Journal of Petroleum Science and Engineering 177 (2019) 41–52

Contents lists available at ScienceDirect

Journal of Petroleum Science and Engineering

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

Embedded discrete fracture modeling for compositional reservoir simulation using corner-point grids

Yifei Xu[a,*], Bruno Ramon Batista Fernandes[a], Francisco Marcondes[b], Kamy Sepehrnoori[a]

[a] Hildebrand Department of Petroleum and Geosystems Engineering, The University of Texas at Austin, Austin, TX, USA
[b] Department of Metallurgical Engineering and Material Science, Federal University of Ceará, Fortaleza, Ceará, Brazil

ARTICLE INFO

Keywords:
Embedded discrete fracture model
Corner-point grid
Complex fractures
Full-permeability tensor formulation
Water flooding

ABSTRACT

Corner-point is an industry-standard type grid for application in reservoir simulation. The flexible gridding in corner-point grids provides several advantages over Cartesian grids for the representation of complex geological features. In this work, an embedded discrete fracture model (EDFM) is extended to corner-point grids with a full-permeability-tensor formulation to simulate complex fractures in this type of grids. The developed model is implemented in an IMPEC, compositional reservoir simulator. We first describe the formulations of the full-permeability-tensor implementation together with the modified governing equations for EDFM simulations. Then, we present methodologies for computing matrix-fracture intersections and fracture-fracture connections in the EDFM considering the various block geometries in corner-point grids. Subsequently, case studies are presented to verify the developed model, where the impacts of grid distortion and cross derivatives on simulation results are discussed. Three-dimensional case studies are also shown to illustrate the influence of natural fractures on secondary recovery. The results of this study demonstrate the reliability of the developed model, and they also show the compatibility of the EDFM with different types of numerical solution schemes in existing simulators.

1. Introduction

Corner-point is widely used in reservoir simulation due to its flexibility to represent geological features, such as faults, and complex boundaries of hydrocarbon reservoirs. Different from Cartesian grids, the locations of corner points for each block are defined using coordinate lines and depths (Wadsley, 1980; Ponting, 1989). For each gridblock, the coordinates of each corner point need to be known to define the gridblock geometry. Most commercial reservoir simulators support this type of grid.

Although corner-point grids give a better representation of the reservoir geometry, the numerical solution scheme used in conjunction with the corner-point grids is much more complicated than the one using Cartesian grids because in general, the grid is non-orthogonal. For three dimensional problems, this typically results in a 19- or 27-point stencil (Aavatsmark et al., 1998; Marcondes et al., 2008). To reduce the computational costs from the use of non-orthogonal grids, most commercial simulators use only two points to evaluate the potential difference, which is known as two-point flux approximation (TPFA), and neglect the additional terms in the Jacobian matrix. TPFA speeds up simulations, but it introduces numerical errors which do not decrease when grid refinement is performed (Marcondes et al., 2005; Aavatsmark, 2007). The reason is that when we neglect terms from the approximate equations, they will not approach the original set of differential equations because the approximate equations represent another set of partial differential equations. We can only neglect the error when using TPFA if the grid is not highly distorted. However, the rule of thumb for modeling complex reservoirs is the use of highly distorted grids. Therefore, full-permeability-tensor formulations need to be implemented to improve the accuracy of simulation results (Hegre et al., 1986; Aavatsmark et al., 1998; Lee et al., 1998; Marcondes et al., 2005, 2008).

Simulation of fractures adds another dimension of complexity to the numerical formulations. As a traditional type of grid, corner-point grids are frequently used for simulating different types of reservoirs, including naturally fractured reservoirs and hydraulically fractured reservoirs. Therefore, it is necessary to develop methods to simulate the fractures in models using corner-point grids. A traditional method for the simulation of naturally fractured reservoirs is the dual-porosity model (Warren and Root, 1963). By treating the fractures and matrix as two continua, the dual-porosity model separates the reservoir into two systems, one with high flow capacity and one with high storage

* Corresponding author.
E-mail address: yifei.xu@utexas.edu (Y. Xu).

https://doi.org/10.1016/j.petrol.2019.02.024
Received 19 June 2018; Received in revised form 15 January 2019; Accepted 8 February 2019
Available online 11 February 2019

capacity. This simplified approach ignores the location or geometry of individual fractures and assumes a well-connected fracture system. However, in realistic reservoirs, the connectivity between fractures may not be guaranteed (Li and Lee, 2008). In such cases, an explicit representation of the contribution from each fracture is desirable.

Discrete fracture models were developed to explicitly simulate the impact of individual fractures. To accurately describe the fracture geometries, unstructured grids were often used in these models (Matthai et al., 2007; Marcondes et al., 2010; Karimi-Fard and Durlofsky, 2016). Although these models were able to capture the detailed geometry of each fracture, several disadvantages exist in these models. The disadvantages include the difficulties of gridding, especially at fracture intersections for 3D simulations, the low computational efficiency together with unstructured grids, and the incompatibility with existing simulators and reservoir models (Fumagalli et al., 2017).

The embedded discrete fracture model (EDFM) was developed as a solution to these issues. The initial idea of EDFM was proposed by Hearn et al. (1997). In their work, additional gridblocks were used to represent the channels, and non-neighboring connections (NNCs) were added to describe the fluid flow between the matrix and fractures wherever a grid block was crossed by a fracture. Li and Lee (2008) further developed the model and proposed the formulations for transmissibility calculations in two-dimensional studies. Moinfar et al. (2014) improved the EDFM and extended it for three-dimensional studies. The EDFM approach has been applied to several in-house and commercial simulators (Panfili and Cominelli, 2014; Shakiba and Sepehrnoori, 2015; Xu, 2015; Xu et al., 2017b; Tang et al., 2017). Studies have shown that the EDFM can accurately simulate different geometries of fractures while keeping a relatively high computational efficiency compared to other methods (Xu et al., 2017a; Du et al., 2017). Also, the compatibility of EDFM with existing reservoir simulators makes it convenient to apply the EDFM in real-life studies.

As the EDFM combines the advantages of conventional reservoir simulators and discrete fracture modeling, it is of great importance to combine the EDFM with different types of grids in reservoir simulation. Most of the previous works using EDFM have been conducted with Cartesian grids (Li and Lee, 2008; Moinfar et al., 2014; Cavalcante-Filho et al., 2015; Du et al., 2017; Cavalcante-Filho and Sepehrnoori, 2017), due to the simplicity of implementation. However, as corner-point grids being the industry standard, it is very beneficial to apply the EDFM to corner-point grids for the study of realistic reservoirs. However, existing studies mentioning corner-point grids with EDFM (Panfili et al., 2015; Fumagalli et al., 2016; de Sousa Junior et al., 2016) have not comprehensively considered the complexities pertaining to the numerical formulations and geometrical calculations.

In this work, the EDFM is implemented in UTCOMP-RS, an IMPEC, multi-phase, multi-component, in-house reservoir simulator (Chang, 1990) with corner-point grids (Fernandes et al., 2014, 2018). Full-permeability-tensor formulation is used for the flux calculation between matrix, and non-neighboring connections are constructed to simulate the fluid flow from the matrix to fractures and from fractures to fractures. The overall objective is to extend the EDFM formulation to corner-point grids with a full-permeability-tensor implementation. We first describe the basic equations of the full-permeability-tensor corner-point grid. After that, the simulator governing equations with the EDFM, the formulations to calculate EDFM transmissibilities, and the issues with geometrical calculations are discussed. Then we present case studies using corner-point grids with different level of distortion to verify the model and study the influence of grid distortion and cross derivatives on simulation results. Finally, the EDFM is applied to highly distorted corner-point grids to simulate the impact of fractures on the recovery processes in geologically complex reservoirs. This study provides a convenient approach to fracture simulation in reservoir models using corner-point grids.

2. Governing equations of UTCOMP-RS

One of the most important equation in the simulator is the material balance equation. Material balance equations describe the mole conservation across the reservoir domain. In UTCOMP-RS, the fluid flow in the porous media is modeled using Darcy's law. Local phase equilibrium is assumed for the hydrocarbon phases, while no mass transfer between the water phase and hydrocarbon phases is considered. Therefore, the material balance equation for each component without considering fractures is given by Eq. (1), where the advective and dispersion fluxes are given by Eqs. (2) and (3), respectively.

$$\frac{\partial N_k}{\partial t} = F_k + D_k + q_k, \quad k = 1, \ldots, n_c + 1, \tag{1}$$

$$F_k = V_b \sum_{j=1}^{n_p} \vec{\nabla} \cdot \left(x_{kj} \xi_j \frac{k_{rj}}{\mu_j} \overrightarrow{K} \cdot (\vec{\nabla} P_j - \rho_j g \vec{\nabla} D) \right), \tag{2}$$

$$D_k = V_b \sum_{j=1}^{n_p} \vec{\nabla} \cdot \left(\phi S_j \xi_j \overrightarrow{\Lambda_{kj}} \cdot \vec{\nabla} x_{kj} \right), \tag{3}$$

where F_k is the molar flow rate of component k due to convection, D_k is the molar flow rate of component k due to physical dispersion, q_k is the injection/production molar rate of component k from wells, V_b is the bulk volume, N_k is the number of moles of component k, t is the time, n_p is the number of fluid phases, x_{kj} is the mole fraction of component k in phase j, ξ_j is the molar density of phase j, k_{rj} is the relative permeability of phase j, μ_j is the viscosity of phase j, $\overrightarrow{K}$ is the permeability tensor, P_j is the pressure of phase j, ρ_j is the mass density of phase j, g is the gravity acceleration, D is the depth, S_j is the saturation of phase j, ϕ is the porosity, $\overrightarrow{\Lambda_{kj}}$ is the dispersion tensor of component k in phase j, and n_c is the number of hydrocarbon components. Component $n_c + 1$ denotes the water component. The phase pressure is computed using the capillary relationships (Eq. (4)) and assuming oil phase as the reference

$$P_j = P + P_{cjo}, \quad j = 1, \ldots, n_p, \tag{4}$$

where P is the pressure of the oil phase (or any other reference phase in the absence of oil) and P_{cjr} is the capillary pressure between the reference phase and phase j. In this work, the oil phase is labeled as phase 2.

The Peng-Robinson equation of state (Peng and Robinson, 1976) is used to compute densities and fugacities. Phase stability analysis is used to determine phase appearance and disappearance. The stationary point location method (Michelsen, 1982) and the Gibbs free energy minimization method (Trangenstein, 1987; Perschke, 1988) are implemented in UTCOMP-RS for phase stability calculation. The determination of phase composition and amount is performed through a flash calculation. In UTCOMP-RS, the Accelerated Successive Substitution (ACSS) (Mehra et al., 1983) and the modified Gibbs free energy minimization (Perschke, 1988) are combined to achieve a better convergence rate.

3. Full-permeability-tensor formulation

3.1. Transformed matrix material balance equation

The non-orthogonal boundary fitted approach (Chu, 1971; Thompson et al., 1974) is used to obtain the approximate equations for the matrix using the finite-volume method. The corner-point approach presented here is based on the work of Marcondes et al. (2005) and Marcondes et al. (2008) but applied to an IMPEC formulation. The first step in this approach is to transform the partial differential equations from the physical domain (x, y,z) to a computational domain (ξ, η, γ) where the calculations can be performed in a regular domain. The coordinates of the transformed plane are all functions of the (x, y, z)

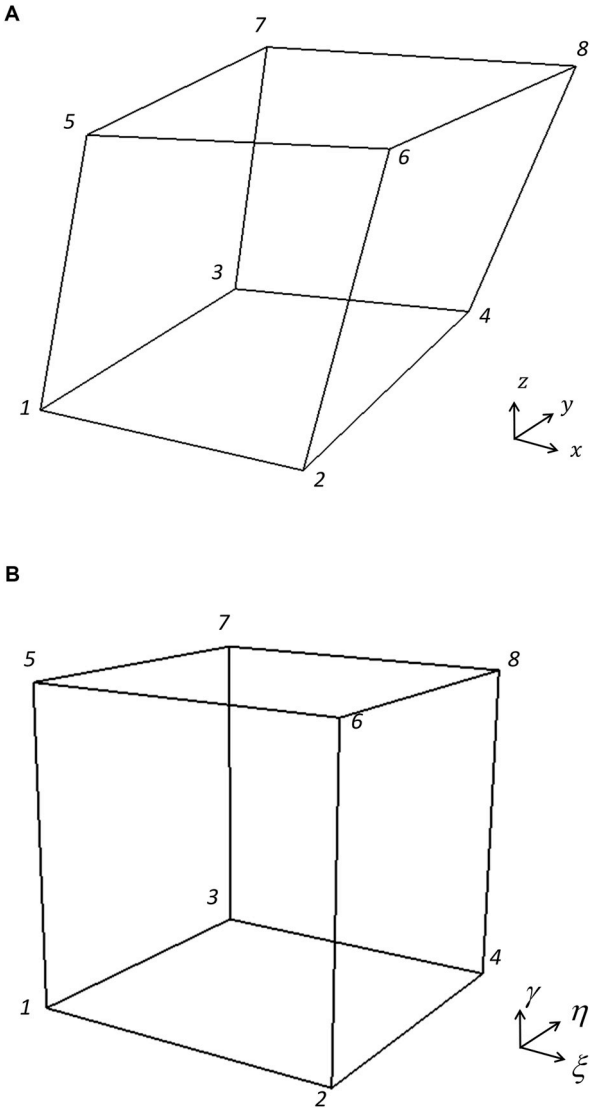

Fig. 1. Transformation from the physical domain (x, y,z) to the computational domain (ξ, η, γ). (a) Physical domain. (b) Computational domain.

coordinates and are designed in such a way that the grid lines are always tangent to ξ, η, and γ, as shown in Fig. 1.

Neglecting the physical dispersion terms for simplification purpose only, the material balance equations in the computational domain for hydrocarbon components in the matrix medium are given by

$$\begin{aligned}
&\frac{\partial N_k}{\partial t} = q_k \\
&+ \frac{\partial}{\partial \xi}\left[D_{11}\sum_{j=1}^{n_p}\left(x_{kj}\xi_j\frac{k_{rj}}{\mu_j}\frac{\partial \Phi_j}{\partial \xi}\right)\right] + \frac{\partial}{\partial \xi}\left[D_{12}\sum_{j=1}^{n_p}\left(x_{kj}\xi_j\frac{k_{rj}}{\mu_j}\frac{\partial \Phi_j}{\partial \eta}\right)\right] \\
&+ \frac{\partial}{\partial \xi}\left[D_{13}\sum_{j=1}^{n_p}\left(x_{kj}\xi_j\frac{k_{rj}}{\mu_j}\frac{\partial \Phi_j}{\partial \gamma}\right)\right] + \frac{\partial}{\partial \eta}\left[D_{21}\sum_{j=1}^{n_p}\left(x_{kj}\xi_j\frac{k_{rj}}{\mu_j}\frac{\partial \Phi_j}{\partial \xi}\right)\right] \\
&+ \frac{\partial}{\partial \eta}\left[D_{22}\sum_{j=1}^{n_p}\left(x_{kj}\xi_j\frac{k_{rj}}{\mu_j}\frac{\partial \Phi_j}{\partial \eta}\right)\right] + \frac{\partial}{\partial \eta}\left[D_{23}\sum_{j=1}^{n_p}\left(x_{kj}\xi_j\frac{k_{rj}}{\mu_j}\frac{\partial \Phi_j}{\partial \gamma}\right)\right] \\
&+ \frac{\partial}{\partial \gamma}\left[D_{31}\sum_{j=1}^{n_p}\left(x_{kj}\xi_j\frac{k_{rj}}{\mu_j}\frac{\partial \Phi_j}{\partial \xi}\right)\right] + \frac{\partial}{\partial \gamma}\left[D_{32}\sum_{j=1}^{n_p}\left(x_{kj}\xi_j\frac{k_{rj}}{\mu_j}\frac{\partial \Phi_j}{\partial \eta}\right)\right] \\
&+ \frac{\partial}{\partial \gamma}\left[D_{33}\sum_{j=1}^{n_p}\left(x_{kj}\xi_j\frac{k_{rj}}{\mu_j}\frac{\partial \Phi_j}{\partial \gamma}\right)\right], \quad k = 1,\ldots,n_c+1,
\end{aligned} \tag{5}$$

where Φ_j is the flow potential of phase j, which is given by

$$\Phi_j = P_j - \rho_j g D. \tag{6}$$

The coefficients of the D tensor involve the direct metrices of the transformation and the absolute permeabilities components, which are given by

$$\begin{aligned}
D_{11} &= \frac{1}{J_t}\left(\xi_x^2K_{xx} + \xi_y^2K_{yy} + \xi_z^2K_{zz} + 2\xi_x\xi_yK_{xy} + 2\xi_x\xi_zK_{xz} + 2\xi_y\xi_zK_{yz}\right); \\
D_{12} = D_{21} &= \frac{1}{J_t}\begin{pmatrix}\xi_x\eta_xK_{xx} + \xi_y\eta_yK_{yy} + \xi_z\eta_zK_{zz} + (\xi_x\eta_y + \xi_y\eta_x)K_{xy} \\ +(\xi_x\eta_z + \xi_z\eta_x)K_{xz} + (\xi_y\eta_z + \xi_z\eta_y)K_{yz}\end{pmatrix}; \\
D_{13} = D_{31} &= \frac{1}{J_t}\begin{pmatrix}\xi_x\gamma_xK_{xx} + \xi_y\gamma_yK_{yy} + \xi_z\gamma_zK_{zz} + (\xi_x\gamma_y + \xi_y\gamma_x)K_{xy} \\ +(\xi_x\gamma_z + \xi_z\gamma_x)K_{xz} + (\xi_y\gamma_z + \xi_z\gamma_y)K_{yz}\end{pmatrix}; \\
D_{22} &= \frac{1}{J_t}\left(\eta_x^2K_{xx} + \eta_y^2K_{yy} + \eta_z^2K_{zz} + 2\eta_x\eta_yK_{xy} + 2\eta_x\eta_zK_{xz} + 2\eta_y\eta_zK_{yz}\right); \\
D_{23} = D_{32} &= \frac{1}{J_t}\begin{pmatrix}\eta_x\gamma_xK_{xx} + \eta_y\gamma_yK_{yy} + \eta_z\gamma_zK_{zz} + (\eta_x\gamma_y + \eta_y\gamma_x)K_{xy} \\ +(\eta_x\gamma_z + \eta_z\gamma_x)K_{xz} + (\eta_y\gamma_z + \eta_z\gamma_y)K_{yz}\end{pmatrix}; \\
D_{33} &= \frac{1}{J_t}\left(\gamma_x^2K_{xx} + \gamma_y^2K_{yy} + \gamma_z^2K_{zz} + 2\gamma_x\gamma_yK_{xy} + 2\gamma_x\gamma_zK_{xz} + 2\gamma_y\gamma_zK_{yz}\right).
\end{aligned} \tag{7}$$

ξ_x, ξ_y, ξ_z, η_x, η_y, η_z, γ_x, γ_y, and γ_z are the direct metrics of the transformation and are evaluated numerically. J_t is the Jacobian of the transformation, which is computed as

$$J_t = \{x_\xi(y_\eta z_\gamma - y_\gamma z_\eta) - x_\eta(y_\xi z_\gamma - y_\gamma z_\xi) + x_\gamma(y_\xi z_\eta - y_\eta z_\xi)\}^{-1}. \tag{8}$$

If we set the volume of each control volume in the computational domain as a unity, the equivalent volume in the physical domain (V_b) is given by

$$V_b = \frac{1}{J_t}. \tag{9}$$

On the right-hand side of Eq. (5), the terms with D_{11}, D_{22}, and D_{33} are direct derivative terms, and the terms with D_{12}, D_{13}, D_{21}, D_{23}, D_{31}, and D_{32} are cross-derivative terms.

3.2. Approximated equations for the matrix

The approximate equation for the matrix can be obtained by integrating the material balance equation (Eq. (5)) in time and space (Fig. 2). For the time integration, we select pressure as an implicit variable while the other variables are treated explicitly. The discretized material balance is then written as

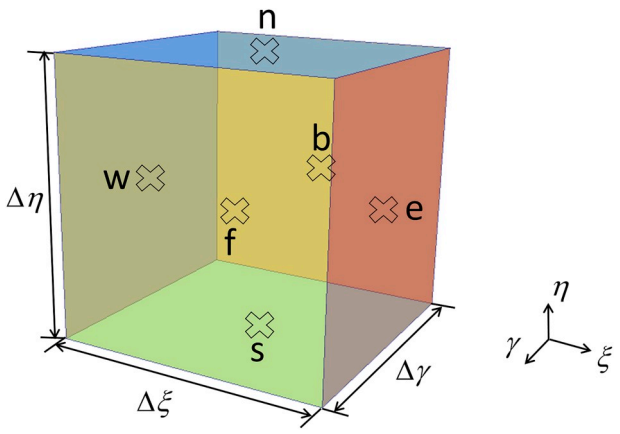

Fig. 2. Illustration of the control volume in the computational domain. Different faces of the control volume ("f", "b", "e", "w", "n", "s") are shown in different colors. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

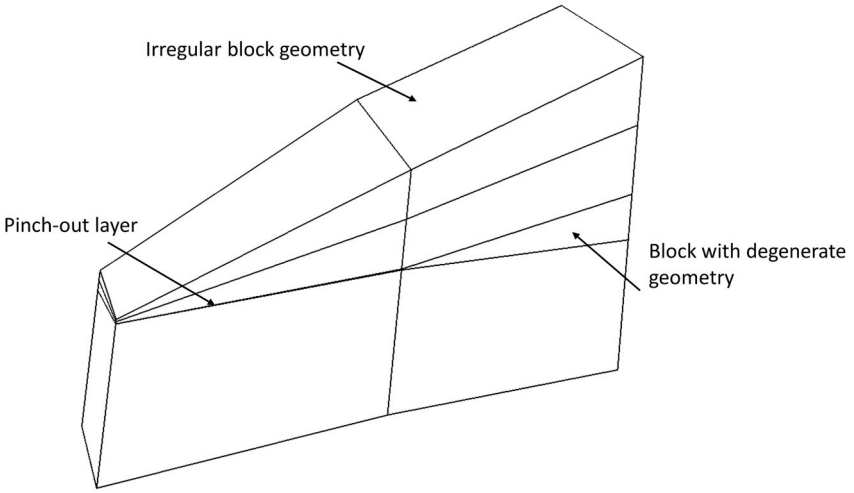

Fig. 3. An example 2×4 corner-point grid with different types of block geometries.

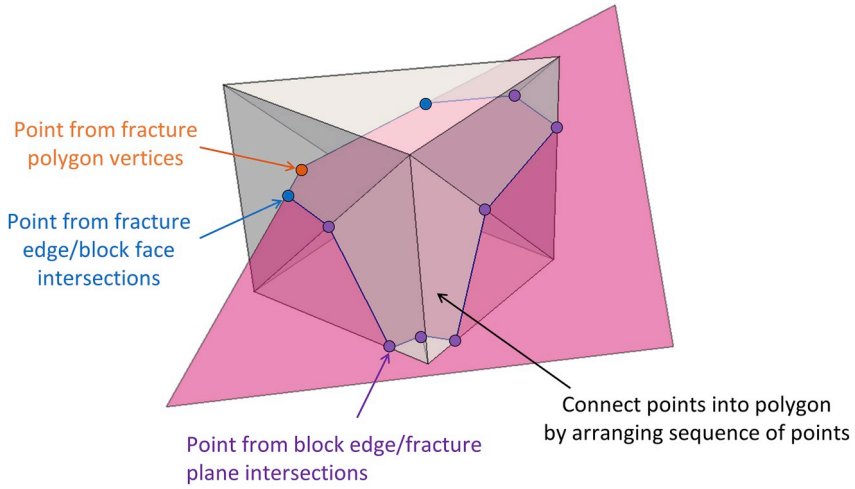

Fig. 4. Illustration of the geometrical calculation of matrix-fracture intersection. The final intersection polygon contains three types of points, as shown in different colors. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

$$
\begin{aligned}
&\frac{N_k^{n+1}-N_k^n}{\Delta t}\Delta\xi\Delta\eta\Delta\gamma = q_k\Delta\xi\Delta\eta\Delta\gamma\\
&+\left[D_{11,e}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n+1}}{\partial\xi}\right)_e - D_{11,w}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n+1}}{\partial\xi}\right)_w\right]\Delta\eta\Delta\gamma\\
&+\left[D_{12,e}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\eta}\right)_e - D_{12,w}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\eta}\right)_w\right]\Delta\eta\Delta\gamma\\
&+\left[D_{13,e}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\gamma}\right)_e - D_{13,w}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\gamma}\right)_w\right]\Delta\eta\Delta\gamma\\
&+\left[D_{21,n}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\xi}\right)_n - D_{21,s}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\xi}\right)_s\right]\Delta\xi\Delta\gamma\\
&+\left[D_{22,n}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n+1}}{\partial\eta}\right)_n - D_{22,s}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n+1}}{\partial\eta}\right)_s\right]\Delta\xi\Delta\gamma\\
&+\left[D_{23,n}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\gamma}\right)_n - D_{23,s}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\gamma}\right)_s\right]\Delta\xi\Delta\gamma\\
&+\left[D_{31,f}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\xi}\right)_f - D_{31,b}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\xi}\right)_b\right]\Delta\xi\Delta\eta\\
&+\left[D_{32,f}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\eta}\right)_f - D_{32,b}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n}}{\partial\eta}\right)_b\right]\Delta\xi\Delta\eta\\
&+\left[D_{33,f}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n+1}}{\partial\gamma}\right)_f - D_{33,b}\sum_{j=1}^{n_p}\left(x_{kj}^n\xi_j^n\frac{k_{rj}^n}{\mu_j^n}\frac{\partial\Phi_j^{n+1}}{\partial\gamma}\right)_b\right]\Delta\xi\Delta\eta,\\
&k=1,\ldots,n_c+1
\end{aligned}
\tag{10}
$$

where the superscript n denotes the previous time level, superscript $n+1$ denotes the next time level, and subscripts "f," "b," "e," "w," "n," and "s" denote faces shown in Fig. 2. It should be noted that to reduce the complexity of the resulting linear system, the cross-derivative terms are treated explicitly in this work.

As it can be noted from the above equations, for every gridblock, there are $n_c + 1$ number of moles and one pressure to be computed at time level $n + 1$, which results in $n_c + 2$ unknowns. However, there are only $n_c + 1$ material balance equations to determine these unknowns, meaning that another equation is required. In this work, the IMPEC formulation proposed by Acs et al. (1985) is used, which uses a volume balance equation for determining the pressure at each gridblock. The pressure equation is obtained through the volume constraint equation at the next time level:

$$\phi^{n+1}V_b = V_T^{n+1}, \tag{11}$$

where V_T^{n+1} is the total fluid volume of the gridblock at time level $n + 1$. Assuming that the porosity is a function of the pressure and the total fluid volume is a function of both the pressure and the number of moles of each component, one can approximate Eq. (11) by Taylor series expansion truncated in the first order derivatives as

$$\left[\phi^n+\left(\frac{\partial\phi}{\partial P}\right)^n(P^{n+1}-P^n)\right]V_b = V_T^n+\left(\frac{\partial V_T}{\partial P}\right)^n(P^{n+1}-P^n)+\sum_{k=1}^{n_c+1}\left(\frac{\partial V_T}{\partial N_k}\right)^n(N_k^{n+1}-N_k^n). \tag{12}$$

Assuming the rock is slightly compressible with compressibility c_f, plugging Eq. (10) into Eq. (12), and rearranging the terms yields

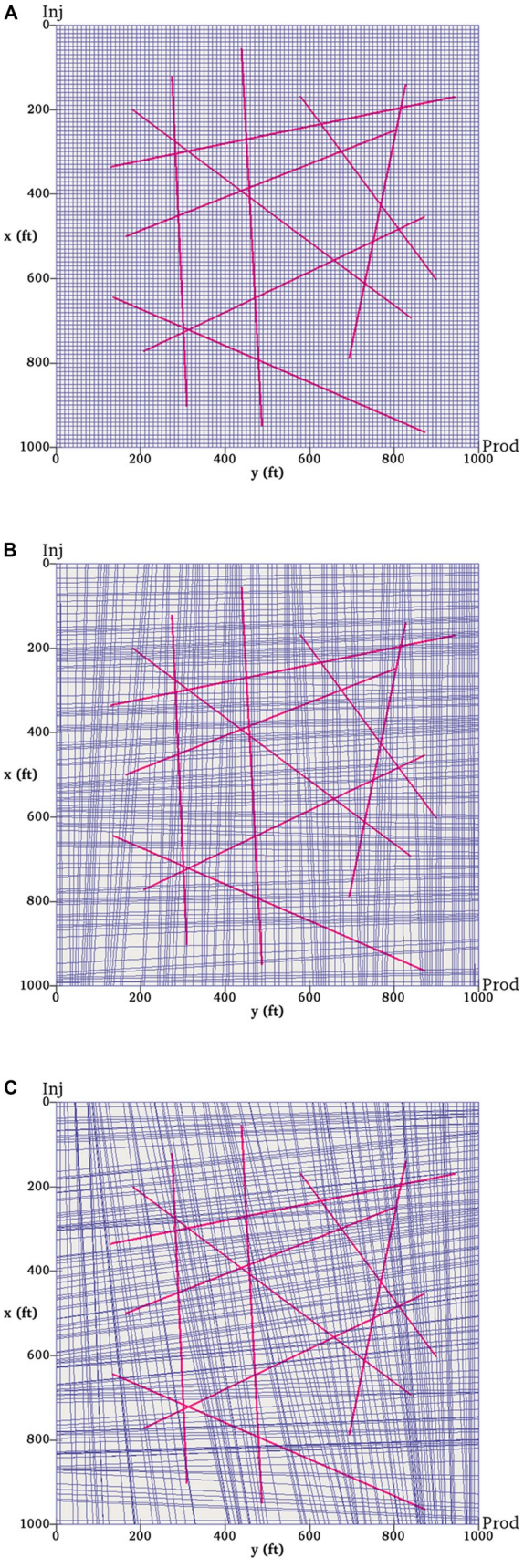

Fig. 5. 2D simulation grids used for the water flooding simulation. (a) A regular Cartesian grid. (b) A slightly distorted corner-point grid. (c) A highly distorted corner-point grid. The red lines represent fractures. The locations of the injector ("Inj") and the producer ("Prod") are also indicated. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

Table 1
Simulation parameters for Case 1.

Parameter	Value	Unit
Matrix permeability	200	md
Reservoir porosity	25%	–
Rock compressibility	5×10^{-5}	psi^{-1}
Reservoir temperature	120	°F
Initial reservoir pressure	4,925	psi
Initial water saturation	0.3	–
Residual water saturation	0.24	–
Residual oil saturation	0.42	–
Water rel. perm. endpoint	0.20	–
Oil rel. perm. endpoint	0.60	–
Water rel. perm. exponent	1.5	–
Oil rel. perm. exponent	3.0	–
Water viscosity	0.79	cp
Producer bottomhole pressure	4,000	psi
Water injection rate	500	STB/day
Wellbore radius	0.5	ft
Simulation time	1,500	day

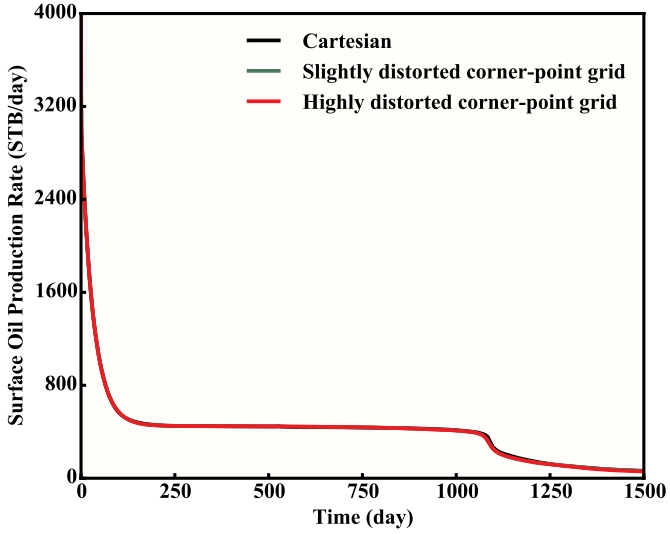

Fig. 6. Comparison of oil production rates for the three simulation grids.

$$\left[\phi^0 c_f V_b - \left(\frac{\partial V_T}{\partial P}\right)^n\right](P^{n+1} - P^n) = \Delta t \sum_{k=1}^{n_c+1}\left(\frac{\partial V_T}{\partial N_k}\right)^n q_k$$
$$+ \Delta t \sum_{k=1}^{n_c+1}\left(\frac{\partial V_T}{\partial N_k}\right)^n \left\{\sum_{j=1}^{n_p}\left[x_{kj}^n \xi_j^n \frac{k_{rj}^n}{\mu_j^n}\left(\frac{D_{11}}{\Delta\xi}\frac{\partial \Phi_j^{n+1}}{\partial \xi} + \frac{D_{12}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \eta} + \frac{D_{13}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \gamma}\right)\right]_e\right.$$
$$- \sum_{j=2}^{n_p}\left[x_{kj}^n \xi_j^n \frac{k_{rj}^n}{\mu_j^n}\left(\frac{D_{11}}{\Delta\xi}\frac{\partial \Phi_j^{n+1}}{\partial \xi} + \frac{D_{12}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \eta} + \frac{D_{13}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \gamma}\right)\right]_w$$
$$+ \sum_{j=2}^{n_p}\left[x_{kj}^n \xi_j^n \frac{k_{rj}^n}{\mu_j^n}\left(\frac{D_{21}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \xi} + \frac{D_{22}}{\Delta\xi}\frac{\partial \Phi_j^{n+1}}{\partial \eta} + \frac{D_{23}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \gamma}\right)\right]_n$$
$$- \sum_{j=2}^{n_p}\left[x_{kj}^n \xi_j^n \frac{k_{rj}^n}{\mu_j^n}\left(\frac{D_{21}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \xi} + \frac{D_{22}}{\Delta\xi}\frac{\partial \Phi_j^{n+1}}{\partial \eta} + \frac{D_{23}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \gamma}\right)\right]_s$$
$$+ \sum_{j=2}^{n_p}\left[x_{kj}^n \xi_j^n \frac{k_{rj}^n}{\mu_j^n}\left(\frac{D_{31}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \xi} + \frac{D_{32}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \eta} + \frac{D_{33}}{\Delta\xi}\frac{\partial \Phi_j^{n+1}}{\partial \gamma}\right)\right]_f$$
$$\left. - \sum_{j=2}^{n_p}\left[x_{kj}^n \xi_j^n \frac{k_{rj}^n}{\mu_j^n}\left(\frac{D_{31}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \xi} + \frac{D_{32}}{\Delta\xi}\frac{\partial \Phi_j^n}{\partial \eta} + \frac{D_{33}}{\Delta\xi}\frac{\partial \Phi_j^{n+1}}{\partial \gamma}\right)\right]_b\right\}, \quad (13)$$

where ϕ^0 is the porosity at reference pressure.

In this work, the matrix pressure is obtained by Eq. (13). After that, the obtained pressure is substituted into Eq. (10) to calculate the number of moles of all components at the new time level.

4. The embedded discrete fracture model (EDFM)

4.1. Numerical formulations

The EDFM discretizes the fractures into fracture segments using the matrix cell boundaries. In the simulator, similar to the matrix, each fracture segment is represented as a control volume. The introduction of new control volumes brings a new term $F_{k,\,nnc}$ in the material balance equation. These new terms consider:

1. Flow between a fracture segment and the matrix gridblock it intersects (matrix-fracture flow)
2. Flow between fracture segments that are connected with each other or intersect each other (fracture-fracture flow)

Both types of flow are represented in the simulator using NNCs. After considering the flow through NNCs, Eq. (1) can be rewritten as

$$\frac{\partial N_k}{\partial t} = F_k + D_k + q_k + F_{k,\,nnc}, \quad k = 1, \ldots, n_c + 1, \tag{14}$$

where $F_{k,\,nnc}$ represents the molar flow rate of component k due to convection through NNCs.

In the EDFM, two-point flux approximation is used to approximate the flow between matrix control volumes and fracture control volumes, or the flow between fracture control volumes. For a matrix or fracture control volume p,

$$F_{k,\,nnc} = \sum_{l=1}^{n} \sum_{j=1}^{n_p} \left[x_{kj} \xi_j \frac{k_{rj}}{\mu_j} \right]_l T_{nnc,i} (P_{j,L} - P_{j,p} - \rho_{j,l} g (D_L - D_p)), \tag{15}$$

where n is the number of NNCs for p, subscript L represents the control volume L that is connected to p via a NNC, subscript l represents the interface between p and L, and $T_{p,L}$ is the interblock transmissibility factor for the flow across the interface l between p and L.

In the EDFM, the general formulation to evaluate the transmissibility factor $T_{p,L}$ is

$$\frac{1}{T_{p,L}} = \frac{d_{p-l}}{k_p A_l} + \frac{d_{L-l}}{k_L A_l}, \tag{16}$$

where k_p and k_l are the permeabilities of p and L, respectively, d_{p-l} and d_{L-l} are the equivalent distances from the centroids of p and L to the interface l, respectively, and A_l is the area of interface l.

For matrix-fracture flow,

$$\frac{1}{T_{p,L}} \approx \frac{d_{f-m}}{\left(\overrightarrow{\overrightarrow{K_m}} \cdot \vec{n} \right) \cdot \vec{n} A_l} = \frac{d_{f-m}}{2 \left(\overrightarrow{\overrightarrow{K_m}} \cdot \vec{n} \right) \cdot \vec{n} A_f}, \tag{17}$$

where $\overrightarrow{\overrightarrow{K_m}}$ is the permeability tensor of the matrix gridblock, $\vec{n}$ is the normal vector of the fracture plane, d_{f-m} is the average normal distance from the matrix gridblock to the fracture segments, and A_f is the area of the fracture segment. $A_l = 2A_f$ considering the flow on both sides of the fracture segments. The assumption involved in the derivation of Eq. (17) is a linear pressure distribution in the matrix block intersected by the fracture (Xu et al., 2017a).

For fracture-fracture flow,

$$\frac{1}{T_{p,L}} = \frac{d_{f1-l}}{k_{f1} A_l} + \frac{d_{f2-l}}{k_{f2} A_l}, \tag{18}$$

where k_{f1} and k_{f2} are the permeability of the fracture segments f_1 and f_2, respectively, d_{f1-l} and d_{f1-l} are the average distances from each fracture segment to the interface between the fracture segments, and is A_l is the area for flow. A_l can be calculated as the product of fracture width and the length of the connection or intersection line between the fracture segments. Please see Moinfar et al. (2014) and Xu et al. (2017a) for more details.

For a fracture control volume, Eq. (14) can be simplified as

$$\frac{\partial N_k}{\partial t} = q_k + F_{k,\,nnc}, \quad k = 1, \ldots, n_c + 1. \tag{19}$$

Only the flow through NNCs and the flow between the fracture and wellbores are considered. When evaluating q_k, a well index is calculated for the fracture segment as (Moinfar et al., 2013)

$$WI_f = \frac{2\pi k_f w_f}{ln\left(\frac{r_e}{r_w}\right)}, \; r_e = 0.14\sqrt{L_s^2 + H_s^2}. \tag{20}$$

where k_f is the fracture permeability, w_f is the fracture width, L_s is the length of the fracture segment, and H_s is the height of the fracture segment. Eq. (20) has a similar form as Peaceman's well model.

After the transmissibility factors and well indices are calculated, the terms pertaining to NNCs are also added to the pressure equation (Eq. (13)) in a similar way as in Eq. (14).

4.2. Geometrical calculation

A challenge for the EDFM in corner-point grids is the geometrical calculation. The calculation of matrix-fracture intersection, fracture-fracture connection, fracture-fracture intersection, and fracture-well intersection requires knowledge of computational geometry. In the literature, the EDFM approach has mainly been applied in Cartesian grids, which has a relatively simple gridblock geometry. The gridblock geometry of corner-point grids might be much more irregular, where the gridblock cannot be represented as a cuboid. In addition, since the eight vertices of a gridblock is defined individually, there is a possibility that two vertices of the gridblock have the same location (degeneracy). The gridblocks may also be pinched-out when they have close-to-zero thickness. Fig. 3 shows a 2×4 corner-point grid with the block geometries mentioned above. It can be observed that the block geometry of corner-point grids is much more complicated compared to that of Cartesian grids, and this pose challenges for the EDFM geometrical calculation.

We first discuss the methodology for the calculation of intersections between matrix gridblocks and fractures. In this work, considering the irregular block geometries and degeneracy, we treat each gridblock as a general polyhedron represented using vertices, edges, and faces, and each fracture as a general polygon represented using vertices and edges. The intersection between the polyhedron and the polygon is also a polygon, and the vertices of the intersection polygon can be found by calculating three types of points:

1. The vertex of the fracture polygon that is inside the polyhedron
2. The intersection point between an edge of the fracture polygon and a face of the polyhedron
3. The intersection point between an edge of the polyhedron and the fracture polygon

The three types of points are shown in Fig. 4. The intersection polygon can be obtained by connecting the three types of points.

The pinch-outs of gridblocks are also considered in the EDFM. When a gridblock is pinched-out, no matrix-fracture intersection is calculated for this gridblock. In addition, in the vertical direction, the fracture segments above and below this gridblock are connected to remain the connectivity of the fractures.

The geometrical calculation is performed in an EDFM Preprocessor developed in this work. After the calculation, the EDFM Preprocessor exports NNC pairs, transmissibilities factors, and well indices calculated

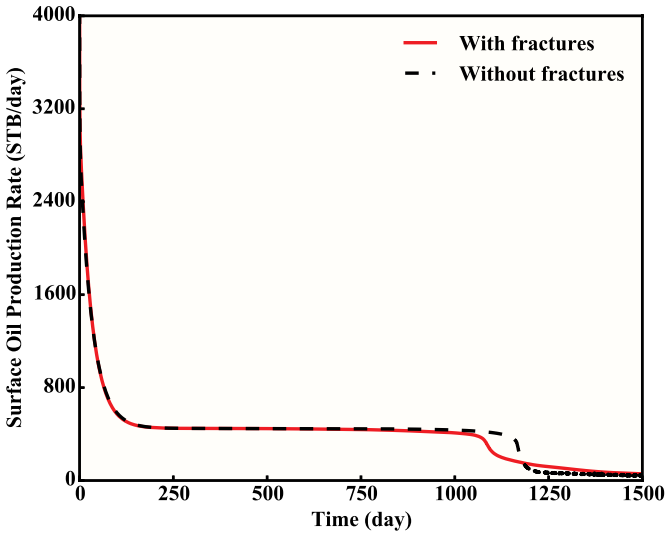

Fig. 7. Comparison of oil production rate for the case with and without fractures (simulated using the highly distorted grid).

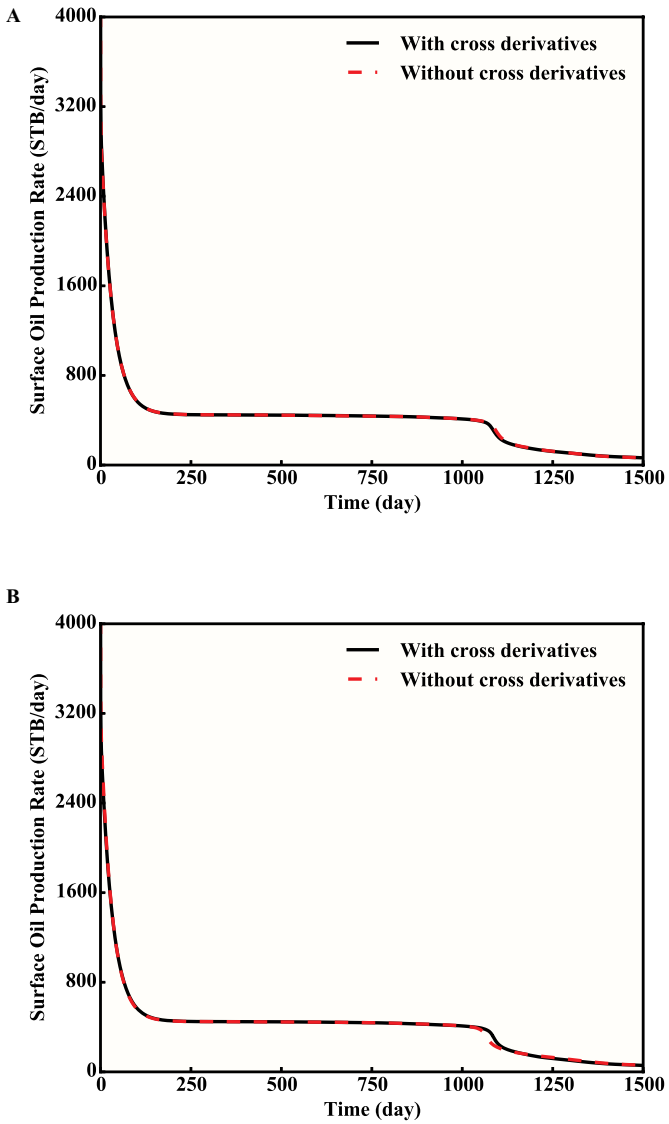

Fig. 8. Comparison of oil production rate for the cases with and without cross derivatives. (a) The simulation results on the grid in Fig. 5b. (b) The simulation results on the grid in Fig. 5c.

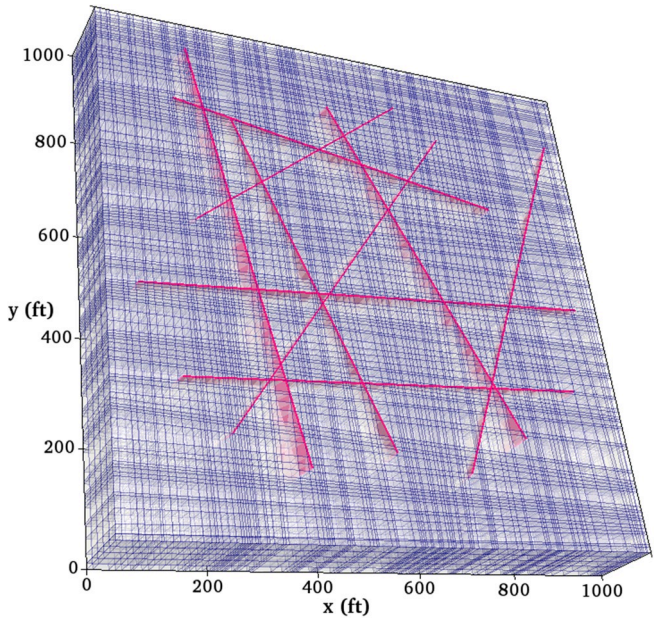

Fig. 9. Simulation grid and the location of fractures for the 3D study with the highly distorted grid.

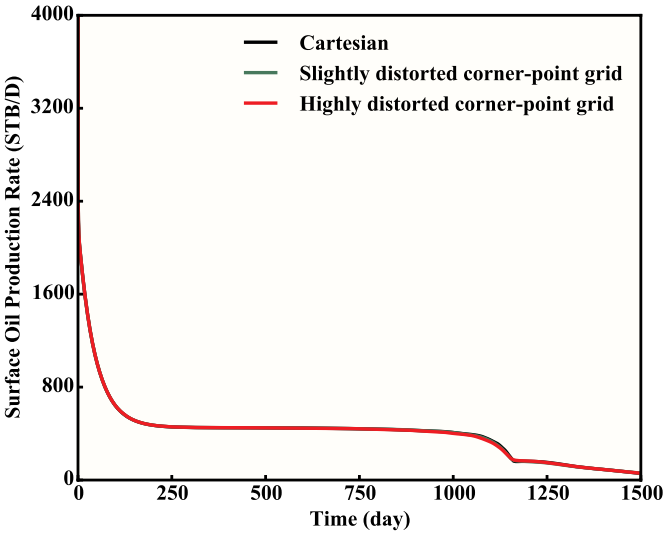

Fig. 10. Comparison of oil production rates for the three-dimensional simulation study.

by Eqs. (17), (18) and (20). The simulator reads these outputs and adds the corresponding terms in the material balance equation and pressure equation.

5. Results and discussion

To illustrate the applicability of the developed model, several synthetic case studies are simulated. Both 2D and 3D corner-point grids are used in these studies.

5.1. Case 1: study on influence of grid distortion

A conventional reservoir with dimensions of 1,000×1,000×50 ft is studied. Three 100×100×1 matrix grids with different levels of distortion are used for this reservoir, as shown in Fig. 5. The grid coordinates for these three grids are provided with this paper. Six hydrocarbon components (C_1, C_2, C_3, C_7, C_{10}, and C_{20}) are considered, with their

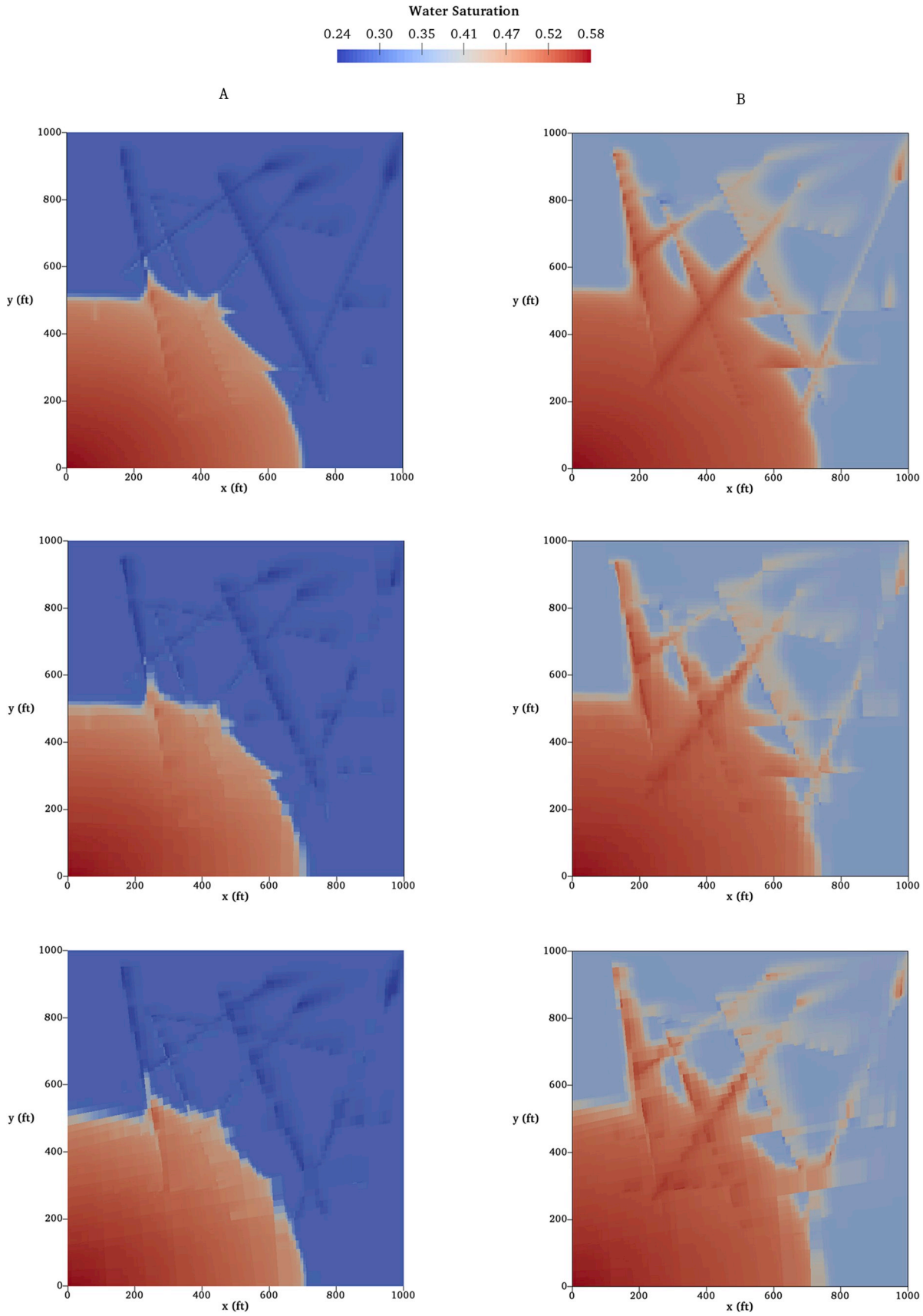

Fig. 11. Water saturation profiles at 400 days. (a) The top layer of the matrix. (b) The bottom layer of the matrix. The top row shows the results using the Cartesian grid, the middle row shows the results using the slightly distorted grid, and the bottom row shows the results using the highly distorted grid.

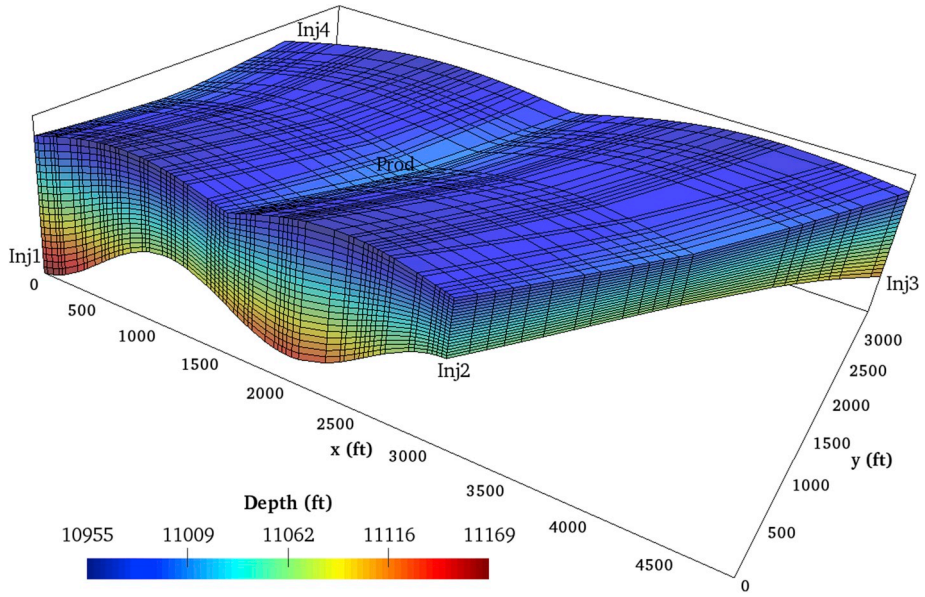

Fig. 12. Reservoir model used for simulation study. The simulation grid is colored by depth. The length in the z-direction is shown five times the real size.

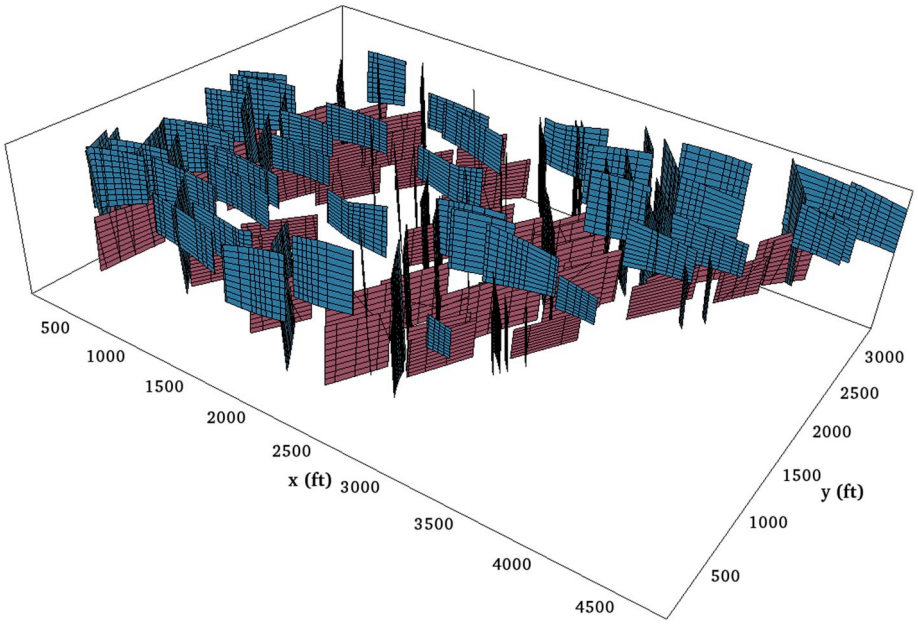

Fig. 13. Fracture segments obtained from EDFM preprocessing. Different sets of fractures are represented using different colors. The length in the z-direction is shown five times the real size. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

initial composition as 0.1, 0.1, 0.1, 0.1, 0.2, and 0.4, respectively. Some important simulation parameters are summarized in Table 1. For the water-oil two-phase relative permeability, Corey's model is used, and the model parameters are also given in Table 1. The capillary pressure effect is ignored in this study. Nine large-scale vertical fractures are placed inside the reservoir using the EDFM approach, as shown in Fig. 5. The fracture conductivities are 10,000 md-ft. For all three matrix grids, the fracture locations and parameters are the same. The total number of fracture segments are 846, 843, and 834, respectively. An injector with a constant injection rate and a producer with a constant bottom hole pressure are placed at two corners of the grids.

Fig. 6 shows the predicted oil rates by EDFM on three different grids. With the full-permeability-tensor formulation and the EDFM method, the model obtains similar results on different grids. As the accuracy of the EDFM with Cartesian matrix grid has been verified in the literature (Moinfar et al., 2014; Shakiba and Sepehrnoori, 2015; Xu et al., 2017a,b; Flemisch et al., 2018), the results confirm the accuracy of the developed model in corner-point grids. Furthermore, the agreement between different simulations also indicates that the accuracy of the EDFM approach is not greatly influenced by matrix gridding when similar numbers of gridblocks are used.

To investigate the influence of fractures on water flooding, Fig. 7 compares the oil rate curves for the cases with and without fractures, using the matrix grid in Fig. 5c. An observation is that the existence of fractures lowers the oil rate after 500 days of injection. The reason is that the highly conductive fractures speed up the movement of the injected fluid (water) from the injector to the producer. Hence the sweep efficiency of the water flooding process is lower when the influence of fractures is taken into account.

The use of full-tensor permeability is of great importance in the simulation of this process if a distorted grid is used. Fig. 8 compares the simulation results with and without considering cross derivatives. For

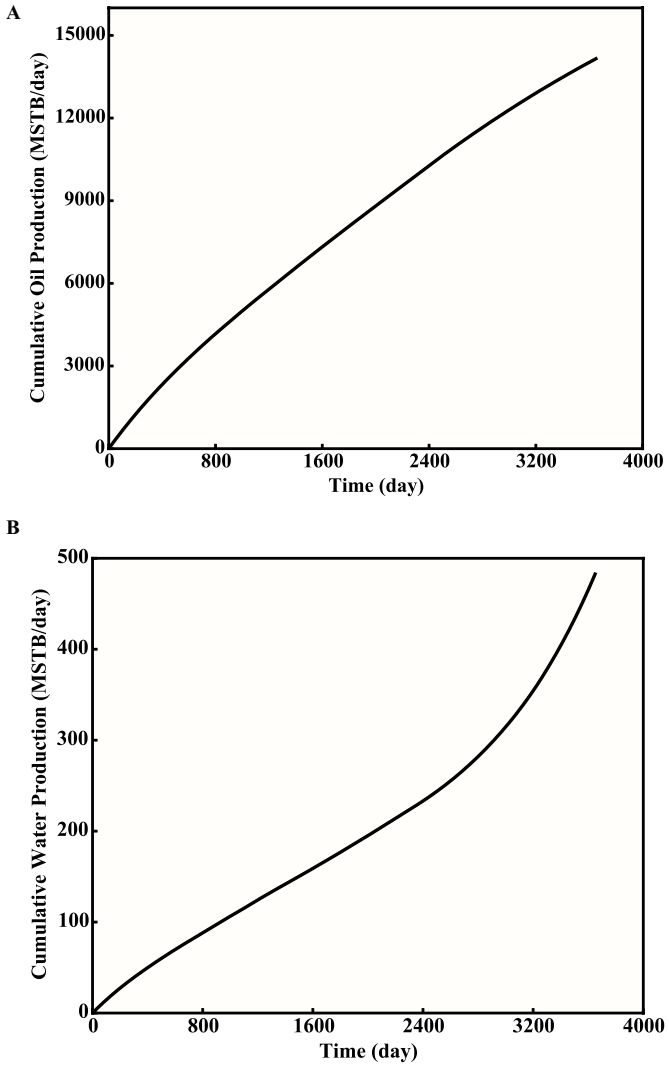

Fig. 14. (a) Cumulative oil production (at surface condition) curve. (b) Cumulative water production (at surface condition) curve.

the slightly distorted grid (Fig. 5b), there is a small error resulting from the non-orthogonality of the grid. For the highly distorted grid (Fig. 5c), the error caused by the ignorance of cross-derivative terms can be clearly seen. As discussed before, the error will not disappear even when grid refining is performed. Therefore, although the reliability of the EDFM is not significantly impacted by the discretization of the matrix, careful selection of formulations is required when the grid is not orthogonal.

5.2. Case 2: 3D matrix grids with inclined fractures

In this case, the three 2D grids used in Case 1 are extended to 3D, and the reservoir is uniformly discretized into five layers in the vertical direction, as shown in Fig. 9. In addition, inclined fractures instead of vertical fractures are placed inside the reservoir, and the dip angles of fractures are 70°, 60°, 90°, 77°, 82°, 55°, 45°, 82°, and 73°, respectively. The matrix permeability in the *x*-, *y*-, and *z*-directions are 200 md, 100 md, and 20 md, respectively. Other simulation parameters and the well-operating conditions are the same as in Case 1.

Fig. 10 shows the predicted oil rates using three different grids. For the 3D cases, the EDFM still renders similar results for the three grids, which again shows the insensitivity of the EDFM to matrix gridding. Fig. 11 gives the water saturation profiles in the matrix at 400 days, where the waterfront in the top and the bottom layers can be easily observed. The gravitational effect can also be clearly seen as the locations of waterfront in different layers are different. Therefore, the reservoir needs to be carefully discretized in the vertical direction when simulating such processes to accurately predict the movement of waterfront under the influence of gravity.

5.3. Case 3: water flooding in a naturally fractured reservoir

This study simulates water flooding in a reservoir with irregular boundary. A 60×20×20 corner-point grid is used for matrix gridding, as shown in Fig. 12. The reservoir thickness ranges from 73 ft to 220 ft. Four injectors (“Inj1”, “Inj2”, “Inj3”, and “Inj4”) and one producer (“Prod”) are placed inside the reservoir. The injection rate of each injector is 1000 STB/day. The producer bottom hole pressure is 4,000 psi. All injectors are perforated in the lower ten layers, and the producer is perforated in the upper ten layers. Other simulation parameters are the same as in Case 2.

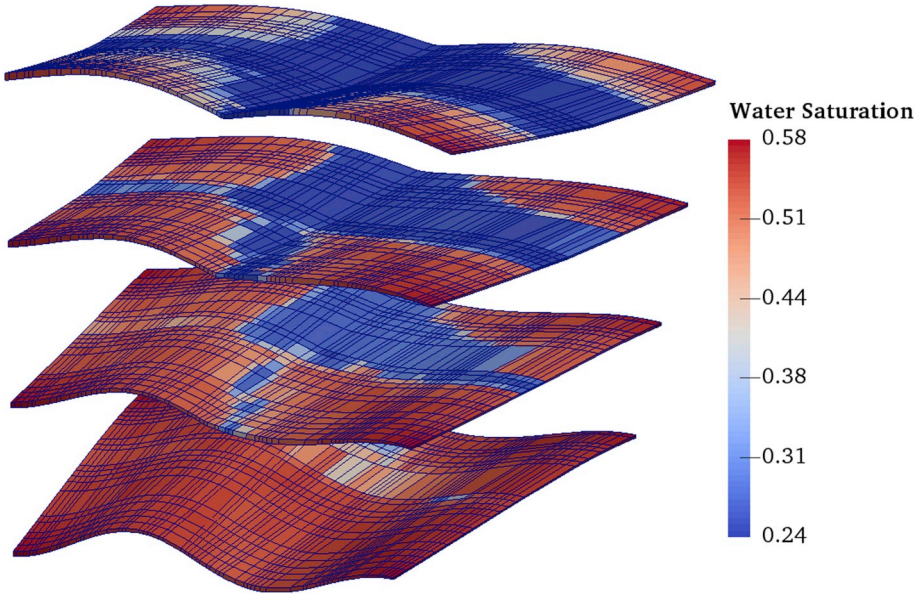

Fig. 15. Water saturation profiles at 3600 days. From top to bottom, the water saturation in four layers (Layers 1, 5, 11, and 18) is shown. The length in the *z*-direction is shown five times the real size.

Three sets of natural fractures with different orientations are placed inside the reservoir. The total number of fractures is 128. The conductivity of fractures is 100,000 md-ft. In naturally fractured reservoirs, the formation of different sets of natural fractures might be related to geological activities at different times. Therefore, the three sets of fractures are restricted in different layers in this study. The first set of fractures only penetrates Layers 1–10, the second set only penetrates Layers 11–20, and the third set penetrates the whole formation in the vertical direction. The fracture segments obtained from EDFM preprocessing are shown in Fig. 13.

The cumulative oil and water production rate curves predicted by the simulation are shown in Fig. 14. To illustrate the effect of gravity, water saturation in four different layers at 3600 days are shown in Fig. 15. It can be observed that the sweeping efficiency in different layers is quite different. In addition, the variation of depth within each layer also has a great impact on the water saturation distribution.

6. Conclusions

In this study, an Embedded Discrete Fracture Model was extended to matrix grids with corner-point geometry to simulate complex fractures in geologically complex reservoirs. The model was implemented in an IMPEC, compositional reservoir simulator with a full-tensor-permeability formulation. Modifications in the simulator governing equations and key points for geometrical calculations were presented in this work.

Several case studies were presented for verification and application of the developed model. It was found that the accuracy of the EDFM approach is not greatly influenced by matrix gridding when similar numbers of gridblocks are used. With the full-permeability-tensor simulation, similar simulation results were obtained on grids with different levels of distortion. The inclusion of cross derivatives in the governing equation was also found to be of great importance when the matrix grid is highly distorted. Therefore, although the EDFM result is not sensitive to grid distortion, numerical solution schemes should be carefully selected when the grid is not orthogonal.

The impact of natural fractures on water-flooding efficiency was also investigated. Serving as high-speed channels, the highly conductive fractures might speed up the movement of the injected fluid from the injector to the producer, lowering the flooding efficiency. The gravity is also playing a significant role in the simulation of water-oil system, and care should be taken for the discretization of simulation grid in the vertical direction to accurately simulate the movement of waterfront through the fracture network in different layers.

This study illustrates the compatibility of the EDFM approach with different types of grids widely used in reservoir simulation, including both Cartesian grid and corner-point grids. As the EDFM requires no special treatment of the matrix grid around fractures, it makes it easy and convenient to place complex fractures in existing reservoir simulation models using these types of grid.

Abbreviations

ACCS	Accelerated Successive Substitution
EDFM	Embedded Discrete Fracture Model
NNC	Non-Neighboring Connection
TPFA	Two-Point Flux Approximation

References

Aavatsmark, I., 2007. Multipoint flux approximation methods for quadrilateral grids. In: 9th International Forum on Reservoir Simulation, Abu Dhabi, UAE.

Aavatsmark, I., Barkve, T., Mannseth, T., 1998. Control-volume discretization methods for 3D quadrilateral grids in inhomogeneous, anisotropic reservoirs. SPE J. 3 (02), 146–154. SPE-38000-PA. https://doi.org/10.2118/38000-PA.

Acs, G., Doleschall, S., Farkas, E., 1985. General purpose compositional model. SPE J. 25 (04), 543–553. SPE-10515-PA. https://doi.org/10.2118/10515-PA.

Cavalcante-Filho, J.S.A., Sepehrnoori, K., 2017. Simulation of planar hydraulic fractures with variable conductivity using the embedded discrete fracture model. J. Petrol. Sci. Eng. 153, 212–222 2017.

Cavalcante-Filho, J.S.A., Shakiba, M., Moinfar, A., et al., 2015. Implementation of a preprocessor for embedded discrete fracture modeling in an IMPEC compositional reservoir simulator. In: SPE Reservoir Simulation Symposium, Houston, Texas, 23-25 February, SPE-173289-MS. https://doi.org/10.2118/173289-MS.

Chang, Y., 1990. Development and Application of an Equation of State Compositional Simulator. Doctoral dissertation. University of Texas, Austin, Texas August 1990.

Chu, W.H., 1971. Development of a general finite difference approximation for a general domain – Part I: machine transformation. J. Comput. Phys. 8 (3), 392–408.

de Sousa Junior, L.C., dos Santos, L.O.S., de Souza Rios, V., et al., 2016. Methodology for geomechanically controlled transmissibility through active natural fractures in reservoir simulation. J. Petrol. Sci. Eng. 147, 7–14. 2016. https://doi.org/10.1016/j.petrol.2016.04.040.

Du, S., Liang, B., Yuanbo, L., 2017. Field study: embedded discrete fracture modeling with artificial intelligence in Permian basin for shale formation. In: SPE Annual Technical Conference and Exhibition, San Antonio, Texas, 9-11 October.

Fernandes, B.R.B., Marcondes, F., Sepehrnoori, K., 2018. Development of a fully implicit approach with intensive variables for compositional reservoir simulation. J. Petrol. Sci. Eng. 169, 317–336 2018.

Fernandes, B.R.B., Drumond Filho, E.P., Gonçalves, A.D.R., et al., 2014. Investigation of cross derivatives in corner point grids formulation in conjunction with compositional reservoir simulation. In: XXXV Iberian Latin-American Congress on Computational Methods in Engineering, Fortaleza, Ceará, Brazil, 23-26 November.

Flemisch, B., Berre, I., Boon, W., et al., 2018. Benchmarks for single-phase flow in fractured porous media. Adv. Water Resour. 111, 239–258 2018.

Fumagalli, A., Pasquale, L., Zonca, S., et al., 2016. An upscaling procedure for fractured reservoirs with embedded grids. Water Resour. Res. 52 (8), 6506–6525. http://doi.org/10.1002/2015WR017729.

Fumagalli, A., Zonca, S., Formaggia, L., 2017. Advances in computation of local problems for flow-based upscaling in fractured reservoirs. Math. Comput. Simulat. 137, 299–324. 2017. https://doi.org/10.1016/j.matcom.2017.01.007.

Hearn, C.L., Al-Emadi, I.A.A., Worley, P.L.H., et al., 1997. Improved oil recovery in a tight reservoir with conductive faults, ISND Shuaiba, Qatar. In: SPE Annual Technical Conference and Exhibition, San Antonio, Texas, 5-8 October, SPE-38908-MS. https://doi.org/10.2118/38908-MS.

Hegre, T.M., Dalen, V., Henriquez, A., 1986. Generalized transmissibilities for distorted grids in reservoir simulation. In: SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, 5-8 October, SPE-15622-MS. https://doi.org/10.2118/15622-MS.

Karimi-Fard, M., Durlofsky, L.J., 2016. A general gridding, discretization, and coarsening methodology for modeling flow in porous formations with discrete geological features. Adv. Water Resour. 96, 354–372 2016.

Lee, S.H., Durlofsky, L.J., Lough, M.F., et al., 1998. Finite difference simulation of geologically complex reservoirs with tensor permeabilities. SPE Reservoir Eval. Eng. 1 (06), 567–574. SPE-52637-PA. https://doi.org/10.2118/52637-PA.

Li, L., Lee, S.H., 2008. Efficient field-scale simulation of black oil in a naturally fractured reservoir through discrete fracture networks and homogenized media. SPE Reservoir Eval. Eng. 11 (4), 750–758. SPE-103901-PA. https://doi.org/10.2118/103901-PA.

Marcondes, F., Han, C., Sepehrnoori, K., 2005. Implementation of corner point mesh into a parallel, fully implicit, equation of state compositional reservoir simulator. In: 18th International Congress of Mechanical Engineering, Ouro Preto, Minas Gerais, Brazil.

Marcondes, F., Han, C., Sepehrnoori, K., 2008. Effect of cross derivatives in discretization schemes in structured non-orthogonal meshes for compositional reservoir simulation. J. Petrol. Sci. Eng. 63 (1–4), 53–60. https://doi.org/10.1016/j.petrol.2008.07.009.

Marcondes, F., Varavei, A., Sepehrnoori, K., 2010. An element-based finite-volume method approach for naturally fractured compositional reservoir simulation. In: 13th Brazilian Congress of Thermal Sciences and Engineering–ENCIT, Uberlândia, MG, Brazil, 5-10 December.

Matthai, S.K., Geiger, S., Roberts, S.G., et al., 2007. Numerical simulation of multi-phase fluid flow in structurally complex reservoirs. Geol. Soc. Lond. Spl. Publ. 292 (1), 405–429. https://doi.org/10.1144/SP292.22.

Mehra, R.K., Heidemann, R.A., Aziz, K., 1983. An accelerated successive substitution algorithm. Can. J. Chem. Eng. 61 (4), 590–596.

Michelsen, M.L., 1982. The isothermal flash problem. Part I. Stability. Fluid Phase Equilib. 9 (1), 1–19.

Moinfar, A., Varavei, A., Sepehrnoori, K., et al., 2013. Development of a coupled dual continuum and discrete fracture model for the simulation of unconventional reservoirs. In: SPE Reservoir Simulation Symposium, the Woodlands, Texas, 18-20 February, SPE-163647-MS. https://doi.org/10.2118/163647-MS.

Moinfar, A., Varavei, A., Sepehrnoori, K., et al., 2014. Development of an efficient embedded discrete fracture model for 3D compositional reservoir simulation in fractured reservoirs. SPE J. 19 (2), 289–303. SPE-154246-PA. https://doi.org/10.2118/154246-PA.

Panfili, P., Cominelli, A., 2014. Simulation of miscible gas injection in a fractured carbonate reservoir using an embedded discrete fracture model. In: Abu Dhabi International Petroleum Exhibition and Conference, Abu Dhabi, 10-13 November, SPE-171830-MS. https://doi.org/10.2118/171830-MS.

Panfili, P., Colin, R., Cominelli, A., et al., 2015. Efficient and effective field scale simulation of hydraulic fractured wells: methodology and application. In: SPE Reservoir Characterisation and Simulation Conference and Exhibition, Abu Dhabi, UAE, 14-16 September, SPE-175542-MS. https://doi.org/10.2118/175542-MS.

Peng, D.Y., Robinson, D.B., 1976. A new two-constant equation of state. Ind. Eng. Chem. Fundam. 15 (1), 59–64.

Perschke, D.R., 1988. Equation-of-State Phase-Behavior Modeling for Compositional Simulation. Ph.D. dissertation. The University of Texas at Austin, Austin, Texas.

Ponting, D.K., 1989. Corner point geometry in reservoir simulation. In: 1st European Conference on the Mathematics of Oil Recovery, Cambridge, UK, 14-16 July.

Shakiba, M., Sepehrnoori, K., 2015. Using embedded discrete fracture model (EDFM) and microseismic monitoring data to characterize the complex hydraulic fracture networks. In: SPE Annual Technical Conference and Exhibition, Houston, Texas, 28-30 September, SPE-175142-MS. https://doi.org/10.2118/175142-MS.

Tang, H., Chai, Z., Yan, B., et al., 2017. Application of multi-segment well modeling to simulate well interference. In: SPE/AAPG/SEG Unconventional Resources Technology Conference, Austin, Texas, July, pp. 24–26.

Thompson, J.F., Thames, F.C., Mastin, C.W., 1974. Automatic numerical generation of body-fitted curvilinear coordinates systems for fields containing any number of arbitrary two-dimensional bodies. J. Comput. Phys. 15 (3), 299–319.

Trangenstein, J.A., 1987. Customized minimization techniques for phase equilibrium computations in reservoir simulation. Chem. Eng. Sci. 42 (12), 2847–2863.

Wadsley, W.A., 1980. Modelling reservoir geometry with non-rectangular coordinate grids. In: SPE Annual Technical Conference and Exhibition, Dallas, Texas, pp. 21–24. September. SPE-9369-MS. https://doi.org/10.2118/9369-MS.

Warren, J.E., Root, P.J., 1963. The behavior of naturally fractured reservoirs. SPE J. 3 (03), 245–255. SPE-426-PA. https://doi.org/10.2118/426-PA.

Xu, Y., 2015. Implementation and Application of the Embedded Discrete Fracture Model (EDFM) for Reservoir Simulation in Fractured Reservoirs. Master thesis. The University of Texas at Austin, Austin, Texas December 2015.

Xu, Y., Cavalcante-Filho, J.S.A., Yu, W., et al., 2017a. Discrete-fracture modeling of complex hydraulic-fracture geometries in reservoir simulators. SPE Reservoir Eval. Eng. 20 (02), 403–422. SPE-183647-PA. https://doi.org/10.2118/183647-PA.

Xu, Y., Yu, W., Sepehrnoori, K., 2017b. Modeling dynamic behaviors of complex fractures in conventional reservoir simulators. In: SPE/AAPG/SEG Unconventional Resources Technology Conference, Austin, Texas, 24-26 July.