

Local Capillary Pressure Estimation Based on Curvature of the Fluid Interface – Validation with Two-Phase Direct Numerical Simulations

Takashi Akai^{1,*}, Branko Bijeljic¹, and Martin Blunt¹

¹Imperial College London, Department of Earth Science and Engineering, London, SW7 2AZ, United Kingdom

Abstract. With the advancement of high-resolution three-dimensional X-ray imaging, it is now possible to directly calculate the curvature of the interface of two phases extracted from segmented CT images during two-phase flow experiments to derive capillary pressure. However, there is an inherent difficulty of this image-based curvature measurement: the use of voxelized image data for the calculation of curvature can cause significant errors. To address this, we first perform two-phase direct numerical simulations to obtain the oil and water phase distribution, the exact location of the interface, and local fluid pressure. We then investigate a method to compute curvature on the oil/water interface. The interface is defined in two ways. In one case the simulated interface which has a sub-resolution smoothness is used, while the other is a smoothed interface which is extracted from synthetic segmented data based on the simulated phase distribution. Computed mean curvature on these surfaces are compared with that obtained from the fluid pressure computed directly in the simulation. We discuss the accuracy of image-based curvature measurements for the calculation of capillary pressure and propose the best way to extract an accurate curvature measurement, quantifying the likely uncertainties.

1 Introduction

Capillary pressure, P_c , is a pressure discontinuity across the interface between oil and water, defined as $P_c = P_o - P_w$, where P_o and P_w are the pressures of oil and water phase, respectively. Traditionally, capillary pressure for oil/water systems has been measured in a laboratory using the porous plate method in which the pressure of each phase is measured using two external pressure transducers. Based on the Young-Laplace equation, capillary pressure locally is defined as:

$$P_c = 2\sigma\kappa_m, \quad (1)$$

where σ is the interfacial tension between two phases and κ_m is the mean curvature of the interface.

With the advancement of high-resolution three-dimensional X-ray imaging, it is now possible to directly measure the curvature of the interface extracted from segmented CT images during two-phase flow experiments to derive capillary pressure. Armstrong et al. [1] demonstrated this approach using synchrotron-based tomographic datasets of oil/water drainage and imbibition cycles on a bead pack structure [2]. They compared the capillary pressure obtained from curvature measurements with that obtained from pressure transducers. Fairly good agreement was obtained for imbibition, whereas the curvature measurement showed a systematically lower value than that obtained from the transducers for drainage cycles. Later, using the same dataset, Li et al. [3] presented that their proposed curvature measurement

method improved the agreement with the transducer based capillary pressure. Using a similar curvature measurement method, Herring et al. [4] estimated the capillary pressure for a range of curvature between 0 and 0.225 voxel⁻¹ based on their air/water drainage and imbibition experiments on a Bentheimer sandstone. However, there is an inherent difficulty of this image-based curvature measurement: the use of voxelized image data for the calculation of curvature can cause significant errors, resulting in a wide range of measured values, with some negative curvature values, which are not expected in a water-wet system. Hence, it is not clear how the distribution of measured curvature values represents the true range of local capillary pressure.

We investigate the accuracy of curvature measurement on the basis of pore-by-pore comparison using direct numerical simulations of two-phase flow. The color-gradient lattice Boltzmann method is employed to generate an oil and water phase distribution in pore structures of a bead pack and Bentheimer sandstone. From the simulated phase distribution, synthetic segmented data is generated, then curvature computation on the interface extracted from this segmented data is performed by employing several smoothing methods. These curvature values are then compared with those obtained from the simulated local fluid pressure. We discuss the accuracy of image-based curvature measurements for the calculation of capillary pressure and suggest the best method with associated errors for capillary pressure estimation.

* Corresponding author: t.akai17@imperial.ac.uk

2 Methods

2.1 The color-gradient lattice Boltzmann method

The color-gradient LB model proposed by Halliday et al. [5] was used. Our LB model was constructed with a 3D19Q lattice model which consists of a set of 19 discrete lattice velocity vectors, \mathbf{e}_i , in three-dimensional space. We defined particle distributions of two immiscible fluids, labeled red and blue, as f_i^r and f_i^b , respectively. The fluid density, ρ_r and ρ_b , and velocity, \mathbf{u} , at position \mathbf{x} and time t are obtained by:

$$\rho^k = \sum_i f_i^k(\mathbf{x}, t), \quad (2)$$

$$\rho \mathbf{u} = \sum_i f_i \mathbf{e}_i, \quad (3)$$

where f_i is the total particle distribution given by $f_i = f_i^r + f_i^b$; ρ is the total fluid density given by $\rho = \rho_r + \rho_b$. The lattice Boltzmann equation for the total particle distribution is written as:

$$f_i(\mathbf{x} + \mathbf{e}_i \delta t, t + \delta t) = f_i(\mathbf{x}, t) + \Omega_i(\mathbf{x}, t) + \phi_i, \quad (4)$$

where t denotes the lattice time step which was set to unity and Ω_i and ϕ_i are the collision operator and the body force term, respectively. For the collision operator, we used the Multiple-relaxation-time (MRT) collision operator [6] expressed as:

$$\Omega_i = -(\mathbf{M}^{-1} \mathbf{S} \mathbf{M})_{i,j} (f_j - f_j^{eq}), \quad (5)$$

where \mathbf{M} and \mathbf{S} are the transformation matrix and the diagonal relaxation matrix, respectively. f_i^{eq} is the equilibrium distribution function which is obtained by a second order Taylor expansion of the Maxwell-Boltzmann distribution with respect to the local fluid velocity. The location of the interface was tracked using a color function ρ^N defined by:

$$\rho^N(\mathbf{x}, t) = \frac{\rho_r(\mathbf{x}, t) - \rho_b(\mathbf{x}, t)}{\rho_r(\mathbf{x}, t) + \rho_b(\mathbf{x}, t)}, \quad -1 \leq \rho^N \leq 1. \quad (6)$$

Using the color function, the interfacial tension between two fluids was computed as a spatially varying body force, \mathbf{F} , based on the continuum surface force (CSF) model [7] given by:

$$\mathbf{F} = -\frac{1}{2} \sigma \kappa \nabla \rho^N, \quad (7)$$

where σ is the interfacial tension and κ is the curvature of the interface. This spatially varying body force \mathbf{F} was incorporated into the lattice Boltzmann equation through the body force term ϕ_i . For the MRT model, this was performed by transforming the forcing term proposed by Guo et al. [8] using the scheme presented in Yu and Fan [9]. After the application of the interfacial tension (\mathbf{F}) to the particle distributions, the recoloring algorithm proposed by Latva-Kokko and Rothman [10] applied to these distributions to ensure phase segregation and

maintain the interface. This results in a slightly diffusive interface whose thickness is about 2 to 3 lattice units. Further details of our LB model can be found in Akai et al. [11, 12]. The only difference is that we used the MRT collision operator, while the single-relaxation-time (SRT) collision operator [13] was used in Akai et al. [11].

At solid-fluid boundary lattice nodes, a full-way bounce back boundary condition was implemented to achieve a non-slip boundary condition. In addition, the wettability of solid surface was modeled by specifying contact angles using the wetting boundary condition presented in Akai et al. [11]. This wetting boundary condition accurately assigns contact angles for 3D arbitrary geometries with smaller spurious currents compared to the widely-used fictitious density boundary condition [11].

2.2 Curvature computation on voxelized images

There are mainly two approaches to compute curvature from voxelized data such as micro-CT images. One approach calculates curvatures from the gradient of 3D float data (e.g., raw or processed grayscale data), while the other approach estimates curvature through the fitting of a quadratic form locally to a surface extracted from voxelized, segmented data [14]. We used the latter approach as this has been presented in previous studies [14, 1, 15, 4, 3, 16].

We started with three-phase segmented label data (oil, water and solid) obtained from raw grayscale CT images. Using the marching cubes algorithm [17], the oil/water interface was extracted from the label data. Since this surface had a staircase shape, it had to be smoothed before computing curvature. In this study, we compare three smoothing methods: Constrained Gaussian smoothing (CGS), Laplacian smoothing (LPS) and boundary preserving Gaussian smoothing (BPGS). CGS is applied when the surface is extracted using the marching cubes algorithm. A Gaussian kernel filter with different kernel sizes is applied to label data, then isosurface is extracted. In this process, the constraint to preserve the location of an original label is imposed.

LPS [18] and BPGS [19] are applied after the extraction of the surface with the marching cubes algorithm. The extracted surface with a staircase shape is modeled as a triangulated surface. Then, the vertices of the triangle elements are moved with these smoothing methods. In LPS, the position of a vertex is moved to the average position of its neighboring vertices. BPGS also moves the position of a vertex based on the position of its neighboring vertices. A scale factor which determines the degree of the movement in one iteration is defined. Two consecutive smoothing steps with a positive and negative scaling factor are performed in one iteration. This smoothing produces a surface which preserves the original boundary without shrinkage [19]. The degree of the smoothing in LPS and BPGS is controlled by the number of iterations. In this study, CGS and BPGS were performed with commercial image processing software, Avizo, while LPS was performed with Paraview.

After the generation of a smoothed triangulated surface, the elemental triangles were fitted by a quadratic form:

$$ax^2 + by^2 + cz^2 + 2exy + 2fyz + 2gzx + 2lx + 2my + 2nz + d = 0. \quad (8)$$

Then, the principal curvature values and directions of principal curvature were obtained from the eigenvalues and eigenvectors of the fitted quadratic form in Eq. 8 [1]. Avizo was used to perform this computation. Users can choose the number of neighboring triangles to be used for the fitting at the center of a triangle. We used a fixed value of 4 neighbors in the following analyses.

3 Results

In section 3.1, we provide validation of our direct numerical simulations for different grid resolutions. Next, in section 3.2, using a single oil droplet test case, we investigate the accuracy of the curvature computation method described in section 2.2. Here, a smoothed interface which was extracted from synthetic segmented data based on the simulated phase distribution is used to compute curvature, then the resultant values are compared with those obtained from the simulated fluid pressure. Finally, in section 3.3, the same approach is applied to the simulated phase distributions in two realistically complex pore structures of a bead pack and Bentheimer sandstone.

3.1 Validation of the two-phase lattice Boltzmann model

To validate our two-phase lattice Boltzmann model and investigate its accuracy as a function of grid resolutions, the oil/water interface in a corner of triangular pore space was simulated. A 2D pore structure with an isosceles triangle shape as shown in Fig. 1 was used. Here, the length of L_x and L_y were set to 70 μm . R is the radius of curvature of the interface; θ is the contact angle; β is the half angle of the corner of the triangle, which is given by: $\tan\beta = L_x/2L_y$. This pore structure was modeled with 4 grid sizes of $\Delta = 1.0, 2.0, 3.5$ and $5.0 \mu\text{m}$.

The identical density and viscosity of the water and oil phases were set to 1,000 kg/m^3 and 1 mPa, respectively. The interfacial tension and contact angle were set to 18 mN/m and 45° , respectively. Initially, the lower part of the pore structure was filled with oil to a specified oil saturation, while the other part at the top corner was filled with water. Then, simulations were performed for 50,000 time steps until they reached equilibrium.

In this pore geometry, the radius of curvature, R , can be analytically derived based on the geometrical relationship, which is given by:

$$R^2 = \frac{S_w L_x^2}{4 \tan \beta} \left(\frac{\cos^2 \alpha \cos \beta}{\sin \beta} + \sin \alpha \cos \alpha - \left(\frac{\pi}{2} - \alpha \right) \right)^{-1} \quad (9)$$

$$\alpha = \theta + \beta,$$

where S_w is the water saturation and the other parameters are shown in Fig 1. This analytically derived radius of curvature was used to compare with the simulated radius of curvature obtained using the input contact angle of $\theta = 45^\circ$. Based on the simulated fluid configurations, the exact location of the interface corresponding to $\rho^N = 0$ (see Eq. 6) was extracted as shown in Fig. 2. Then, the radius of the curvature of this interface was obtained by fitting a circle to the interface, since in 2D in equilibrium the analytical shape of the interface is a part of a circle. Table 1 shows the comparison between the radius of curvature obtained from the analytical solution and that obtained with a circle fit to the simulated interface for different grid sizes. The relative error in the radius of curvature is less than 3% for these grid resolutions. This suggests that good agreement in capillary pressure is also obtained since the capillary pressure is directly linked to the radius of curvature through Eq. 1.

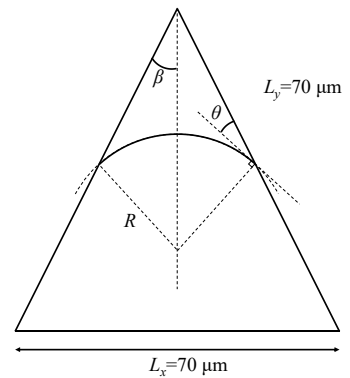


Fig. 1. A schematic of the isosceles triangle pore used for the simulations.

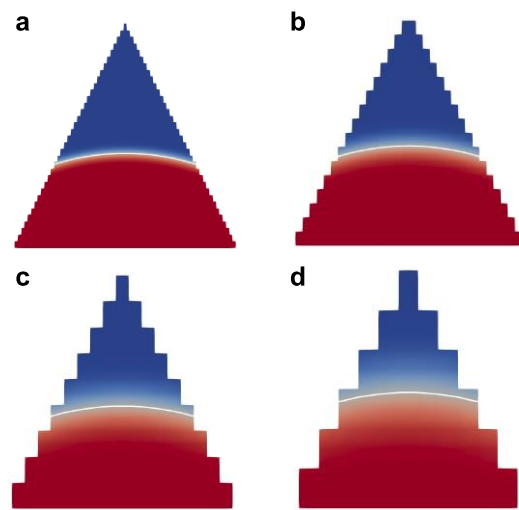


Fig. 2. Simulated fluid configurations for grid sizes of (a) $\Delta = 1.0 \mu\text{m}$, (b) $\Delta = 2.0 \mu\text{m}$, (c) $\Delta = 3.5 \mu\text{m}$ and (d) $\Delta = 5.0 \mu\text{m}$. Here, oil and water are shown in red and blue, while the exact location of the oil/water interface is shown by the white line.

Table 1. Comparison between the radius of curvature obtained from the analytical solution and that obtained with a circle fit to the simulated interface for different grid sizes.

grid size μm	S_w %	analytical	simulated	%Diff. %
		R μm	R μm	
1.0	35.9%	70.3	70.9	0.9%
2.0	35.6%	69.9	68.2	-2.4%
3.5	36.0%	70.3	68.3	-3.0%
5.0	32.7%	67.0	65.9	-1.7%

3.2 Curvature of a single oil droplet in triangular pore space

To investigate the curvature computation algorithm described in section 2.2, a simple 3D test case was performed. A cylindrical pore structure with the length of $171.5 \mu\text{m}$ with an isosceles triangular cross-section as in the previous test case was used. This pore structure was modeled with a grid size of $3.5 \mu\text{m}$. The same fluid properties and contact angle as in the previous section were used. Initially, oil was placed in the central region of the pore space as shown in Fig. 3a. Then, simulations were performed for 50,000 time steps until they formed a single oil droplet after equilibrium (Fig. 3b).

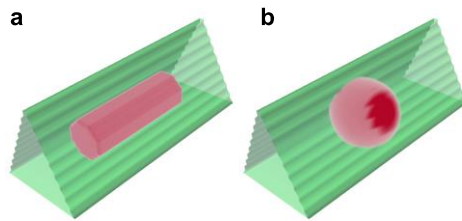


Fig. 3. An oil droplet in a 3D triangular pore space. (a) Initial condition. (b) Equilibrium condition. Here, solid is shown in transparent green and the oil phase is shown in red.

Two types of the oil/water interface were prepared based on the simulation results: a “simulated interface” and a “smoothed interface”. The simulated interface was obtained by extracting the contour line of the color function $\rho^N=0$. This surface originally had a sub-resolution smoothness. The other surface, the smoothed interface, was obtained from synthetic voxelized label data. The simulated distribution of the color function was segmented into label data with the threshold of $\rho^N=0$, then the oil/water interface was extracted using the marching cubes algorithm. This surface has a staircase shape due to the shape of a voxelized grid system. Therefore, it needs to be smoothed before computing curvature. As discussed in section 2.2, three smoothing methods were applied: Constrained Gaussian smoothing (CGS), Laplacian smoothing (LPS) and boundary preserving Gaussian smoothing (BPGS). The parameters used for the smoothing are summarized in Table 2.

Based on the simulated fluid pressure, the capillary pressure was obtained using:

Table 2. Parameters used for Constrained Gaussian smoothing (CGS), Laplacian smoothing (LPS) and boundary preserving Gaussian smoothing (BPGS).

smoothing method	parameters	smoothing level		
		level 1	level 2	level 3
CGS	kernel size	1	3	9
LPS	number of iterations	200	600	1800
BPGS	number of iterations	50	150	450

$$P_c^{sim} = P_o^{avg} - P_w^{avg} = \frac{1}{N_o} \sum_{\rho^N > 0.9} P - \frac{1}{N_w} \sum_{\rho^N < -0.9} P \quad (10)$$

where P_o^{avg} and P_w^{avg} are the average fluid pressure in the oil and water phases, respectively; N_o and N_w are the number of grid blocks having $\rho^N > 0.9$ and $\rho^N < -0.9$, respectively; P is the simulated fluid pressure in each grid block. Here, to exclude the interface region, we computed an average pressure for each phase for $|\rho^N| > 0.9$. Then, P_c^{sim} was converted to the simulated mean curvature using Eq. 1. We refer to this mean curvature as the fluid pressure derived mean curvature, κ_m^P .

Fig. 4 shows the oil/water interface after smoothing, colored by computed mean curvature. For all of the smoothed surfaces, we see a variation in the computed mean curvature, although the correct mean curvature should give a uniform value as shown in the curvature computed on the simulated interface, Fig. 4(j), because the oil droplet is in capillary equilibrium. Furthermore, we see errors around the edge of the interface (close to the three-phase contact lines) for all of the smoothed interfaces, Fig. 4(a)~(i). Therefore, we decided to discard the data points whose distance from solid surface is fewer than 3 voxels. This 3 voxels distance cutoff was used for all of the following analyses. Table 3 summarizes the average and standard deviation of the computed mean curvature and the relative difference to the fluid pressure derived mean curvature, $\kappa_m^P = 0.133 \text{ voxel}^{-1}$ for the smoothing level 2 as shown in Fig. 4(b), (e), (h). We note that Li et al. [3] proposed that in addition to the distance cutoff, taking an average weighted by the distance from the solid surface further improves the accuracy of curvature measurement using synthetic test cases of the interface in a cylindrical structure with a circular-cross section; however, taking a distance-weighted average did not improve the estimation of curvature in our cases as we can conclude from the overestimated curvature values observed in the central region of the oil droplet in Fig. 4(a) ~ (i).

In fact, the optimum smoothing method and its level of smoothing is dependent on the shape and the size of an object. Therefore, here, we only provide the qualitative features of the three smoothing methods. CGS preserves the shape of an object, but significant smoothing cannot be applied even with increasing the kernel size, because the resultant surface is constrained by the original voxel data. As a result, CGS tends to give a wider range of variation. LPS can apply significant smoothing by

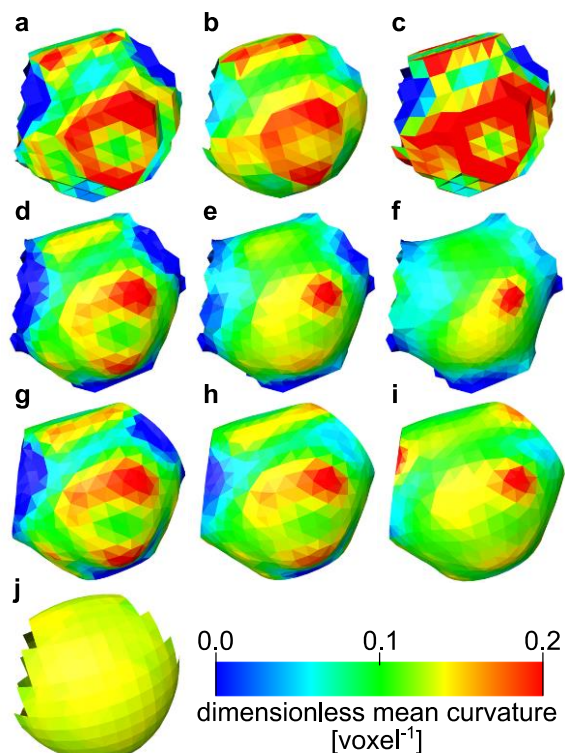


Fig. 4. The oil/water interface after the application of the several smoothing methods with different smoothing levels. CGS with a kernel size of (a) 1 voxel, (b) 3 voxels and (c) 9 voxels. LPS with (d) 200 iterations, (e) 600 iterations and (f) 1800 iterations. BPGS with (g) 50 iterations, (h) 150 iterations and (i) 450 iterations. (j) The simulated interface. Here, the surface is colored by the value of mean curvature. The correct surface should present uniform mean curvature as seen in the simulated interface since the droplet is in capillary equilibrium for which the capillary pressure is uniform.

increasing the number of iterations; however, this could change the shape of the interface as shown in a top part of the interface in Fig. 4(f) where we see the bending of the interface towards the opposite direction. This is caused by the propagation of errors in the three-phase contact line through many iterations. However, combining with the distance cutoff, this erroneous part can be effectively removed while preserving a well smoothed surface in the middle of the interface. BPGS gives the similar results to LPS, while better keeping the shape of an object compared to LPS.

For the simulated interface which was directly obtained from the simulated color function without any smoothing, the computed curvature showed good agreement with the fluid pressure derived curvature with a smaller standard deviation (Table 3). Moreover, since this surface did not show the errors close to the three-phase contact line, the average and standard deviation without the distance cutoff also gave similar values, i.e., the average of 0.132 voxel^{-1} and standard deviation of $3.99 \times 10^{-3} \text{ voxel}^{-1}$.

Table 3. Voxel based interfaces with the smoothing level 2 (see Table 2) and a distance cutoff of 3 voxels. Here, the relative difference was computed to the fluid pressure derived mean curvature of $\kappa_m^P = 0.133 \text{ voxel}^{-1}$.

surface	computed curvature	std.	relative difference
	avg. [voxel^{-1}]	[voxel^{-1}]	
simulated interface	0.134	3.23×10^{-3}	0.7%
smoothed with CGS	0.149	3.23×10^{-2}	12.1%
smoothed with LPS	0.132	2.40×10^{-2}	-0.1%
smoothed with BPGS	0.139	2.68×10^{-2}	5.0%

3.3 Local capillary pressure estimation for complex porous media

Two realistically complex pore structures were used: a synthetic bead pack pore structure and micro-CT images of Bentheimer sandstone. For these structures, simulation domains were composed of $256 \times 256 \times 256$ voxels with a grid size of $3.56 \mu\text{m}$ as shown in Fig 5. For the analysis of the simulation results, the pore structures were divided into pore regions using the separate pore algorithm in Avizo, resulting in 402 and 272 pore regions for the bead pack and Bentheimer sandstone, respectively. The mean pore radius which accounts for 50% of the pore volume was $37 \mu\text{m}$ and $30 \mu\text{m}$ for the beadpack and Bentheimer sandstone, respectively.

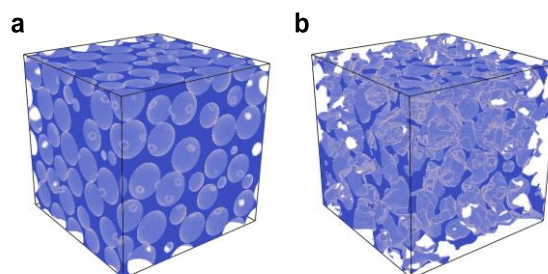


Fig. 5. Pore structures of (a) the bead pack and (b) the Bentheimer sandstone.

The identical density and viscosity of water and oil phase were set to $1,000 \text{ kg/m}^3$ and 1 mPa , respectively. The interfacial tension and contact angle were set to 25 mN/m and 45° , respectively. All 6 faces of the cubic simulation domain were covered with solid voxels, i.e., a no-slip boundary condition was applied to all 6 faces. Initially, 50% of oil and 50% of water were randomly placed in pore voxels, then the simulations were performed with no external force for 250,000 time steps (corresponding to 0.1 seconds) until they reached equilibrium conditions.

Fig. 6 shows the simulated phase distribution at the equilibrium condition. Similar to the analysis presented in the previous section, the simulated interface and the smoothed interface were prepared from the simulation results. Curvature computation was performed on these surfaces. Figs. 7 and 8 show the histogram of the computed curvature for the bead pack and Bentheimer sandstone, respectively. The fluid pressure derived mean

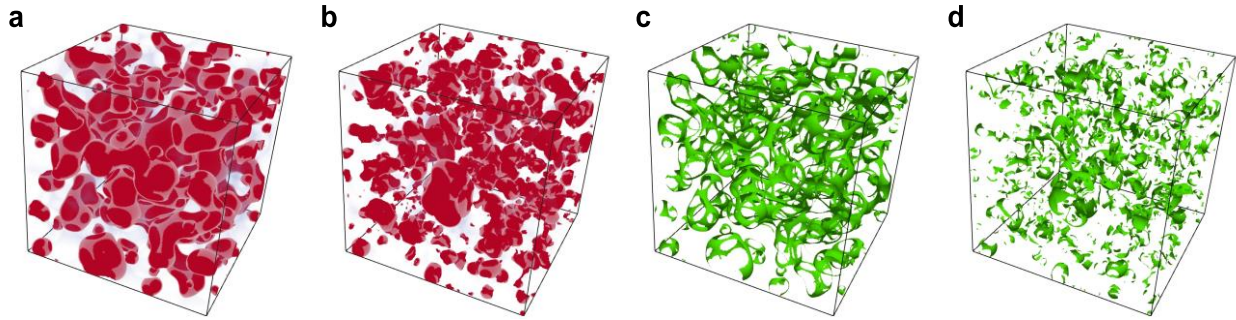


Fig. 6. The simulated phase distribution of (a) the bead pack pore structure and (b) the Bentheimer sandstone pore structure. Here, only oil is shown in red, while water and solid are transparent. The extracted interface based on the simulation results for (c) the bead pack and (d) the Bentheimer sandstone. This was performed extracting the contour line of $p^N=0$.

curvature, κ_m^P , obtained with Eq. 10 is also shown by the vertical line. In both figures, the histograms obtained from the smoothed interface without and with the distance cutoff are shown in (a) and (b), while those obtained from the simulated interface without and with the distance cutoff are shown in (c) and (d). The average and standard deviation of these distributions are summarized in Table 4 for the bead pack and Table 5 for the Bentheimer sandstone.

Here, we discuss the following two key observations. First, the computed curvatures obtained with the smoothed interface show the wide range of the distribution when the cutoff is not applied, while the curvatures obtained with the simulated interface show a much narrower range. For the histograms without the cutoff, the difference between the distributions obtained with the smoothed interfaces and the simulated interface suggests that there are many erroneous values in curvature computation on the smoothed surfaces. The histograms

obtained with CGS appeared to have the most similar distribution to that obtained with the simulated interface. However, as shown in Table 4 and 5, their average values are much higher than that obtained from the fluid pressure and simulated interface because of the long tails of their distribution toward values higher than 0.3, which are not shown in Figs. 7 and 8. Second, after the application of the distance cutoff, all the histograms obtained with the smoothed interface become similar to that obtained with the simulated interface. However, in both the smoothed and the simulated interface, the data points of high curvature values have been lost. This is because the distance cutoff removes the data points of not only the edges of the interface but also the entire parts of the interface in small pores, which tend to give a high curvature value. Although a high curvature value is not captured when the distance cutoff is applied, the smoothed interface seems to provide a good estimate of local mean curvature values of the interface. This will be

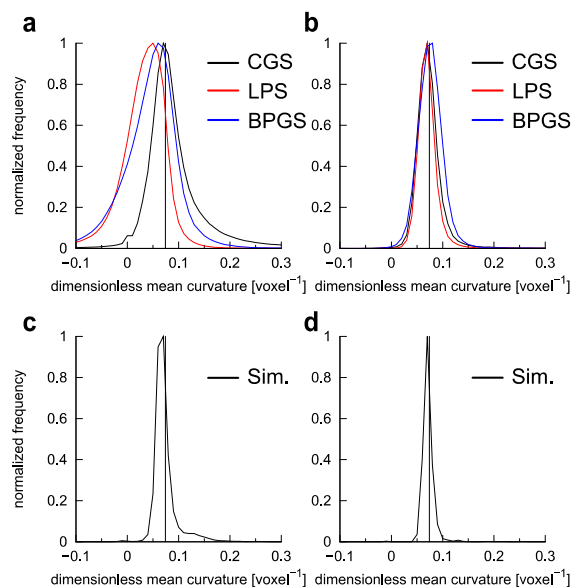


Fig. 7. The histogram of computed mean curvature for the bead pack. The histograms computed on the smoothed interface without and with the distance cutoff (a and b). The histograms computed on the simulated interface without and with the distance cutoff (c and d). The fluid pressure derived mean curvature obtained with Eq. 10 is also shown by the vertical line.

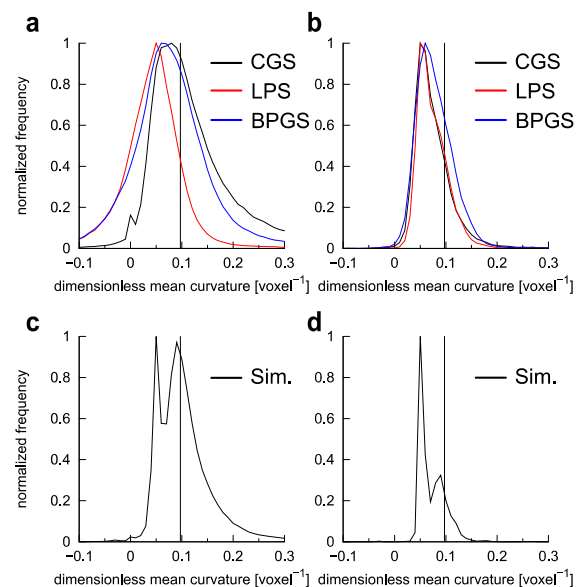


Fig. 8. The histogram of computed mean curvature for the Bentheimer sandstone. The histograms computed on the smoothed interface without and with the distance cutoff (a and b). The histograms computed on the simulated interface without and with the distance cutoff (c and d). The fluid pressure derived mean curvature obtained with Eq. 10 is also shown by the vertical line.

Table 4. The average and standard deviation of the distribution of computed mean curvature for the bead pack shown in Fig. 7.

	computed curvature	
	avg. [voxel ⁻¹]	std. [voxel ⁻¹]
simulated fluid pressure	7.40×10^{-2}	N/A
without the cutoff		
simulated interface	7.36×10^{-2}	4.25×10^{-2}
smoothed with CGS	11.5×10^{-2}	16.1×10^{-2}
smoothed with LPS	3.06×10^{-2}	5.63×10^{-2}
smoothed with BPGS	4.54×10^{-2}	6.87×10^{-2}
with the cutoff		
simulated interface	7.12×10^{-2}	1.22×10^{-2}
smoothed with CGS	8.02×10^{-2}	8.43×10^{-2}
smoothed with LPS	7.01×10^{-2}	2.03×10^{-2}
smoothed with BPGS	7.62×10^{-2}	2.54×10^{-2}

Table 5. The average and standard deviation of the distribution of computed mean curvature for Bentheimer sandstone shown in Fig. 8.

	computed curvature	
	avg. [voxel ⁻¹]	std. [voxel ⁻¹]
simulated fluid pressure	9.73×10^{-2}	N/A
without the cutoff		
simulated interface	11.1×10^{-2}	8.20×10^{-2}
smoothed with CGS	19.3×10^{-2}	27.0×10^{-2}
smoothed with LPS	4.24×10^{-2}	7.64×10^{-2}
smoothed with BPGS	8.51×10^{-2}	10.8×10^{-2}
with the cutoff		
simulated interface	7.17×10^{-2}	2.51×10^{-2}
smoothed with CGS	8.16×10^{-2}	8.48×10^{-2}
smoothed with LPS	7.45×10^{-2}	3.12×10^{-2}
smoothed with BPGS	7.81×10^{-2}	3.60×10^{-2}

further discussed in the following analysis. Among the three smoothing methods, when the cutoff is applied, LPS gave the closest average value of the mean curvature to that obtained from the simulated interface with the smallest standard deviation.

To further investigate the accuracy of curvature computation on the interface, the comparison was made on a pore-by-pore basis. First, we computed local capillary pressure applying Eq. 10 for each pore region. This local capillary pressure was converted to the mean curvature for each pore region using Eq. 1. Next, we also obtained the mean curvature for each pore region by taking the average of the computed curvature values on the interfaces in that pore region. Figs. 9 and 10 show the

comparison between the dimensionless mean curvature obtained from the simulated fluid pressure and that obtained from the computed curvature on the interface for the bead pack and Bentheimer sandstone, respectively. In both, the curvature computed on the simulated interfaces gave consistent values with those obtained from the fluid pressure. This means that when the interface is reasonably smooth, the curvature computation gives an accurate estimation of local capillary pressure. For the curvature computed on the smoothed interface, CGS shows highly overestimated curvature values for some pore regions, while LPS and BPGS estimate curvature within a range of about $\pm 20\%$ difference from the curvatures obtained with the fluid pressure for the mean curvature smaller than

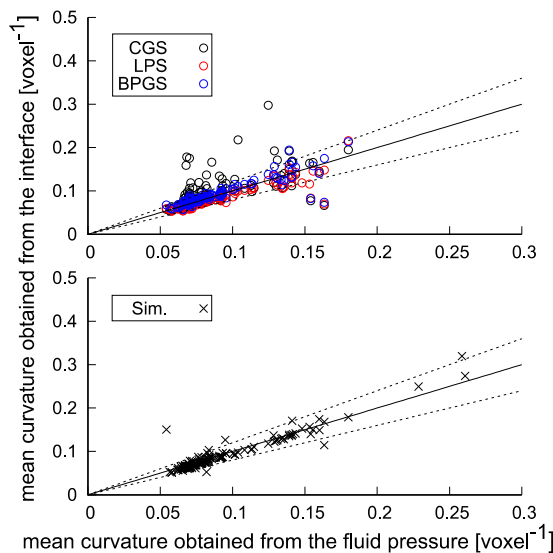


Fig. 9. Comparison of local mean curvature for the bead pack on pore-by-pore basis. X-axis shows the dimensionless mean curvature obtained from the simulated fluid pressure, while the Y-axis shows that computed on the interface. A unit slope indicating perfect agreement in these values is shown by a black solid line, while, $\pm 20\%$ difference is shown by black dotted lines.

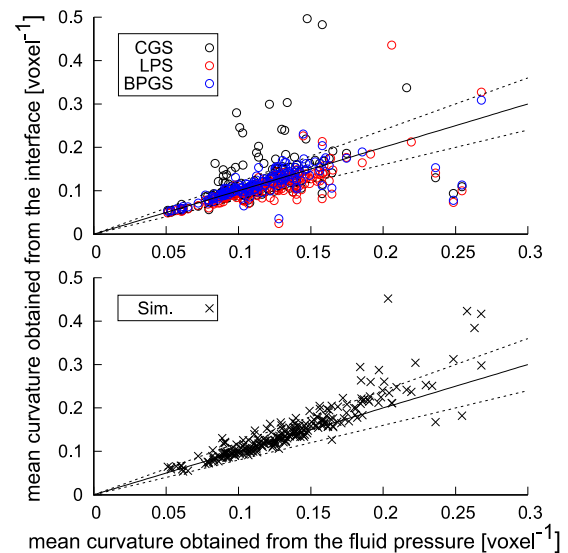


Fig. 10. Comparison of local mean curvature for the Bentheimer sandstone on pore-by-pore basis. X-axis shows the dimensionless mean curvature obtained from the simulated fluid pressure, while the Y-axis shows that computed on the interface. A unit slope indicating perfect agreement in these values is shown by a black solid line, while, $\pm 20\%$ difference is shown by black dotted lines.

about 0.15 voxel^{-1} . Therefore, with LPS or BPGS incorporating the distance cutoff of 3 voxels, local capillary pressure can be measured with a $\pm 20\%$ error up to 0.15 voxel^{-1} at this resolution.

4 Conclusions

Direct numerical simulations were conducted to obtain oil and water phase distribution, the exact location of the oil/water interface and local fluid pressure. From the simulation results, two types of the oil/water interface were generated. One is a simulated interface which has a sub-resolution smoothness, while the other is a smoothed interface. The smoothed interfaces were generated by applying three types of smoothing methods to a staircase shape surface extracted from synthetic segmented data based on the simulated phase distribution. Then, curvature computation was performed on these surfaces. The resultant mean curvature value from these surfaces was compared with that obtained from the simulated fluid pressure. This was performed to find the best way to extract accurate curvature measurement, and to quantify the likely uncertainties.

First, we tested our approach using a single oil droplet in triangular pore space. Computed curvature values on the simulated interface showed a narrow distribution whose average value was consistent with that obtained from the fluid pressure. This means that when the surface is sufficiently smooth, the fitting of a quadratic form equation to the surface properly computes local curvatures. On the other hand, computed curvature values on the smoothed interface showed a wide distribution with erroneous values close to the three-phase contact lines. Therefore, we discarded values whose distance from solid surface was fewer than 3 voxels. After the application of the distance cutoff, the average of computed mean curvature values became closer to that obtained from the fluid pressure. However, their standard deviation was on the order of $10^{-2} \text{ voxel}^{-1}$ against the mean curvature value of 0.133 voxel^{-1} , which was 10 times higher than that obtained from the simulated interface. These deviated values were caused by error, since an oil droplet in capillary equilibrium should give a uniform value of mean curvature.

Next, realistically complex pore structures of a bead pack and Bentheimer sandstone were used to simulate oil droplets in capillary equilibrium. When the distance cutoff was not applied, curvature computed on the simulated interface gave an average value of mean curvature consistent with that obtained from the fluid pressure for both the bead pack and Bentheimer sandstone. However, when the distance cutoff was applied, the average value became lower than that from the fluid pressure. This is because the distance cutoff removes the data points of not only the edges of the interface but also the entire parts of the interface in small pores, which tend to give a high curvature value. For the smoothed interface, the distributions obtained without the cutoff were quite different from that obtained from the simulated interface. Many negative values of computed mean curvatures suggest significant errors in these cases.

After the application of the distance cutoff, the distribution became closer to that obtained with the simulated interface.

Finally, we compared computed mean curvature on a pore-by-pore basis. As opposed to the test case performed on the single oil droplet, oil droplets with different mean curvature values corresponding to different pore sizes can be obtained in these simulations. Therefore, we need to investigate whether the range of computed mean curvature is caused by error or it reflects a variation in local capillary pressure. Good agreement between mean curvature obtained from the fluid pressure and that computed from the simulated surface for each pore region suggested that the range of distribution observed in the simulated surface properly captures variation in local capillary pressure. For the smoothed interface with the cutoff, LPS and BPGS estimated curvature within a range of about $\pm 20\%$ difference from the curvatures obtained with the fluid pressure for mean curvatures smaller than about 0.15 voxel^{-1} , while CGS showed highly overestimated curvature values for some pore regions.

In conclusion, the application of the distance cutoff is necessary to remove the errors close to the three-phase contact line. Among the three tested smoothing methods, LPS appeared to be the best method since it gave the closest average of mean curvature values to that obtained from the simulated interface with the smallest standard deviation. In addition, it gave good estimates of the local mean curvature for each pore with a $\pm 20\%$ error up to 0.15 voxel^{-1} at this resolution.

Currently, we are using the similar approach to quantify the accuracy of curvature computation for drainage and imbibition events on the bead pack and Bentheimer sandstone. This will be discussed in a future publication.

Acknowledgements

The authors gratefully thank Japan Oil, Gas and Metals National Corporation (JOGMEC)

References

1. R. T. Armstrong, M. L. Porter, and D. Wildenschild, "Linking pore-scale interfacial curvature to column-scale capillary pressure," *Adv. Water Resour.*, vol. 46, no. 6, pp. 55–62, Sep. 2012, doi: 10.1016/j.advwatres.2012.05.009.
2. M. L. Porter, D. Wildenschild, G. Grant, and J. I. Gerhard, "Measurement and prediction of the relationship between capillary pressure, saturation, and interfacial area in a NAPL-water-glass bead system," *Water Resour. Res.*, vol. 46, no. 8, pp. 1–10, Aug. 2010, doi: 10.1029/2009WR007786.
3. T. Li, S. Schlüter, M. I. Dragila, and D. Wildenschild, "An improved method for estimating capillary pressure from 3D microtomography images and its application to

- the study of disconnected nonwetting phase,” *Adv. Water Resour.*, vol. 114, pp. 249–260, 2018, doi: 10.1016/j.advwatres.2018.02.012.
4. A. L. Herring, J. Middleton, R. Walsh, A. Kingston, and A. Sheppard, “Flow rate impacts on capillary pressure and interface curvature of connected and disconnected fluid phases during multiphase flow in sandstone,” *Adv. Water Resour.*, vol. 107, pp. 460–469, 2017, doi: 10.1016/j.advwatres.2017.05.011.
 5. I. Halliday, A. P. Hollis, and C. M. Care, “Lattice Boltzmann algorithm for continuum multicomponent flow,” *Phys. Rev. E*, vol. 76, no. 2, p. 026708, Aug. 2007, doi: 10.1103/PhysRevE.76.026708.
 6. D. D’Humières, I. Ginzburg, M. Krafczyk, P. Lallemand, and L.-S. Luo, “Multiple-relaxation-time lattice Boltzmann models in three dimensions,” *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, vol. 360, no. 1792, pp. 437–451, Mar. 2002, doi: 10.1098/rsta.2001.0955.
 7. J. Brackbill, D. Kothe, and C. Zemach, “A continuum method for modeling surface tension,” *J. Comput. Phys.*, vol. 100, no. 2, pp. 335–354, Jun. 1992, doi: 10.1016/0021-9991(92)90240-Y.
 8. Z. Guo, C. Zheng, and B. Shi, “Discrete lattice effects on the forcing term in the lattice Boltzmann method,” *Phys. Rev. E*, vol. 65, no. 4, p. 046308, Apr. 2002, doi: 10.1103/PhysRevE.65.046308.
 9. Z. Yu and L.-S. Fan, “Multirelaxation-time interaction-potential-based lattice Boltzmann model for two-phase flow,” *Phys. Rev. E*, vol. 82, no. 4, p. 046708, Oct. 2010, doi: 10.1103/PhysRevE.82.046708.
 10. M. Latva-Kokko and D. H. Rothman, “Diffusion properties of gradient-based lattice Boltzmann models of immiscible fluids,” *Phys. Rev. E*, vol. 71, no. 5, p. 056702, May 2005, doi: 10.1103/PhysRevE.71.056702.
 11. T. Akai, B. Bijeljic, and M. J. Blunt, “Wetting boundary condition for the color-gradient lattice Boltzmann method: Validation with analytical and experimental data,” *Adv. Water Resour.*, vol. 116, no. March, pp. 56–66, Jun. 2018, doi: 10.1016/j.advwatres.2018.03.014.
 12. T. Akai, A. M. Alhammadi, M. J. Blunt, and B. Bijeljic, “Modeling Oil Recovery in Mixed-Wet Rocks: Pore-Scale Comparison Between Experiment and Simulation,” *Transp. Porous Media*, vol. 127, no. 2, pp. 393–414, Mar. 2019, doi: 10.1007/s11242-018-1198-8.
 13. Y. H. Qian, D. D’Humières, and P. Lallemand, “Lattice BGK models for Navier-Stokes equation,” *EPL (Europhysics Lett.)*, vol. 17, no. 6, p. 479, 1992.
 14. R. T. Armstrong, C. H. Pentland, S. Berg, J. Hummel, D. Lichau, and L. Bernard, “Estimation of curvature from micro-CT liquid-liquid displacement studies with pore scale resolution,” *Int. Symp. Soc. Core Anal.*, vol. SCA2012-55, no. August, p. 6, 2012, doi: 10.1109/CVPR.1994.323794.
 15. M. Andrew, B. Bijeljic, and M. J. Blunt, “Pore-by-pore capillary pressure measurements using X-ray microtomography at reservoir conditions: Curvature, snap-off, and remobilization of residual CO₂,” *Water Resour. Res.*, vol. 50, no. 11, pp. 8760–8774, Nov. 2014, doi: 10.1002/2014WR015970.
 16. Q. Lin, B. Bijeljic, R. Pini, M. J. Blunt, and S. Krevor, “Imaging and Measurement of Pore-Scale Interfacial Curvature to Determine Capillary Pressure Simultaneously With Relative Permeability,” *Water Resour. Res.*, vol. 54, no. 9, pp. 7046–7060, Sep. 2018, doi: 10.1029/2018WR023214.
 17. W. E. Lorensen and H. E. Cline, “Marching cubes: A high resolution 3D surface construction algorithm,” in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques - SIGGRAPH ’87*, 1987, pp. 163–169, doi: 10.1145/37401.37422.
 18. L. R. Herrmann, “Laplacian-isoparametric grid generation scheme,” *J. Eng. Mech. Div.*, vol. 102, no. 5, pp. 749–907, 1976.
 19. G. Taubin, “Curve and surface smoothing without shrinkage,” in *Proceedings of IEEE International Conference on Computer Vision*, 1995, pp. 852–857, doi: 10.1109/ICCV.1995.466848.