Journal of Computational and Applied Mathematics 265 (2014) 139–150

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Adaptive moving grid methods for two-phase flow in porous media

Hao Dong [a], Zhonghua Qiao [b], Shuyu Sun [c], Tao Tang [d,*]

[a] Institute for Computational and Theoretical Studies, Hong Kong Baptist University, Kowloon, Hong Kong

[b] Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

[c] Division of Physical Science and Engineering, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia

[d] Department of Mathematics & Institute for Computational and Theoretical Studies, Hong Kong Baptist University, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 26 October 2012
Received in revised form 22 March 2013

Keywords:
Two-phase flow
Porous medium
Mixed finite element method
Finite volume method
Moving mesh method

ABSTRACT

In this paper, we present an application of the moving mesh method for approximating numerical solutions of the two-phase flow model in porous media. The numerical schemes combine a mixed finite element method and a finite volume method, which can handle the nonlinearities of the governing equations in an efficient way. The adaptive moving grid method is then used to distribute more grid points near the sharp interfaces, which enables us to obtain accurate numerical solutions with fewer computational resources. The numerical experiments indicate that the proposed moving mesh strategy could be an effective way to approximate two-phase flows in porous media.

1. Introduction

Various models of two-phase flows have been widely used in many fields, such as power generation and chemical and petroleum exploitation. Flows of such type are important for the design of steam generators, internal combustion engines, jet engines, refrigeration systems, pipelines for transport of gas and oil mixtures, etc.

Many numerical methods have been applied to solve two-phase flow, including conventional finite difference methods [1,2], discontinuous Galerkin methods [3–5], mixed finite element methods [6–8], higher-order finite element methods [9,10], finite element methods with adaptive mesh refinements [11–13], lattice Boltzmann methods [14,15] and cell-centered finite difference methods with multipoint flux approximation [16–18].

There are two main approaches to model two-phase flow in porous media. The first one solves the balance equations individually for each of the fluids, while the second involves manipulation and combination of those equations into several alternate forms with various choices of primary dependent variables. For the latter method, the choice of equation form and the primary solution defined by variables have considerable implications for the mathematical analysis and the numerical method used to solve these equations. When the second approach above is applied, the governing equations are usually written in terms of a fractional flow formulation, i.e., in terms of the saturation and the global pressure. The main reason for this fractional flow approach is that one can then use efficient numerical methods to take advantage of many physical properties inherent in the flow equations.

Some relevant works choose global pressure as one of the variables of the fractional flow formulation; see, e.g., [19–21] and the references therein. It has been demonstrated that this fractional flow approach is far more efficient than the original

* Corresponding author.
E-mail addresses: donghao103@hotmail.com (H. Dong), zqiao@polyu.edu.hk (Z. Qiao), shuyu.sun@kaust.edu.sa (S. Sun), ttang@math.hkbu.edu.hk, ttang@lsec.cc.ac.cn (T. Tang).

0377-0427/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.cam.2013.09.027

two-pressure approach from the computational point of view [21]. However, when the mixed finite element (MFE) method is employed, this kind of fractional flow formulation becomes inconsistent [22,20,23–25]. The reason is that the MFE method requires the primary variable and its derivative to be continuous at the grid–block interface, while the global pressure is discontinuous in heterogeneous media [26] due to saturation discontinuity. To avoid the drawback of the fractional flow implementation, Hoteit and Firoozabadi [27] provide a new hybrid MFE formulation in two-dimensional (2D) and three-dimensional (3D) heterogeneous media. Instead of using the global pressure, they employ the wetting-phase pressure as a primary variable, as the wetting-phase pressure is always continuous as long as none of the phases is immobile. Such formulation has also been applied in more complicated three-phase flow problems by Sun and Firoozabadi [10].

The most important characteristic of two-phase flow in porous media is the existence of sharp saturation fronts, and sometimes saturation jumps, in addition to permeability and porosity jumps that separate different media in heterogeneous media cases. The main challenge lies in the fact that extremely fine meshes are required over thin fronts or jumps of the physical domain in order to produce physically correct results. While using uniform meshes in such cases is quite ineffective, adaptive mesh refinement can reduce the computational costs greatly. However, mesh refinement increases the number of total elements, requiring a storage reallocation process, which is quite complex for programming. In this work, instead of local mesh refinement, we extend the moving mesh methods developed in [28,29,12] to increase the numerical efficiency. The method can cluster more grid points to the areas around the sharp interfaces without changing the mesh structure.

The paper is organized as follows. First, we will review the governing equations of two-phase flow, and the new formulation proposed by Hoteit and Firoozabadi [27]. We will then use an iterative implicit pressure explicit concentration (IMPEC) scheme based on the mixed finite element–finite volume (MFE–FV) method [27,10,30] to discretize the governing equations. The moving mesh method will be introduced in Section 4. Finally, several one-dimensional and two-dimensional numerical experiments are carried out to demonstrate the effectiveness of the proposed scheme.

2. Mathematical formulation

Two-phase flow in porous media is usually modeled from a representative elementary volume (REV) scale or Darcy scale point of view. In the Darcy scale, two-phase flow modeling equations can be broken into two parts. The first part is the conservation law, which is exact. For the wetting phase, the mass conservation law reads as

$$\phi \frac{\partial(\rho_w S_w)}{\partial t} + \nabla \cdot (\rho_w v_w) = \hat{F}_w = F_w \rho_w, \tag{2.1}$$

where ϕ is the porosity of the medium, and S_w, F_w, v_w, and ρ_w are the saturation, external volumetric flow rate, volumetric velocity, and density of the wetting phase, respectively. Assuming an incompressible fluid, (2.1) is simplified to the conservation of volume:

$$\phi \frac{\partial S_w}{\partial t} + \nabla \cdot v_w = F_w. \tag{2.2}$$

Similarly, the conservation of volume for the incompressible non-wetting phase yields

$$\phi \frac{\partial S_n}{\partial t} + \nabla \cdot v_n = F_n, \tag{2.3}$$

where S_n, F_n, and v_n are the saturation, external volumetric flow rate, and volumetric velocity of the non-wetting phase, respectively.

To close the system, we need constitutive equations to link the Darcy velocity with pressure, which are given by Darcy's law:

$$v_\alpha = -\frac{k_{r\alpha}}{\mu_\alpha} \boldsymbol{K} (\nabla p_\alpha + \rho_\alpha g \nabla z), \quad \alpha = n, w, \tag{2.4}$$

where the subscripts n and w denote the non-wetting and wetting phases, respectively, $\boldsymbol{K}$ is the absolute permeability tensor, g is the gravity acceleration constant, z is the depth, and p_α, $k_{r\alpha}$, and μ_α are the pressure, relative permeability, and viscosity of phase α, respectively. By definition, the summation of the two saturations must be unity:

$$S_n + S_w = 1. \tag{2.5}$$

In porous media, surface tension and wettability usually cause significant capillary pressure (i.e., the difference of the pressure in non-wetting and wetting phases). Here, we assume that the capillary pressure is a given function of saturation:

$$p_n - p_w = p_c = p_c(S_w). \tag{2.6}$$

Hoteit and Firoozabadi [27] provided a mixed formulation using the wetting-phase pressure as a primary variable. The flow potential Φ_α of the α-phase and the capillary potential Φ_c are defined as follows:

$$\Phi_\alpha = p_\alpha + \rho_\alpha g z, \quad \alpha = n, w, \tag{2.7}$$

$$\Phi_c = \Phi_n - \Phi_w. \tag{2.8}$$

The total velocity v_t can be written in form

$$v_t = v_n + v_w = -\lambda_n \boldsymbol{K} \nabla \Phi_n - \lambda_w \boldsymbol{K} \nabla \Phi_w, \tag{2.9}$$

where $\lambda_\alpha = k_{r\alpha}/\mu_\alpha$ is the mobility of the α-phase.

With some simple manipulations, Eq. (2.9) becomes

$$v_t = -\lambda_t \boldsymbol{K}\nabla\Phi_w - \lambda_n \boldsymbol{K}\nabla\Phi_c = v_a + v_c, \tag{2.10}$$

where $\lambda_t = \lambda_n + \lambda_w$, $v_a = -\lambda_t \boldsymbol{K}\nabla\Phi_w$, and $v_c = -\lambda_n \boldsymbol{K}\nabla\Phi_c$. The velocity variable v_a has the same driving force as the wetting-phase velocity, but with a smoother mobility λ_c than the wetting-phase mobility.

With the above notation, the wetting-phase velocity v_w can be written as

$$v_w = -\lambda_w \boldsymbol{K}\nabla\Phi_w = \frac{\lambda_w}{\lambda_t}(-\lambda_t \boldsymbol{K}\nabla\Phi_w) = f_w v_a, \tag{2.11}$$

where the wetting-phase fractional function f_w is denoted by

$$f_w = \frac{\lambda_w}{\lambda_t} = \frac{\mu_n k_{rw}}{\mu_n k_{rw} + \mu_w k_{rn}}. \tag{2.12}$$

Using Eqs. (2.5) and (2.10), the balance equation of both phases is obtained by adding the mass balance equations of the wetting phase and the non-wetting phase together:

$$\nabla\cdot(v_a + v_c) = F_n + F_w. \tag{2.13}$$

Moreover, the saturation equation of the wetting phase can be expressed as

$$\phi\frac{\partial S_w}{\partial t} + \nabla\cdot(f_w v_a) = F_w. \tag{2.14}$$

Consequently, the two-phase flow model in porous media can be written as

$$\nabla\cdot(v_a + v_c) = F_n + F_w, \tag{2.15}$$

$$\phi\frac{\partial S_w}{\partial t} + \nabla\cdot(f_w v_a) = F_w, \tag{2.16}$$

$$v_a = -\lambda_t \boldsymbol{K}\nabla\Phi_w, \tag{2.17}$$

$$v_c = -\lambda_n \boldsymbol{K}\nabla\Phi_c, \tag{2.18}$$

and the boundary conditions are as follows:

$$p_w = p^D \quad \text{on } \Gamma^D, \tag{2.19}$$

$$(v_a + v_c)\cdot\mathbf{n} = q^N, \quad \text{on } \Gamma^N, \tag{2.20}$$

$$S_w = S^0 \quad \text{in } \Omega, \tag{2.21}$$

$$S_w = S^N \quad \text{on } \Gamma^N, \tag{2.22}$$

where Γ^D and Γ^N are non-overlapping boundaries corresponding to Dirichlet and Neumann boundary conditions, $\mathbf{n}$ is the outward unit normal vector, q^N and p^D are the imposed volumetric injection rate and pressure at Γ^N and Γ^D, respectively, S^0 is the initial saturation, and S^N is the boundary saturation of the injected fluid at Γ^N.

3. Numerical method

Let the domain Ω be partitioned into $\Sigma_h = \{E_i, i = 1, 2, \ldots, N_E\}$, where N_E is the number of elements and $\bigsqcup_{i=1,\ldots,N_E}\bar{E}_i = \Omega$. Similarly, we partition the time $[0, T]$ into $\{0 = t_0, t_1, t_2, \ldots, t_M = T\}$.

We first apply forward Euler's time stepping to system (2.15)–(2.18):

$$\nabla\cdot(v_a^{(k+1)} + v_c^{(k)}) = F_n^{\left(k+\frac{1}{2}\right)} + F_w^{\left(k+\frac{1}{2}\right)}, \tag{3.1}$$

$$v_a^{(k+1)} = -\lambda_t(S_w^{(k)})\mathbf{K}\nabla\Phi_w^{(k+1)}, \tag{3.2}$$

$$v_c^{(k)} = -\lambda_n(S_w^{(k)})\mathbf{K}\nabla\Phi_c(S_w^{(k)}), \tag{3.3}$$

$$\phi\frac{S_w^{(k+1)} - S_w^{(k)}}{t_{k+1} - t_k} + \nabla\cdot(f_w^{(k)} v_a^{(k+1)}) = F_w^{\left(k+\frac{1}{2}\right)}. \tag{3.4}$$

We then apply the mixed finite element method to the pressure equation to solve Φ_w and v_a: seek $\Phi_w \in W_h$ and $v_a \in V_h$ such that

$$\begin{cases} -(v_a^{(k+1)} + v_c^{(k)}, \nabla\cdot u) = \left(F_n^{\left(k+\frac{1}{2}\right)} + F_w^{\left(k+\frac{1}{2}\right)}, u\right), \quad \forall u \in V_h, \\ (\lambda_t^{-1}(S_w^{(k)})\mathbf{K}^{-1}v_a^{(k+1)}, w) = \langle\Phi_w^{(k+1)}, w\rangle_{\Gamma^D} + (\Phi_w^{(k+1)}, \nabla w), \quad \forall w \in W_h, \end{cases} \tag{3.5}$$

where $W_h \subset L^2(\Omega)$ and $V_h \subset H(\mathrm{div}, \Omega)$ are the Raviart–Thomas (RT) spaces, and $L^2(\Omega)$ and $H(\mathrm{div}, \Omega)$ are the usual space setting for a mixed formulation. [In implementation, we choose the lowest order RT_0 space [6], and consequently, $(\Phi^{(k+1)},$

∇w) term above vanishes. We refer to [27,30] for the detailed description on how to implement the lowest order RT_0 element for space discretization of (3.5) are the usual space setting for mixed formulation.] In implementation, we choose the lowest-order RT_0 space [6], and consequently the $(\Phi^{(k+1)}, \nabla w)$ term above vanishes. We refer to [27,30] for a detailed description of how to implement the lowest-order RT_0 element for space discretization of (3.5). A similar mixed finite element method is used to compute $v_c^{(k)}$, but based on the saturation data from the previous time step.

A standard upwind finite volume (FV) method is applied to solve the nonlinear convection equation (3.4). The FV method is a numerical method for solving partial differential equations (PDEs) that calculates the values of the conserved variables averaged across the volume. We use the cell-centered finite volume approach in this paper. The method defines each cell to be the control volume, and defines the numerical solution as the average of the conservative quantity over each cell. Integrating both sides of Eq. (2.14) over any cell K and using the divergence theorem yields

$$\frac{\partial}{\partial t}\int_K \phi S_w \vec{n} + \oint_{\partial K} f_w v_a = \int_K F_w, \tag{3.6}$$

where $\vec{n}$ is the unit outward normal to ∂K. Note that, when expressing $v_{a,K}$ in terms of RT_0 basis functions, the velocity $v_{a,K}$ over a mesh element K can be written as follows:

$$v_{a,K} = \sum_{E\in\partial K} q_{a,K,E} w_{K,E}, \tag{3.7}$$

where $q_{a,K,E}$ is the flux variable across face E of element K, and $w_{K,E}$ is the RT_0 basis function. Thus we can simply use the explicit Euler scheme (3.4) to calculate S_w at time step $k+1$, based on S_w and v_a in terms of RT_0 basis functions at time step k:

$$S_{w,K}^{(k+1)} = S_{w,K}^{(k)} + \frac{1}{\phi|K|}(t_{k+1}-t_k)\left[\int_K F_w^{\left(k+\frac{1}{2}\right)} - \left(\sum_{E\in\partial K} f_w^{(k)} q_{a,K,E}|E|\right)\right], \tag{3.8}$$

where $|K|$ and $|E|$ are the volume of cell K and the length of edge E, respectively. The upwind scheme is employed to discretize f_w. More details of the FV method can be found in LeVeque's book [31].

4. Moving mesh strategy

There have been several useful moving mesh strategies for speeding up the solution convergence for nonlinear PDEs with interfaces. Relevant works for the moving mesh method can be found in [32–36] and the recent book [37]. In this work, we will follow the procedure proposed in [28,29] to construct our moving mesh method for two-phase flow in porous media. In this section, we briefly outline the moving mesh strategy of [28].

Assume that we have obtained a numerical approximation of $S_w^{(k)}$ (saturation of the wet phase) at $t = t_k$ on the mesh $\Gamma^{(k)}$ using the MFE–FV method presented in Section 2. The outline of the moving mesh algorithm is as follows.

Step 1: Solve the Euler–Lagrange equation

$$\frac{\partial}{\partial x_i}\left(G^{ij}\frac{\partial \xi^*}{\partial x^j}\right) = 0, \quad x \in \Omega, \tag{4.1}$$

$$\vec{\xi}|_{\partial\Omega} = \vec{\xi}_b, \tag{4.2}$$

to obtain a logical mesh $\vec{\xi}^*$, where $G = (G^{ij}) = M^{-1}$. M is a piecewise constant called the monitor function, and how to obtain a monitor function is discussed in Section 4.1.

Step 2: Judge if the L_2-norm of $\vec{\xi}^* - \vec{\xi}^0$ is small enough, where $\vec{\xi}^0$ is the fixed initial mesh. If yes, the iteration is over. Otherwise, do the following procedure.

Step 3: Use the difference $\vec{\xi}^* - \vec{\xi}^0$ to compute the mesh moving vector $\delta\vec{x}$ in the physical domain. Then select a suitable ration parameter $\lambda \in [0, 1]$, and move the old mesh $\Gamma^{(k),s}$ in the physical domain to the new one $\Gamma^{(k),s+1}$ by using

$$\vec{x}_0^{s+1} = \vec{x}_0^s + \lambda\delta\vec{x}_0.$$

Step 4: Update the numerical approximation at the new mesh $\Gamma^{(k),s+1}$. We used the conservative interpolation in [29], where van Leer's slope limiter [38] is applied to approximate the left and right data states in second-order accuracy. In the case of unstructured 2D meshes, a vertex value reconstruction technique is applied, as proposed in [39,40].

4.1. Monitor function

It is known that monitor functions have played a very important role in moving mesh schemes; see, e.g., [41,42]. The standard adopt gradient based monitor function is formulated as

$$M = \sqrt{1+\beta|\nabla S_w|^2}, \tag{4.3}$$

where β is some non-negative constant. The moving rate of the mesh gets larger as β increases. Currently we do not have any method to determine the value of β accurately; thus have to choose it by experience. Monitor functions of this type

have been applied in many problems; see e.g. [37,28,29,12] and the references therein. Most recently, this arc-length type monitor function was used in [43] for the time adaptivity in a simulation of thin-film epitaxial growth, and in [44] for solving the Cahn–Hilliard equation.

It is actually not necessary to find the exact value of $|\nabla S_w|$. As the average value of saturation S_w on each cell is already obtained by the FV method, we can take S'^2 in place of $|\nabla S_w|^2$ in Eq. (4.3), where

$$S' = \frac{\sum_{K\cap K'\neq\emptyset} |S_w(K) - S_w(K')|\,|K\cap K'|}{|K|}. \tag{4.4}$$

Here, K and K' are elements with a common edge (face), and $K\cap K'$ is the common edge (face) of K and K'.

In addition, to prevent very singular meshes and large approximation error around the sharp faces, some smoothing steps for the monitor function should be applied. In this paper, we use a smooth technique proposed by [28].

Step 1: Interpolate the monitor function from a piecewise constant function to a piecewise linear function

$$(\pi_h M)|_{\vec{X}_i} = \frac{\sum_{\vec{X}_i\in K} |K|\,M|_K}{\sum_{\vec{X}_i\in K} |K|}, \tag{4.5}$$

where K is an element and $\vec{X}_i$ is its node.

Step 2: Project the piecewise linear function back to the piecewise constant

$$M|_K = \frac{1}{d+1}\sum_{\vec{X}_i\in K} (\pi_h M)|_{\vec{X}_i}, \tag{4.6}$$

where d is the dimension of the physical domain.

However, some of our numerical experiments indicate that such a monitor function may not lead to satisfactory grids. Fig. 1 shows the mesh of initial value produced by monitor (4.3). The true value of S_w is

$$S_w(x) = \begin{cases} 1 & \text{if } x \le 0.15, \\ 0 & \text{if } x > 0.15. \end{cases}$$

Compared with the whole physical domain, the initial saturated region is very small, and the wetting front is very sharp. Even with a value of β as large as 10^4, we are still not able to concentrate enough grid points around the wetting front. If we simply further increase the value of β, the mesh quality around the wetting front becomes quite bad, resulting in the deterioration of numerical solutions after several time steps.

We want to drag as many as grid points away from the wetting front to the region around the sharp interface without harming the mesh quality too much. To achieve this, the strategy proposed by Wang et al. [12] can be applied. Instead of monitor (4.3), a new monitor function is used:

$$M = \sqrt{1 + \beta|\nabla S_w|^2 + \alpha\tilde{S}}, \tag{4.7}$$

where α is some non-negative constant, and the diffused monitor $\tilde{S}$ is obtained by solving the following problem:

$$(1 - \mu\delta)\tilde{S} = |\nabla S_w|^2, \tag{4.8}$$

where μ is a positive parameter, which should be modified to make sure that the result of $\tilde{S}$ is greater than machine epsilon for all mesh grids. In practical computations, it is unnecessary to compute the exact value of the diffused monitor by solving Eq. (4.8); instead we only used two or three algebraic multigrid iterations to get an approximation of $\tilde{S}$. Thus the long-range diffusion effect can be achieved with very high efficiency.

In our work, borrowing the idea from [34], we further modify (4.7) to

$$M = \sqrt{1 + \beta M_0 + \alpha M_1}, \tag{4.9}$$

with

$$M_0 = \frac{|\nabla S_w|^2 - \min(|\nabla S_w|^2)}{\max(|\nabla S_w|^2) - \min(|\nabla S_w|^2)}, \qquad M_1 = \frac{\tilde{S} - \min(\tilde{S})}{\max(\tilde{S}) - \min(\tilde{S})},$$

where $\max(\bullet)$ and $\min(\bullet)$ denote the maximum and minimum of $\bullet$ over the whole compute space. With the above modification, it is much easier to adjust the values of α and β.

Fig. 2 shows the mesh of initial value produced by monitor (4.9). Compared to Fig. 1, it is obvious that the new monitor function helps a lot in improving the mesh quality, and even grids far away from the wetting front are moved to the area near the wetting front.

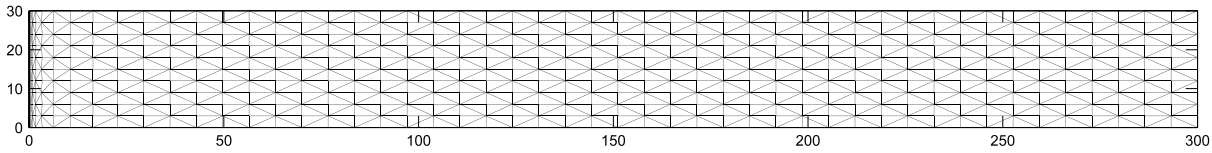

Fig. 1. The initial mesh produced by monitor (4.3).

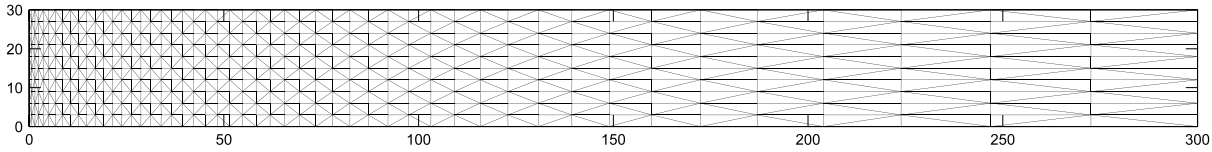

Fig. 2. The initial mesh produced by monitor (4.9).

5. Numerical tests

In this section, we use several examples to test our algorithms. Our simulations are based on the adaptive finite element library AFEPack [45].

5.1. Buckley–Leverett problem in one dimension

We first take the Buckley–Leverett problem [46] in a homogeneous medium as an example to verify the effectiveness of our method. Consider a 1D horizontal domain of length 300 m filled with oil at t_0. Water is injected with a constant flow rate at one end to displace oil to the other end. The pressure is kept constant at the product end and the capillary pressure is neglected. We assume the same viscosity for the oil and water phases for one case and change the viscosity ratio for the other cases, and we use linear relative permeability functions. The relative permeabilities are given by

$$k_{rw} = S_e^m; \qquad k_{rn} = (1 - S_e)^m, \tag{5.1}$$

where $m = 1$ (or $m = 2$) for linear (or quadratic) relative permeabilities, and S_e is the normalized saturation, defined as

$$S_e = \frac{S_w - S_{rw}}{1 - S_{rw} - S_{rn}}, \tag{5.2}$$

where S_{rw} and S_{rn} are the residual saturations for the wetting and non-wetting phases, respectively. Other relevant data are listed in Table 1.

As for the moving mesh procedure, after some experiments, β and α in (4.9) are chosen to be quite large; in our case they are 30 000 and 500, respectively, and μ in Eq. (4.8) is chosen to be 500. The monitor function (4.9) is further smoothed by repeatedly applying Eqs. (4.5) and (4.6) a few times (normally about three times), to prevent very singular meshes and to avoid large approximation error around the sharp interfaces.

In Fig. 3, the solutions generated by using the MFE–FV method with and without the moving mesh algorithm are plotted. Both solutions are obtained on a mesh of 50 cells. It is easy to see that the MFE–FV method with the moving mesh strategy generates sharper wetting fronts, as expected.

Fig. 4 shows the solutions generated by using the MFE–FV method with and without the moving mesh algorithm, obtained on meshes with various cell numbers. In our computations, the linear relative permeability function is used, together with $\mu_w/\mu_n = 2$. It is easy to see that the MFE–FV method with moving grids generates sharper wetting-phase fronts, as expected. The L_1-error of each solution is provided in Table 2. According to the data, very fine grids are needed for the uniform mesh computations in order to reduce the L_1-error to be below 0.5. However, this can be achieved by using fewer than 50 grid points with the moving mesh method.

5.2. Heterogeneous media problem in two dimensions

Consider a 2D horizontal domain of size 100 m by 100 m. The lower-right half of the domain ($x \geq y$) contains a low-permeability (50 md) medium. The remaining part of the domain is filled with a high-permeability (100 md) medium. Water is uniformly injected across the lower-left corner of the domain, replacing initially saturated oil. The production is across the opposite upper-right corner. The relevant data can be found in Table 3.

We choose β and α in monitor function (4.9) to be 6400 and 100, respectively. μ in Eq. (4.8) is set to 100. Eq. (4.9) is further smoothed by repeatedly applying Eqs. (4.5) and (4.6) 16 times.

Fig. 5 plots the mesh structures and contours of S_w at different times. We can easily see the effect of heterogeneous permeability on two-phase flow problems. The injected water flows faster in the more permeable region, as expected.

Table 1
Relevant data for the example in Section 5.1.

Domain dimensions	300 m × 1 m × 1 m
Rock properties	$\phi = 0.2,\ k = 1$ md
Fluid properties	$\mu_w(cP)/\mu_n(cP) = 1/1, 2/1, 2/3$
	$\rho_w = \rho_n = 1000$ kg/m^3
Relative permeabilities	Eq. (4.1)
Capillary pressure	Neglected
Residual saturations	$S_{rw} = 0,\ S_{rn} = 0.2$
Injection rate	5×10^{-4} PV/day
Mesh size	50 cells

Table 2
L_1-error of the MFE–FV method without and with the moving mesh strategy applied to the Buckley–Leverett problem as given in Fig. 4.

MFE–FV		Moving mesh	
Grid–blocks	L_1-error	Grid–blocks	L_1-error
50	1.8886	50	0.4164
100	1.0669		
200	0.4746		
400	0.1786		

Table 3
Relevant data for the example in Section 5.2.

Domain dimensions	100 m × 100 m × 1 m
Rock properties	$\phi = 0.2,\ k = 100$ md
Fluid properties	$\mu_w(cP)/\mu_n(cP) = 1/0.45$
	$\rho_w = \rho_n = 1000$ kg/m^3
	$\rho_n = \rho_n = 660$ kg/m^3
Relative permeabilities	Eq. (4.1), $m = 2$
Capillary pressure	Neglected
Residual saturations	$S_{rw} = 0,\ S_{rn} = 0$
Injection rate	0.10 PV/year
Mesh size	2.5 m, 3702 triangles

To check the accuracy of our method, we refine the mesh to $N = 128^2$, and carry out the same simulation without the moving mesh procedure. The result obtained with such a fine uniform mesh is then compared to that generated by the moving mesh method. In Fig. 6, we plot both of the contours of wetting-phase saturation at 5 years. As we can see, the shapes of the wetting-phase fronts are similar, and so are the contour lines.

However, there are still noticeable differences between the two figures in Fig. 6, especially in regions near the line $x = y$, which is the interface of the high-permeability and low-permeability media. Since the monitor function is based on the value of the wetting-phase saturations, the mesh grids will only cluster to the flow interface, and thus cannot well resolve the media interface. This may be the main reason for the disagreement of the two contours. To overcome such a problem, we can refine the mesh a little bit, for example, refine it to $N = 80^2$.

We present the change of the location of the wetting-phase fronts with time in Fig. 7. The x-axis represents the positions of the wetting-phase fronts on the lower boundary, while the y-axis represents the PVI (pore volume injected). The dotted and dashed lines represent the values associated with the $N = 20^2$ and $N = 40^2$ moving meshes, respectively. These results are fairly close to the results obtained with the $N = 128^2$ uniform mesh, which is shown by the solid line.

5.3. Heterogeneous media problem with a circular poor permeable region

This example is almost identical to the example in the last section, except that the 2D horizontal domain consists of two permeable media in a different manner. The circular region, whose radius is 20 m and center is (50, 50), comprises a poor-permeability (1 md) medium. The remaining part of the domain is filled with a high-permeability (100 md) medium. Water is uniformly injected across the lower-left corner of the domain, replacing initially saturated oil. The production is across the opposite upper-right corner. Other relevant data are the same as those in Table 3.

According to the new permeability condition, it may be unwise to consider very long-range diffusion, so we set μ in Eq. (4.8) to 10 instead of 100. β and α in monitor function (4.9) are set to 6400 and 100, respectively, without change. Eq. (4.9) is further smoothed by repeatedly applying Eqs. (4.6) and (4.7) 16 times.

Fig. 8 shows the mesh structure and solution contours of S_w at different times. Since the capillary pressure is neglected, there is no cross-flow between different media. Therefore, the injected water flows faster in the high-permeability region, and almost completely circumvents the poor-permeability region.

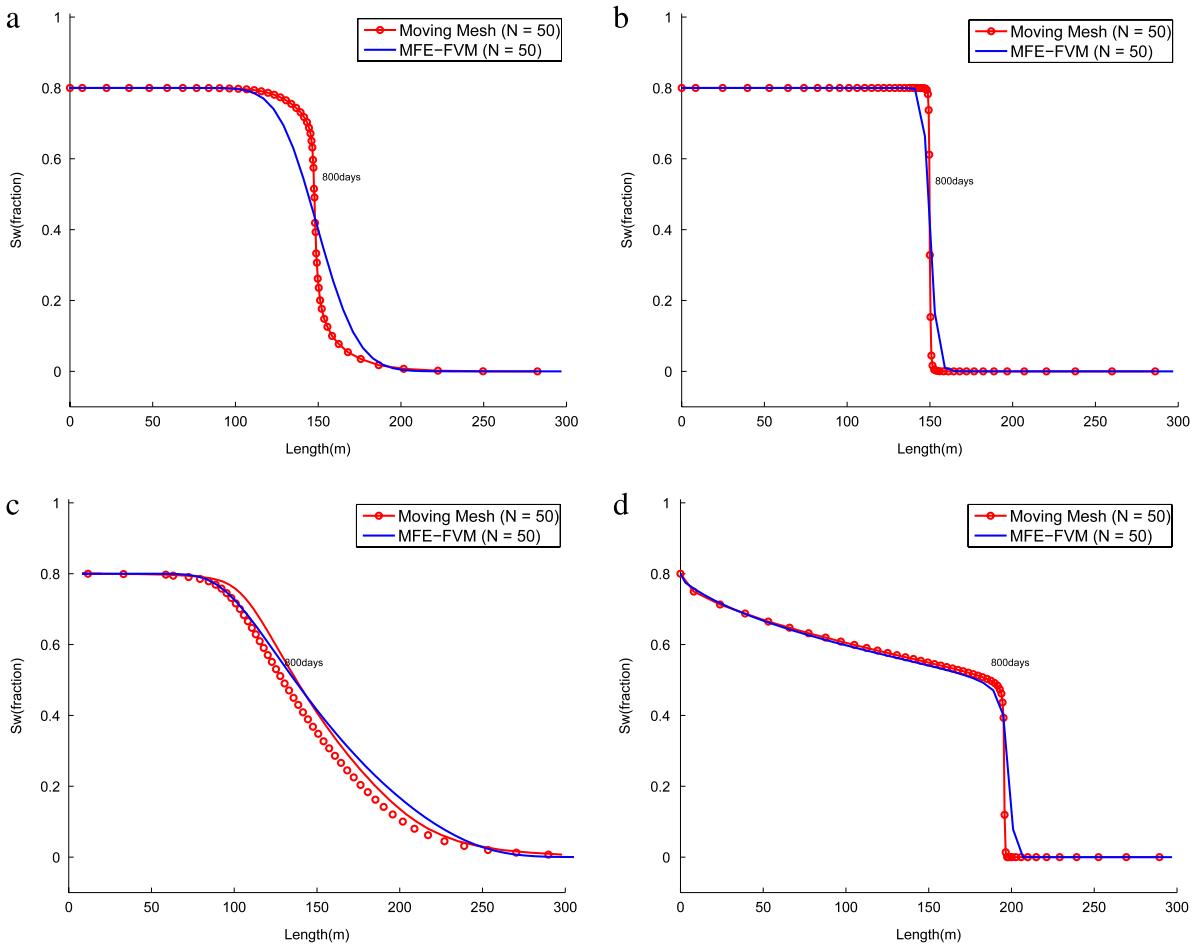

Fig. 3. Solution of the Buckley–Leverett problem with different relative permeabilities and viscosity ratios. (a) Linear relative permeabilities: $\mu_w/\mu_n = 1$. (b) Linear relative permeabilities: $\mu_w/\mu_n = 2$. (c) Linear relative permeabilities: $\mu_w/\mu_n = 2/3$. (d) Quadratic relative permeabilities: $\mu_w/\mu_n = 2/3$.

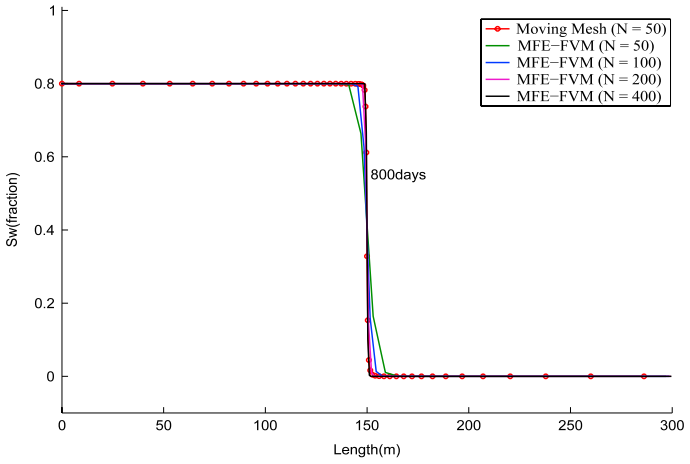

Fig. 4. Solution of the Buckley–Leverett problem with linear relative permeabilities and μ_w/μ_n as 2.

To check the accuracy of our method, we refine the mesh to $N = 128^2$, and carry out the same simulation without the moving mesh procedure. The result obtained with such a fine uniform mesh is then compared to that generated by the moving mesh method. In Fig. 9, we plot both of the contours of wetting-phase saturation at 5 years. It is easy to see that the wetting-phase front shapes of the uniform and moving meshes are very similar to each other.

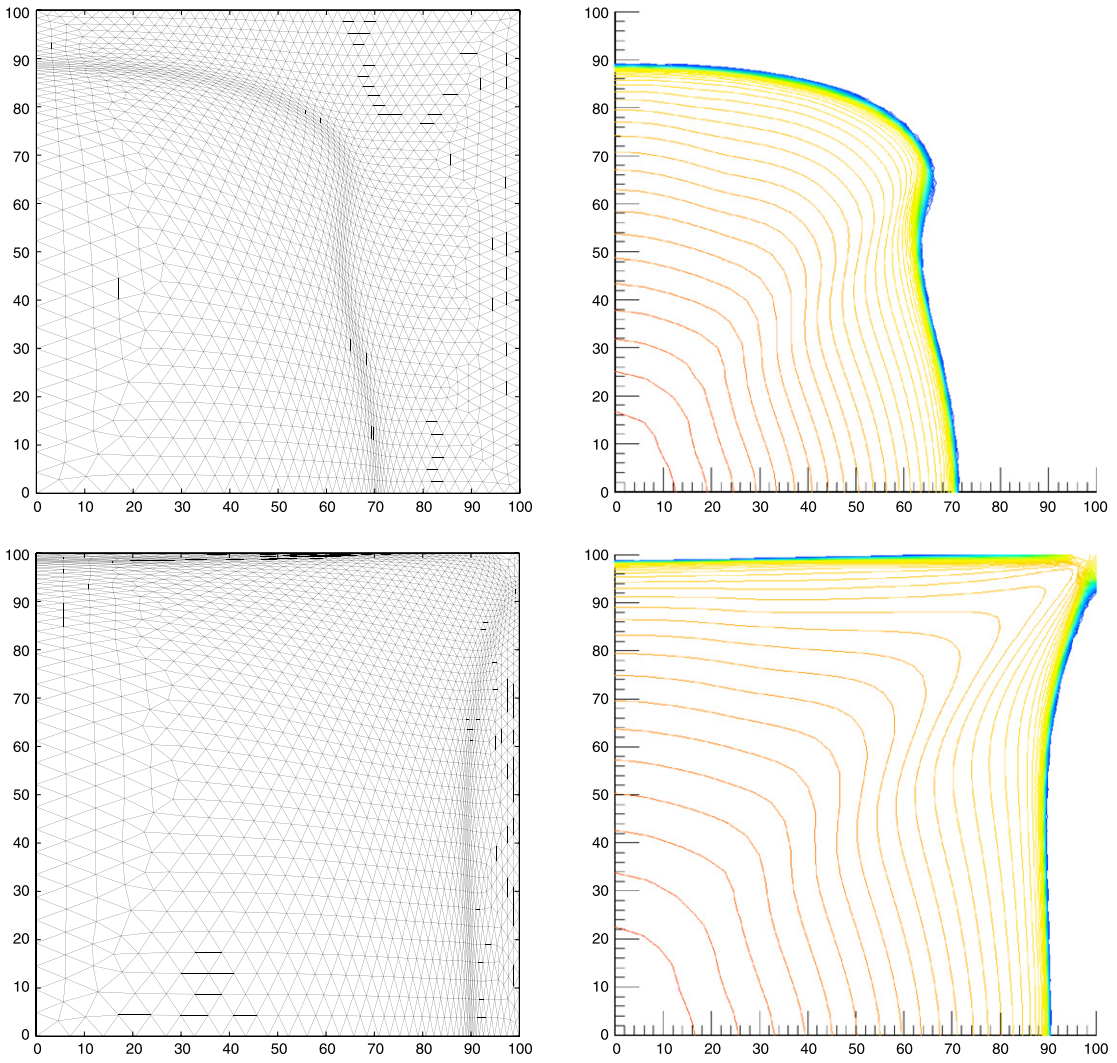

Fig. 5. The adaptive mesh and the contours of the wetting-phase saturation for the example in Section 5.2. Upper: PVI = 0.5. Lower: PVI = 0.9.

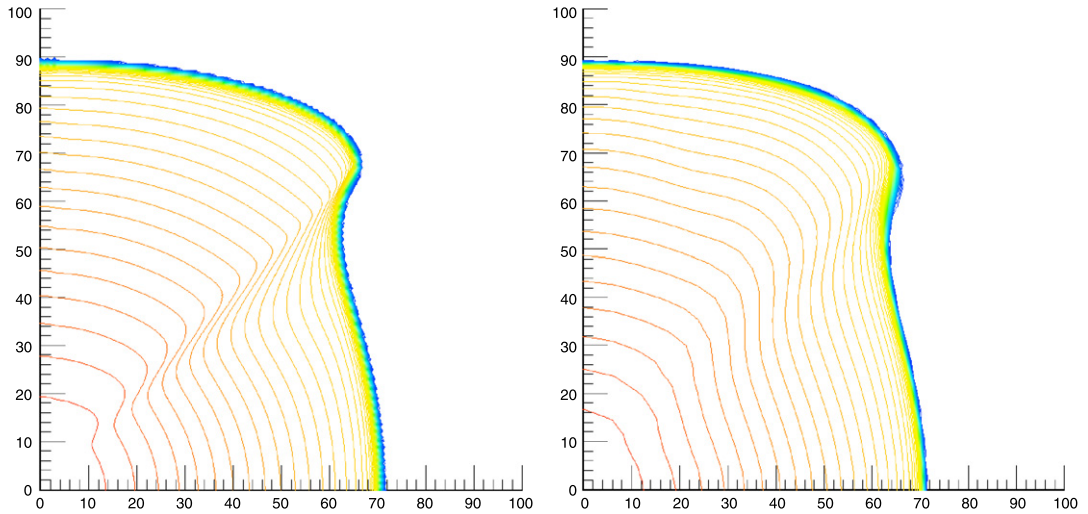

Fig. 6. The contours of the wetting-phase saturation for the example in Section 5.2 at 0.5/0.10 years. The left figure is obtained with a fine uniform mesh, while the right one is generated by the moving mesh method.

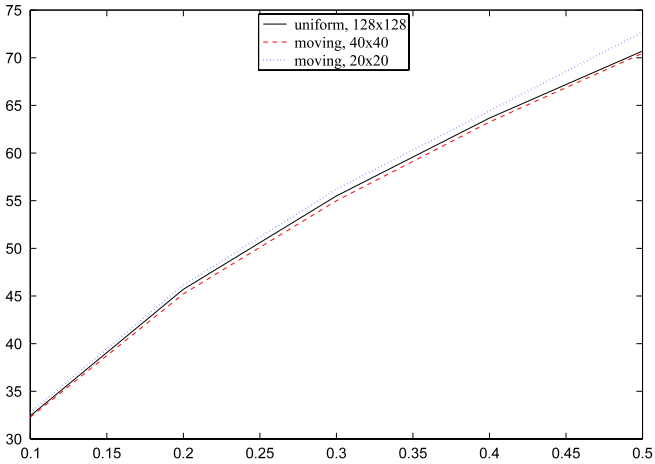

Fig. 7. Example in Section 5.2: the change in the location of the wetting-phase front with time.

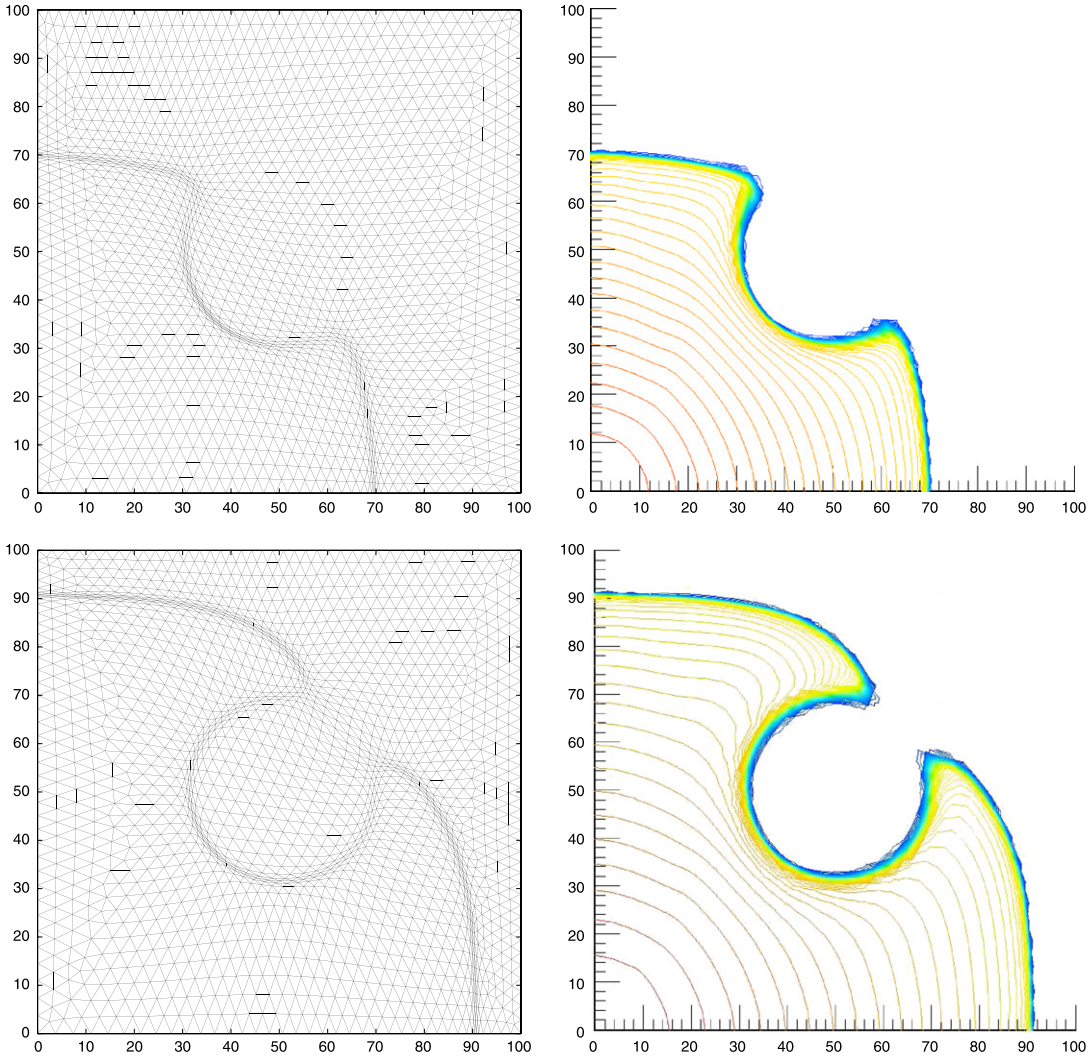

Fig. 8. The adaptive mesh and the contours of the wetting-phase saturation for the example in Section 5.3. Upper: PVI = 0.3. Lower: PVI = 0.5.

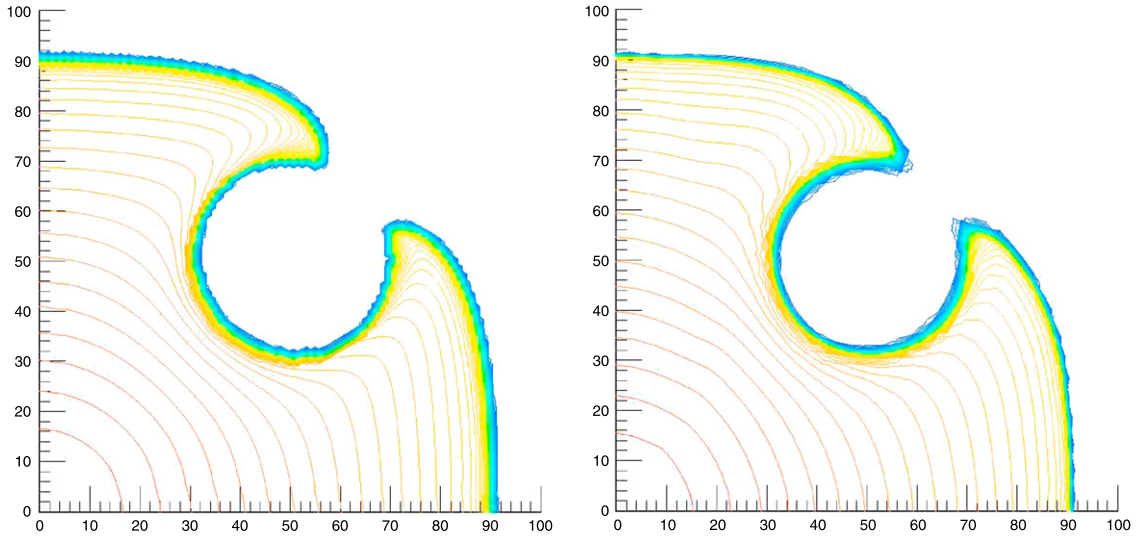

Fig. 9. The contours of the wetting-phase saturation for the example in Section 5.3 at 5 years. The left figure is obtained with a fine uniform mesh, while the right one is generated by the moving mesh method.

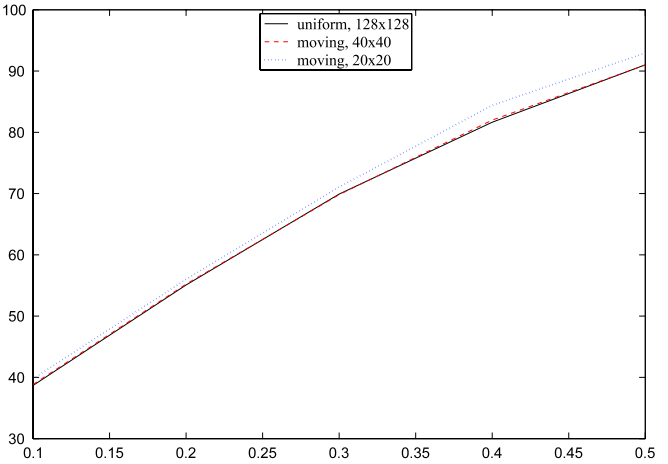

Fig. 10. Example in Section 5.3: the change in the location of the wetting-phase fronts with time.

Compared to Fig. 6, it is found that the difference between the two figures obtained in Fig. 9 is much smaller. The main reason may be the extremely low permeability in the center circular region, which causes almost no water to flow into the center. Consequently, the mesh can resolve the flow interface and the media interface at the same time. Such a case, although rare in practical computations, provides a valuable idea for solving problems with more complicated permeability conditions. By choosing suitable monitor functions, it might be possible to move more grid points to both the flow interface and the media interface.

We close this section by plotting the locations of the wetting-phase fronts against time in Fig. 10. The x-axis represents the positions of the wetting-phase fronts on the lower boundary, while the y-axis represents the PVI. The dotted and dashed lines represent the values associated with the $N = 20^2$ and $N = 40^2$ moving meshes, respectively. While the dotted line is close to the uniform fine mesh result (the solid line), the dashed line nearly overlaps the solid line, showing the good accuracy of our method.

6. Conclusions

We have applied the moving mesh method to the mixed finite element–finite volume (MFE–FV) method for two-phase flow problems. We solve the governing equations by discretizing the velocity equation and the volumetric balance equation individually, based on a hybrid mixed finite element approach, together with a finite volume method for solving the saturation equation. After obtaining the solutions at each time level, the solutions are used to redistribute the mesh grids based on a moving mesh strategy. The main idea of the moving mesh method is to employ some suitable monitor functions and

moving mesh equations to move more grid points near the improvement of the moving mesh strategy over the fixed mesh approach has been demonstrated by several numerical experiments. It is observed that using the moving mesh approach can give accurate numerical solutions with fewer degrees of freedoms.

Acknowledgments

The research of H. Dong and T. Tang is mainly supported by Hong Kong Research Council GRF Grants and Hong Kong Baptist University FRG grants. The research of Z. Qiao is partially supported by the Hong Kong RGC grant PolyU 2021/12P. S. Sun is partially supported by KAUST's GRP University Research Funds.

References

[1] W. Brand, J. Heinemann, K. Aziz, The grid orientation effect in reservoir simulation, in: Symposium on Reservoir Simulation, No. SPE21228, 1991.
[2] E. Nacul, K. Aziz, Use of irregular grid in reservoir simulation, in: Annual Technical Conference and Exhibition, No. SPE22886, 1991.
[3] G. Chavent, A. Salzano, A finite-element method for the 1-D water flooding problem with gravity, J. Comput. Phys. 45 (1982) 307–344.
[4] P. Song, S. Sun, Contaminant flow and transport simulation in cracked porous media using locally conservative schemes, Adv. Appl. Math. Mech. 4 (2012) 389–421.
[5] S. Sun, M.F. Wheeler, Anisotropic and dynamic mesh adaptation for discontinuous Galerkin methods applied to reactive transport, Comput. Methods Appl. Mech. Engrg. 195 (2006) 3382–3405.
[6] F. Brezzi, M. Fortin, Mixed and Hybrid Finite Element Methods, Environmental Engineering, Springer-Verlag, New York, 1991.
[7] L.J. Jiang, I.D. Mishev, Mixed multiscale finite volume methods for elliptic problems in two-phase flow simulations, Commun. Comput. Phys. 11 (2012) 19–47.
[8] P. Raviart, J. Thomas, A Mixed Hybrid Finite Element Method for the Second Order Elliptic Problem, in: Lectures Notes in Mathematics, vol. 606, Springer-Verlag, 1977.
[9] J. Moortgat, S. Sun, A. Firoozabadi, Compositional modeling of three-phase flow with gravity using higher-order finite element methods, Water Resour. Res. 47 (2011) W05511.
[10] S. Sun, A. Firoozabadi, Compositional modeling in three-phase flow for CO_2 and other fluid injections using higher-order finite element methods, in: 2009 SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 4–7 October 2009.
[11] L.V. Branets, S.S. Ghai, S.L. Lyons, X.H. Wu, Challenges and technologies in reservoir modeling, Commun. Comput. Phys. 6 (2009) 1–23.
[12] H.Y. Wang, R. Li, T. Tang, Efficient computation of dendritic growth with r-adaptive finite element methods, J. Comput. Phys. 227 (2008) 5984–6000.
[13] P. Yue, C. Zhou, J.J. Feng, C.F. Ollivier-Gooch, H.H. Hu, Phase-field simulations of inter-facial dynamics in viscoelastic fluids using finite elements with adaptive meshing, J. Comput. Phys. 219 (2006) 47–67.
[14] Koji Moriyama, Takaji Inamuro, Lattice Boltzmann simulations of water transport from the gas diffusion layer to the gas channel in PEFC, Commun. Comput. Phys. 9 (2011) 1206–1218.
[15] Toshiro Murayama, Masato Yoshino, Tetsuo Hirata, Three-dimensional lattice Boltzmann simulation of two-phase flow containing a deformable body with a viscoelastic membrane, Commun. Comput. Phys. 9 (2011) 1397–1413.
[16] I. Aavatsmark, An introduction to multipoint flux approximations for quadrilateral grids, Comput. Geosci. 6 (2002) 405–432.
[17] R. Juanes, J. Kim, S. Matringe, K. Thomas, Implementation and application of a hybrid multipoint flux approximation for reservoir simulation on corner-point grids, in: Annual Technical Conference and Exhibition, No. SPE95928, 2005.
[18] J. Nordbotten, J. Brunsgt, Discretization on quadrilateral grids with improved monotonicity properties, J. Comput. Phys. 203 (2002) 744–760.
[19] S.N. Antontsev, A.V. Kazhikhov, V.N. Monakhov, Boundary Value Problems in Mechanics of Nonhomogeneous Fluids, North-Holland, 1990.
[20] G. Chavent, J. Jaffré, Mathematical Models and Finite Elements for Reservoir Simulation, in: Studies in Mathematics and its Applications, Springer-Verlag, 1986.
[21] Z. Chen, G. Huan, Y. Ma, Computational Methods for Multiphase Flows in Porous Media, SIAM, 2006.
[22] G. Chavent, G. Cohen, J. Jaffré, R. Eyard, R. Dominique, L. Weill, Discontinuous and mixed finite elements for two-phase incompressible flow, SPE Reservoir Eng. 5 (1990) 567–575.
[23] Z. Chen, R. Ewing, From single-phase to compositional flow: applicability of mixed finite elements, Transp. Porous Media 57 (1997) 225–242.
[24] Z. Chen, R. Ewing, Q. Jiang, A. Spagnuolo, Degenerate two-phase incompressible flow V: characteristic finite element methods, J. Numer. Math. 10 (2002) 87–107.
[25] R. Ewing, R. Heinemann, Incorporation of mixed finite element methods in compositional simulation for reduction of numerical dispersion, in: Reservoir Simulation Symposium, No. SPE12267, 1983.
[26] G. Chavent, J. Jaffré, J. Roberts, Generalized cell-centered finite volume methods: application to two-phase flow in porous media, in: Computational Science for the 21st Century, Chichester, 1997, pp. 231–241.
[27] H. Hoteit, A. Firoozabadi, Numerical modeling of two-phase flow in heterogeneous permeable media with different capillarity pressures, Adv. Water Resour. 31 (2008) 56–73.
[28] R. Li, P. Zhang, T. Tang, Moving mesh methods in multiple dimensions based on harmonic maps, J. Comput. Phys. 170 (2001) 562–588.
[29] H.Z. Tang, T. Tang, Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws, SIAM J. Numer. Anal. 41 (2001) 487–515.
[30] S. Sun, A. Firoozabadi, J. Kou, Numerical modeling of two-phase binary fluid mixing using mixed finite elements, Comput. Geosci. 16 (2012) 1101–1124.
[31] R.J. LeVeque, Finite Volume Methods for Hyperbolic Problems, Cambridge University Press, Cambridge, 2002.
[32] M.J. Baines, M.E. Hubbard, P.K. Jimack, Velocity-based moving mesh methods for nonlinear partial differential equations, Commun. Comput. Phys. 10 (2011) 509–576.
[33] C.-Q. Jin, K. Xu, S. Chen, A three dimensional gas-kinetic scheme with moving mesh for low-speed viscous flow computations, Adv. Appl. Math. Mech. 2 (2010) 746–762.
[34] Z. Qiao, Numerical investigations of the dynamical behaviors and instabilities for the Gierer–Meinhardt system, Commun. Comput. Phys. 3 (2008) 406–426.
[35] P.A. Zegeling, I. Lagzi, F. Izsk, Transition of Liesegang precipitation systems: simulations with an adaptive grid PDE method, Commun. Comput. Phys. 10 (2011) 867–881.
[36] Y.-B. Zhang, T. Tang, Simulating three-dimensional free surface viscoelastic flows using moving finite difference schemes, Numer. Math. Theory Methods Appl. 4 (2011) 92–112.
[37] W.Z. Huang, R.D. Russell, Adaptive Moving Mesh Methods, Springer Science, 2011.
[38] B. van Leer, Towards the ultimate conservative difference schemes V. A second-order sequel to Godunov's method, J. Comput. Phys. 32 (1979) 101–136.
[39] G.X. Chen, H.Z. Tang, P.W. Zhang, Second order accurate Godunov scheme for multicomponent flows on moving triangular meshes, J. Sci. Comput. 34 (2008) 64–86.
[40] P. He, H.Z. Tang, An adaptive moving mesh method for two-dimensional relativistic hydrodynamics, Commun. Comput. Phys. 11 (2012) 114–146.
[41] A. van Dam, P.A. Zegeling, Balanced monitoring of flow phenomena in moving mesh methods, Commun. Comput. Phys. 7 (2010) 138–170.
[42] G. Hu, Z. Qiao, T. Tang, Moving finite element simulations for reaction–diffusion systems, Adv. Appl. Math. Mech. 4 (2012) 365–381.
[43] Z. Qiao, Z. Zhang, T. Tang, An adaptive time-stepping strategy for the molecular beam epitaxy models, SIAM J. Sci. Comput. 33 (2011) 1395–1414.
[44] Z. Zhang, Z. Qiao, An adaptive time-stepping strategy for the Cahn–Hilliard equation, Commun. Comput. Phys. 11 (2011) 1261–1278.
[45] R. Li, W.B. Liu, http://dsec.pku.edu.cn/~rli/software.php#AFEPack.
[46] S. Buckley, M. Leverett, Mechanism of fluid displacement in sands, Trans. AIME 146 (1942) 187–196.