A Continuous Projection-Based EDFM Model for Flow in Fractured Reservoirs

Harun U. Rashid[1] and **Olufemi Olorode**[1]*

[1]Lousiana State University

Summary

Various researchers have evaluated different methods of modeling fractured reservoirs accurately and efficiently. Of these methods, the embedded discrete fracture model (EDFM) is one of the most popular because it does not require the mesh for the simulation domain to conform to the orientation of the natural fractures. However, it is limited because it cannot accurately model low-conductivity fractures. Although the projection-based EDFM (pEDFM) was developed to address this limitation of the EDFM, recent studies show that pEDFM still cannot accurately model low-conductivity fractures that are neither parallel to the simulation grid nor cutting through the matrix cell diagonals. In these cases, it has been observed that reservoir fluids can flow around low- or even zero-conductivity fractures.

This paper presents a rigorous analysis, which reveals that the error when modeling inclined low-conductivity fractures with pEDFM is because the projections of the fracture on the gridblock faces are discontinuous. This discontinuity implies that the matrix cell interfaces along the path of the fractures only project a fraction of the fracture area on these interfaces in each direction. So, none of these faces captures the sealing potential of these interfaces, and the pore fluid can flow around even zero-conductivity fractures. We present a robust algorithm that ensures that the projections of inclined fractures on cell faces are continuous in 3D. We refer to the model based on this algorithm as the continuous projection-based EDFM (CPEDFM) and show that it directly solves the pEDFM limitation.

We present the simulation of several cases and their selected projection faces to demonstrate why CPEDFM works. We also verify the CPEDFM method by comparing the CPEDFM and pEDFM model results to high-resolution simulation results. To demonstrate the feasibility of modeling complex, realistic systems using CPEDFM, we simulate a 3D compositional Eagle Ford shale reservoir with 75 low-conductivity and 75 high-conductivity fractures. The results show that pEDFM overestimates production because it does not fully account for the sealing effects of inclined low-conductivity fractures. In conclusion, this paper presents a novel numerical model for accurately and efficiently simulating reservoirs containing fractures of arbitrary conductivity, size, and orientation in 3D.

Introduction

Over the last two decades, there has been an increased interest in studying fractured rocks. This can be attributed to the increased interest in developing unconventional oil and gas (UOG) reservoirs and enhanced geothermal systems (EGS), which are both tight and naturally fractured. However, most subsurface rocks are naturally fractured to an extent. Before UOG reservoirs and EGS, petroleum engineers developed other fractured reservoirs like fractured carbonates. To model such reservoirs, which typically have densely interconnected fractures, the dual porosity (Warren and Root 1963) and dual permeability models (Gilman and Kazemi 1983) were used. Unfortunately, these models' assumptions (of dense fracture distribution and homogeneous fracture properties) do not apply to fractured tight rocks like EGS and UOG reservoirs (Norbeck et al. 2016). As shown in **Fig. 1** (from Olorode and Rashid 2022), UOG reservoirs are known to have multiscale fractures with preferential fracture orientations that are dependent on the prevailing stress state over the geologic history of the rock (Shafiei et al. 2018). Studies suggest that the most productive regions in the reservoir, referred to as sweet spots, could be attributed to well-connected and conductive natural fracture networks (Zoback and Kohli 2019). Additionally, Curtis (2002) discussed the attributes and importance of sealing natural fractures in several shale plays, while Gale et al. (2014) reviewed several publications on cores and outcrops from different shale plays in the US. They indicated the presence of sealing (and conductive) fractures in tight rocks like the Eagle Ford, Barnett, Marcellus, Haynesville, and Woodford shale plays.

Modeling the fractures in tight rocks like EGS and UOG reservoirs is particularly important because these fractures play a significant role in the flow of fluids in such ultralow matrix permeability systems. To this end, several numerical fracture models have been developed, as summarized in Olorode et al. (2021). Considering the importance of computational efficiency and the need to account for each fracture individually in fractured tight rocks with heterogeneous fracture properties, the EDFM (Lee et al. 2000, 2001) has become one of the most commonly used fracture models in these systems. This is because, unlike the explicit and discrete fracture models, which also account for the fractures individually, EDFM does not require the mesh for the matrix to conform to the geometry of the natural fractures in the system. However, the limitations of the EDFM include its inaccuracy when modeling low-conductivity fractures and transient flow between a tight rock matrix and a fracture.

Considering the ultralow matrix permeability of these UOG reservoirs, some authors have modified the standard EDFM to account for the transient flow expected near the fractures. To this end, some have proposed numerical modifications based on local grid refinement (Rao et al. 2019a) and boundary integral equations (Rao et al. 2019b) to account for the matrix/fracture flow term in tight rocks. In contrast, for improved computational efficiency, Olorode and Rashid (2022) presented a transient matrix-fracture nonneighboring flux instead of the pseudosteady state assumption implicit in the standard EDFM.

The pEDFM was developed to address the limitations of EDFM in modeling low-conductivity fractures. It involves projecting a fracture on the interface between the matrix which hosts the fracture and its immediate neighboring cells. The neighbor that shares this projected face with the host matrix is called the projection matrix cell. In the original 2D pEDFM proposed by Tene et al. (2017), the pEDFM approach involves selecting a projection matrix cell in the x- and y-directions. The algorithm to determine which projection matrix cell is selected in each direction was given by Jiang and Younis (2017). However, the 2D pEDFM algorithm is limited to vertical and fully penetrating fractures and cannot be applied to modeling realistic 3D fractured reservoirs.

*Corresponding author; email: femorode@yahoo.com

Original SPE manuscript received for review 31 May 2023. Revised manuscript received for review 21 August 2023. Paper (SPE 217469) peer approved 30 August 2023.

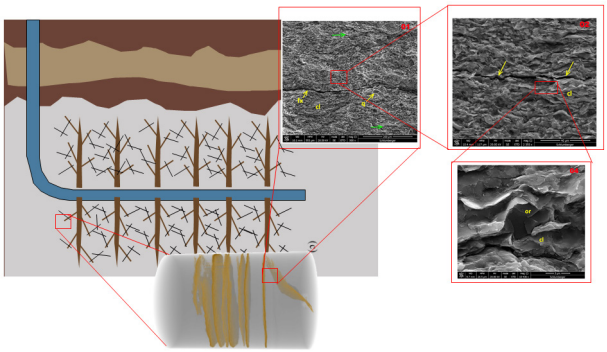

Fig. 1—This sketch from Olorode and Rashid (2022) illustrates the presence of fractures at multiple scales in tight source rocks. The man-made hydraulic fractures are on the order of millimeters and tens/hundreds of meters in aperture and length, whereas the microcracks are on the order of micrometers in length. The natural fractures are at an intermediate scale between these two and could be on the order of meters in length.

Olorode et al. (2020) presented a fully 3D pEDFM algorithm for simulating fractured reservoirs with any arbitrary orientations in a 3D space and demonstrated its application in modeling a compositional Eagle Ford reservoir with hundreds of stochastic fractures of arbitrary orientation. This 3D pEDFM was iteratively coupled with the extended finite element method to model hydraulic fracture propagation in a fractured rock (Rashid et al. 2022). In addition to applying pEDFM in modeling UOG reservoirs, some authors (Rao et al. 2022) have demonstrated its application in modeling geothermal systems. However, Rao and Liu (2022) pointed out that pEDFM cannot effectively model inclined low-conductivity fractures because the pEDFM projections are not "topologically homeomorphic to the fracture." In simple terms, this pEDFM limitation is because the area projected on the interfaces between the host and projection matrix cells is always less than the total area of these interfaces when the fractures are inclined and do not cut through the diagonals of the matrix cells. Although pEDFM has been verified in low-conductivity fractures that are either parallel to the spatial axes (Tene et al. 2017) or cutting through the matrix cell diagonals (Jiang and Younis 2017), a simple pEDFM algorithm that selects one of the immediate neighbors of the host matrix cell in each direction is inadequate at orientations other than these two unlikely cases. In this work, we present a clear illustration of the erroneous fluid leakage observed when modeling arbitrarily inclined low- or zero-conductivity fractures using pEDFM. We also present a robust 3D algorithm that provides a practical solution to this leakage problem.

Governing Equations for Modeling Fractured Reservoirs with pEDFM

Although several approaches have been proposed to model fluid flow in naturally fractured reservoirs, we focus on pEDFMs because they can model fractures of any conductivity and orientation. Like EDFM, pEDFM simulates fractures as $(n-1)$-dimensional cells. The matrix is discretized independently of the fractures using structured n-dimensional cells. This independence in the meshing of the matrix and fractures results in the embedment of the fracture cells in the matrix cells. Both pEDFM and EDFM account for the exchange of fluids between the matrix and fractures using the concept of nonneighboring connections (NNCs) and transmissibilities. In standard reservoir simulation, NNCs are used when a cell needs to exchange fluids with another cell that is not its topological neighbor. For the compositional simulations presented in this work, we add a nonneighboring mass rate q_i^{nnc} to the governing equation for compositional simulation (Eq. A-11, which is discussed in Appendix A), as follows:

$$\frac{V}{\Delta t}\left[\left(\phi\rho^l S^l X_i^l + \phi\rho^v S^v X_i^g\right)^{n+1} - \left(\phi\rho^l S^l X_i^l + \phi\rho^v S^v X_i^g\right)^n\right] + \mathrm{div}(\rho^l X_i^l \vec{v}_l + \rho^v X_i^g \vec{v}^v)^{n+1} - (\rho^l X_i^l q^l + \rho^v X_i^g q^v)^{n+1} + q_i^{nnc} = R_i^{n+1}, \quad (1)$$

where q_i^{nnc} is the mass rate of component i that is exchanged through the NNC (in units of mass per time). It is given as follows:

$$q_i^{nnc} = \sum_{m=1}^{N_{nnc}} A_m^{nnc} \sum_{\alpha=1}^{n_p} \frac{k_m^{nnc} k_{r\alpha}}{\mu^\alpha} \rho^\alpha X_i^\alpha \left[\frac{(p^\alpha - \rho^\alpha gz) - (p^\alpha - \rho^\alpha gz)_m^{nnc}}{d_m^{nnc}}\right], \quad (2)$$

The subscript m represents an index that ranges from 1 to the total number of NNCs for each cell (N_{nnc}). In this context, the flow potential of a cell is given as $(p^\alpha - \rho^\alpha gz)$, while that of its nonneighboring cell is given as $(p^\alpha - \rho^\alpha gz)^{nnc}$. To compute the transmissibility factor (T^{nnc}) between any two cells connected through NNCs, we need to estimate the area (A^{nnc}), permeability (k^{nnc}), and distance (d^{nnc}) of these connections. We can calculate the transmissibility factor using the following equation:

$$T^{nnc} = \frac{k^{nnc} A^{nnc}}{d^{nnc}}, \quad (3)$$

where subscript j represents the current NNC and ranges from one to the total number of NNCs (N_{nnc}). The flow potentials of the current cell and that of its nonneighboring cell are represented by the two quantities in the numerator of the last term, whereas k^{nnc}, A^{nnc}, and d^{nnc} represent the permeability, area, and distance of the NNCs, respectively. The expressions for k^{nnc}, A^{nnc}, and d^{nnc} are specific to each type of NNC. In EDFM, there can be up to four distinct transmissibilities. These include the standard transmissibility between neighboring cells in the matrix, standard transmissibility between neighboring cells in the same fracture plane, nonneighboring transmissibility between a matrix cell and an embedded fracture cell, and the nonneighboring transmissibility between two cells in different fracture planes. For further details on these EDFM NNCs, the reader is referred to Moinfar et al. (2014). We only present the equations required to calculate the nonneighboring transmissibilities here. The nonneighboring transmissibilities are computed as follows:

1. *Matrix-Fracture Connectivity:* The expressions for k^{nnc}, A^{nnc}, and d^{nnc} for this connectivity are given as

$$A^{nnc} = 2A_f, \tag{4}$$

$$k^{nnc} = \frac{k_m k_f}{k_m + k_f}, \tag{5}$$

$$d^{nnc} = \frac{\int_v x_n dv}{V}, \tag{6}$$

where k_m is the matrix permeability, k_f is the fracture permeability, and A_f is the fracture area. The symbols x_n, dv, and v represent the normal distance of the element from the fracture, the volume element, and cell volume, respectively. The k^{nnc}, A^{nnc}, and d^{nnc} values computed from these equations are then used to compute the corresponding nonneighboring transmissibility using Eq. 3.

2. *Intersecting Fracture Connectivity:* The nonneighboring transmissibility for this connectivity is given as follows:

$$T^{nnc} = \frac{T_1 T_2}{T_1 + T_2}, \tag{7}$$

where T_1 and T_2 are the half transmissibilities of the two intersecting fractures:

$$T_1 = \frac{k_{f1}\omega_1 L_{int}}{d_{f1}}, \tag{8}$$

$$T_2 = \frac{k_{f2}\omega_2 L_{int}}{d_{f2}}. \tag{9}$$

In these two equations, L_{int} represents the length of the line formed when two fracture cells intersect in a matrix cell, whereas ω_f and k_f are the fracture aperture and permeability, respectively. The symbols d_{f1} and d_{f2} represent the distances from the centroids of Fracture Cells 1 and 2 to the intersection line.

The pEDFM attempts to model low-conductivity fractures by adding an NNC between a fracture and one of its two neighboring matrix cells in each direction. An algorithm is needed to determine which matrix cell out of a pair of cells (in each direction) should be selected. Jiang and Younis (2017) presented this algorithm for 2D systems, whereas Olorode et al. (2020) presented an algorithm for 3D systems. The matrix cells chosen using this algorithm are known as "projection cells," while the matrix cells that contain the fracture cells are called "host cells." In addition to the standard connectivities of EDFM, the pEDFM has two additional NNCs, which are as follows:

1. *Projection Matrix/Fracture Transmissibility:* The projection matrix/fracture (pM-F) transmissibility is the nonneighboring transmissibility between a projection matrix cell and a fracture cell. Tene et al. (2017) provide the expression for this transmissibility as follows:

$$T^{nnc}_{\mathrm{pM-F}} = \frac{A_{if\perp\vec{x}} k^{nnc}_{\mathrm{pM-F}}}{d^{nnc}_{\mathrm{pM-F}}}, \tag{10}$$

where,

$$k^{nnc}_{\mathrm{pM-F}} = \frac{k_{\mathrm{pM}} k_f}{k_{\mathrm{pM}} + k_f}. \tag{11}$$

In this equation, $A_{if\perp\vec{x}}$ is the area of the fracture projections along the spatial directions. The symbol $\vec{x}$ represents the spatial coordinates (X and Y in 2D or X, Y, and Z in 3D), and d^{nnc}_{pM-F} is the distance between the centroid of the fracture and that of the projection cell.

2. *Projection Matrix/Matrix Transmissibility:* The projection matrix/matrix (pM-M) transmissibility is between a host matrix cell and its corresponding projection matrix cells. It is given as follows:

$$T^{nnc}_{\mathrm{pM-M}} = k\frac{A_{ii} - A_{if\perp x}}{\Delta\vec{x}_e}, \tag{12}$$

where $\Delta\vec{x}_e$ refers to the gridblock sizes in each spatial direction.

The pEDFM approach has several advantages over other fracture modeling approaches, including its low computation cost and ability to account for the low conductivity of fractures. However, the next section shows that the pEDFM cannot account for low-conductivity fractures that are inclined with respect to the grid coordinates.

Limitations of pEDFM. This section illustrates the source of the limitations of pEDFM regarding the modeling of slanted sealing fractures. **Fig. 2** presents a simple two-phase oil/water system with an injector/producer pair separated by a slanted low-conductivity fracture. Although the fracture is vertical or parallel to the z-axis, it is at an angle to the x- and y-axes. So, the system cannot be modeled effectively using Cartesian grids, which cannot conform to the orientation of this fracture. **Fig. 2** shows the pEDFM grid on the left and a high-resolution perpendicular bisection (PEBI) grid on the right. The PEBI grid can conform to the orientation of this single inclined fracture and is provided to obtain a high-resolution reference solution for comparison against the pEDFM simulation results. The input parameters for the simulation model studied in this section are summarized in **Table 1.**

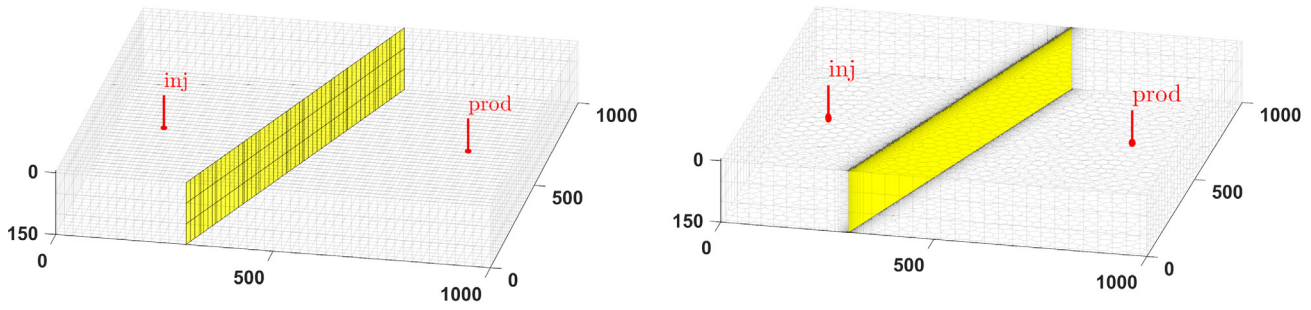

Fig. 2—The images show the simulation domain with a slanted sealing fracture, which isolates the producer from the injector. The left image shows the grid used in the pEDFM simulation, whereas the image on the right presents the PEBI grid used to obtain the high-resolution reference solution.

Input Data	Value	Unit
Number of grids	50×50×3	–
Physical dimensions	1000×1000×150	m
Matrix porosity	0.1	–
Matrix permeability	1×10^{-3}	darcies
Fracture porosity	0.01	–
Fracture permeability	1×10^{-9}	darcies
Fracture aperture	5	mm
Fluid viscosity	1	cp
Fluid compressibility	2.76×10^{-6}	1/psi
Initial pressure	2,900	psi
Well radius	0.0328	ft
Injection rate	1,086.8793	B/D
Bottomhole pressure	1,450.4	psi
Initial saturation	[0.2, 0.8, 0]	–

Table 1—Model parameters for the performance analysis of pEDFM when modeling slanted low-conductivity fractures.

We simulated water injection into the injector on the left of the fracture while producing oil from the production well on the right. **Fig. 3** presents the pressure profiles after simulating simultaneous injection and production for 15 years. The PEBI grid pressure solution shows a sharp contrast on either side of the fracture. This is because the sealing fracture curtails the flow of the injected fluid across it. So, the pressure builds up due to the injected fluid on the left of the fracture, whereas the pressure drops on the right of the fracture due to the withdrawal of oil from the well. In contrast, the pEDFM solution shows pressure communication across the fracture, which erroneously allows the injected fluid to provide pressure support for the producer. This difference between the pEDFM pressure and the reference pressure solutions highlights the limitation of the pEDFM in modeling slanted low-conductivity fractures.

Fig. 4 presents the corresponding oil production rate and cumulative oil production for the pEDFM and PEBI grid simulation cases. The results show that the oil production from the producer drops significantly due to the sharp drop in pressure of the blue region in **Fig. 3**. The pEDFM model overestimates oil production by a staggering 3,662.79% in this slanted, low-conductivity fracture scenario.

To understand why pEDFM yields erroneous results for slanted, low-conductivity fractures, **Fig. 5** illustrates the projection of the slanted sealing fracture (in yellow) on the interfaces between the matrix cells using the standard projection algorithm in Jiang and Younis (2017). The areas projected on the selected vertical projection faces are colored blue, whereas the corresponding areas projected on the horizontal faces are green. It is easy to see that although the yellow sealing fracture is expected to prevent the flow of the injected fluid across it, the partial areas projected on these interfaces allow the fluid to leak through the matrix cells. For instance, the fluid in Cell 11 can flow into Cell 12 through the portion of the shared face that is not colored blue. The fluid can then move from Cell 12 to Cells 4 and 13 toward the producer. A similar fluid leakage will be observed between Cells 1 and 2, 21 and 22, and so on.

Fig. 5 shows that the yellow sealing fracture to be modeled is continuous, but the union of the blue and green projected areas is discontinuous and unable to prevent the flow of fluids across it. Although our illustration of this pEDFM limitation was done in 2D for simplification, similar fluid leakage occurs in all three directions for sealing fractures inclined with respect to all three axes. This pEDFM limitation is significant because virtually all fractures in nature will have a nonzero angle of inclination with respect to the three spatial directions. To address this limitation, the next section discusses a new 3D algorithm for selecting and projecting the areas of slanted fractures such that their union forms a sealing interface that can prevent fluid flow when the fractures are sealing.

CPEDFM Algorithm

This section presents a pragmatic solution to the problem of fluid leakage across sealing fractures in the standard pEDFM method. To this end, we developed a robust 3D algorithm that can account for the sealing effect of fractures/faults of any arbitrary orientation in three

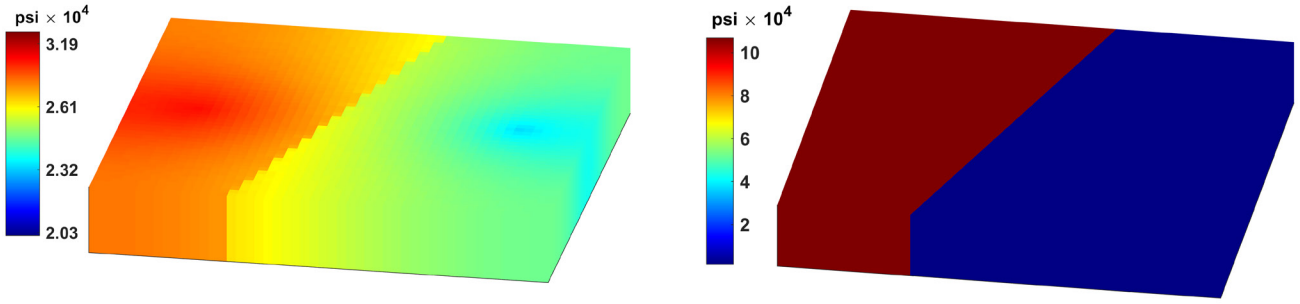

Fig. 3—Pressure profile (in psi) after 15 years of simulated production using pEDFM (left). The image on the right presents the corresponding reference solution.

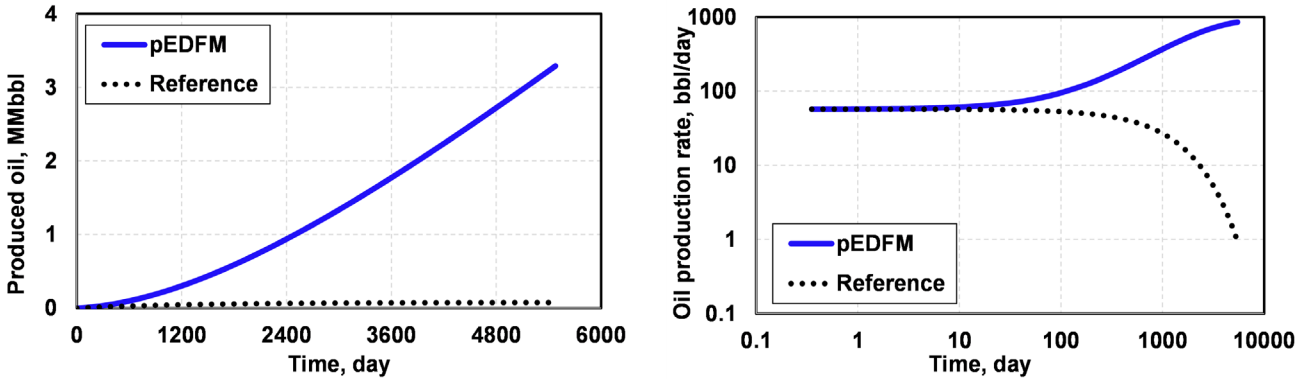

Fig. 4—Comparison of the pEDFM simulation results with the reference solution shows significant errors in the simulated (a) oil rate and (b) cumulative oil production.

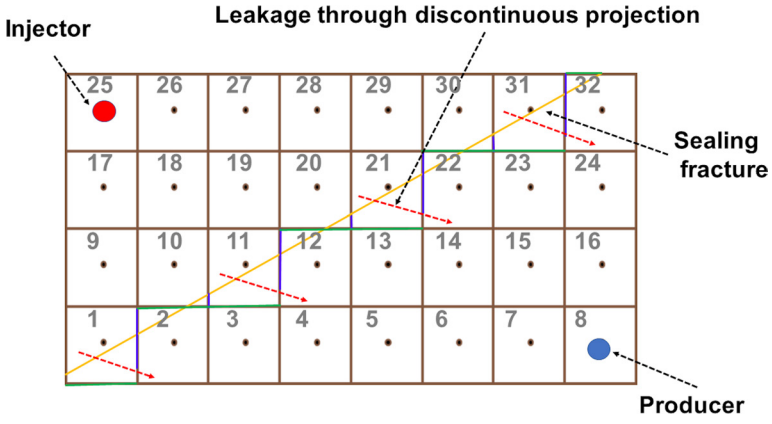

Fig. 5—This sketch illustrates the source of the erroneous fluid leakage through slanted sealing fractures when using the pEDFM projection algorithm. The matrix cells are numbered to illustrate the interfaces through which fluid leaks across the fracture. For instance, fluid leaks across the portion of the interface between Cells 11 and 12, which is not colored blue.

dimensions. By carefully inspecting the images of the selected faces from the pEDFM algorithm presented in the previous section, we observe that the problem can be fixed by ensuring that the projection faces selected are continuous. This can be achieved by projecting inclined fractures on stair-step interfaces in three dimensions. The 3D algorithm thus developed uses fracture slopes (in all three directions) to determine the cell interfaces into which the entire fracture plane is projected. This contrasts with the standard pEDFM algorithm, which finds the projection faces for each fracture segment independently of the other fracture segments of the same fracture plane in other cells.

Fig. 6, which was modified from Olorode et al. (2021), illustrates the fracture orientation angles in the 3D domain. To specify the orientation of a plane for the proposed algorithm, three measurements are required—dip, dip direction, and the z-component of the plane normal. The dip of a plane represents its inclination, which measures the steepest possible slope in the plane, while dip direction indicates the direction in which the plane slopes downhill (Waldron and Snyder 2020). A surface normal for a 3D domain at a given point is a vector perpendicular to the surface's tangent plane. It should be noted that for EDFM, pEDFM, or CPEDFM, which are developed for coplanar fractures, the surface normal for each segment of a fracture plane is constant.

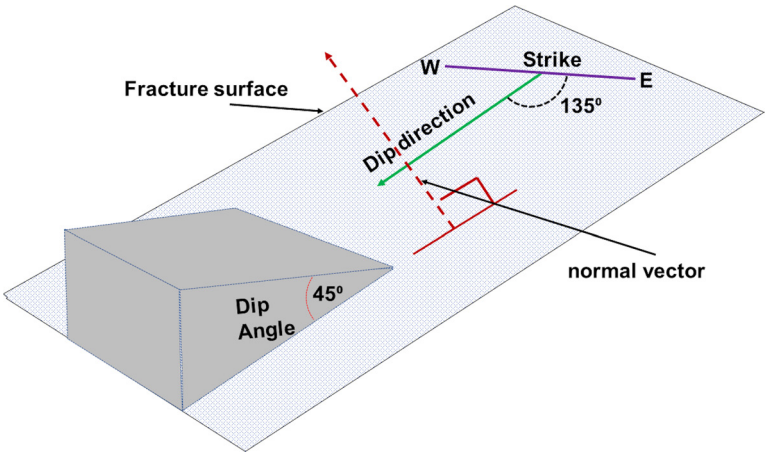

Fig. 6—Orientation of a fracture plane in a 3D domain (modified from Olorode et al. 2021).

To facilitate the selection of the faces on which the slanted fractures are projected, **Fig. 7** presents an ordering scheme for the local IDs of each face of a host matrix cell. In a 3D gridblock , the six faces are identified with local IDs ranging from one to six. The faces in the *X*-direction are denoted as left and right and are assigned local face IDs 1 and 2, respectively. Similarly, the two faces in the *Y*-direction are the front (with an ID of 3) and back (ID 4), whereas the faces in the *Z*-direction are the bottom (ID 5) and top (ID 6).

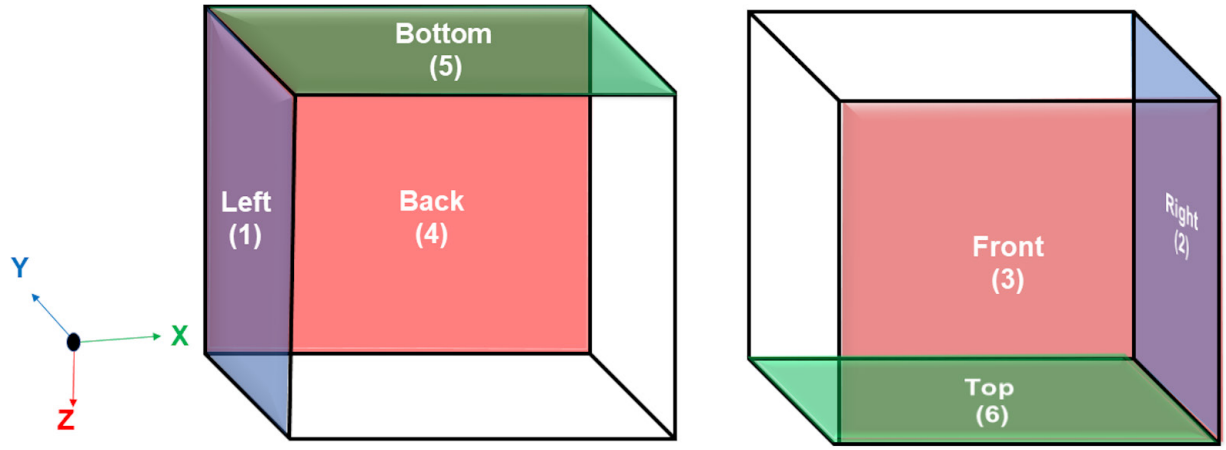

Fig. 7—Identification or indexing of faces in a 3D gridblock.

Fig. 8 presents a flow chart that illustrates the algorithm for projecting slanted fractures onto continuous matrix faces. The flow chart shows that the algorithm's first step involves computing the fracture plane's orientation angles (dip, dip direction, and normal) using the fracture vertices. Based on the orientation angles, we select three of the six faces for each matrix cell (in the simulation domain) containing segments of the fracture plane. The three chosen faces must be in different directions, so we can only select the left or right face, the front or back face, and the top or bottom face. For instance, if the dip direction lies between 0° and 90°, and if the *Z*-component of the normal vector is positive, we select Faces 1, 3, and 5 of each host matrix cell. The procedure for selecting the corresponding faces at different values of orientation angles is summarized in the other branches in the first four rows of this flow chart.

Considering that the selected faces from the first four steps of the algorithm could lead to discontinuous fracture projection areas, we need a procedure to ensure that the projection of these areas is continuous. So, we introduce the concept of a "local neighborhood" based on the coordinates of the centroids of all the selected faces for each fracture plane. The idea is to combine the partial projection areas of each host matrix cell within a region or local neighborhood into a full matrix face. To this end, we group all the faces selected for each fracture plane by the coordinates of their centroids such that all faces chosen with centroids having the same *Y*- and *Z*-coordinates, *X*- and *Z*-coordinates, and *X*- and *Y*-coordinates belong to the same *X*-, *Y*-, and *Z*-local neighborhoods, respectively. For example, **Fig. 9** shows that Faces 1, 2, and 3 belong to the same *X*-neighborhood.

The final step of the algorithm involves projecting all of the partial projection areas in each local neighborhood onto either the first or the last matrix face in the neighborhood, depending on the orientation angles. This selection of the first or last face and the projection of the total area are illustrated in **Fig. 10.** It shows the pEDFM partial projection areas of each fracture segment in each matrix cell on the left, whereas the continuous projection areas from CPEDFM are shown on the right. It is worth noting that the flow chart in **Fig. 8** does not consider the trivial case where the fractures are completely horizontal or vertical. This is because when fractures are completely vertical or horizontal, the entire fracture area is projected on the face between the two matrix cells on either side of the fracture. So, there is neither a local neighborhood nor a partial projection of areas. Therefore, CPEDFM uses the standard pEDFM algorithm, which simplifies into DFM and accurately models low-conductivity fractures in these cases.

To illustrate the continuous projection of fracture areas using CPEDFM, we present subvertical and inclined fractures as yellow planes in **Figs. 11a and 11c.** The corresponding projections of these fractures on the selected interfaces are shown in **Figs. 11b and 11d**. The

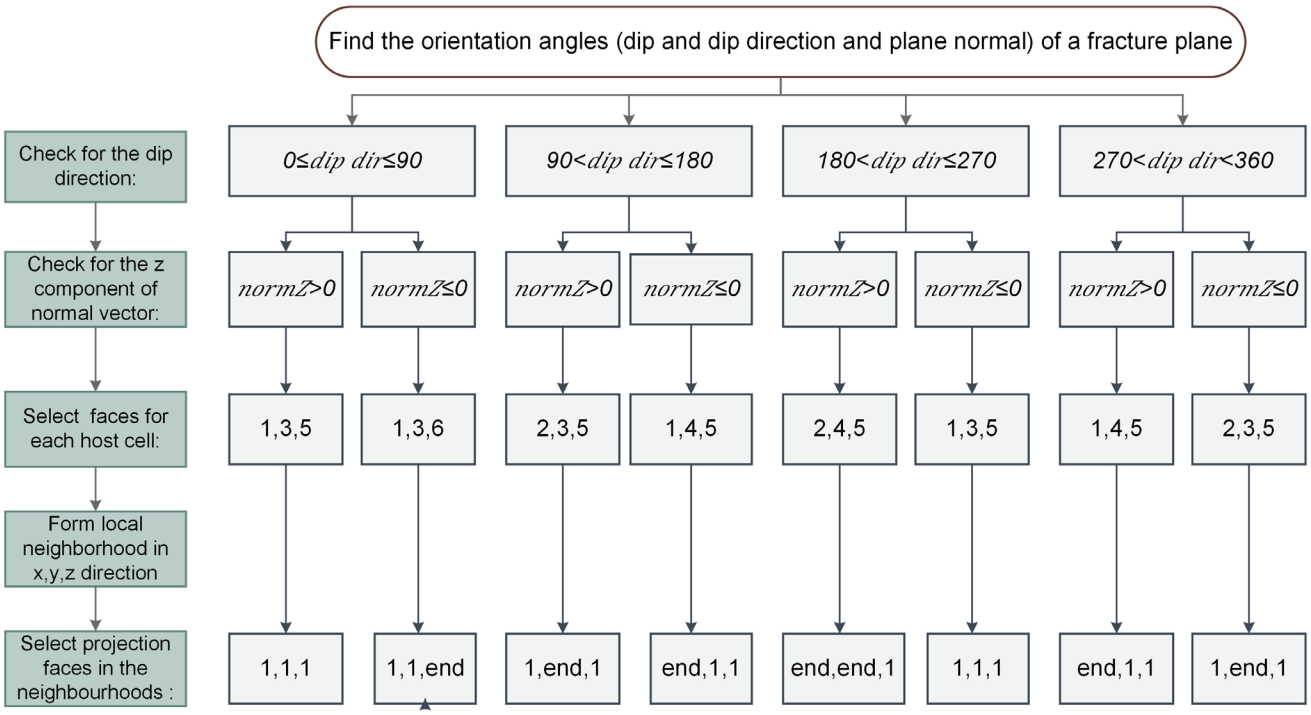

Fig. 8—Proposed algorithm for the continuous projection of fracture planes to prevent leakage through sealing fractures of any orientation.

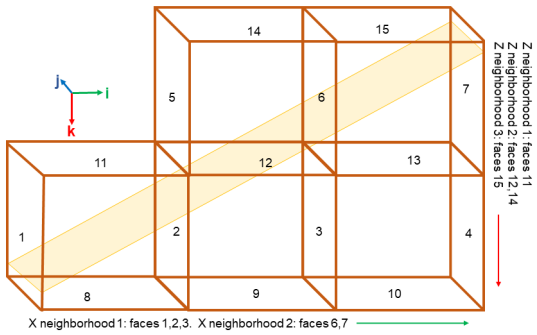

Fig. 9—Illustration of the formation of local neighborhoods.

selected faces in the *X*-, *Y*-, and *Z*-directions are shown in blue, red, and green, respectively. These images show that the projection of the fracture areas on the matrix cell interfaces is continuous.

We further illustrate the robustness of the CPEDFM algorithm by generating several stochastic realizations of fracture orientations using the open-source Alghalandis Discrete Fracture Network Engineering package (Fadakar Alghalandis 2017). **Fig. 12** illustrates a few

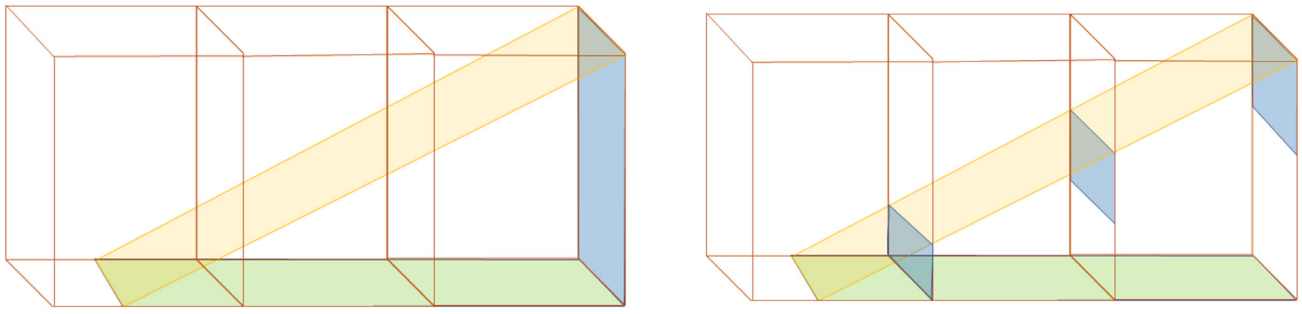

Fig. 10—Left sketch illustrates the partial areas (in blue) projected by the yellow fracture on the vertical matrix interfaces using pEDFM, whereas the sketch on the right illustrates the complete or continuous area projected using CPEDFM.

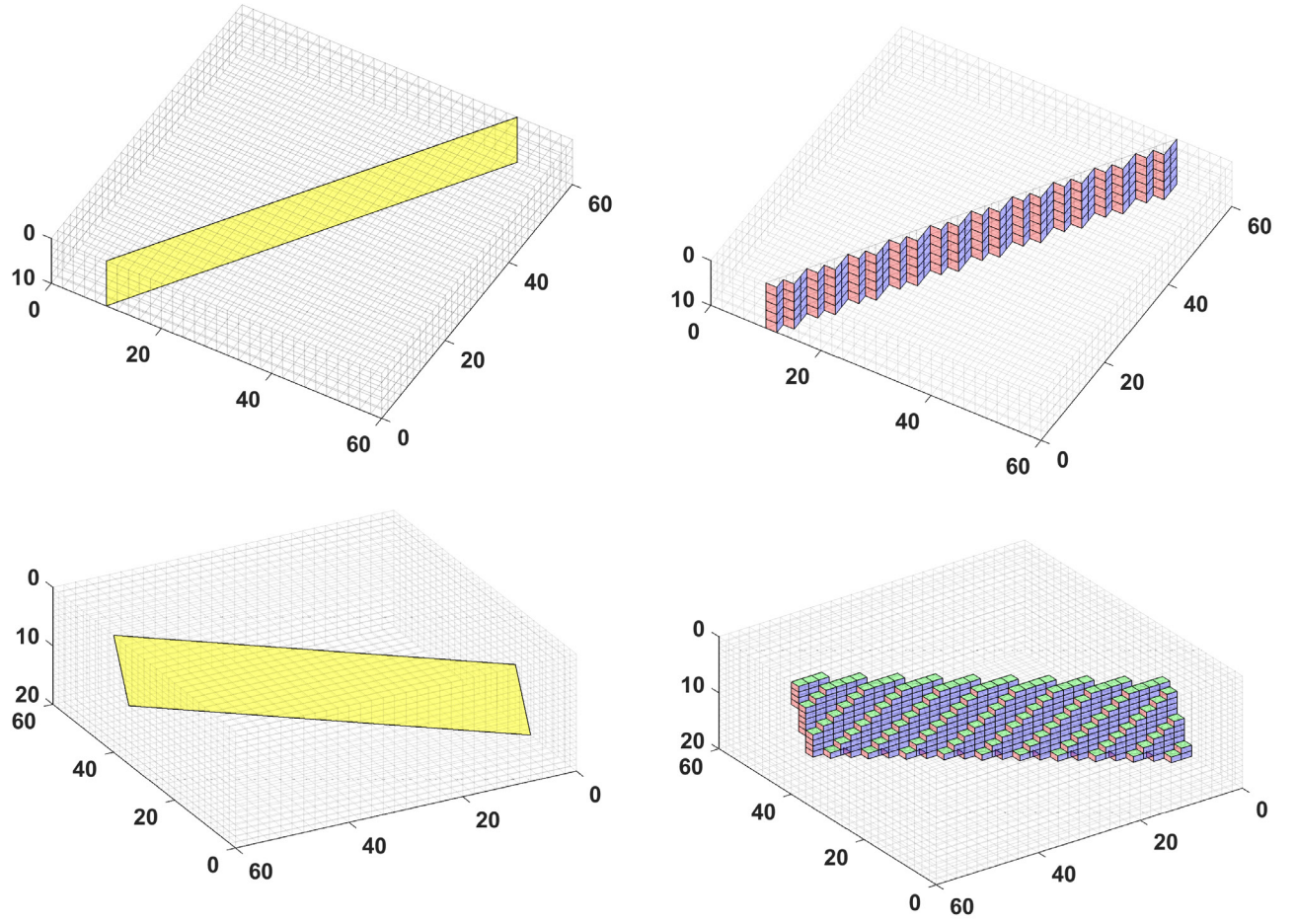

Fig. 11—CPEDFM generates continuous projections for slanted fractures. The matrix grids containing the fracture are presented on the left, whereas the corresponding fracture projections are given on the right.

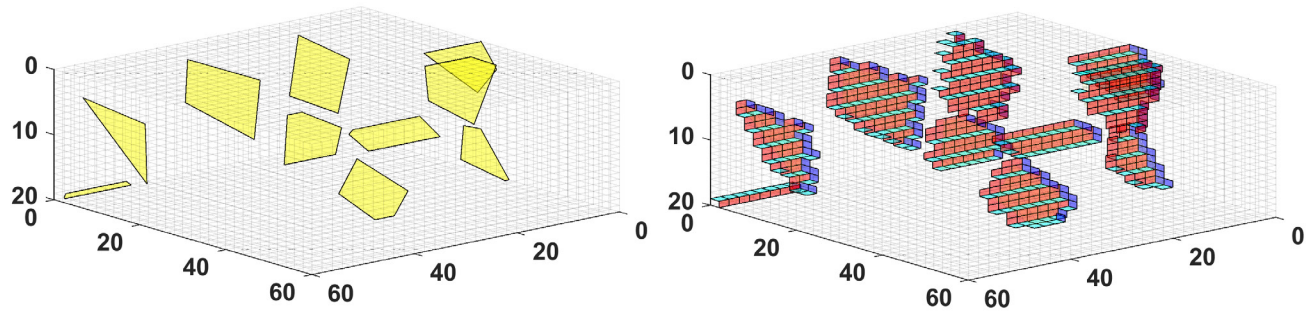

Fig. 12—CPEDFM generates continuous projections for randomly oriented stochastic natural fractures.

of these realizations of the fracture network. The inclined fractures are shown as yellow planes, whereas their projections on the *X*-, *Y*-, and *Z*-matrix faces are shown in blue, red, and green, respectively. A close inspection of the CPEDFM projections for all 10 fractures (as well as other realizations not presented for brevity) shows that the projections are continuous. So, CPEDFM addresses this limitation of the standard pEDFM method.

Given the continuous projection faces obtained from CPEDFM, we also need an algorithm to accurately calculate the transmissibility of the selected continuous faces. An inspection of the projection faces selected from the CPEDFM algorithm in **Figs. 10 and 13** reveal that the chosen projection faces for a fracture neighborhood can consist of multiple fracture cells. So, we need to compute the pM-F segment transmissibility for each neighborhood using Eq. 10, but with a redefinition of the area and distance. For example, the area to be used for the horizontal transmissibility across the projection matrix Cell 9 and the red fracture segment equals the area of the blue-colored interface between Cells 9 and 10. The distance in this equation is illustrated as the dotted gray arrow between matrix Cell 9 and the centroid of the red fracture segment. Finally, the nonneighboring permeability is essentially the harmonic average of the projection matrix cell and fracture permeability values.

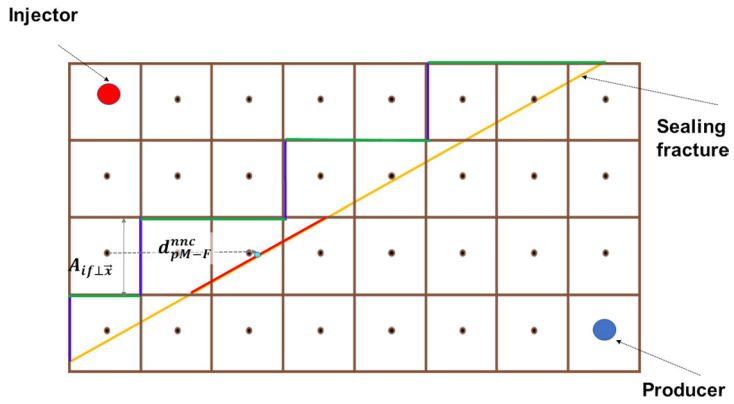

Fig. 13—This sketch illustrates the computation of the CPEDFM NNC distance (d^{nnc}_{pM-F}) and projection areas ($A_{if\perp\vec{x}}$) in Eq. 10.

Verification of CPEDFM. We revisit the classic test case presented in Tene et al. (2017) and Olorode et al. (2020) to verify CPEDFM against a high-resolution explicit or fully dimensional fracture model. The original classic test case involved flowing water from the west to the east face of the simulation domain by applying a high pressure on the west face of the mesh presented in **Fig. 14a** and a low pressure on the east face. Considering our focus on validating CPEDFM for low-conductivity fractures, we only simulate the cases where these intersecting fractures are of low conductivity. In **Fig. 14a**, the two fractures intersect at 90° and are parallel to the *X*- and *Y*-axes of the mesh. However, to validate the sealing effect of slanted fractures, we rotated the north-south fracture anticlockwise by 6.5° but left the east-west fracture unchanged, as shown in the Cartesian mesh in **Fig. 14b**. This Cartesian grid is used in the EDFM, pEDFM, and CPEDFM simulations discussed in this section but cannot be used to model the high-resolution, fully dimensional, reference solution because the north-south fracture is not parallel to the *X*- and *Y*-axes of the mesh. So, we use the high-resolution PEBI grid presented in **Fig. 14c** for the reference solution.

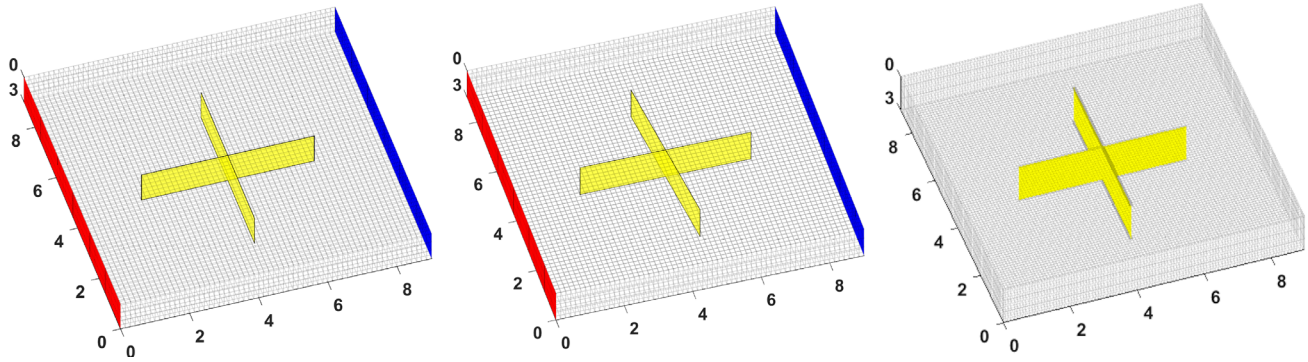

Fig. 14—The image in (a) is the Cartesian mesh used for EDFM, pEDFM, and CPEDFM simulation of the orthogonal intersecting fracture case. The mesh presented in (b) is used for EDFM, pEDFM, and CPEDM simulation of the case with the north-south fracture rotated anticlockwise by 6.5°. The high-resolution PEBI mesh in (c) is used to obtain the reference solution for the case with the north-south fracture rotated anticlockwise by 6.5°.

The results in the first row (a-d) of **Fig. 15** are for the standard case presented in Olorode et al. (2020), where the two low-conductivity fractures intersect at 90°. As expected, EDFM cannot accurately model the sealing fracture, whereas pEDFM and CPEDFM match the high-resolution fully dimensional model results. With the fractures intersecting at 90°, CPEDFM essentially simplifies into pEDFM because all the projections of the fractures are continuous in this case. However, when the north-south fracture is rotated anticlockwise by 6.5°, the projections of the segments of this fracture on the matrix-matrix interfaces are discontinuous, resulting in the partial fluid leakage that can be seen in the pEDFM solution in **Fig. 15f**. The fluid leakage essentially creates discontinuities in the coin-like shape of the pressure profile at the location of the slanted fracture. Also, it reduces the size of the coin in comparison to the reference solution. This is because the fluid flows around the fracture, resulting in a less steep pressure drop across it.

In contrast, CPEDFM still yields the expected sharp pressure drop in the reference solution **(Fig. 15h)**, whereas EDFM cannot accurately model any sealing fracture. It is worth mentioning that the appearance of the reference solution is a little different because it requires a different plotting functionality that essentially triangulates the solutions in the unstructured/PEBI grid to render the pressure profile. It is also expected to be more accurate because several fine meshes were used near the slanted fractured surface, whereas the meshes used in the EDFM, pEDFM, and CPEDFM cases are more coarse. The fine PEBI grid used in the reference solution is shown in **Fig. 14c**, whereas the coarse mesh used in the other cases is shown in **Fig. 14b**.

Fig. 16 provides a more quantitative comparison of the EDFM, pEDFM, and CPEDFM simulation results to the high-resolution reference solution for the intersecting orthogonal fracture case and the case where the north-south fracture is rotated anticlockwise by 6.5°. The pressures shown in this figure were obtained by extracting the pressure in the middle cells (in the *Y*-direction) running from the west to the east of the simulation domain.**Fig. 16a** is consistent with the results we presented in Olorode et al. (2020), where the two intersecting fractures were orthogonal to each other. It shows that both pEDFM and CPEDFM match the reference solution in this case. However, when the north-south fracture is rotated anticlockwise by 6.5°, the results in **Fig. 16b** show that pEDFM is inaccurate because the fluid

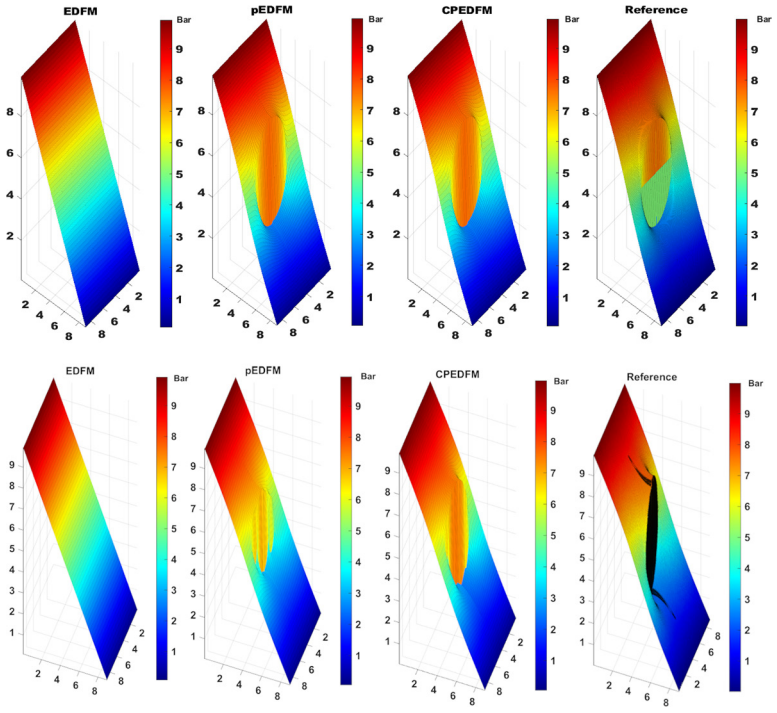

Fig. 15—Pressure profile at the end of the simulation for the intersecting orthogonal fracture cases (a–d) and for the cases with the north-south fracture rotated anticlockwise by 6.5° (e–h).

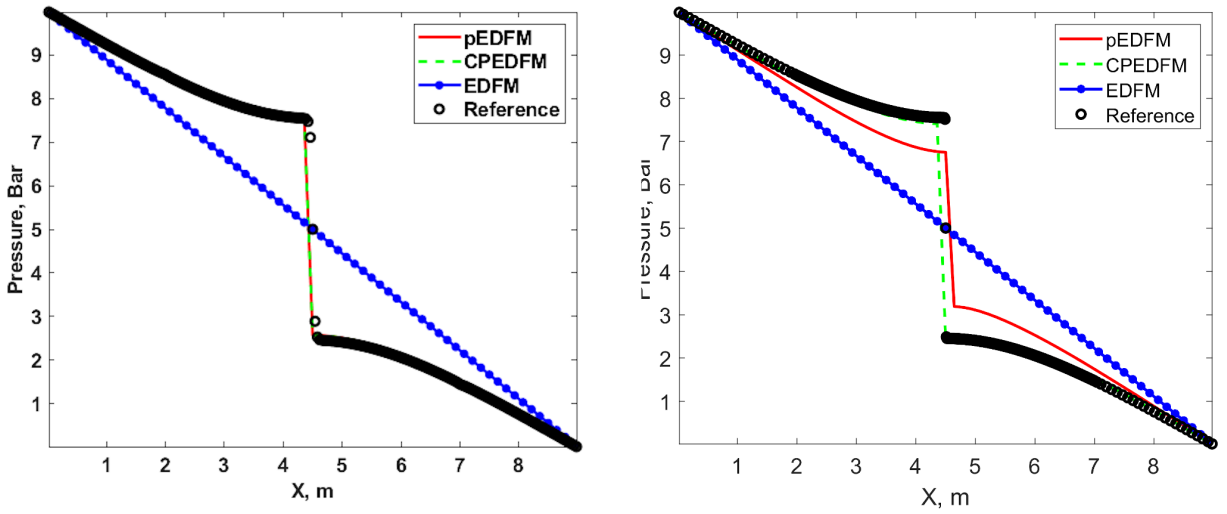

Fig. 16—Comparison of pressure along the fracture line shows that pEDFM and CPEDFM match the reference solution in the straight fracture case (a). In contrast, only CPEDFM matches the reference solution in the slanted fracture case (b). EDFM cannot match the reference solution in both cases because the fracture is of low conductivity.

partially leaks through the fracture, resulting in less steep pressure drop across the slanted fracture. The results also show that CPEDFM matches the reference solution in this slanted fracture case.

In addition to validating the intersecting low-conductivity fracture cases, we also present verified CPEDFM against the high-resolution PEBI simulation of the slanted sealing fracture system presented in **Fig. 2**. We compared the CPEDFM pressure profiles and production against the reference solutions presented in **Figs. 3 and 4**. The images presented in **Figs. 17a and 17b** represent the pressure profiles at the end of the simulation for the reference and CPEDFM cases. The results show that the pressure profile for CPEDFM matches that of the reference solution, but the interface is a stair step instead of the straight line obtained from using high-resolution PEBI grids in the reference solution. This is because CPEDFM projects the fracture on the interface between matrix cells, allowing the fractures to be meshed independently of the matrix cells.

Figs. 17c and 17d compare the oil production rates and cumulative oil production from CPEDFM to the reference solution. Unlike pEDFM, which cannot account for the sealing effect of an inclined fracture, CPEDFM shows a close match of the production rates and volumes to the reference solution. The potential of CPEDFM lies in its accuracy and increased computational efficiency because it does

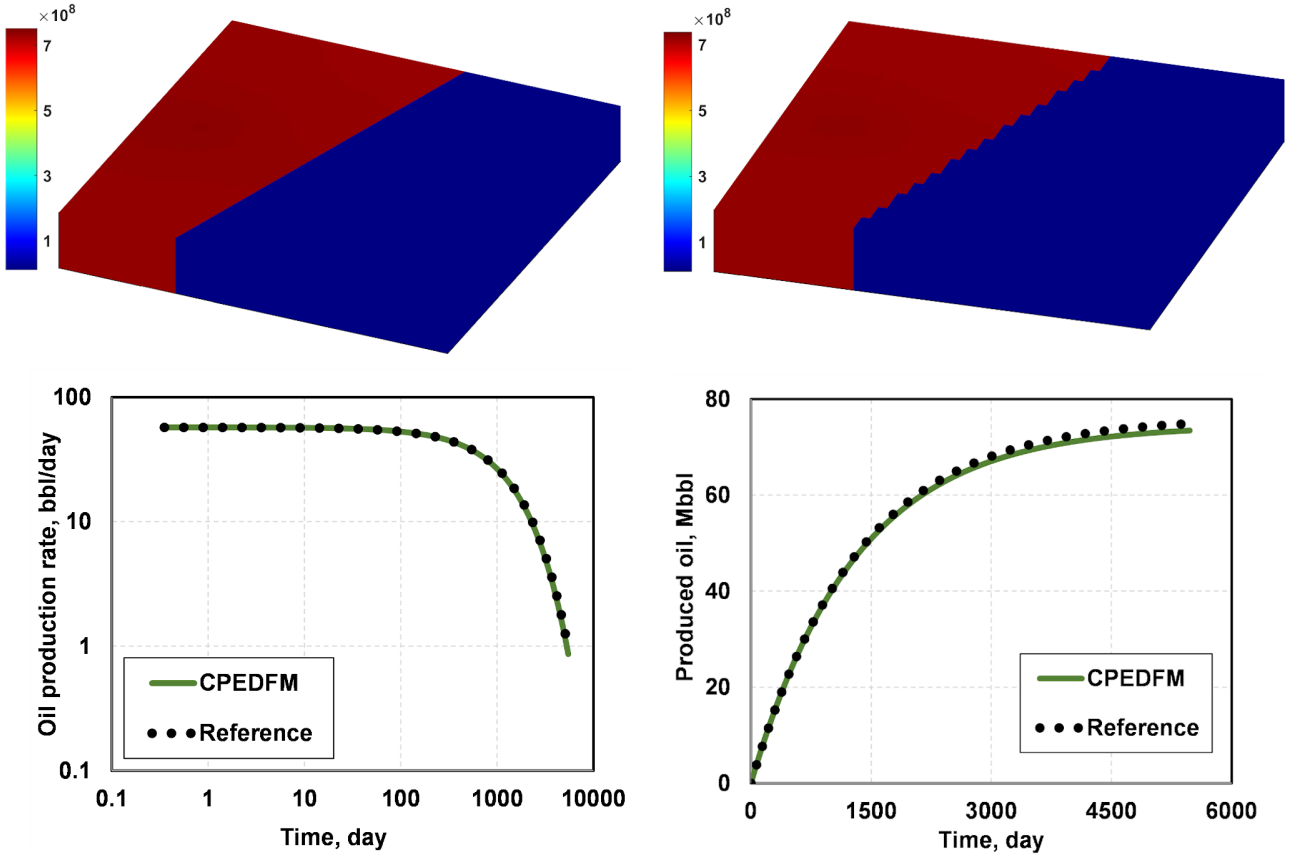

Fig. 17—The images in (a) and (b) represent the pressure profiles (in Pa) for the reference and CPEDFM solutions after 15 years. These pressure profiles and the plots of oil rate and cumulative oil production in (c) and (d) show that CPEDFM matches the reference solution.

not require the meshes for the matrix to conform to the orientation of all fractures in a fractured reservoir. The next section presents the application of CPEDFM to a representative Eagle Ford shale stencil with hundreds of natural fractures in different orientations.

Application of CPEDFM to Representative Fractured Eagle Ford Shale

This section uses CPEDFM to simulate a representative Eagle Ford shale with stochastic fractures of different orientations and conductivity values. Considering the symmetry expected in the numerical solution of multistage fractured horizontal wells with biwing planar fractures, we model only a smaller repeatable fraction of the entire fractured well simulation domain. This fraction of the domain of a fractured well is referred to as a stencil. Its ability to capture the behavior of a complete system of multiple fractures is demonstrated in Olorode et al. (2013). As shown in **Fig. 18,** the stencil used in the Eagle Ford simulation studies discussed in this section covers a quarter of the simulation domain around one fracture stage. To obtain the production rates and volumes for the entire fractured well, we multiply these values by the number of times the stencil occurs within the whole system. This allows us to obtain high-resolution simulation results for fractured well simulation domains with thousands of fractures.

Table 2 summarizes the model parameters used to simulate the representative Eagle Ford shale well. The composition and binary interaction coefficients of the fluid simulated in the Eagle Ford fractured well are summarized in **Tables B-1 and B-2** of Appendix B. Although UOG reservoirs are known to have ultralow matrix permeability values, it could be misleading to assume that all the natural fractures in these tight rocks will have high-conductivity values. This is because many of these fractures were mechanically active in the past but are no longer active in the current state of in-situ rock stress. In addition, the "critically-stressed-fault hypothesis" indicates that mechanically active faults act as fluid flow conduits, while mechanically inactive faults act as flow barriers (Zoback 2010). So, half of the 150 fractures in the stencil simulated had high conductivity, while the other half had low conductivity. These conductivity values are provided in **Table 2**.

We simulated the fractured Eagle Ford well system for 30 years using EDFM, pEDFM, and CPEDFM. **Fig. 19** presents the simulated cumulative oil and gas corresponding to these three models. As expected, EDFM overestimates the cumulative production of all the reservoir fluids because it cannot account for the low-conductivity fractures present in the simulation domain. Although pEDFM attempts to model these low-conductivity fractures by adding two more nonneighboring connectivities, the fluid still leaks through the partial areas projected onto the matrix faces in systems with inclined fractures. This explains why pEDFM overestimates the cumulative production compared to CPEDFM for this system, which contains stochastic natural fractures.

The stochasticity of the natural fractures implies that it is practically impossible to generate random fractures that will be exactly parallel to one of the spatial coordinates or run exactly through the matrix cell diagonals. So, every stochastic fracture in the simulation domain is essentially inclined to varying degrees, leading to errors in the low-conductivity fractures modeled with pEDFM. However, the difference between the pEDFM and CPEDFM results in **Fig. 19** are negligible because the system simulated is under primary production, and the natural fracture networks do not effectively block off the flow of fluids toward the hydraulically fractured well. The fact that half

Input Data	Value	Unit
Stencil model dimension	492.1×984.3×125.0	ft
Initial reservoir pressure	8,125	psi
Reservoir temperature	270	°F
Reservoir thickness	250.0	ft
Matrix permeability	9×10^{-7}	darcies
Matrix porosity	0.12	–
Hydraulic fracture porosity	0.5	–
Hydraulic fracture half-length	590.6	ft
Hydraulic fracture aperture	3.048	mm
Hydraulic fracture height	150	ft
Hydraulic fracture permeability	1	darcy
Natural fracture aperture	1×10^{-4}	mm
Conductive natural fracture permeability	100×10^{-6}	darcies
Sealing natural fracture permeability	1×10^{-12}	darcies
Number of fractures per stage	3	–
Cluster spacing	10	m
Well radius	0.1	m
Initial water saturation	0.17	–
Flowing bottomhole pressure	1,000	psi
Number of natural fractures	150	–

Table 2—Parameters used in the simulation of a representative Eagle Ford shale well.

of the stochastic natural fractures are conductive also facilitates the flow of reservoir fluids through the low-conductivity fractures in the simulation domain.

Our analysis of the results presented in **Fig. 19** reveals that pEDFM overestimates the expected ultimate oil recovery by 2.12% compared to the results from CPEDFM. With an increase in the size and number of sealing fractures in a reservoir, the magnitude of the error in the pEDFM model could be much higher. These sealing fractures could also curtail production more significantly during cyclic gas enhanced oil recovery, where they could curtail the efficient recovery of oil from certain parts of the reservoir.

Fig. 20 presents the pressure profile after simulating 30 years of production using (a) EDFM, (b) pEDFM, and (c) CPEDFM. As expected, we do not see the effect of the sealing of natural fractures near the hydraulic fractures in EDFM. The pEDFM results show some sealing effect, but comparing the pressure profile on the outer sides of these sealing fractures to those from CPEDFM indicates a less steep drop in pressure in the pEDFM solution. These results again confirm the ability to effectively model sealing fractures of any orientation

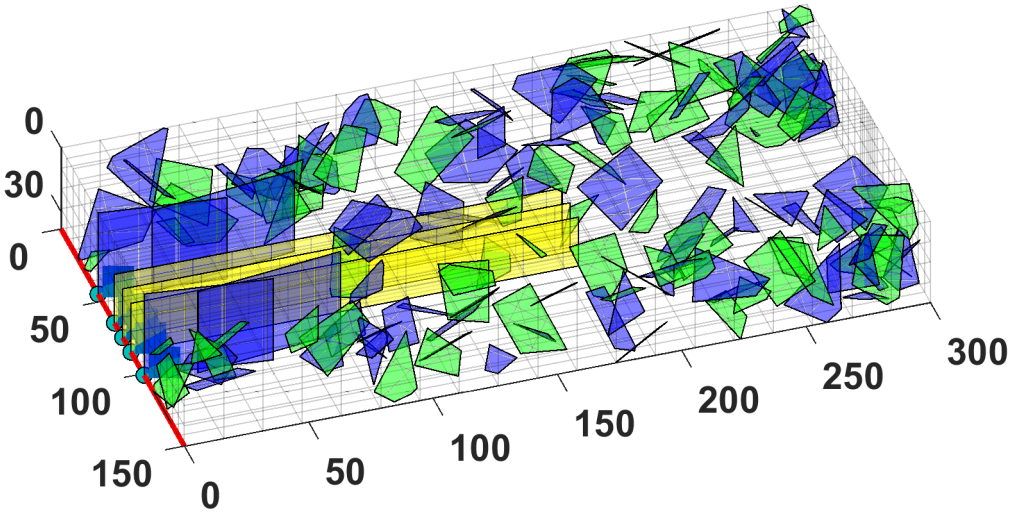

Fig. 18—This image illustrates the stencil used to simulate a representative Eagle Ford fractured well. The hydraulic, sealing, and conductive natural fractures are colored yellow, blue, and green, respectively. This stencil is essentially a quarter of the simulation domain around one fracture stage.

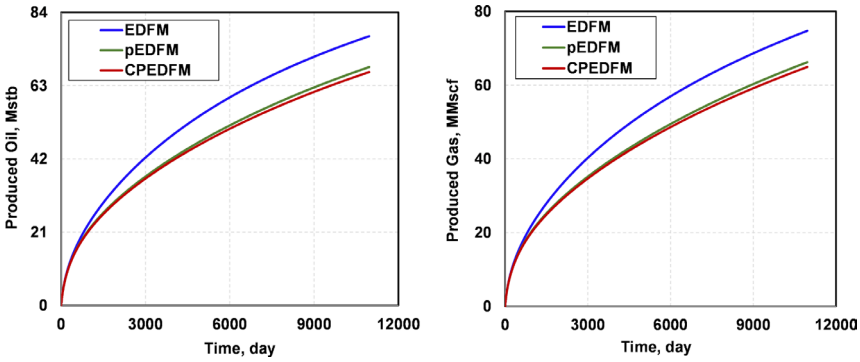

Fig. 19—The plots of (a) cumulative oil production and (b) cumulative gas production for the EDFM, pEDFM, and CPEDFM simulations of the fractured Eagle Ford well stencil. It shows that EDFM and pEDFM overestimate production in tight rocks with inclined sealing fractures.

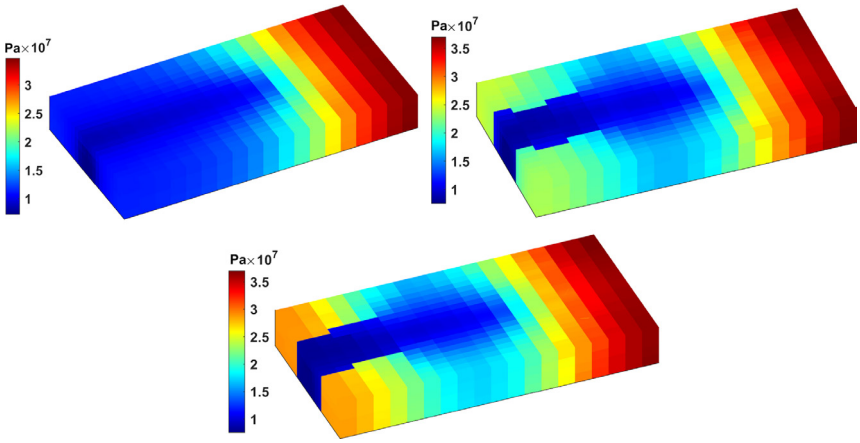

Fig. 20—The pressure profile after 30 years of production using (a) EDFM, (b) pEDFM, and (c) CPEDFM. The large slanted natural fractures near the hydraulic fractures create a flow barrier that is not captured using EDFM. The pEDFM method tries to capture this flow barrier but shows flow leakage that is evident in the lack of contrast between the dark blue and green colors near these natural fractures. Conversely, CPEDFM shows the expected sharp contrast in the pressure profile near the location of the flow barrier.

using the proposed CPEDFM model. It is essential to note that the results presented in this section are based on the stencil and have not been multiplied by the number of times the stencil occurs in the actual multistage fractured horizontal well system.

Conclusions

This work presents the development and validation of a 3D CPEDFM, enabling fast and accurate modeling of fractured tight reservoirs that typically have fractures of different orientations and conductivities. We show that pEDFM yields inaccurate results for low-conductivity fractures that are not parallel to the spatial axes because it treats each fracture cell independently, resulting in a discontinuous projection of the fracture on the matrix faces. To address this pEDFM limitation, we present a robust algorithm that ensures a continuous projection of fractures on matrix faces, as well as the modification of the terms in the pEDFM NNCs.

To demonstrate the accuracy of the proposed CPEDFM model, we verified it by rotating the north-south fracture in the classical intersecting fracture problem by 6.5°. We also verified it against a high-resolution reference solution obtained by explicitly modeling a slanted sealing fracture with PEBI grids. We compared CPEDFM and pEDFM model results to these reference solutions and confirmed that pEDFM cannot account for slanted low-conductivity fractures, whereas CPEDFM can. We demonstrate the practical application of CPEDFM to the compositional simulation of the production from a realistic fractured Eagle Ford shale well with stochastic fractures of low and high conductivity. The results show that EDFM overestimates the production of reservoir fluids (by ~13%) because it cannot account for the low conductivity of the fractures. In contrast, pEDFM slightly overestimates production (by ~2%) because it cannot accurately account for the inclined low-conductivity fractures in the simulation domain. The CPEDFM model presented can enhance our understanding of the production of fluids from fractured reservoirs and the injection of fluids like carbon dioxide and hydrogen into fractured/faulted subsurface rocks. This improved understanding is essential to efficient and optimal fluid production from or injection into these subsurface rocks.

Nomenclature

$Aif\perp x$ = area of fracture projections along each dimension, L^2, m^2
A^p_{ij} = projection area, L^2, m^2
A^{nnc} = area of a nonneighboring connection, L^2, m^2
d^{nnc} = nonneighboring connection distance, L, m

k_f = fracture permeability, L^2, m^2
k_m = matrix permeability, L^2, m^2
k^{nnc} = permeability of a nonneighboring connection, L^2, m^2
K_i = vapor-liquid equilibrium constant for component i
R = residual, $M^{-1}L^{-3}$, $kg^{-1}m^{-3}$
S^α = saturation of phase, α
T_i = half transmissibility, L^3, m^3
T^{nnc} = transmissibility factor for a nonneighboring connection, L^3, m^3
t = time,T, seconds
q^α = volumetric flow rate of phase α, L^3/T, m^3/s
$\vec{x}_e$ = gridblock sizes in the X-, Y-, and Z-directions, L, m
$\vec{x}$ = X-, Y-, and Z-coordinates, L, m
x_i = mole-fraction of component, i in the liquid phase
X_i^α = mass-fraction of component i in phase, α
Y_i = mass-fraction of component, i in the gas phase
y_i = mole-fraction of component, i in the gas phase
Z_i = overall mass-fraction of component, i
ϕ = porosity
ρ = density, M/L^3, kg/m^3

Subscripts

α = fluid phase
f = fracture
if = interaction between matrix cell i and fracture cell f
i, j = cell indices
m = matrix
M−M = interaction between two different host matrix cells
pM−F = interaction between a projection matrix and a fracture cell

Superscripts

$k+1$ = current timestep
nnc = nonneighboring connections

Acknowledgments

This work was facilitated using the software packages available in the Craft and Hawkins Department of Petroleum Engineering at Louisiana State University and the computational resources at the LSU Center for Computation and Technology.

References

Coats, K. H. 1980. An Equation of State Compositional Model. *SPE J.* **20** (5): 363–376. https://doi.org/10.2118/8284-PA.

CurtisJ. B.2002. Fractured Shale-Gas Systems. *AAPG Bulletin* **86** (11): 1921–1938. https://doi.org/10.1306/61EEDDBE-173E-11D7-8645000102C1865D.

Fadakar Alghalandis, Y. 2017. ADFNE: Open Source Software for Discrete Fracture Network Engineering, Two and Three Dimensional Applications. *Comput Geosci* **102**: 1–11. https://doi.org/10.1016/j.cageo.2017.02.002.

Firoozabadi, A. 2015. *Thermodynamics and Applications of Hydrocarbon Energy Production*. New York, USA: McGraw Hill Professional.

Gale, J. F. W., Laubach, S. E., Olson, J. E. et al. 2014. Natural Fractures in Shale: A Review and New Observations. *AAPG Bulletin* **98** (11): 2165–2216. https://doi.org/10.1306/08121413151.

Gilman, J. R. and Kazemi, H. 1983. Improvements in Simulation of Naturally Fractured Reservoirs. *SPE J.* **23** (4): 695–707. https://doi.org/10.2118/10511-PA.

Jiang, J. and Younis, R. M. 2017. An Improved Projection-Based Embedded Discrete Fracture Model (pEDFM) for Multiphase Flow in Fractured Reservoirs. *Adv Water Resour* **109**: 267–289. https://doi.org/10.1016/j.advwatres.2017.09.017.

Lee, S. H., Jensen, C. L., and Lough, M. F. 2000. Efficient Finite-Difference Model for Flow in a Reservoir With Multiple Length-Scale Fractures. *SPE J.* **5** (3): 268–275. https://doi.org/10.2118/65095-PA.

Lee, S. H., Lough, M. F., and Jensen, C. L. 2001. Hierarchical Modeling of Flow in Naturally Fractured Formations with Multiple Length Scales. *Water Resour Res* **37** (3): 443–455. https://doi.org/10.1029/2000WR900340.

Lie, K.-A. 2019. An Introduction to Reservoir Simulation Using MATLAB/GNU Octave. In *An Introduction to Reservoir Simulation Using MATLAB/GNU Octave: User Guide for the MATLAB Reservoir Simulation Toolbox (MRST)*. Chennai, Tamil Nadu: Cambridge University Press. https://doi.org/10.1017/9781108591416.

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. https://doi.org/10.2118/154246-PA.

Møyner, O. and Tchelepi, H. 2017. A Multiscale Restriction-Smoothed Basis Method for Compositional Models. Paper presented at the SPE Reservoir Simulation Conference, Montgomery, Texas, USA, 20–22 February. https://doi.org/10.2118/182679-MS.

Norbeck, J. H., McClure, M. W., Lo, J. W. et al. 2016. An Embedded Fracture Modeling Framework for Simulation of Hydraulic Fracturing and Shear Stimulation. *Comput Geosci* **20** (1): 1–18. https://doi.org/10.1007/s10596-015-9543-2.

Olorode, O. M. M., Freeman, C. M. M., Moridis, G. J. J. et al. 2013. High-Resolution Numerical Modeling of Complex and Irregular Fracture Patterns in Shale-Gas Reservoirs and Tight Gas Reservoirs. *SPE Res Eval & Eng* **16** (4): 443–455. https://doi.org/10.2118/152482-PA.

Olorode, O. and Rashid, H. 2022. Analytical Modification of EDFM for Transient Flow in Tight Rocks. *Sci Rep* **12** (1). https://doi.org/10.1038/s41598-022-26536-w.

Olorode, O., Wang, B., and Rashid, H. U. 2020. Three-Dimensional Projection-Based Embedded Discrete-Fracture Model for Compositional Simulation of Fractured Reservoirs. *SPE J.* **25** (4): 2143–2161. https://doi.org/10.2118/201243-PA.

Olorode, O., Wang, B., and Rashid, H. 2021. Numerical Modeling of Fractured Unconventional Oil and Gas Reservoirs. In *Advanced Modeling with the Matlab Reservoir Simulation Toolbox (MRST)*, 409–453. Chennai, Tamil Nadu: Cambridge University Press. https://doi.org/10.1017/9781009019781.

Peng, D.-Y. and Robinson, D. B. 1976. A New Two-Constant Equation of State. *Ind. Eng. Chem. Fund* **15** (1): 59–64. https://doi.org/10.1021/i160057a011.

Rao, X., Cheng, L., Cao, R. et al. 2019a. A Modified Embedded Discrete Fracture Model to Improve the Simulation Accuracy during Early-Time Production of Multi-Stage Fractured Horizontal Well. Paper presented at the SPE/IATMI Asia Pacific Oil & Gas Conference and Exhibition, Bali, Indonesia, 29–31 October. https://doi.org/10.2118/196263-MS.

Rao, X., Cheng, L., Cao, R. et al. 2019b. An Efficient Three-Dimensional Embedded Discrete Fracture Model for Production Simulation of Multi-Stage Fractured Horizontal Well. *Eng Anal Bound Elem* **106**: 473–492. https://doi.org/10.1016/j.enganabound.2019.06.006.

Rao, X., Xin, L., He, Y. et al. 2022. Numerical Simulation of Two-Phase Heat and Mass Transfer in Fractured Reservoirs Based on Projection-Based Embedded Discrete Fracture Model (pEDFM). *J Pet Sci Eng* **208**: 109323. https://doi.org/10.1016/j.petrol.2021.109323.

Rao, X. and Liu, Y. 2022. A Numerical Modelling Method of Fractured Reservoirs with Embedded Meshes and Topological Fracture Projection Configurations. *Comput Model Eng Sci* **131** (3): 1403–1429. https://doi.org/10.32604/cmes.2022.018879.

Rashid, H., Olorode, O., and Chukwudozie, C. 2022. An Iteratively Coupled Model for Flow, Deformation, and Fracture Propagation in Fractured Unconventional Reservoirs. *J Pet Sci Eng* **214**: 110468. https://doi.org/10.1016/j.petrol.2022.110468.

Shafiei, A., B. Dusseault, M., Kosari, E. et al. 2018. Natural Fractures Characterization and In Situ Stresses Inference in a Carbonate Reservoir—An Integrated Approach. *Energies* **11** (2): 312. https://doi.org/10.3390/en11020312.

Tene, M., Bosma, S. B. M., Al Kobaisi, M. S. et al. 2017. Projection-Based Embedded Discrete Fracture Model (pEDFM). *Adv Water Resour* **105**: 205–216. https://doi.org/10.1016/j.advwatres.2017.05.009.

Waldron, J., Snyder, M. et al. 2020. *Geological Structures: A Practical Introduction*. Edmonton, Alberta, Canada: University of Alberta. https://doi.org/10.29173/oer3.

Warren, J. E. and Root, P. J. 1963. The Behavior of Naturally Fractured Reservoirs. *SPE J.* **3** (3): 245–255. https://doi.org/10.2118/426-PA.

Yu, W., Zhang, Y., Varavei, A. et al. 2019. Compositional Simulation of CO2 Huff 'n' Puff in Eagle Ford Tight Oil Reservoirs With CO2 Molecular Diffusion, Nanopore Confinement, and Complex Natural Fractures. *SPE Res Eval & Eng* **22** (2): 492–508. https://doi.org/10.2118/190325-PA.

Zoback, M. D. 2010. *Reservoir Geomechanics*. Chennai, Tamil Nadu: Cambridge University Press.

Zoback, M. D. and Kohli, A. H. 2019. *Unconventional Reservoir Geomechanics*. Chennai, Tamil Nadu: Cambridge University Press. https://doi.org/10.1017/9781316091869.

Appendix A—Governing Equations for Compositional Reservoir Simulation

Without accounting for the presence of natural fractures, the governing equations for the mass conservation of each hydrocarbon component (i in the liquid (l) and vapor (v) phases are given as

$$\frac{\partial}{\partial t}\left(\phi[\rho^l S^l X_i^l + \rho^v S^v X_i^g]\right) + \nabla\cdot(\rho^l X_i^l \vec{v}^l + \rho^v X_i^g \vec{v}^v) - (\rho^l X_i^l q^l + \rho^v X_i^g q^v)/V = 0. \quad \text{(A-1)}$$

Similarly, the mass conservation equation for water (w) in the aqueous phase is given as

$$\frac{\partial}{\partial t}(\phi\rho^w S^w) + \nabla\cdot(\rho^w \vec{v}^w) - \rho^w q^w/V = 0, \quad \text{(A-2)}$$

where ϕ, ρ^α, S^α, and q^α represent the matrix porosity, mass density, saturation, and volumetric withdrawal/injection rate of phase α, respectively. The symbols X_i^l and X_i^g represent the mass fractions of component i in the liquid and vapor phases, while $\vec{v}_l$ and $\vec{v}_v$ are the Darcy velocities for the liquid and vapor hydrocarbon phases, respectively. Note that the division of the source/sink term in Eqs. A-1 and A-2 by bulk volume V is needed for dimensional consistency.

We obtain the phase velocities in Eqs. A-1 and A-2 from Darcy's equation as follows:

$$\vec{v}^\alpha = -\mathbf{K}\frac{k^\alpha(S)}{\mu^\alpha}(\nabla p^\alpha - \rho^\alpha g\nabla z), \quad \text{(A-3)}$$

where μ^α and $\mathbf{K}$ represent the phase viscosity and absolute matrix permeability, respectively. In the natural variables composition approach (Coats 1980), which is used in this work, the primary variables are pressure, vapor, and liquid composition of all but the last component, and water saturation (p, x_1^l,x_1^g, x_{n-1}^l, x_{n-1}^g, and S_w), respectively. The auxiliary thermodynamic equations and constraints needed for compositional simulation are summarized as follows:

$$f_i^g(p,T,y_1,\ldots,y_n) - f_i^l(p,T,x_1,\ldots,x_n) = 0, \quad \text{for } i\in 1,\ldots,n_c, \quad \text{(A-4)}$$

$$z_i - Lx_i - (1-L)y_i = 0, \quad \text{for } i\in 1,\ldots,n_c, \quad \text{(A-5)}$$

$$\textstyle\sum_{i=1}^{n_c} x_i = 1, \quad \text{for } i\in 1,\ldots,n_c, \quad \text{(A-6)}$$

$$\textstyle\sum_{i=1}^{n_c} y_i = 1, \quad \text{for } i\in 1,\ldots,n_c, \quad \text{(A-7)}$$

$$S^w + S^l + S^v = 1.0. \quad \text{(A-8)}$$

In these equations, f_i^g and f_i^l are the fugacities of each component in the gas and liquid phases, respectively. Eq. A-4 ensures that the fugacity of each component in the vapor phase is equal to that of the same component in the liquid phase (which is required at chemical equilibrium), Eq. A-5 ensures that the sum of the number of moles of each component in the liquid and gas phases is equal to its corresponding overall composition, while Eqs. A-6 through A-8 ensure that all mole fractions and saturations sum up to one.

We use the Peng-Robinson equation of state (Peng and Robinson 1976) to compute the fugacities and phase compressibility factors (Z^g and Z^l). Firoozabadi (2015) provides more details on the equation of state, flash procedure, and the equations to compute the fugacities and compressibility factors. To solve the continuous equations in Eqs. A-1 and A-2 numerically, we first perform a temporal discretization using the backward Euler scheme as follows:

$$\frac{1}{\Delta t}\left[\left(\phi\rho^l S^l X_i^l + \phi\rho^v S^v X_i^g\right)^{n+1} - \left(\phi\rho^l S^l X_i^l + \phi\rho^v S^v X_i^g\right)^n\right] + \nabla\cdot(\rho^l X_i^l \vec{v}^l + \rho^v X_i^g \vec{v}^v) - (\rho^l X_i^l q^l + \rho^v X_i^g q^v)/V = R_i, \tag{A-9}$$

$$\frac{1}{\Delta t}\left[\left(\phi\rho^w S^w\right)^{n+1} - \left(\phi\rho^w S^w\right)^n\right] + \nabla\cdot(\rho^w \vec{v}^w) - \rho^w q^w/V = R^w. \tag{A-10}$$

In the above equations, $n+1$ represents the current timestep, while n represents the previous timestep. Note that all other terms without these superscripts are evaluated at the current timestep. We then discretize the space flux terms using the finite volume method with two-point flux approximation (TPFA). The TPFA method involves integrating Eqs. A-9 and A-10 over a control volume, after which the divergence theorem is applied. In this work, we use the discrete divergence (div) and gradient (grad) operators, which are discussed in the MATLAB reservoir simulation book (Lie 2019) and implemented as functions in the MATLAB reservoir simulation toolbox (MRST). The resulting discretized form of Eqs A-9 and A-10 can be written as

$$\frac{V}{\Delta t}\left[\left(\phi\rho^l S^l X_i^l + \phi\rho^v S^v X_i^g\right)^{n+1} - \left(\phi\rho^l S^l X_i^l + \phi\rho^v S^v X_i^g\right)^n\right] + \mathrm{div}(\rho^l X_i^l \vec{v}^l + \rho^v X_i^g \vec{v}^v)^{n+1} - (\rho^l X_i^l q^l + \rho^v X_i^g q^v)^{n+1} = R_i^{n+1}, \tag{A-11}$$

and

$$\frac{V}{\Delta t}\left[\left(\phi\rho^w S^w\right)^{n+1} - \left(\phi\rho^w S^w\right)^n\right] + \mathrm{div}(\rho^w \vec{v}^w)^{n+1} - (\rho^w q^w)^{n+1} = R_w^{n+1}, \tag{A-12}$$

where

$$\vec{v}^\alpha = -T_{ik}\lambda_\alpha^{n+1}\left[\mathrm{grad}(p_\alpha^{n+1}) - \rho_\alpha^{n+1} g\,\mathrm{grad}(z)\right], \tag{A-13}$$

$$T_{ik} = \left[T_{i,k}^{-1} + T_{k,i}^{-1}\right]^{-1}, \tag{A-14}$$

and

$$T_{i,k} = A_{i,k}\mathbf{K}_i \frac{\vec{c}_{i,k}\cdot\vec{n}_{i,k}}{|\vec{c}_{i,k}|^2}. \tag{A-15}$$

Here, V and $A_{i,k}$ refer to the cell volumes and face areas, respectively. The symbol $\vec{n}_{i,k}$ is the unit normal in the direction from the centroid of cell i toward the face between cells i and k, while $\vec{c}_{i,k}$ is the vector from the cell centroid to the face centroid. Additionally, T_{ik} is face transmissibility, while $T_{i,k}$ is the contribution of a cell to face transmissibility. This transmissibility ($T_{i,k}$) is a half-transmissibility because a pair of cells contribute to the transmissibility of each face in the TPFA formulation. Note that the temporal and spatial discretizations of the continuous partial differential equations lead to a mass imbalance, which is represented by the residual (R) in Eqs. A-9 through A-12. The Newton-Raphson method involves applying the Taylor expansion to the residual at the current timestep and current Newtonian iteration to obtain

$$\frac{\partial R^{k+1}}{\partial X}\Delta X = -R^{k+1}(X), \tag{A-16}$$

where X denotes the primary variables. The matrix that contains the partial derivatives of the residuals with respect to each of these primary variables ($\frac{\partial R^{k+1}}{\partial X}$) is referred to as the Jacobian matrix. The setup of this matrix is facilitated using automatic differentiation in MRST, and more details on the solution of the system of equations for compositional flow are provided in Møyner and Tchelepi (2017).

Appendix B—Compositional Fluid Data

Tables B-1 and B-2 provide compositional data inputs for a representative Eagle Ford shale oil reservoir.

Components	Components Mole Fraction	Critical Pressure (atm)	Critical Temperature (K)	Critical Volume (L/mol)	Molar Weight (g/gmol)	Acentric Factor	Parachor Coefficient
CO_2	0.01183	72.80	304.20	0.0940	44.01	0.225	78.01
N_2	0.00161	33.50	126.20	0.0895	28.01	0.040	41.0
C_1	0.11541	45.40	190.6	0.0990	16.04	0.008	77.0
C_2–C_5	0.26438	36.5	274.74	0.2293	52.02	0.1723	171.07
C_6–C_{10}	0.38089	25.08	438.68	0.3943	103.01	0.2839	297.42
C_{11+}	0.22588	17.55	740.29	0.8870	267.15	0.6716	661.45

Table B-1—Compositional data for Eagle Ford shale formation, culled from Yu et al. (2019).

Components	CO_2	N_2	C_1	C_2–C_5	C_6–C_{10}	C_{11+}
CO_2	0	0.02	0.1030	0.1299	0.15	0.15
N_2	0.02	0	0.031	0.082	0.12	0.12
C_1	0.1030	0.031	0	0.0174	0.0462	0.111
C_2-C_5	0.1299	0.082	0.0174	0	0.0073	0.0444
C_6–C_{10}	0.15	0.12	0.0462	0.0073	0	0.0162
C_{11+}	0.15	0.12	0.111	0.0444	0.0162	0

Table B-2—Binary interaction coefficients for Eagle Ford shale formation, culled from Yu et al. (2019).