero elements in row i are the neighbors of particle i. Therefore, the required details of the sparse matrix are available through the neighbor search, which is carried out anyway.

In solving Eq. (28) for the REV size sample studied in Section 5, on average the solver converges to the specified tolerance 5×10^{-7} Pa^2 in 4 iterations while reducing the initial residual error by three orders of magnitude.

4. Validation

The predictions of the MMPS model were rigorously compared against three groups of data available in the literature: (1) the analytical solution of Poiseuille flow, (2) the finite element numerical solutions of flow in capillary tubes with various cross-sectional shapes in [41], and (3) the experimental data in [42] for flow through a packed bed of spheres, measured using MRI technique for different Reynolds numbers (12.17–204.74).

4.1. Poiseuille flow

We use MMPS to simulate flow of an incompressible Newtonian fluid between two parallel plates (Poiseuille flow). The fluid is water at ambient conditions with ρ = 997 Kg m^{-3} and μ = 0.001 Kg m^{-1} s^{-1}. We use a three-dimensional system with a high $\frac{L_x}{L_y}$ ratio to emulate two parallel plates with infinite depth in the x direction and flow in the z direction (i.e., perpendicular to $x-y$ plane). The dimensions of the system are $L_x = 0.226$ mm, $L_y = 0.034$ mm, and $L_z = 0.064$ mm. The analytical solution for the velocity of fluid in the the z direction is given by [37]:

$$v_z(y,t) = \frac{F}{2\nu} y(y - L_y) + \sum_{i=0}^{\infty} \frac{4FL_y^2}{\nu\pi^3(2i+1)^3} \sin\left(\frac{\pi y}{L_y}(2i+1)\right) \exp\left(-\frac{(2i+1)^2\pi^2\nu}{L_y^2} t\right) \tag{33}$$

where F is the driving force in ms^{-2}, ν is the kinematic viscosity in m^2 s^{-1}, L_y is the distance between the two parallel plates in m and t is time in s. We apply 1 Pa pressure difference across the length of the system resulting in $F = -\frac{\Delta P}{\rho L_z} = -15.672$ and a Reynolds number of $Re = \frac{V_0 L}{\nu} = 0.077$ where V_0 is the peak velocity obtained from our simulation and is equal to 2.26×10^{-3}. One should note that Eq. (33) gives the same peak velocity at $t \to \infty$.

In order to compare our results with the analytical solution, we take a $y-z$ slice of the system at $x = 0.5L_x$ to avoid the edge effects. We then plot an average velocity over the entire length in the z direction for all the particles at any given y. The particle spacing of 2 μm has allowed us to obtain an excellent agreement with the analytical solution, see Fig. 6. The steady-state is reached at $t_{ss} = 0.001$ s. Also, the dynamic pressure during the flow simulation is in the order of 1.0×10^{-3} Pa. Small dynamic pressure is expected in this simulation as the flow domain has straight boundaries with no complexities. This confirms the correctness of the boundary conditions explained in Section 3.4 for systems with simple fluid/solid boundaries.

4.2. Flow in capillary tubes

In this section, we test the capability of our model to predict single-phase hydraulic conductivities in three groups of capillary tubes with circular/elliptical, square/rectangular and triangular cross-sections. The results are compared against the finite element numerical solutions presented in [41]. The cross-sectional shape of capillary tubes in each group is varied using a shape factor defined as $G = \frac{A}{p^2}$ where A is the cross-sectional area and p is the perimeter of the capillary tube cross section.

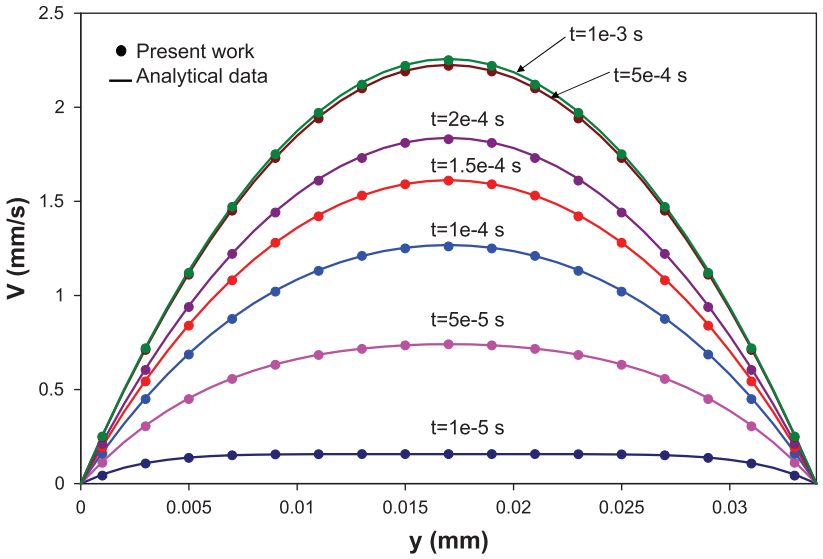


Fig. 6. Comparison of velocities predicted using MMPS model with a particle spacing of 2 μm against the analytical solution of Poiseuille flow given by Eq. (33).

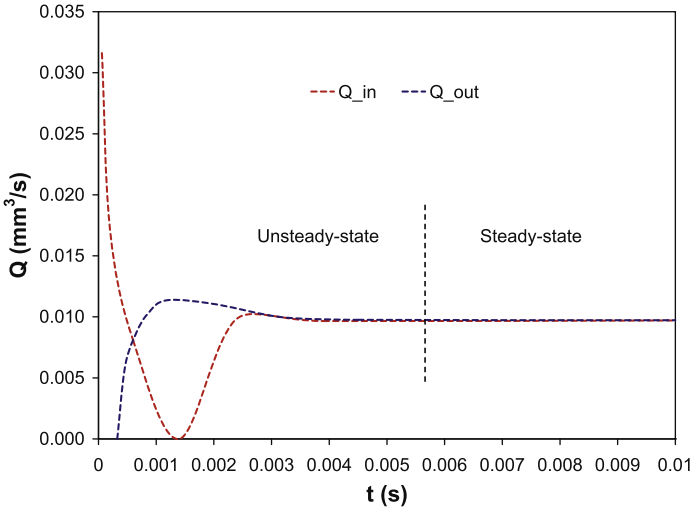


Fig. 7. Inlet and outlet flow rates versus time for flow in a capillary tube with square cross-section, i.e., $G = 0.0625$.

The simulations are carried out in capillary tubes with different shape factors, but we keep the inscribed radius in all cases constant, 14 μm. This is the average pore and throat inscribed radius in the Berea sandstone network used in [43]. We build a particle representation of each capillary tube, which is then read by our model as input. Dimensionless hydraulic conductance of a capillary tube is calculated using $\hat{g} = \frac{Q\mu L}{\Delta P A^2}$ where $\hat{g}$ is the dimensionless hydraulic conductance, Q is flow rate in $m^3\ s^{-1}$, μ is viscosity in $Kg\ m^{-1}\ s^{-1}$, L is length of the tube in m, and ΔP is pressure difference between the upstream and downstream sections in Pa. The fluid is water at 25 °C with $\rho = 997\ Kg\ m^{-3}$ and $\mu = 0.001\ Kg\ m^{-1}\ s^{-1}$.

We assume the system is initially at rest at atmospheric pressure in all cases. At $t = 0$ a step increase in the upstream pressure is introduced, while keeping the downstream pressure constant at atmospheric ($P = 0$). The inlet and outlet flow rates are equal at the steady-state which is reached after a short transient state, see Fig. 7. The steady-state flow rate is then used to calculate the dimensionless hydraulic conductance. In all the cases, length of the tube, particle spacing, and kernel size are 100, 2, and 8 μm, respectively. One should note that 100 μm length includes 8 μm of upstream section and 8 μm of downstream section.

Fig. 8 shows the comparison with the finite element results presented in [41]. We obtain a good agreement in all three groups of capillary tubes. This confirms the accuracy of our model in simulating three-dimensional single-phase flow in

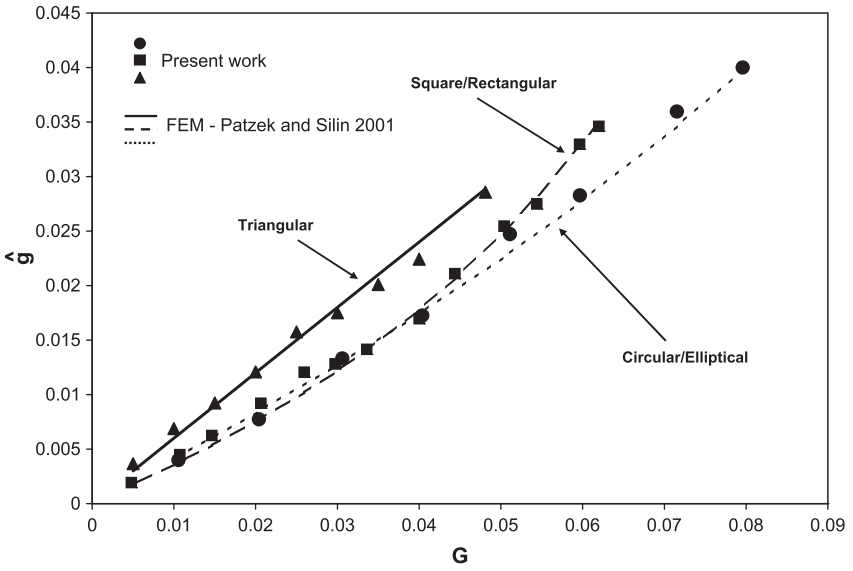


Fig. 8. Dimensionless hydraulic conductance versus shape factor for flow in capillary tubes with circular/elliptical, square/rectangular and triangular cross-sections.

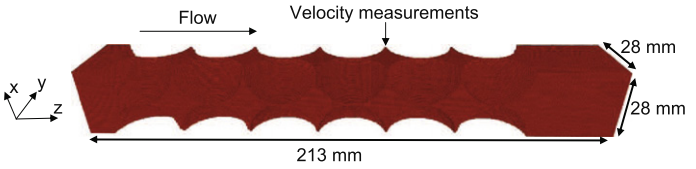


Fig. 9. Particle representation of the pore space in the packed bed of spheres studied in this work.

capillary tubes with various cross-sectional shapes. The fact that a constant steady-state flow rate is observed along the length of the tubes indicates that $\nabla \cdot \mathbf{v} = 0$ holds true for all the cases and hence the accuracy of the open boundaries as well as the fluid/solid boundaries.

4.3. Flow in a packed bed of spheres

To validate the accuracy of the model in simulating flow of incompressible fluids in systems with complex fluid/solid boundaries, here we compare our results against the experimental data in [42] for flow through a packed bed of spheres, measured using MRI technique for different Reynolds numbers (12.17–204.74). Shown in Fig. 9 is the pore space of the packed bed used in the experimental setup with porosity $\phi = 0.476$. We use a particle spacing of 0.5 mm and kernel size of 1.5 mm to simulate the flow in this system. The simulations are carried out for five cases of laminar flow corresponding to five different Reynolds numbers of 12.17, 28.88, 59.78, 105.57 and 204.74 studied in the experimental work. Figs. 10 and 11 show the comparison between the predicted normalized z-velocity $\left(\frac{V_z}{V_{\text{mean}}}\right)$ profiles along the x axis, at $z = 125$ mm and $y = 14$ mm, and their experimental counterparts for five different Reynolds numbers. The z-velocity is normalized with $V_{\text{mean}} = \frac{V_{z,\text{ave}}}{\phi}$. The Reynolds number used both here and in [42] is $Re = \frac{\rho V_{\text{mean}} d}{\mu}$ where $d = 28$ mm is diameter of the packed bed of spheres. We obtain an encouraging agreement for all cases, cf. [44]. The visualization of normalized velocity profiles at $z = 125$ mm are shown in Fig. 12 for the five cases studied here. As the Reynolds number increases a transition from creeping flow to inertial flow and then to unsteady laminar flow occurs.

The system of packed bed, studied here, does not need the use of our splitting/merging scheme as there are no narrow fluid channels. However, to test the reliability of our splitting/merging scheme discussed in Section 3.5 and its ability to conserve mass and momentum, we present the results of a simulation for Re = 59.78 with the threshold of splitting set at about 10%. This means that about 10% of the fluid particles would experience splitting. Fig. 11 compares the predicted values against the experimental data. The results are almost identical to those obtained without using splitting/merging scheme. This confirms the conservation of mass and momentum when the splitting/merging scheme is used. Also, to test the reliability of the model with a different particle resolution, we simulate the Re = 59.78 case with the initial particle spacing of 1 mm. As shown in Fig. 11, the simulation leads to a very similar results to both the experimental data and also the high-resolution simulation with $D_p = 0.5$ mm.

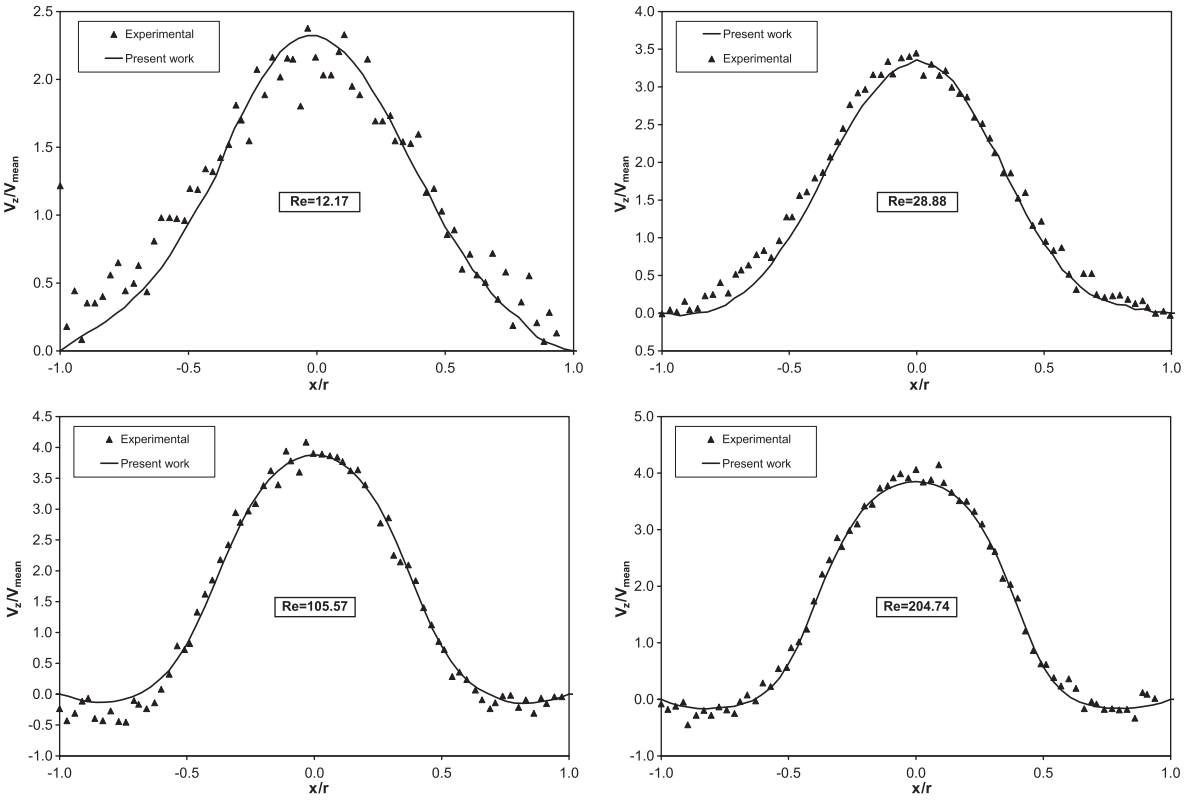


Fig. 10. Comparison of the predicted normalized velocities with their experimental counterparts at various Reynolds numbers for flow through a packed bed of spheres, $r = \frac{d}{2} = 14$ mm. Particle spacing (and resolution) is 0.5 mm.

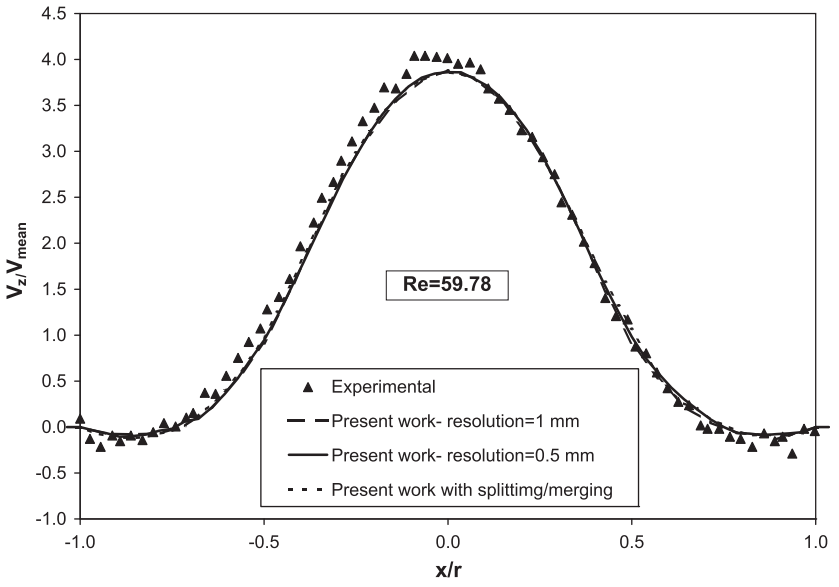


Fig. 11. Comparison of the predicted normalized velocities with their experimental counterparts at $Re = 59.78$ for flow through a packed bed of spheres, $r = \frac{d}{2} = 14$ mm.

5. Flow in a naturally-occurring sandstone

In this section, the validated model is further used to study the flow of an incompressible fluid in an REV size naturally-occurring sandstone. We study the flow of water at 25 °C through a reservoir sandstone, i.e., sandstone S_9 from [8]. This

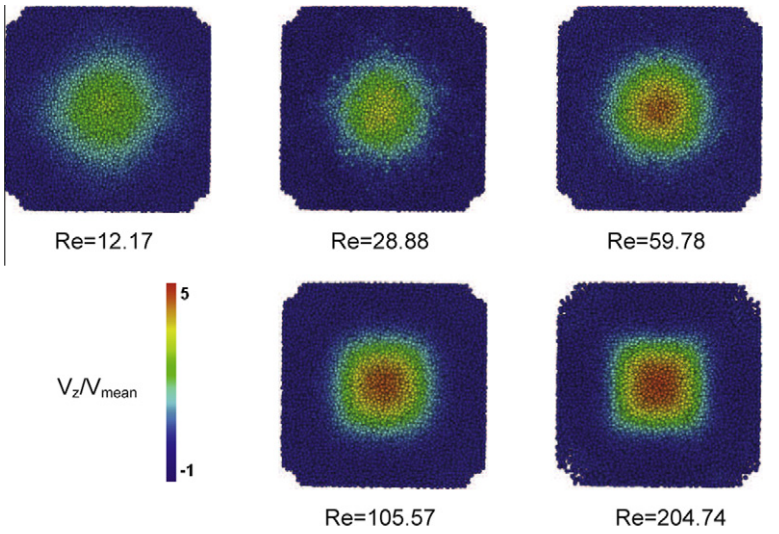


Fig. 12. Visualization of normalized velocity profile at $z = 125$ mm for different Reynolds numbers.

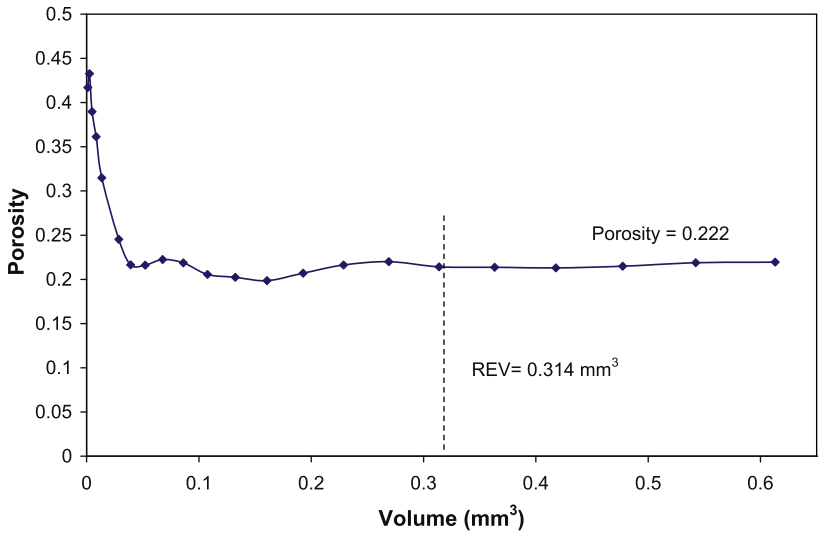


Fig. 13. Representative elementary volume (REV) of sandstone S_9.

sandstone has a 22.2% porosity with a Lattice-Boltzmann permeability of 2,224 mD and an average formation factor of 18.02. The original X-ray image includes $300 \times 300 \times 300$ voxels with 3.398 μm resolution. As shown in Fig. 13, for sample sizes above $200 \times 200 \times 200$ voxels and a resolution of 3.398 μm, porosity changes by less than 1%. Therefore, REV for this sandstone is about $0.6796 \times 0.6796 \times 0.6796$ mm ($V = 0.3138$ mm^3).

To lower the computational cost, we first coarsen the entire image by combining the adjacent voxels. An REV size sample is then chosen from the coarsened image. Also, as discussed in the penultimate paragraph of Section 3.4.1, a total of 95.144 μm of stabilizer is added to the system along the main flow direction to prevent the flow disturbances reach the US and DS sections, which is then augmented by a total of 40.776 μm for the US and DS sections. The resulting system is $0.6796 \times 0.6796 \times 0.81552$ mm, which corresponds to $100 \times 100 \times 120$ voxels with 6.796 μm resolution, see Fig. 14(a). Furthermore, as we discussed in Section 3.4.1, we form a shell of solid particles around the sample to create a no-flow boundary condition. The shell has the thickness of the kernel size, i.e., $h = 20.388$ μm, and covers the porous sample in all sides except where the upstream and downstream pressures are applied, i.e., open boundaries. This setup leads to 1,399,680 total number of particles.

At $t = 0$, a 5 Pa pressure increase is introduced to the upstream section while the downstream pressure is kept constant at 0. This pressure difference, which manifests itself in the static pressure profile shown in Fig. 14(a), develops quickly across

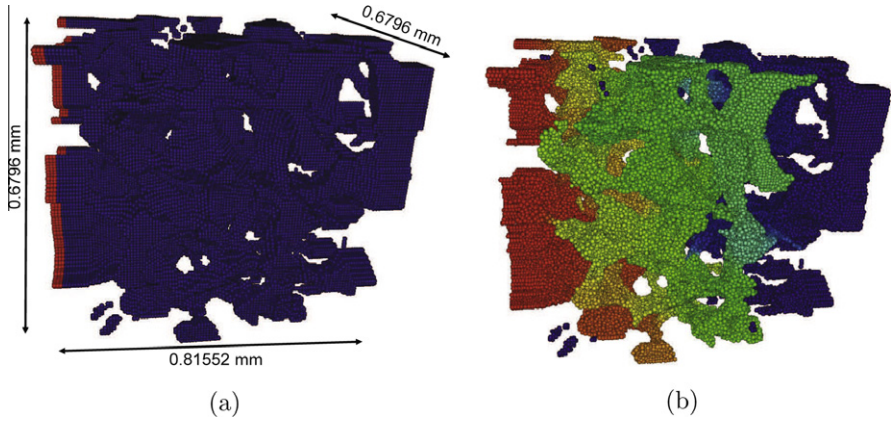


Fig. 14. 0 2.5 5 Static pressure profile for a flow simulation in sandstone sample S_9. The upstream and downstream sections are included in the visualizations: (a) $t = 0$, (b) $t > 0$.

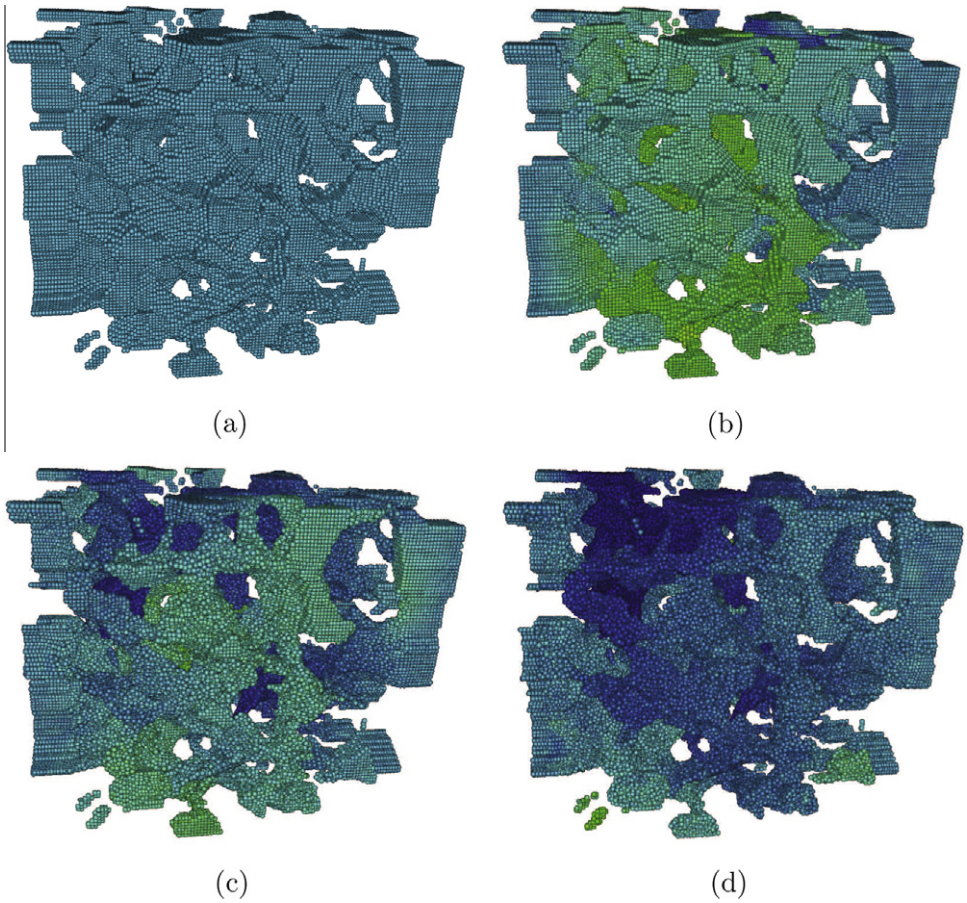
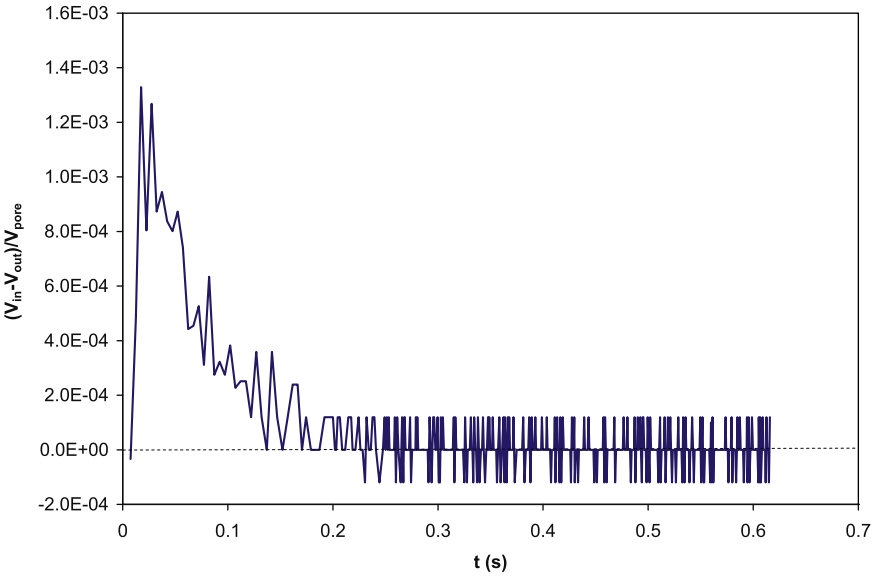
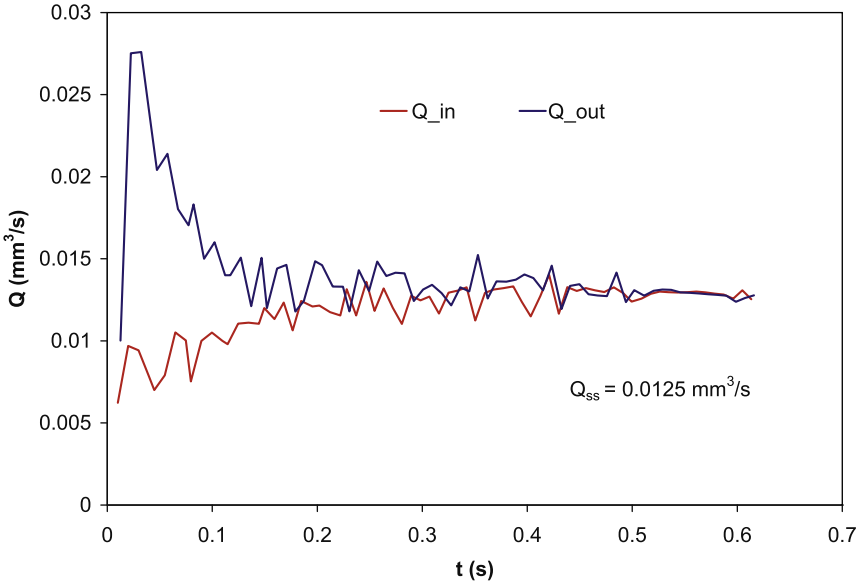


Fig. 15. -1.3 0.65 2.6 Dynamic pressure profile for a flow simulation in sandstone sample S_9. The upstream and downstream sections are included in the visualizations: (a) $t = 0$, (b) $t = 0.005$ s, (c) $t = 0.0492$ s, (d) $t = 0.4781$ s.

the system. Being independent from velocity, the static pressure *profile* does not change as the fluid particles move. The false connectivity rules discussed earlier keep the isolated pores, as seen in Fig. 14(b), in their initial condition throughout the simulation as they do not experience any of the applied pressure difference. The dynamic pressure profile, however, is a function of velocity and follows its transition to the steady-state seen in Fig. 15. Note the changes in the dynamic pressure of the isolated pores are due to the influence from the dynamic pressure of their surrounding solid particles through Eq. (28)



(a)



(b)

Fig. 16. (a) The normalized difference between the incoming and outgoing volume fluxes per total pore volume versus time in sample S_9. (b) Inlet and outlet flow rates for flow simulation in sample S_9.

and not the result of their movement. These dynamic pressures, however, are negligible and their impact are neutralized by the solid particles in the surrounding. Also, shown in Fig. 16(a) is the normalized difference between the fluid volumes entered and left the medium as the system is exposed to the step increase in the static pressure. A transient period of about 0.3 s is required by the system to stabilize and build up the dynamic pressure profile needed to compensate the 5 Pa step increase in P_s. During this transient (initialization) period, particle distribution in the system must adapt with the distribution of P_d which results in some particles leaving the boundaries of the domain.

The entire simulation took about 16.5 h on 216 processors (2.3 GHz each). Fig. 16(b) shows a flow rate of 1.25×10^{-2} mm^3 s^{-1} at steady-state which results in the absolute permeability of 4,250 mD. We, however, did not have a core sample of sandstone S_9 available to measure its permeability. But, our calculated permeability is comparable with those reported in [8] for the entire sample of $300 \times 300 \times 300$ voxels that are 2,220 mD using Lattice-Boltzmann and 3,640 mD using a

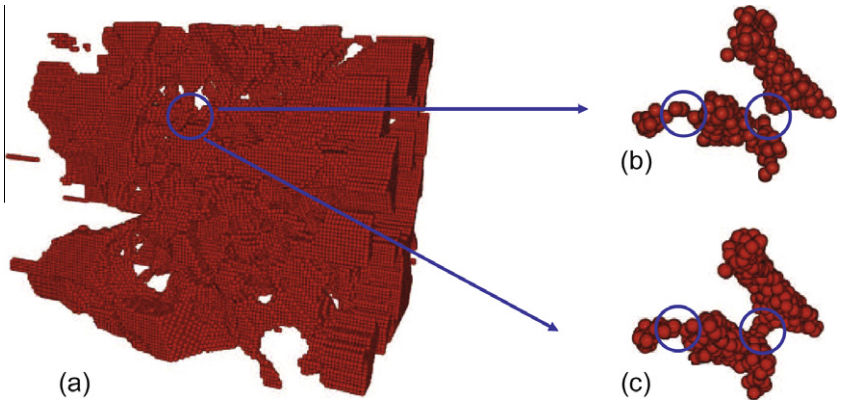


Fig. 17. Visualization of fluid particles of sample S_9. (a) The entire sample at $t = 0$. (b) Two narrow channels at $t = 0.1$ s without splitting and merging. (c) The same narrow channels at $t = 0.1$ s with splitting and merging; particles in the channels are smaller than the rest of fluid particles.

reconstructed pore network. One should note that the sample studied here is much smaller (about 3 times) than the sample used in the Lattice-Boltzmann and network modeling approaches mentioned above, which may have contributed to the difference in the predicted permeabilities. This is consistent with the trend of the permeabilities we obtained by running a similar simulation using a smaller size, i.e., $0.38 \times 0.38 \times 0.38$ mm, of the same sandstone which resulted in the permeability of 7,380 mD.

Also, in Fig. 17 one of the narrow throats in the above-mentioned sandstone is chosen to illustrate the impact of splitting and merging. Without splitting and merging, this narrow throat gets choked, as shown in Fig. 17(b), after 0.1 s. However, by splitting the fluid particles according to the rules described in Section 3.5, this channel stays connected throughout the entire simulation, see Fig. 17(c). It is important to note that splitting and merging of the fluid particles lead to more stable simulations, which in turn allows larger time steps and consequently lower computational costs. Simulations on a smaller size sample from the same sandstone with a higher resolution, i.e. 3.398 μm, did not show any sensitivity to splitting and merging. We believe the narrow throats at high resolution are represented by enough particles to preserve the connectivity.

6. Parallelization and scalability

Due to the nature of the particle-based methods and also because of the required resolution for flow in porous media problems, simulations in large samples, e.g., REV size, are computationally expensive. As it was mentioned earlier, for problems in astrophysics, parallel particle-based algorithms, mainly based on SPH, have already been developed and used successfully [17]. These parallel algorithms use tree search [36] for the neighbor search step, which is computationally the most demanding step. However, based on our experience, keeping the kernel size constant will generate reliable results for the fluid flow problems. Therefore, it is more suitable to use the linked-list algorithm [45] in these cases which has the complexity of $O(N)$. Consequently, as there are no parallel algorithms handy for MMPS simulations of fluid flows in porous media, we have developed a customized, efficient and scalable parallel scheme, described next.

The efficiency of the parallel code is dependent on partitioning, i.e., decomposing the medium into subvolumes and allocating them to different processes. A partitioning is considered efficient only if it leads to: (1) optimum load balance and (2) minimum communication between the processes. Since the computational load depends on the number of particles in each subvolume, we divide the system into parts with equal subvolumes to fulfil the load balance condition for the porous systems larger than REV. For the systems smaller than REV, the solid particles may not be distributed uniformly among the processes. Therefore, since the computational work for solid particles is less compared to the fluid particles, a non-uniform distribution of the computational load is expected for these systems. The second requirement of an efficient partitioning is a minimum communication between processes. By defining *particle reallocation* as an operation that performs the transfer of particles from one process to another, major communication points in parallel MMPS are those related to calculation of the pressure fields, particle reallocation, communication for neighbor search and writing in the output files. We minimize these communications by overlapping them with the computations.

In Fig. 18 the time scales for different computation and communication operations in one average step of MMPS are shown. The results belong to a flow simulation in the same sample studied in Section 5. The simulation was run on 48 processors with $4 \times 3 \times 4$ partitioning scheme in x, y, and z directions and with 1 Pa pressure difference. The amount of time for each operation was recorded in each time step, and the recorded times were averaged over 1,000 time steps. In descending order, neighbor search, calculation of V_I, detection of false connectivity, communication for neighbor search and calculation of V_E are the most time-consuming operations in each time step. Furthermore, the rate of particle reallocation is not significant regardless of the way we partition the porous system. Overall, the effect of communication on each time step is not significant and the emphasis should be given to load balancing when partitioning the system. Recalling from Section 3.2,

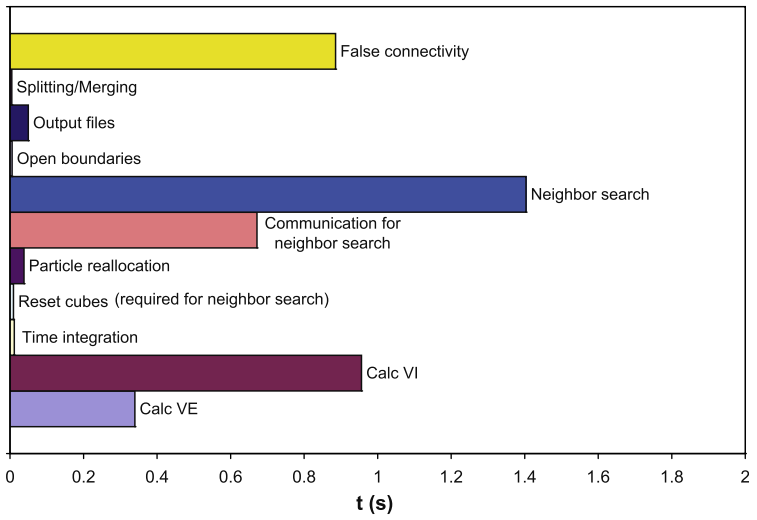


Fig. 18. Average time scales in one time step of a flow simulation in sample S_9. The simulation was performed on 48 processors with the $4 \times 3 \times 4$ partitioning scheme and the time scales are averaged over 1,000 time steps.

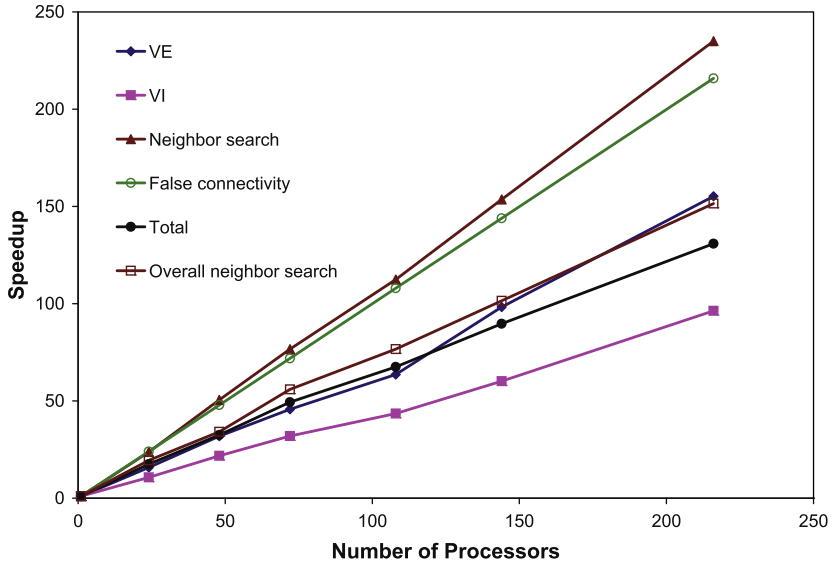


Fig. 19. Scalability of the most time-consuming operations in each time step of MMPS. The simulations were performed in sample S_9.

we know the calculation of P_s and P_d are the main part of V_E and V_I calculations, respectively. Knowing that the pressure calculations and also the neighbor search are only dependent on the number of particles in each processor, the optimum partitioning for this system is when it is partitioned to equal subvolumes along all x, y and z axes.

6.1. Scalability

In this section, we present a comprehensive study of the scalability of the computationally expensive operations discussed earlier. Speedup is defined as [46]:

$$\text{Speedup} = \frac{\text{time (1 process)}}{\text{time (n processes)}}, \tag{34}$$

where 'time' denotes the computational time on 1 or n processors.

Here we first address the scalability of the most time-consuming operations of the code in each time step, namely the calculation of V_E, V_I, neighbor search and detection of false connectivities. To do this, flow simulations are performed in

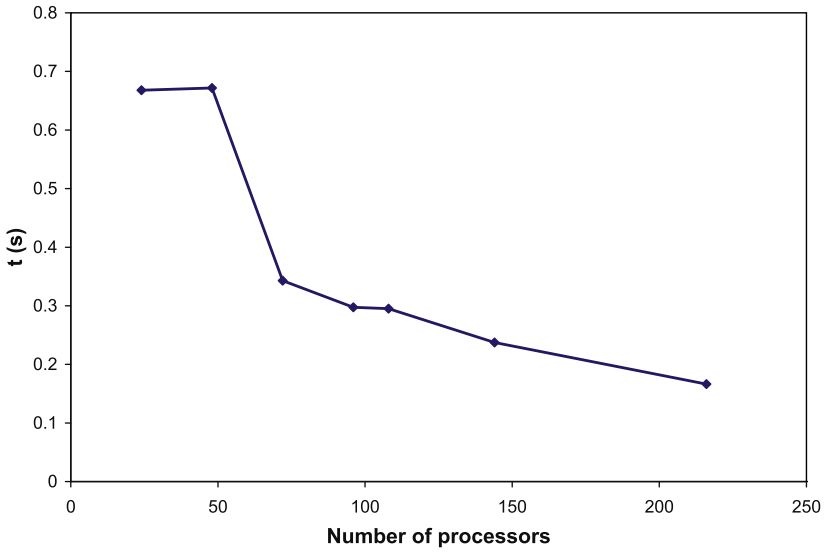


Fig. 20. Communication time for receiving/sending information to the neighboring processes in the neighbor search step versus the number of processors. The simulations were performed in sample S_9.

the sample studied in Section 5 with the pressure difference of 1 Pa and the stagnant initial conditions. For each simulation with a given number of processors, the amount of time taken by each operation in a time step was recorded and then averaged over 1,000 time steps. The results are shown in Fig. 19.

The V_E and V_I calculations are dominated by the static and dynamic pressure calculations, respectively. Each of these pressure calculations involves solving a system of linear equations. Consequently, the more uniform is the distribution of the solid/fluid particles, the more load balanced is the system, which leads to a higher scalability. Besides load balance, each iteration in the Bi-CGSTAB solver requires four global reduction operations which contributes to its low scalability due to the excessive synchronization of the processes. This should be the focus of the future work as in [47]. The overall performance of the pressure calculations and hence the explicit and implicit velocity calculations is therefore controlled by the solver. Despite this, due to the optimizations performed on the code, calculation of both V_E and V_I, and hence P_s and P_d, achieve almost linear scalabilities but with lower performances than that of neighbor search. The calculation of V_E is more scalable because the calculation of P_s is computationally straightforward and in fewer iterations of the solver the residual would be in an acceptable range (5×10^{-7} Pa^2). However, P_d calculations are more difficult and require more iterations which is why the performance is lower than all the other operations. Both the neighbor search and detection of false connectivities, which is integrated in the neighbor search step, show a high performance with linear scalability. Note that both of these operations are pure computation. The time spent on communication for neighbor search is shown in Fig. 20 for different number of processors. The plot shows a declining trend with increase in the number of processors. Due to the impact of communication time on the overall neighbor search operation, the performance is lowered but still the overall neighbor search is highly scalable (linear scalability). Overall, the code exhibits an encouraging speedup of 131 on 216 processors for the REV size sample studied here.

7. Conclusions

We presented a direct technique for particle-based pore-level modeling of incompressible fluid flow in disordered porous media that can be used as a platform to study flow and transport in various random porous material. The model combines five critical elements: (1) the most faithful description of the pore space in porous medium, (2) solutions of incompressible Navier–Stokes equations for each particle, (3) particle size adaptivity, (4) realistic fluid properties, and (5) scalable parallelization to obtain both unsteady- and steady-state solutions.

The moving particle semi-implicit method was modified (MMPS) through calculation of dynamic pressure directly from the continuity equation for incompressible fluids, which resulted in more stable simulations. Also, new techniques were developed to automatically split and merge fluid particles to protect flow through very narrow conduits, and therefore pave the way for more accurate and less expensive particle-based simulations. Novel procedures were presented for handling open (upstream and downstream) and fluid/solid boundaries as well as detection of unrelated fluid particles (false connectivity). A scalable parallel version of MMPS was implemented to allow flow simulations in REV size samples. The model was successfully validated against experimental, numerical, and analytical data available in the literature. The validated model was then used to directly simulate the flow of an incompressible fluid through an REV size reservoir sandstone to obtain unsteady- and steady-state flow behaviors and compute permeability.

Acknowledgements

We gratefully acknowledge financial support of EnCana and The School of Energy Resources and the Enhanced Oil Recovery Institute at the University of Wyoming. We also thank P.K. Smolarkiewicz, National Center for Atmospheric Research, Boulder, CO, USA; and anonymous reviewers for their scientific insight and the valuable comments. Hu Dong (Numerical Rocks) is gratefully thanked for sharing his micro-CT data with us.

References

[1] R. Lenormand, C. Zarcone, A. Sarr, Mechanisms of the displacement of one fluid by another in a network of capillary ducts, J. Fluid Mech. 135 (1983) 337–353.
[2] P.E. Øren, W.V. Pinczewski, The effect of wettability and spreading coefficients on recovery of waterflood residual oil by immiscible gasflooding, SPE Formation Eval. 9 (2) (1994) 149–156.
[3] D.P. Lymberopoulos, A.C. Payatakes, Derivation of topological, geometrical, and correlational properties of porous media from pore-chart analysis of serial section data, J. Colloid Interf. Sci. 150 (1) (1992) 61–80.
[4] J.T. Fredrich, B. Menendez, T.-F. Wong, Imaging the pore structure of geomaterials, Science 268 (1995) 276–279.
[5] P.M. Adler, J.-F. Thovert, Real porous media: local geometry and macroscopic properties, Appl. Mech. Rev. 51 (1998) 537–585.
[6] P.E. Øren, S. Bakke, Process based reconstruction of sandstones and prediction of transport properties, Transport Porous Med. 46 (2–3) (2002) 311–343.
[7] J. Coenen, E. Tchouparova, X. Jing, Measurement parameters and resolution aspects of micro X-ray tomography for advanced core analysis, in: Proceedings of the International Symposium of the Society of Core Analysts, Abu Dhabi, UAE, 2004.
[8] Hu Dong, Micro-CT Imaging and Pore Network Extraction, Ph.D. Dissertation, Department of Earth Science and Engineering, Imperial College, London, 2007.
[9] M.J. Blunt, M.D. Jackson, M. Piri, P.H. Valvatne, Detailed physics, predictive capabilities and macroscopic consequences for pore-network models of multiphase flow, Adv. Water Resour. 25 (8–12) (2002) 1069–1089.
[10] P.E. Øren, S. Bakke, O.J. Arntzen, Extending predictive capabilities to network models, SPE J. 3 (4) (1998) 324–336.
[11] T.W. Patzek, Verification of a complete pore network simulator of drainage and imbibition, SPE J. 6 (2) (2001) 144–156.
[12] P.H. Valvatne, M. Piri, X. Lopez, M.J. Blunt, Predictive pore-scale modeling of single and multiphase flow, Transport Porous Med. 58 (1–2) (2005) 23–41.
[13] M. Piri, M.J. Blunt, Three-dimensional mixed-wet random pore-scale network modeling of two- and three-phase flow in porous media. I. Model description, Phys. Rev. E 71 (026301) (2005), doi:10.1103/PhysRevE.71.026301.
[14] V.S. Suicmez, M. Piri, M.J. Blunt, Effects of wettability and pore-level displacement on hydrocarbon trapping, Adv. Water Resour. 31 (2008) 503512.
[15] W. Fourie, R. Said, P. Young, D.L. Barnes, The simulation of pore scale fluid flow with real world geometries obtained from X-ray computed tomography, in: Proceedings of the COMSOL Conference, Boston, 2007.
[16] P.K. Smolarkiewicz, L. Winter, Pores resolving simulation of Darcy flows, J. Comput. Phys. 229 (2010) 31213133.
[17] G.R. Liu, M.B. Liu, Smoothed Particle Hydrodynamics: A Meshfree Particle Method, World Scientific Publishing Co. Pte. Ltd., 2005.
[18] T. Inamuro, T. Ogata, S. Tajima, N. Konishi, A Lattice Boltzmann method for incompressible two-phase flows with large density differences, J. Comput. Phys. 198 (2004) 628644.
[19] H. Li, C. Pan, C.T. Miller, Pore-scale investigation of viscous coupling effects for two-phase flow in porous media, Phys. Rev. E 72 (2005) 026705.
[20] Yi Zhu, Patrick J. Fox, Joseph P. Morris, A pore-scale numerical model for flow through porous media, Int. J. Numer. Anal. Meth. Geomech. 23 (1999) 881–904.
[21] A.M. Tartakovsky, P. Meakin, A smoothed particle hydrodynamics for miscible flow in three-dimensional fractures and two-dimensional Rayleigh–Taylor instability, J. Comput. Phys. 207 (2005) 610–624.
[22] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Mon. Not. R. Astron. Soc. 181 (1997) 375–389.
[23] S. Koshizuka, H. Tamako, Y. Oka, A particle method for incompressible viscous flow with fluid fragmentation, Comput. Fluid Dyn. J. 4 (1) (1995) 29–46.
[24] Simon Premože, Tolga Tasdizen, James Bigler, Aaron Lefohn, Ross Whitaker, Particle-based simulation of fluids, Comput. Graph. Forum 22 (2003) 3.
[25] F.H. Harlow, The particle-in-cell method for numerical solution of problems in fluid dynamics, Proc. Symp. Appl. Math. (1963).
[26] J.U. Brackbill, H.M. Ruppel, FLIP: a method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions, J. Comput. Phys. 65 (1986) 314–343.
[27] W. Hong, D.H. House, J. Keyser, Adaptive particles for incompressible fluid simulation, Vis. Comput. 24 (2008) 535–543.
[28] J. Salles, J.F. Thovert, P.M. Adler, Reconsttructed porous media and their application to fluid flow and solute transport, J. Contam. Hydrol. 13 (1993) 3–22.
[29] R. Peyret, T.D. Taylor, Computational Methods for Fluid Flow, Springer-Verlag, 1983.
[30] A.M. Tartakovsky, P. Meakin, Pore scale modeling of immiscible and miscible fluid flows using smoothed particle hydrodynamics, Adv. Water Resour. 29 (2006) 1464–1478.
[31] C.H. Arns, M.A. Knackstedt, W.V. Pinczewski, E.J. Garboczi, Computation of linear elastic properties from microtomographic images: methodology and agreement between theory and experiment, Geophysics 67 (2002) 1396–1405.
[32] C.H. Arns, M.A. Knackstedt, W.V. Pinczewski, N.S. Martys, Virtual permeametry on microtomographic images, J. Petrol. Sci. Eng. 45 (2004) 41–46.
[33] P.-E. Øren, S. Bakke, R. Held, Direct pore-scale computation of material and transport properties for North Sea reservoir rocks, Water Resour. Res. 43 (2007) W12S04.
[34] J.T. Fredrich, A.A. DiGiovanni, D.R. Noble, Predicting macroscopic transport properties using microscopic image data, J. Geophys. Res. 111 (2006) B03201.
[35] Y. Zhu, P.J. Fox, Simulation of pore-scale dispersion in periodic porous media using smoothed particle hydrodynamics, J. Comput. Phys. 182 (2002) 622–645.
[36] L. Hernquist, N. Katz, TreeSPH – a unification of SPH with the hierarchcal tree method, Astrophys. J. Suppl. Ser. 70 (1989) 419–446.
[37] J.P. Morris, P.J. Fox, Y. Zhu, Modeling low Reynolds number incompressible flows using SPH, J. Comput. Phys. 136 (1997) 214–226.
[38] P.W. Randles, L.D. Libersky, Smoothed particle hydrodynamics some recent improvements and applications, Comput. Meth. Appl. Mech. Eng. 139 (1996) 375–408.
[39] T. Belytschko, Y. Krongauz, J. Dolbow, C. Geralch, On the completeness of meshfree particle methods, Int. J. Numer. Meth. Eng. 4395 (1998) 785–819.
[40] H.A. van der Vorst, Efficient and reliable iterative methods for linear systems, J. Comput. Appl. Math. 149 (1) (2002) 251–265.
[41] T.W. Patzek, D.B. Silin, Shape factor and hydraulic conductance in noncircular capillaries. I. One-phase creeping flow, J. Colloid Interf. Sci. 236 (2001) 295–304.
[42] T. Suekane, Y. Yokouchi, S. Hirai, Inertial flow structures in a simple-packed bed of spheres, AIChE J. 49 (2003) 10–17.
[43] M. Piri, M.J. Blunt, Three-dimensional mixed-wet random pore-scale network modeling of two- and three-phase flow in porous media. II. Results, Phys. Rev. E 71 (2005) 026302.
[44] P.R. Gunjal, V.V. Ranade, R.V. Chaudhari, Computational study of a single-phase flow in packed beds of spheres, AIChE J. 51 (2005) 365–378.
[45] J.J. Monaghan, Particle methods for hydrodynamics, Comput. Phys. Rep. 3 (2) (1985) 71–124.

[46] D.E. Culler, J.P. Singh, A. Gupta, Parallel Computer Architecture: A Hardware/Software Approach, Morgan Kaufman Publishers Inc., San Francisco, 1999.
[47] L.T. Yang, R.P. Brent, The Improved BiCGSTAB method for large and sparse unsymmetric linear systems on parallel distributed memory architectures, in: Proceedings of the 5th International Conference on Algorithms and Architectures for Parallel Processing, ICA3PP'02, 2002.