

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/349508463>

History Matching of Naturally Fractured Reservoirs Using a Deep Sparse Autoencoder

Article in SPE Journal · February 2021

DOI: 10.2118/205340-PA

CITATIONS
89

READS
405

9 authors, including:



Kai Zhang
China University of Petroleum
196 PUBLICATIONS 2,334 CITATIONS

[SEE PROFILE](#)



Jinding Zhang
China University of Petroleum
3 PUBLICATIONS 95 CITATIONS

[SEE PROFILE](#)



Xiaopeng Ma
Xi'an Shiyou University
23 PUBLICATIONS 339 CITATIONS

[SEE PROFILE](#)



Chuanjin Yao
China University of Petroleum
74 PUBLICATIONS 1,147 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Production optimization [View project](#)



History Matching [View project](#)

History Matching of Naturally Fractured Reservoirs Using a Deep Sparse Autoencoder

Kai Zhang*, Jinding Zhang, Xiaopeng Ma, Chuanjin Yao, Liming Zhang, Yongfei Yang, Jian Wang, and Jun Yao, China University of Petroleum; and Hui Zhao, Yangtze University

Summary

Although researchers have applied many methods to history matching, such as Monte Carlo methods, ensemble-based methods, and optimization algorithms, history matching fractured reservoirs is still challenging. The key challenges are effectively representing the fracture network and coping with large amounts of reservoir-model parameters. With increasing numbers of fractures, the dimension becomes larger, resulting in heavy computational work in the inversion of fractures. This paper proposes a new characterization method for the multiscale fracture network, and a powerful dimensionality-reduction method by means of an autoencoder for model parameters. The characterization method of the fracture network is dependent on the length, orientation, and position of fractures, including large-scale and small-scale fractures. To significantly reduce the dimension of parameters, the deep sparse autoencoder (DSAE) transforms the input to the low-dimensional latent variables through encoding and decoding. Integrated with the greedy layer-wise algorithm, we set up a DSAE and then take the latent variables as optimization variables. The performance of the DSAE with fewer activating nodes is excellent because it reduces the redundant information of the input and avoids overfitting. Then, we adopt the ensemble smoother (ES) with multiple data assimilation (ES-MDA) to solve this minimization problem. We test our proposed method in three synthetic reservoir history-matching problems, compared with the no-dimensionality-reduction method and the principal-component analysis (PCA). The numerical results show that the characterization method integrated with the DSAE could simplify the fracture network, preserve the distribution of fractures during the update, and improve the quality of history matching naturally fractured reservoirs.

Introduction

The complex and irregular fractures in reservoirs play a role in controlling fluid flow and accounting for production behavior. It is of paramount importance to identify the characteristics of fracture distribution and then predict future reservoir performance. According to the core image, borehole logs, microseismic monitoring, and so forth, we might not have an accurate estimation of the fracture distribution. History matching, as a method to solve inverse problems, could help to properly assess the parameters of fractures by using available production data and a forward simulator. However, there are still some challenges in history matching fractured reservoirs. Because of the complex structure of the fracture network, it is difficult to choose what parameters to represent fractures and obtain an accurate and fast simulation. Moreover, solving the inverse problem usually needs iterations of model parameters, resulting in numerous and time-consuming simulations. Therefore, it is important to set up an efficient characterization method for fractures aiming at reducing computational cost in history matching.

Building an exact model to simulate the fluid flow in fractures is fundamental for history matching. Nevertheless, directly describing the motion of the fluid is difficult using conventional mathematical methods because of the irregular structure of fractures. Thus, simplifying the geological models of fractured reservoirs is necessary. Traditionally, the dual-porosity model is applied to the simulation of fractured reservoirs, which regards the matrix and fractures as a homogeneous and orthogonal system (Warren and Root 1963). The uniform fractures are parallel to the principal axes, which can be set with different fracture spacing and width for anisotropy. However, the fluid only flows in fractures, and the matrix stores the fluid. Considering gravity effects, the dual-porosity/dual-permeability model allows the fluid to flow between the matrix blocks (Fung and Collins 1991). Although the simulation using dual-porosity methods is fairly quick, it is not suitable for modeling large-scale fractures or multiscale fracture networks. The discrete-fracture network (DFN), using line elements for fractures, considers the fracture/fracture, matrix/matrix, and matrix/fracture connections for the 2D or 3D system (Karimi-Fard et al. 2003). Zhang et al. (2018) applied the DFN to simulate hydraulic fractures of tight reservoirs and predict parameters by history matching. However, the DFN needs unstructured grids to describe fractures, which is almost impossible for complex fracture networks. The embedded discrete-fracture model (EDFM) represents fractures with structured grids, reducing the difficulty of simulation and computational cost (Li and Lee 2008; Moinfar 2013; Shah et al. 2016). Xia et al. (2018) combined EDFM with the extended-finite-element-method (XFEM) to simulate fluid flow in fractures. The results of numerical examples demonstrate the robustness of the method. For the actual naturally fractured reservoirs, it is obvious that the dual-porosity approaches and the DFN are unsuitable for simulating fractures. By contrast, the EDFM is an alternative way to simulate the complex and multiscale fracture network because of its computational efficiency.

Besides the simulation of fractures, optimization algorithms for solving the objective function are also an essential part of history matching. In recent decades, many methods have been developed to avoid getting into the local optimum and to speed up history matching, such as the Markov chain Monte Carlo (MCMC) methods, ensemble-based methods, and evolutionary algorithms. The MCMC is a sequence-sampling method to estimate the posterior probability distribution using a proposal distribution (Oliver et al. 1997; Robert and Casella 2013). The MCMC usually needs simulations of all samples, which is time consuming and expensive. The high rejection rate of the MCMC, however, increases the computational cost. The two-stage MCMC was applied to increase the acceptance rate and reduce the computational cost (Ma et al. 2008). In the first stage, they use the streamline-derived sensitivities to estimate some samples and change the proposed distribution. In the second stage, they run the simulations of the samples that pass the first stage. The two-stage MCMC is more applicable for history matching compared with the traditional MCMC, despite losing some accuracy. Another

*Corresponding author; email: zhangkai@upc.edu.cn

Copyright © 2021 Society of Petroleum Engineers

Original SPE manuscript received for review 5 August 2020. Revised manuscript received for review 10 December 2020. Paper (SPE 205340) peer approved 21 December 2020.

alternative way is to use the ensemble-based approaches, such as the ensemble Kalman filter, the ES, and ES-MDA. The ensemble Kalman filter updates the ensemble that includes the model parameters and the state parameters according to the forecast result and observed data (Evensen 1994). However, because of the change of the dynamic system state, restarting the recurrent simulation in the data-assimilation process is necessary, which might prevent its application. Only considering model parameters, the ES assimilates the data simultaneously through a global update (Evensen and Van Leeuwen 2000). The disadvantage of the ES is that it might not give acceptable data matches. The ES-MDA assimilates the data multiple times sequentially to find the optimal solution (Emerick and Reynolds 2012). The ES-MDA can provide better data matches than the ensemble Kalman filter and the ES while reducing simulation run time. In addition to these methods, the evolution algorithms, such as the genetic algorithm, the differential evolution algorithm, and particle-swarm optimization, have received more attention in history matching. Through the different updating strategies, evolutionary algorithms can solve high-dimensional problems. Nevertheless, the computational cost is still expensive with a large number of iterations. Combined with data-driven and multiobjective methods, the optimization algorithms will have stronger applicability and robustness in the future (Zhang et al. 2019a; Zhao et al. 2020; Ma et al. 2020).

To reduce the computational cost, we can reduce the number of parameters through the characterization methods for fractures. Ping and Zhang (2013) proposed a vector-based level-set method for the inversion of fractures, which represents fractures with the level-set function, the fracture length, and the fracture angle. However, setting the nodes of fractures is difficult and can lead to more parameters. Lu and Zhang (2015) proposed a Hough-transform-based parameterization to history match fractured reservoirs. They defined each fracture using its angle, algebraic distance in facies map, and length and axis displacement on the Hough space. It is an efficient method to estimate the probabilities of fractures. Zhang et al. (2017) tested different combinations of parameters for the inversion of fractures. The results show that using the midpoint coordinate, azimuth, and length to represent fractures increases stability in history matching. Zhang et al. (2019a) used the fractal DFN to history match the complex fracture network. Ma et al. (2020) set up a multiscale parameterization method for fractured reservoirs, in which the major fractures are defined by length, orientation, and position and the minor fractures follow the fractal-scaling law. Through setting the parameters of the minor fractures, this method can generate a complex and multiscale fracture network. However, the parameters of the major fractures and the minor fractures are different, decreasing flexibility for the inversion of fractures.

Dimensionality-reduction methods can be used for solving a high-dimensional problem. Sarma et al. (2007) applied kernel PCA to model permeability fields characterized by multipoint geostatistics. The kernel PCA can reproduce the complex geology with high-order statistics of random fields. Chen et al. (2014) integrated cumulative-distribution-function-based PCA with streamlined information for history matching channelized reservoirs. It can reconstruct the realizations of non-Gaussian fields, and the quality of the reconstructed model is quite satisfactory. Zhang et al. (2014) proposed a multiparameter optimal method to history matching using singular-value decomposition. The singular-value-decomposition method simplifies the calculation of the covariance matrix and reduces the dimension. Recently, the artificial neural network has become more popular in many fields. An autoencoder can learn efficient data coding, which reduces the dimension and reconstructs the data by encoding and decoding. With the nonlinear encoder and decoder, the autoencoder can learn a more powerful generalization than the PCA (Goodfellow et al. 2016). Deep learning is applied to history matching of facies models integrated with ES-MDA (Canchumuni et al. 2017). The results show that the deep autoencoder can reconstruct facies models according to initial learning. Liu and Grana (2019) proposed a deep convolutional autoencoder to represent seismic data and the model parameters. It performs the data assimilation in the low-dimensional data space, which is helpful to avoid the ensemble collapse and reduce the computational cost. The parameters involved in fracture modeling are different from the parameters of the facies models, so it is uncertain that the autoencoder can reconstruct the complex fracture network with low-dimensional data.

In this work, we propose a new characterization method and a method to reduce dimensionality for history matching of naturally fractured reservoirs. The forward simulator is modeled after the EDFM given its computational efficiency. The fracture network can be represented with length, orientation, and position, including large-scale fractures and small-scale fractures. According to the range of parameters, we randomly generate samples of fracture networks. With the samples, we train a DSAE through greedy layer-wise pre-training and fine tuning. This autoencoder can generate parameters for complex fracture networks using low-dimension latent variables with acceptable accuracy. We use the ES-MDA to solve the objective function of history matching. Finally, we present three cases to illustrate the efficiency of our approach, compared with PCA and the no-dimensionality-reduction method. The first case with five large-scale fractures is to demonstrate the feasibility of our approach and to compare the history-match results using three methods for a simple fracture network. The second case has 50 vertical large-scale fractures, which is to compare the performance of preserving the distribution of fractures using three methods during the update. The third case with 10 large-scale fractures and 300 small-scale fractures is to test the performance of the complex and high-dimensional problem using DSAE.

The remainder of this paper is organized as follows. We first pose the characterization method for the 2D DFN. Then we introduce the process of setting up a DSAE and generating samples using the hidden layer codes. Next, we present the objective function and the ES-MDA algorithm. After that, we design three cases to demonstrate the efficiency of our approach. Finally, our discussion and conclusions are presented.

Methods

2D DFN Model. It is a challenge to represent the fracture network because of its irregular and multiscale structure. An alternative way is representing the fracture as a 1D element, a line, characterized by length L , orientation θ , and the midpoint coordinate (x_0, y_0) . Fig. 1a shows the fracture network represented by lines and Fig. 1b shows the parameters of one fracture.

Running the EDFM simulator needs parameters such as the endpoints, the aperture, and so forth. The endpoints of the fracture, (x_1, y_1) and (x_2, y_2) , can be calculated using Eqs. 1 through 4.

$$x_1 = x_0 - \frac{L}{2} \cos\left(\frac{\theta}{180} \cdot \pi\right), \quad \dots \dots \dots \quad (1)$$

$$y_1 = y_0 - \frac{L}{2} \sin\left(\frac{\theta}{180} \cdot \pi\right), \quad \dots \dots \dots \quad (2)$$

$$x_2 = x_0 + \frac{L}{2} \cos\left(\frac{\theta}{180} \cdot \pi\right), \quad \dots \dots \dots \quad (3)$$

$$y_2 = y_0 + \frac{L}{2} \sin\left(\frac{\theta}{180} \cdot \pi\right). \quad \dots \dots \dots \quad (4)$$

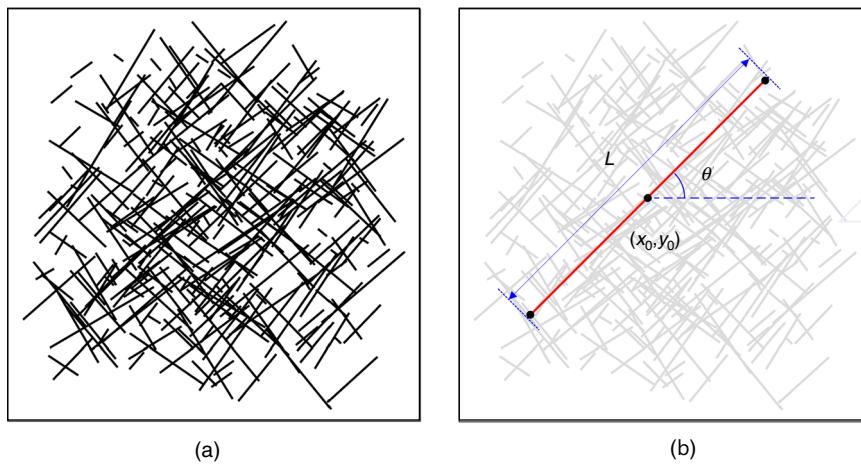


Fig. 1—(a) The fracture network and (b) the parameters for one fracture.

According to the cubic law (Witherspoon et al. 1980), we can calculate the permeability of the fracture blocks with the aperture. The aperture of the fracture is related to the length. Eqs. 5 and 6 show the relation of the permeability, the aperture, and the length.

$$k_f = \frac{h^2}{12}, \quad \dots \dots \dots \quad (5)$$

$$h = c_a \cdot L, \quad \dots \dots \dots \quad (6)$$

where k_f denotes the permeability of the fractures; h represents the aperture; and c_a is a constant. If we can obtain the aperture information of fractures from field data, it is unnecessary to calculate the aperture using the length of fractures. However, sometimes the relationship between the fracture length and the fracture aperture is unavailable, so we can use the empirical value for c_a , which can be set between 10^{-1} and 10^{-3} (Klimczak et al. 2010).

Based on our previous work (Ma et al. 2020), the multiscale fracture network consists of large-scale fractures and small-scale fractures, and there is a linear relationship between the minimum length of large-scale fractures and the maximum length of small-scale fractures. In this work, we propose a method for different scale fractures, which represents fractures using the same parameters. The different scales of fractures are defined similarly. We assume that there is a linear relationship between the minimum length of large-scale fractures and the expectation value of the length distribution of small-scale fractures.

We can obtain the parameter range of large-scale fractures using measurements such as petrophysical interpretation, in-situ-stress data analysis, log data, and so forth. Then samples can be generated randomly within the parameter range. Eqs. 7 through 10 show the process of calculating the parameters of large-scale fractures.

$$L_{li} = L_{l\min} + \text{rand}_i \times (L_{l\max} - L_{l\min}), \quad \dots \dots \dots \quad (7)$$

$$\theta_{li} = \theta_{l\min} + \text{rand}_i \times (\theta_{l\max} - \theta_{l\min}), \quad \dots \dots \dots \quad (8)$$

$$x_{0li} = x_{0l\min} + \text{rand}_i \times (x_{0l\max} - x_{0l\min}), \quad \dots \dots \dots \quad (9)$$

$$y_{0li} = y_{0l\min} + \text{rand}_i \times (y_{0l\max} - y_{0l\min}), \quad \dots \dots \dots \quad (10)$$

where the subscript l denotes the large-scale fractures; the subscript i denotes the i th sample; $L_{l\min}$ and $L_{l\max}$ are the range of the length, $\theta_{l\min}$ and $\theta_{l\max}$ denote the range of the orientation; $x_{0l\min}$, $x_{0l\max}$, $y_{0l\min}$, and $y_{0l\max}$ are the range of the midpoint coordinates; rand_i represents a random number between zero and unity; and L_{li} , θ_{li} , x_{0li} , and y_{0li} denote the parameters of the i th sample.

For the small-scale fractures, we also represent fractures using the length, the orientation, and the midpoint coordinates. However, the generation of the samples is different. We divide the region of the fracture network into four regions. In this work, the fracture number denotes the number of small-scale fractures in a certain region. Then, the proportion of the fracture number can be defined as the ratio of the fracture number of a subregion to the total fracture number of the whole region. We can obtain the proportion of the fracture number of each region using four random numbers. The proportion of the fracture number can be generated using Eq. 11.

$$\mathcal{D}_j = \frac{r_j}{4}, \quad \sum_{j=1}^4 r_j, \quad \dots \dots \dots \quad (11)$$

where \mathcal{D}_j denotes the proportion of the fracture number for the j th region and r_j represents a random number between 0.1 and 1. Given the total number of small-scale fractures, we can calculate the number of small-scale fractures of each region using \mathcal{D}_j . The fracture density is defined as the surface area of fracture per unit volume of rock. The surface area of fractures can be calculated using the length and aperture of each fracture. Thus, the fracture density is closely related to the proportion of the fracture number \mathcal{D}_j .

We use a log-normal distribution to generate the length (Zadheesh et al. 2014), a normal distribution to generate the orientation, and a uniform distribution to generate the midpoint coordinates. The length of the i th sample can be obtained using Eq. 12.

$$L_{si} = f_1(\mu_{Li}, \sigma_{Li}), \quad \dots \dots \dots \quad (12)$$

where the subscript i denotes the i th sample of the small-scale fractures; f_1 is the function of the log-normal distribution; μ_{L_i} and σ_{L_i} denote parameters of the log-normal distribution of the i th sample. We assume that the range of μ_{L_i} and σ_{L_i} is known. The μ_{L_i} and σ_{L_i} can be obtained in Eqs. 13 and 14.

$$\mu_{L_i} = \mu_{L_{\min}} + \text{rand}_i \times (\mu_{L_{\max}} - \mu_{L_{\min}}), \quad \dots \dots \dots \quad (13)$$

$$\sigma_{L_i} = \sigma_{L_{\min}} + \text{rand}_i \times (\sigma_{L_{\max}} - \sigma_{L_{\min}}), \quad \dots \dots \dots \quad (14)$$

where $\mu_{L_{\max}}$, $\mu_{L_{\min}}$, $\sigma_{L_{\max}}$, and $\sigma_{L_{\min}}$ denote the parameter range. The expectation of length of small-scale fractures $E[L_{si}] = \alpha_i L_{l_{\min}}$, where $E[\cdot]$ denotes the expectation, α_i is a scale factor, and $L_{l_{\min}}$ denotes the minimum value of the length of large-scale fractures.

Similarly, the orientation of the small-scale fractures can be generated in Eqs. 15 through 17.

$$\theta_{si} = f_2(\mu_{\theta_i}, \sigma_{\theta_i}), \quad \dots \dots \dots \quad (15)$$

$$\mu_{\theta_i} = \mu_{\theta_{\min}} + \text{rand}_i \times (\mu_{\theta_{\max}} - \mu_{\theta_{\min}}), \quad \dots \dots \dots \quad (16)$$

$$\sigma_{\theta_i} = \sigma_{\theta_{\min}} + \text{rand}_i \times (\sigma_{\theta_{\max}} - \sigma_{\theta_{\min}}), \quad \dots \dots \dots \quad (17)$$

where f_2 is the function of the normal distribution; μ_{θ_i} and σ_{θ_i} denote the mean value and the standard deviation of the i th sample; $\mu_{\theta_{\max}}$, $\mu_{\theta_{\min}}$, $\sigma_{\theta_{\max}}$, and $\sigma_{\theta_{\min}}$ denote the range of the mean value and the standard deviation.

For each region, according to the uniform distribution of the midpoint coordinate, we can generate the midpoint coordinate of the i th sample, x_{0si} , y_{0si} . Through the two different methods, we obtain the samples of the large-scale and small-scale fractures, and then combine the large-scale and small-scale fractures. The parameters of the fracture network ϕ_i can be represented in Eq. 18. **Fig. 2** shows the process of generating the 2D fracture network.

$$\phi_i = [L_{li}, L_{si}, \theta_{si}, \theta_{li}, x_{0li}, x_{0si}, y_{0li}, y_{0si}]. \quad \dots \dots \dots \quad (18)$$

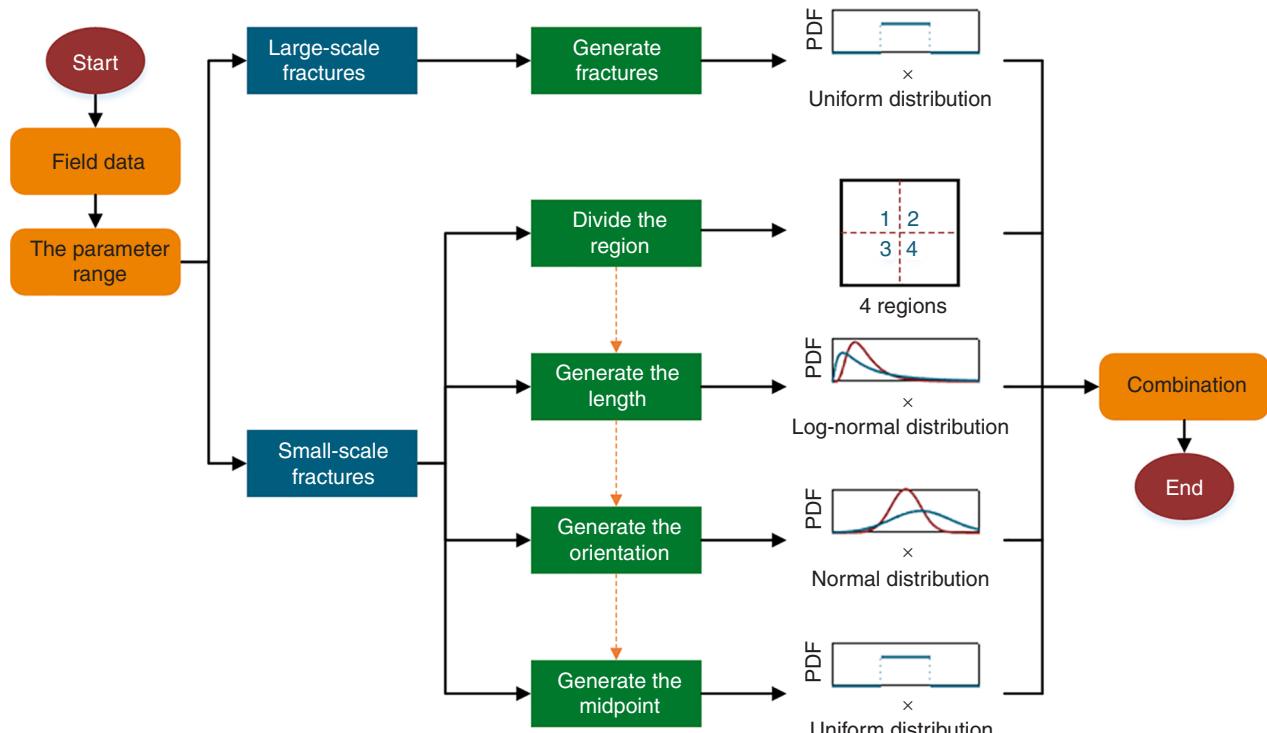


Fig. 2—The workflow of generating the 2D fracture network; PDF = probability-density function.

DSAE. With the popularity of the artificial neural network, the autoencoder gets more attention in dimensionality reduction. Through constructing a neural-network structure with an encoder and a decoder, it can copy its input and its output. The basic architecture of the autoencoder is shown in Eqs. 19 and 20.

$$\mathcal{H} = \delta^{(1)}[W^{(1)}\mathcal{X} + b^{(1)}], \quad \dots \dots \dots \quad (19)$$

where \mathcal{X} denotes the input vector; \mathcal{H} denotes the hidden-layer vector; $W^{(1)}$ and $b^{(1)}$ denote the weight matrix and the bias vector, respectively, of the encoder; and $\delta^{(1)}$ represents the activation function of the encoder.

$$\mathcal{X}' = \delta^{(2)}[W^{(2)}\mathcal{H} + b^{(2)}], \quad \dots \dots \dots \quad (20)$$

where \mathcal{X}' denotes the output vector; $W^{(2)}$ and $b^{(2)}$ denote the weight matrix and the bias vector, respectively, of the decoder; and $\delta^{(2)}$ denotes the activation function of the decoder.

In this work, we apply the hyperbolic tangent activation function to the encoder and the decoder. Initially, the weight matrix and the bias vector are assigned random values. To obtain a network with generalization capability, the number of training samples is at least 10 times the number of degrees of freedom (Menshawy 2018). Through training to find proper values of the weight and the bias, the reconstructed errors are reduced significantly. Using the sparse autoencoder (Ng 2011), the cost function represents the reconstructed errors shown in Eq. 21.

$$E = \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K (\mathcal{X}_{nk} - \mathcal{X}'_{nk})^2 + \lambda \cdot \Omega_w + \beta \cdot \Omega_s, \quad \dots \dots \dots \quad (21)$$

where E is the cost function of the autoencoder; N represents the number of the training samples, K denotes the dimension of the data; \mathcal{X}_{nk} and \mathcal{X}'_{nk} denote the input and output value, respectively, of the k -dimension in the n th sample; λ and β are the coefficients; Ω_w represents the L_2 regularization; and Ω_s represents the sparse regularization.

To improve the generation ability of the autoencoder and avoid overfitting, the L_2 regularization and the sparse regularization are applied. The L_2 regularization prevents the increasing value of weights shown in Eq. 22.

$$\Omega_w = \frac{1}{2} \sum_l^C \sum_j^N \sum_i^K [W_{ji}^{(l)}]^2, \quad \dots \dots \dots \quad (22)$$

where C denotes the number of hidden layers and $W_{ji}^{(l)}$ represents the weights of the i th dimension of the j th sample in the l th layer.

The sparse regularization can be represented in Eqs. 23 and 24.

$$\hat{\rho}_i = \frac{1}{N} \sum_{j=1}^N \mathcal{H}_i(\mathcal{X}_j), \quad \dots \dots \dots \quad (23)$$

$$\Omega_s = \sum_{i=1}^D KL(\rho || \hat{\rho}_i) = \sum_{i=1}^D \rho \log\left(\frac{\rho}{\hat{\rho}_i}\right) + (1 - \rho) \log\left(\frac{1 - \rho}{1 - \hat{\rho}_i}\right), \quad \dots \dots \dots \quad (24)$$

where $\hat{\rho}_i$ represents the average value of the hidden unit i ; N denotes the number of training examples; and ρ denotes the value that we set for the autoencoder. The Kullback-Leibler divergence KL measures the errors between the desired value $\hat{\rho}_i$ and ρ , and D denotes the number of hidden nodes.

At this point, we have set up a sparse autoencoder to reconstruct the input data. With one hidden layer, the autoencoder might not reconstruct the input data accurately with a lower dimension. One alternative choice is to expand one hidden layer to multiple hidden layers for the autoencoder. Training this deep autoencoder includes pretraining and fine tuning. Pretraining means that we train a sparse autoencoder, then use the latent variables of the hidden layer as the input data for the next autoencoder. After greedy layer-wise training, we stack all the autoencoders and then train the stacked sparse autoencoder, which is called fine tuning. **Fig. 3** shows the process of setting up a DSAE.

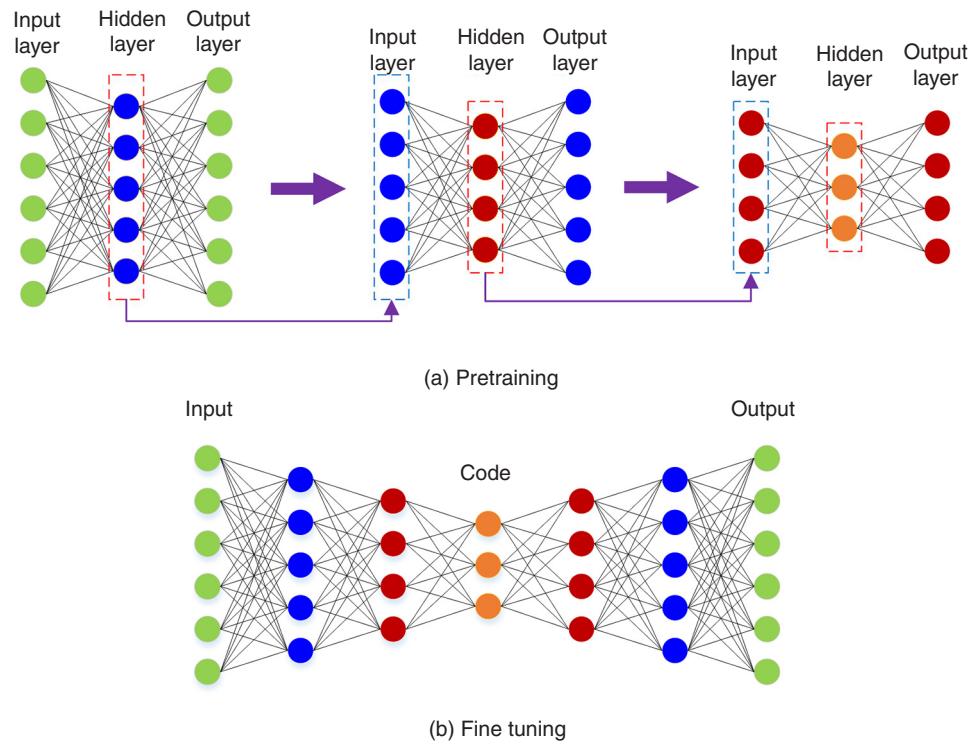


Fig. 3—The process of training a DSAE: (a) pretraining and (b) fine tuning.

Reparameterization of the Fracture Network by DSAE. The parameters of the fracture network increase linearly with the number of fractures, which brings a high-dimensional problem for inversion. Solving this high-dimensional problem usually needs more time-consuming simulations. Therefore, this paper uses the DSAE to reparametrize the fracture network and then reduce the dimension.

In terms of current detection techniques, there is great uncertainty for the fracture number. For the large-scale fracture, we consider that the fracture number is available. However, it is difficult to determine the number of small-scale fractures. Therefore, we randomly delete some fractures in small-scale fractures by mending zero in the matrix ϕ_i . Thus, the number of fractures is changing but the dimension of the problem is unchanged, which is useful to train an autoencoder. We can vary the fracture number with a random number shown in Eq. 25.

$$n_{si} = n_{smin} + \text{rand}_i \times (n_{smax} - n_{smin}), \quad \dots \dots \dots \quad (25)$$

where n_{si} denotes the number of the i th sample and n_{smin} and n_{smax} represent the minimum and the maximum number, respectively, of the small-scale fractures.

After training a DSAE, we extract the codes of the hidden layer and calculate the maximum and minimum values of the codes. We then generate several codes randomly using the uniform distribution within the range of the codes. We can obtain fracture networks by decoding these codes. Although more training samples are more helpful to the deep autoencoder, it causes more training time. The training samples could not include all possible fracture networks, so what we focus on is not more training samples but the generation of the DSAE. **Fig. 4** shows the process of generating samples with the DSAE.

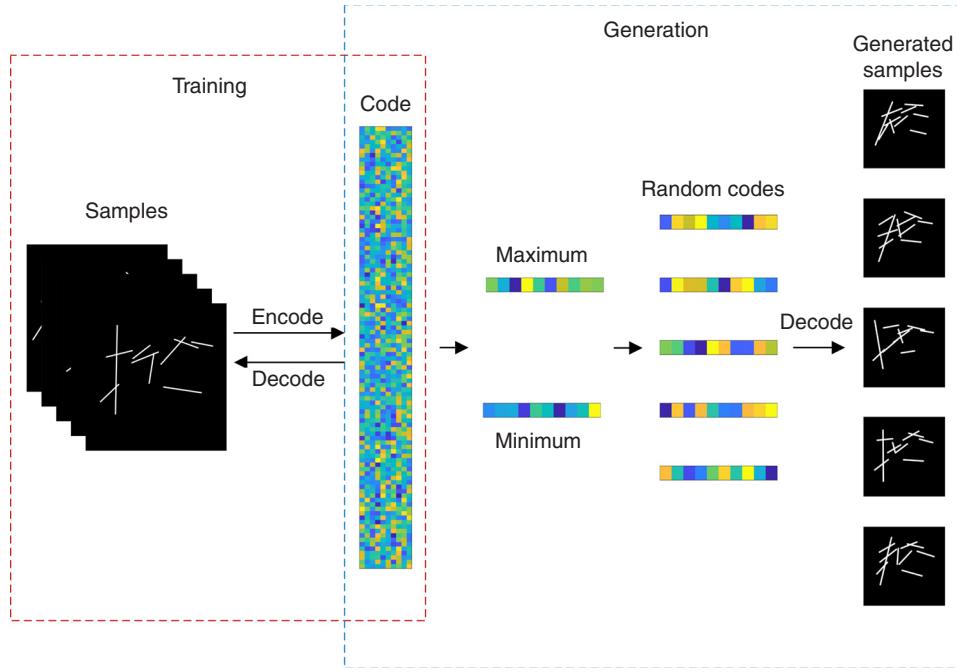


Fig. 4—The process of generating samples using the random codes from the DSAE.

Integrated History-Matching Approach. History matching is a typical inversion problem, which uses the observed reservoir behavior and a simulator of reservoirs to ascertain the parameters of the true reservoir model. The objective function of history matching represents the deviation between the prediction data of simulation and the observed data. According to Bayesian theory, the posterior PDF for the parameters m conditional to the observed production data d_{obs} is given by

$$p(m|d_{\text{obs}}) \propto p(d_{\text{obs}}|m)p(m), \quad \dots \dots \dots \quad (26)$$

where $p(m|d_{\text{obs}})$ denotes the posterior PDF; $p(d_{\text{obs}}|m)$ denotes the likelihood function; and $p(m)$ denotes the prior PDF.

For history matching, there are many available methods, which can mainly fall into three categories: MCMC methods (Oliver et al. 1997; Robert and Casella 2013), optimization algorithms (Romero and Carter 2001; Schulze-Riegert et al. 2002), and ensemble-based methods (Evensen 1994; Emerick and Reynolds 2013a). The number of parameters needed to be calibrated becomes numerous as the number of fractures increases. Ensemble-based methods can estimate a large number of model variables and assimilate different types of data (Oliver and Chen 2011). Thus, we adopt the ES-MDA (Emerick and Reynolds 2013a) for history matching in this work.

ES-MDA. As a modification of the ES, the ES-MDA method applies multiple assimilations at every updated step. The formula is shown in Eq. 27.

$$m_j^{n,a} = m_j^{n,f} + C_{MD}^{n,f} (C_{DD}^{n,f} + C_D^n)^{-1} (d_{uc,j}^n - d_j^{n,f}), \quad \dots \dots \dots \quad (27)$$

where $j = 1, 2, 3, \dots, N_e$ and N_e denotes the ensemble members; $n = 1, 2, 3, \dots, N_a$ and N_a represents the number of the data assimilation; f represents the forecast and a represents the analysis; m denotes the vector of model parameters, $\in R^{N_m}$; $C_{MD}^{n,f}$ is the cross-covariance matrix between the model parameters and predicted data, $\in R^{N_m \times N_d}$; $C_{DD}^{n,f}$ is the auto-covariance matrix of predicted data, $\in R^{N_d \times N_d}$; C_D^n denotes the measurement errors of observation, $\in R^{N_d \times N_d}$; $d_j^{n,f}$ denotes the predicted data; and $d_{uc,j}^n$ denotes the observed data with a noise. Eq. 28 presents the form of $d_{uc,j}^n$.

$$d_{uc,j}^n = d_{\text{obs}} + \sqrt{\alpha_i} C_D^{1/2} Z_d, \quad \dots \dots \dots \quad (28)$$

where Z_d denotes the random values that follow the normal distribution; α_i is a coefficient; and C_D is replaced by $\alpha_i C_D$ at every data assimilation step. For more details, see Emerick and Reynolds (2013a).

The coefficient α_i must be selected such that $\sum_{i=1}^{N_a} \frac{1}{\alpha_i} = 1$. The simplest choice is to set α_i to the value of N_a . However, the results indicate that the coefficients α_i with a decreasing order can improve the performance of ES-MDA. In this work, we use a decreasing order method for α_i . For four data assimilations, the coefficients α_i can be set to [9.333, 7.0, 4.0, 2.0]. For eight data assimilations, the coefficients α_i can be set to [20.719, 19.0, 17.0, 16.0, 15.0, 9.0, 5.0, 2.5]. For 10 data assimilations, the coefficients α_i can be set to [57.017, 35.0, 25.0, 20.0, 18.0, 15.0, 12.0, 8.0, 5.0, 3.0]. For more details, see Emerick and Reynolds (2013b).

ES-MDA with DSAE. The ES-MDA is used to solve the objective function, where the optimization variables are the hidden-layer codes of the DSAE. Fig. 5 shows the workflow of the ES-MDA with the DSAE for history matching. The workflow starts with the range of parameters from the field data. Then, we generate the training samples of the fracture network by our multiscale method. Next, we use the generated training samples with the deep autoencoder to reduce dimensionality. After that, we can calculate the range of codes from the hidden layer and randomly generate codes. Through decoding these codes, we can obtain the parameters of fracture networks. For obtaining the simulated data, we run the EDFM simulator with the parameters of the fracture network. Next, we evaluate the objective function with the simulated data and observed data. If we meet the stopping criteria, we should stop history matching, or use ES-MDA to update the codes and then continue the preceding process.

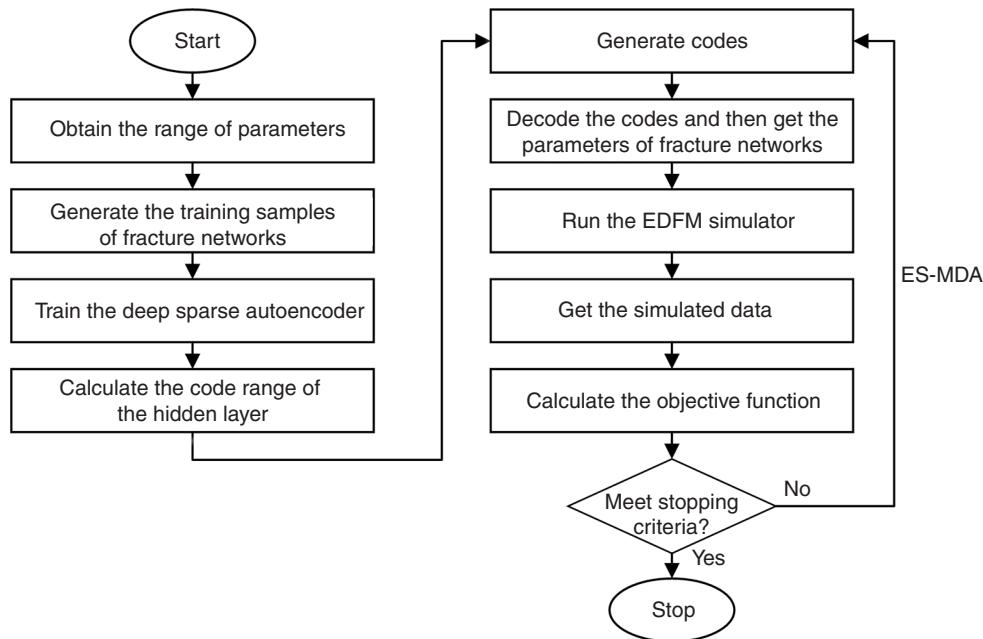


Fig. 5—The workflow of the ES-MDA with DSAE for history matching.

Case Studies

Case 1: Five Large-Scale Fractures Only. The 2D fracture network has five large-scale fractures to verify the feasibility of the DSAE for the inversion. We compare the performance using DSAE, PCA, and the no-dimensionality-reduction method. We assume that the range of parameters is known. We set the parameters of the reference model and the initial models shown in Table 1. The 2D waterflooding problem is considered. The size of the reservoir is 200×200 m, which is discretized into 40×40 gridblocks. The permeability and porosity for the matrix are considered as constants of 3 md and 0.2, respectively. For the fractures, the porosity is set to 0.3 and the permeability follows the cubic law, which is related to the aperture of fractures. Five-spot well pattern is applied. The injector located in the center injects water at a constant rate of $20 \text{ m}^3/\text{d}$. Four producers located at the corners are at a constant well bottomhole pressure (WBHP) of 10 MPa. The duration of the production is 300 days, which is divided into 50 timesteps. The observed data contain the WBHP, well water cut (WWCT), and well oil-production rate (WOPR). Assume that the errors of observed data follow the normal distribution. The mean is set to zero and the standard deviations of WBHP, WWCT, and WOPR are set to 1%, 1%, and 3% of the observed data, respectively. Fig. 6a shows the fracture network, the well pattern, and the grids. Using the structured grids, the EDFM method divides fractures into matrix grids and then adds nonneighboring connections (Moinfar 2013). The EDFM simulator can be implemented in the MATLAB Reservoir Simulation Toolbox (Lie et al. 2012). With the EDFM simulator and the parameters of fractures, we can obtain the residual oil saturation at the 300th day, shown in Fig. 6b. It is obvious that the fracture network has a great influence on fluid flow. The fluid first flows in fractures and then gradually expands to the matrix because of the difference in permeability, porosity, and other properties.

For five large-scale fractures, the number of model parameters is 20. The energy or contribution criterion of the PCA is defined as the ratio of a certain number of cumulative eigenvalues to the total eigenvalues. The energy criterion determines how many dimensions should be remained in the low-dimensional space. For more details regarding PCA, see Abdi and Williams (2010). We set the threshold value of the energy criterion as 90%. We want to choose the value of the remaining dimension as small as possible while achieving a reasonably high value of the energy. In this case, we use the PCA to reduce the dimension from 20 to 17, which can meet the requirements of the energy criterion. The dimension of the latent variables of the DSAE is also set to 17 for comparison. The number of fractures is fixed in history matching. According to the dimension, we set the number of training samples as 2,000. Empirically, we stack two autoencoders to generate the DSAE, which consists of an encoder with layers of (20)-30-17 and a symmetric decoder. Note that the number of the hidden layer might have more nodes than the input layer. The coefficients λ and β in Eq. 21 are set to 0.01 and 1, respectively. The coefficient ρ in Eq. 24 is set to 0.1. Through the pretraining and fine-tuning process, we completed the training of the

DSAE. Through encoding the input data, we can obtain the codes of the hidden layer. Then, through decoding these codes, we can obtain the reconstructed samples. In history matching, we use the latent variables as the optimization variables. During the update by ES-MDA, the variables are changed randomly. We therefore compare the original samples with the generated samples using random latent variables for both DSAE and PCA in **Fig. 7**. Note that fracture networks in Figs. 7b and 7c are not reconstructed fracture networks of fracture networks in Fig. 7a. They are not one-to-one relationships. Fracture networks in Figs. 7b and 7c are generated using random latent variables. It is clear that both DSAE and PCA can maintain diversities of fracture networks for history matching using random latent variables.

Fracture	Length		Orientation		x-Coordinate		y-Coordinate	
	True Value	Initial Range	True Value	Initial Range	True Value	Initial Range	True Value	Initial Range
1	139	130–170	29	20–70	99	90–130	71	30–80
2	123	50–130	8	0–50	99	90–130	156	130–160
3	138	130–170	122	120–160	113	50–120	118	100–150
4	71	20–80	58	50–100	66	20–80	97	50–100
5	38	10–50	76	50–100	136	100–150	124	120–160

Table 1—Parameters of the reference model and the initial range of fractures.

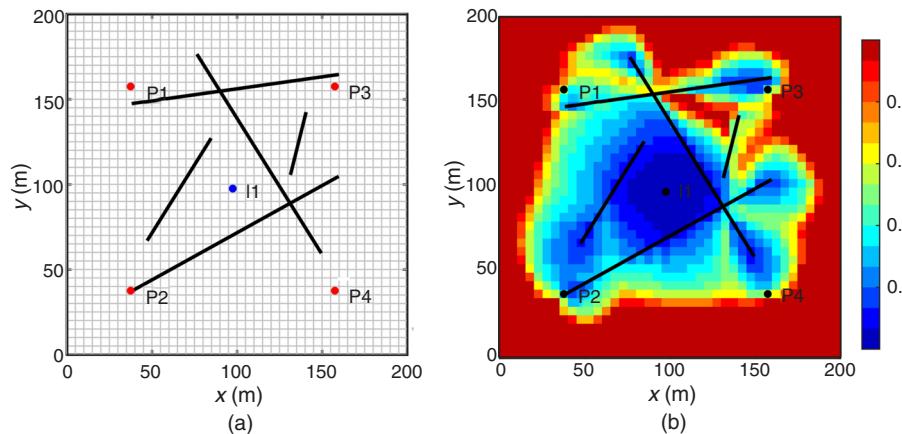


Fig. 6—The reference model for Case 1: (a) the fracture network; (b) the oil-saturation map at the 300th day.

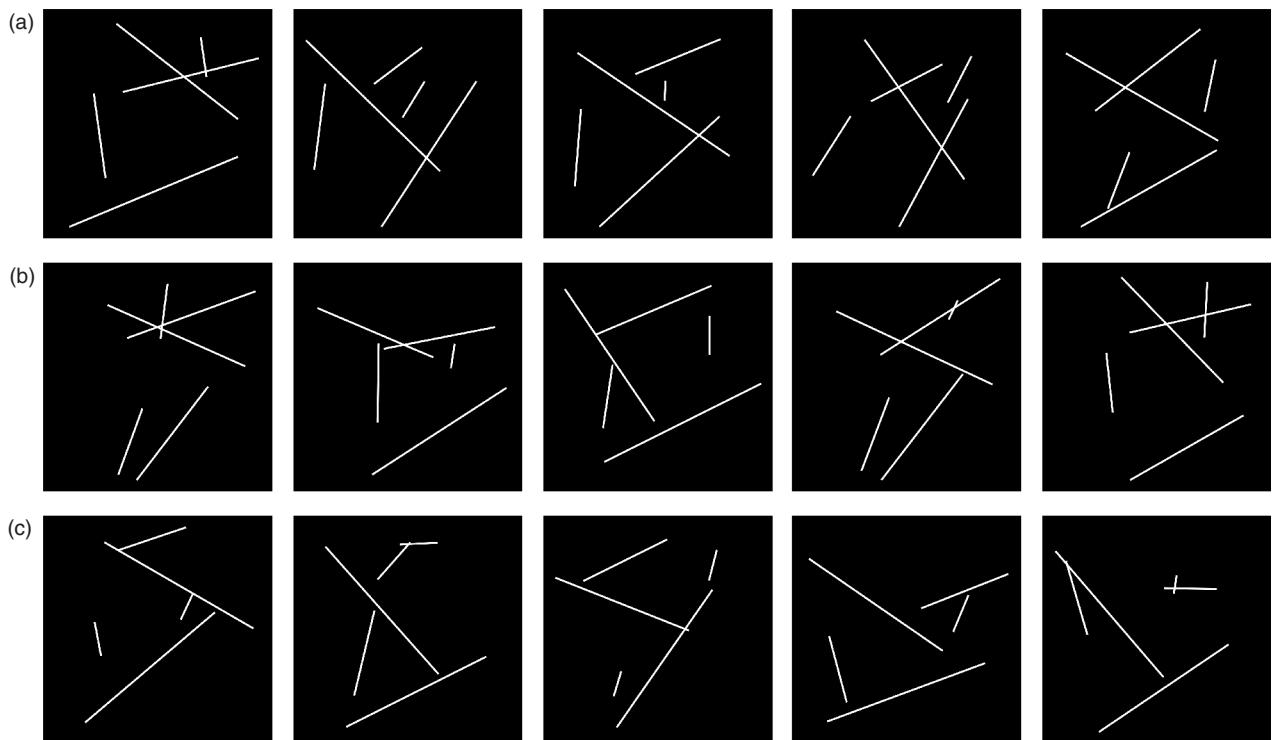


Fig. 7—Comparison of (a) original fracture networks, (b) fracture networks generated by DSAE, and (c) fracture networks generated by PCA for Case 1.

After training the DSAE, we calculated the range of the codes of the hidden layer. The ES-MDA is used to find the optimal solution of history matching. The ensemble size is set to 100. The dimension of the variables from 20 changes to 17 through dimensionality-reduction methods. Through decoding the codes, we can obtain corresponding fracture networks. Next, we run the EDFM simulator with the parameters of the fracture networks. Thus, we can obtain the observed data. The objective function is set up with the observed data and the simulated data according to the Bayesian theory. The ES-MDA evaluates the objective function and then updates the codes until reaching the maximum number of iterations. **Fig. 8** shows the convergence behavior of the objective function for three methods. The initial models are selected randomly from the prior models. We use the same initial models for three methods. We use DSAE to encode the initial models and obtain latent variables. Then, we set latent variables as initial models for DSAE. For PCA, we use the same method to set the initial models. Although we use the same initial models, the objective-function values at the first iteration for the three methods are different because of different reconstruction performances. The objective functions drop greatly in the first four iterations. After seven iterations, the objective functions using the three methods converge. The final objective-function values by three methods are nearly the same. On the whole, the behaviors of objective functions by three methods are similar.

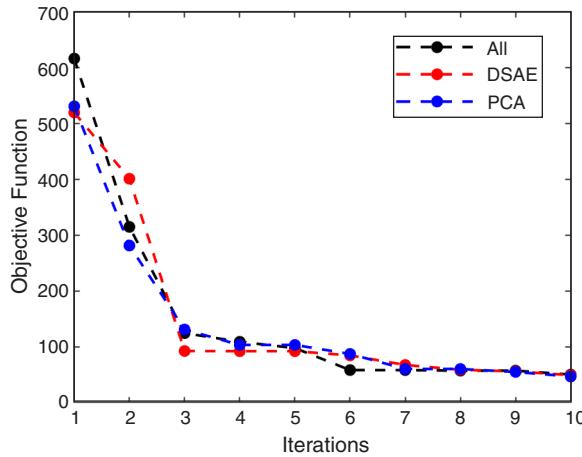


Fig. 8—Convergence behavior of the objective function for Case 1. The black line represents the method that uses ES-MDA to directly estimate all parameters; the red line represents the method that uses DSAE; the blue line represents the method that uses PCA.

Figs. 9, 10, and 11 show the observed data-matching results using DSAE and PCA. For the initial ensemble, the simulated data including WWCT and WOPR have a wide distribution range, which is far from the observed data of the reference model. After iterations, the simulated data as a whole are close to the observed data for the three methods. The data-matching results using ES-MDA directly, DSAE, and PCA are also similar.

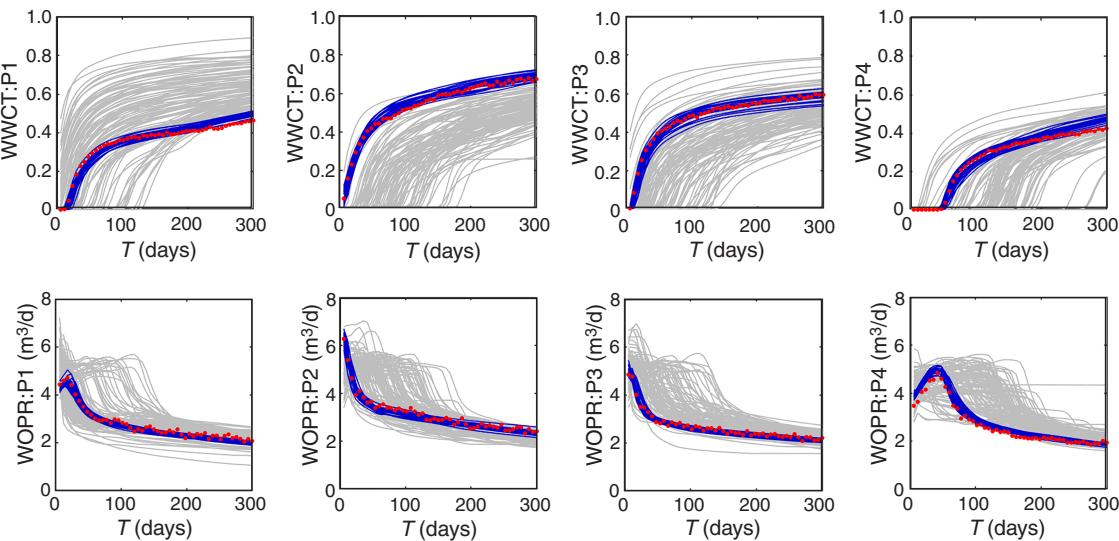


Fig. 9—History-match results using ES-MDA directly for Case 1. The first row shows the water cut (WWCT) and the second row shows the oil-production rate (WOPR). The gray represents the simulated data of the initial models and the blue represents the simulated data of the history-matched models. The red dots represent the observed data.

The structure of the fracture network can be identified accordingly. **Fig. 12** shows the probability map of the initial models and the history-matched models using ES-MDA directly, DSAE, and PCA. We selected some initial fracture networks and then overlapped them in Figs. 12a, 12b, and 12c, which have large uncertainties. Because of the different reconstruction performance between using DSAE and PCA, the initial models are not totally the same. Some history-matched models are overlapped in Figs. 12d, 12e, and 12f,

and the range of uncertainties of history-matched models is reduced. The fracture networks of the history-matched models using the three methods are also close to the reference model. Fig. 13 shows the comparison of oil-saturation maps at the 300th day. The history-match model for saturation-map plots in Fig. 13 is the model that has the lowest objective function. The first row shows the oil-saturation map of the history-matched model and the second row shows the difference of the oil-saturation map with the reference model. The differences are calculated using the subtraction of the saturation value of each grid between the history-matched model and the reference model. Then, we calculate the mean-squared errors of oil-saturation maps between history-matched models and the reference model. The mean-squared errors of oil-saturation maps for using ES-MDA directly, DSAE, and PCA are 0.0445, 0.0340, and 0.0447, respectively. The oil-saturation map using DSAE is the closest to the reference model, followed by ES-MDA directly and PCA. However, the differences between the three methods are negligible. The results of Case 1 show that with history matching of the simple fracture network, it is unnecessary to use dimensionality-reduction methods such as the DSAE and PCA. Using ES-MDA directly could obtain a satisfactory history-match result.

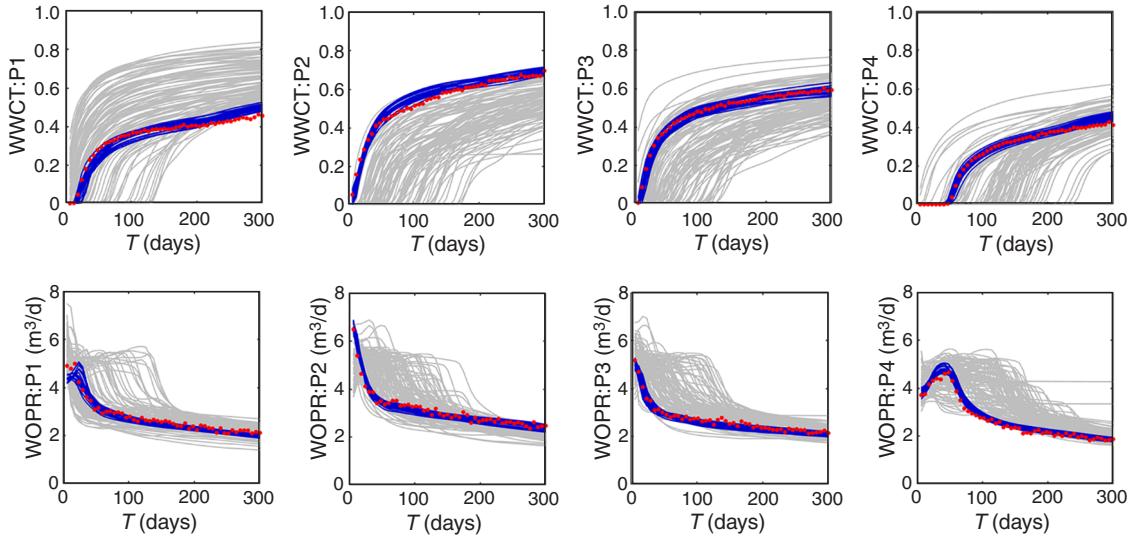


Fig. 10—History-match results using DSAE for Case 1. The first row shows the water cut (WWCT) and the second row shows the oil-production rate (WOPR). The gray represents the simulated data of the initial models and the blue represents the simulated data of the history-matched models. The red dots represent the observed data.

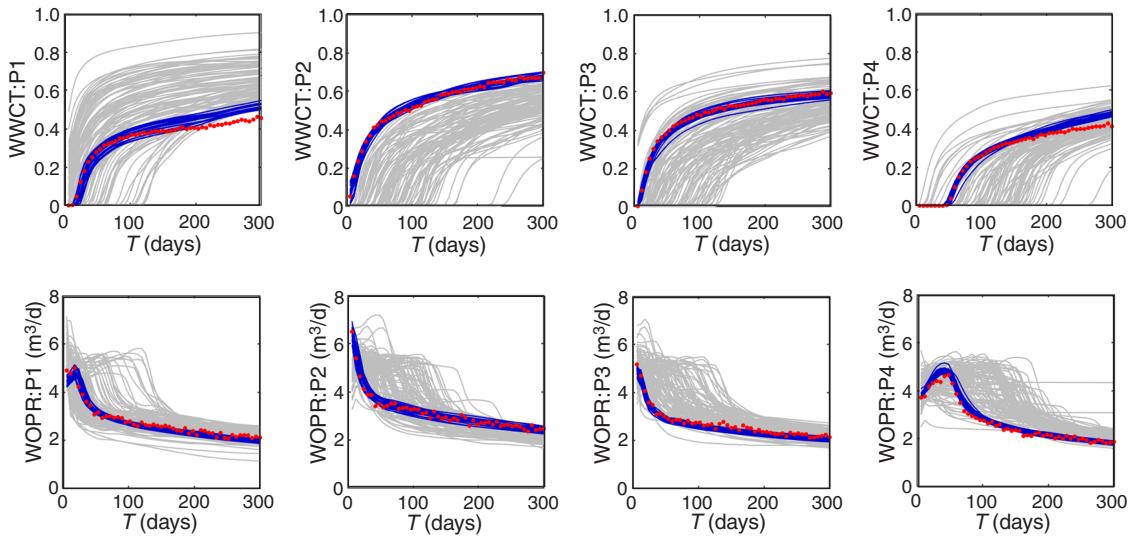


Fig. 11—History-match results using PCA for Case 1. The first row shows the water cut (WWCT) and the second row shows the oil-production rate (WOPR). The gray represents the simulated data of the initial models and the blue represents the simulated data of the history-matched models. The red dots represent the observed data.

Case 2: The Vertical Fracture-Network Model with 50 Large-Scale Fractures. In the environment where most petroleum fields occur, the stress field influenced by the upper crust is anisotropic and can be resolved into three orthogonal principal components. Cores from vertical and horizontal wells and image logs show that fractures are vertical or nearly vertical (perpendicular to bedding) (Narr et al. 2006). Thus, in this case, we designed 50 vertical large-scale fractures in this model. The length of each fracture is 150 m for the reference model, and the initial range is 100 to 200 m. The orientations of fractures that are divided into two sets follow the normal distribution. The mean and standard deviation for one set are 45 and unity, and the mean and the standard deviation of the other set are 135 and unity. Assume that the positions of fractures follow a uniform distribution in the region. The reference model and the

oil-saturation map are shown in **Fig. 14**. The size of the reservoir is 1200×1200 m, which is discretized into 120×120 gridblocks. The permeability and porosity of the matrix are considered as constants of 3 md and 0.2. For fractures, the porosity is set to 0.3 and the permeability follows the cubic law. Nine-spot well pattern is applied. The injector located in the center injects water at a constant rate of $500 \text{ m}^3/\text{d}$. Eight producers located at the corners are at a constant WBHP of 15 MPa. The duration of the production is 300 days, which is divided into 30 timesteps. The duration for future prediction is 200 days. The observed data contain the WBHP, WWCT, and WOPR. We assume that the error of observation data follows the normal distribution. The mean is set to be zero, and the standard deviations of WBHP, WWCT, and WOPR are set to 1%, 1%, and 3% of the observed data, respectively.

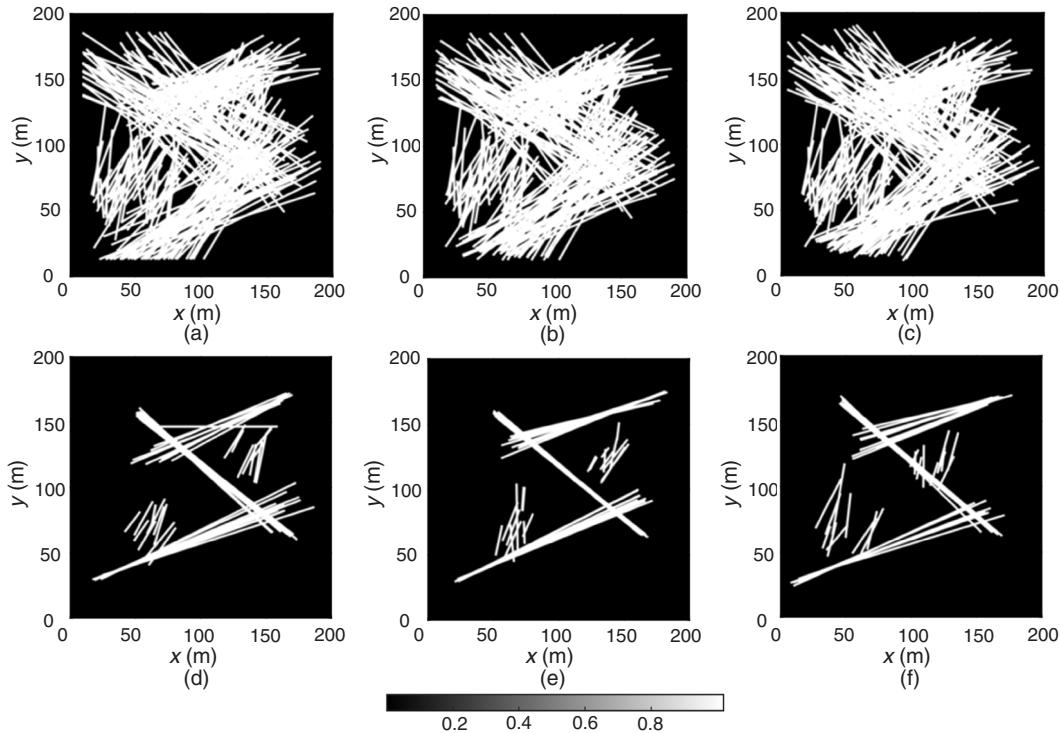


Fig. 12—The fracture-probability maps for Case 1: (a) initial models by ES-MDA directly; (b) initial models by DSAE; (c) initial models by PCA; (d) history-matched models by ES-MDA directly; (e) initial models by DSAE; (f) history-matched models by PCA.

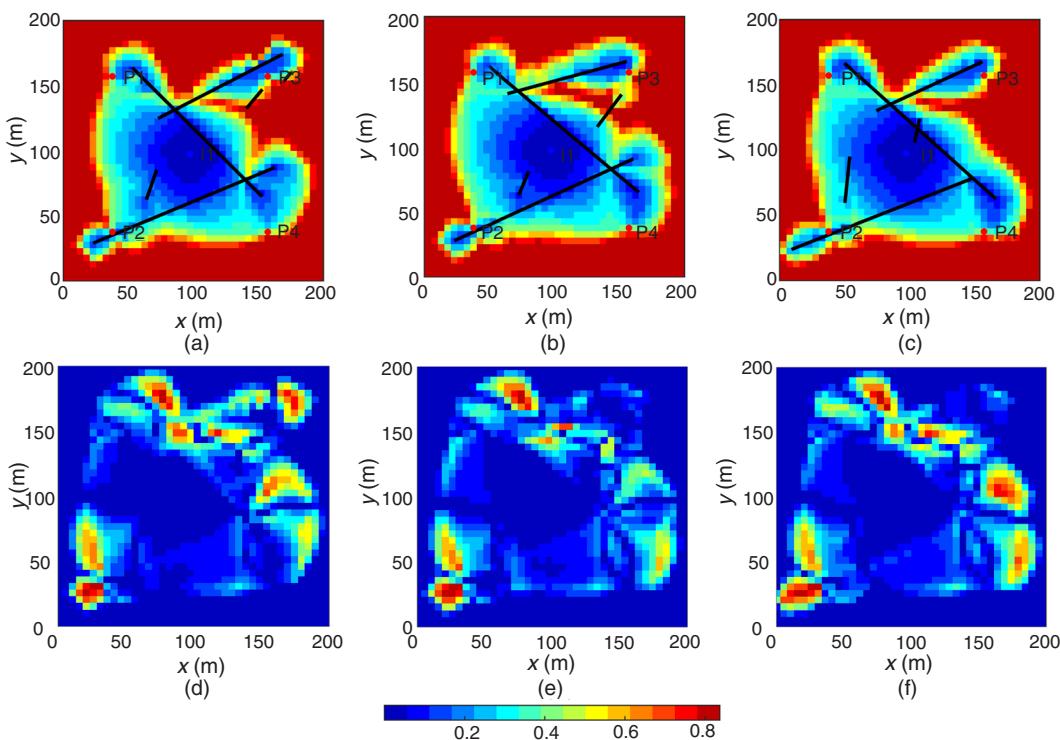


Fig. 13—Comparison of the oil-saturation maps for Case 1. The first row shows the oil-saturation map of the history-matched model: (a) using ES-MDA directly; (b) using DSAE; (c) using PCA. The second row shows the difference of oil-saturation maps with the reference model: (d) using ES-MDA directly; (e) using DSAE; (f) using PCA.

In this case, it is a 200-dimension optimization problem for 50 vertical fractures. With the energy criterion and the threshold value of 90%, we used the DSAE and PCA to reduce the dimension from 200 to 158. We stacked two autoencoders for DSAE, which consists of an encoder with layers of (200)-200-158 and a symmetric decoder. The coefficients λ , β , and ρ are the same as in Case 1. After training the DSAE, we compared the original samples with the generated samples using random latent variables for both DSAE and PCA in **Fig. 15**. In Fig. 15b, most of the fractures generated by DSAE are orthogonal and only a few fractures are unrealistic. However, in Fig. 15c, the orientations of fractures generated by PCA are random, which has a great difference with samples. The results show that the PCA is not suitable to parameterize fracture networks that follow a certain distribution. The structures of fracture networks generated by DSAE are closer to samples.

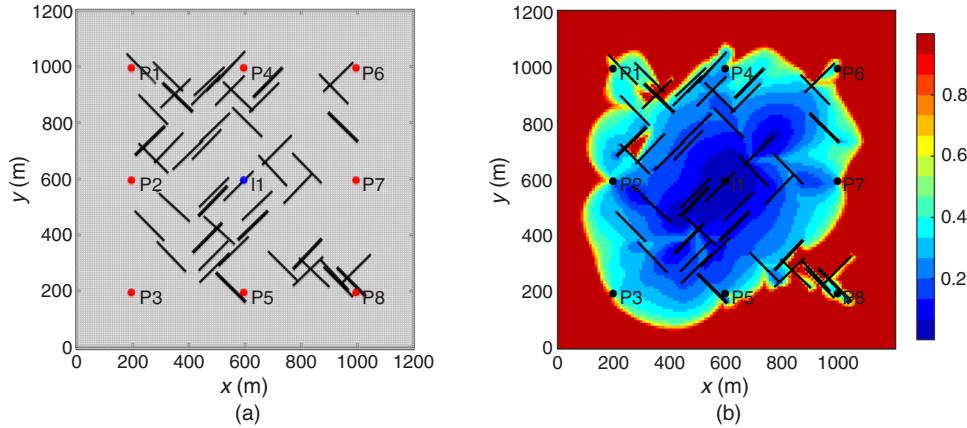


Fig. 14—The reference model for Case 2: (a) the fracture network; (b) the oil-saturation map at the 300th day.

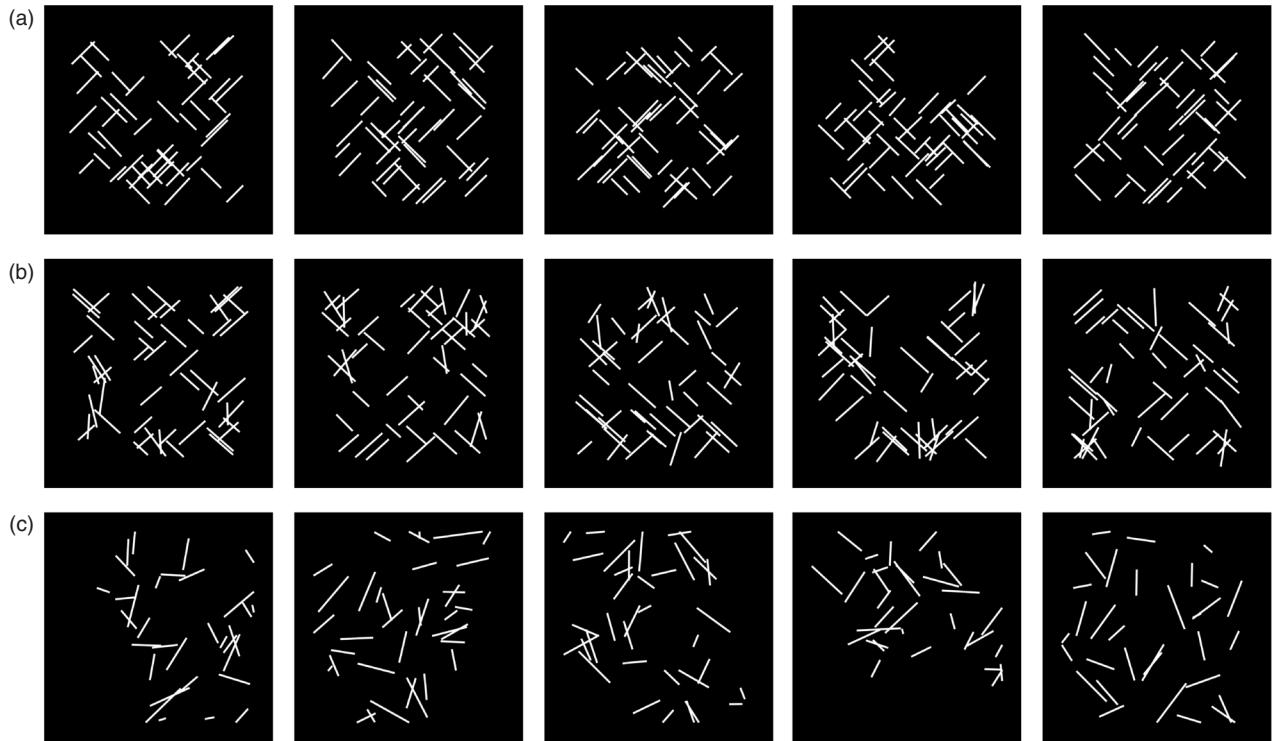


Fig. 15—Comparison of (a) original fracture networks, (b) fracture networks generated by DSAE, and (c) fracture networks generated by PCA for Case 2.

We use the DSAE, PCA, and ES-MDA directly to solve the problem. The objective functions are shown in **Fig. 16**. The objective function of using DSAE converges fast and has the lowest value. However, using ES-MDA directly to solve parameters does not have the same performance as Case 1. We compared fracture networks using the DSAE, PCA, and ES-MDA directly after the first data assimilation in **Fig. 17**. Fracture networks updated by ES-MDA directly or PCA do not preserve geometric shapes of samples in Figs. 17a and 17c. Fracture networks updated by DSAE could maintain the orthogonal properties after updating in Fig. 17b. Only a few fractures are unrealistic compared with samples. Because fracture networks after updating using PCA or ES-MDA directly are unrealistic, the performance of objective functions is unsatisfactory.

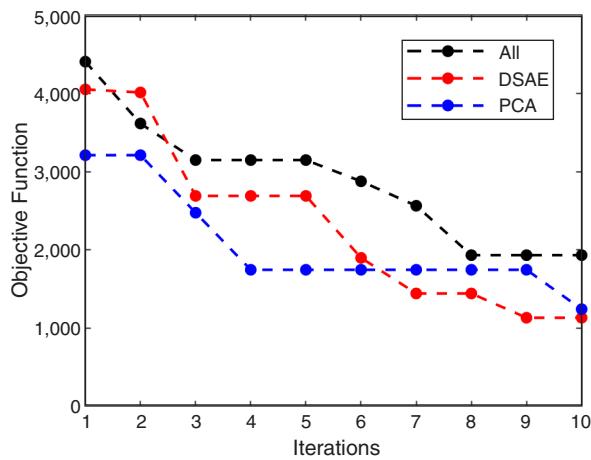


Fig. 16—Convergence behavior of the objective function for Case 2. The black line represents the method that uses ES-MDA to directly estimate all parameters; the red line represents the method that uses DSAE; the blue line represents the method that uses PCA.

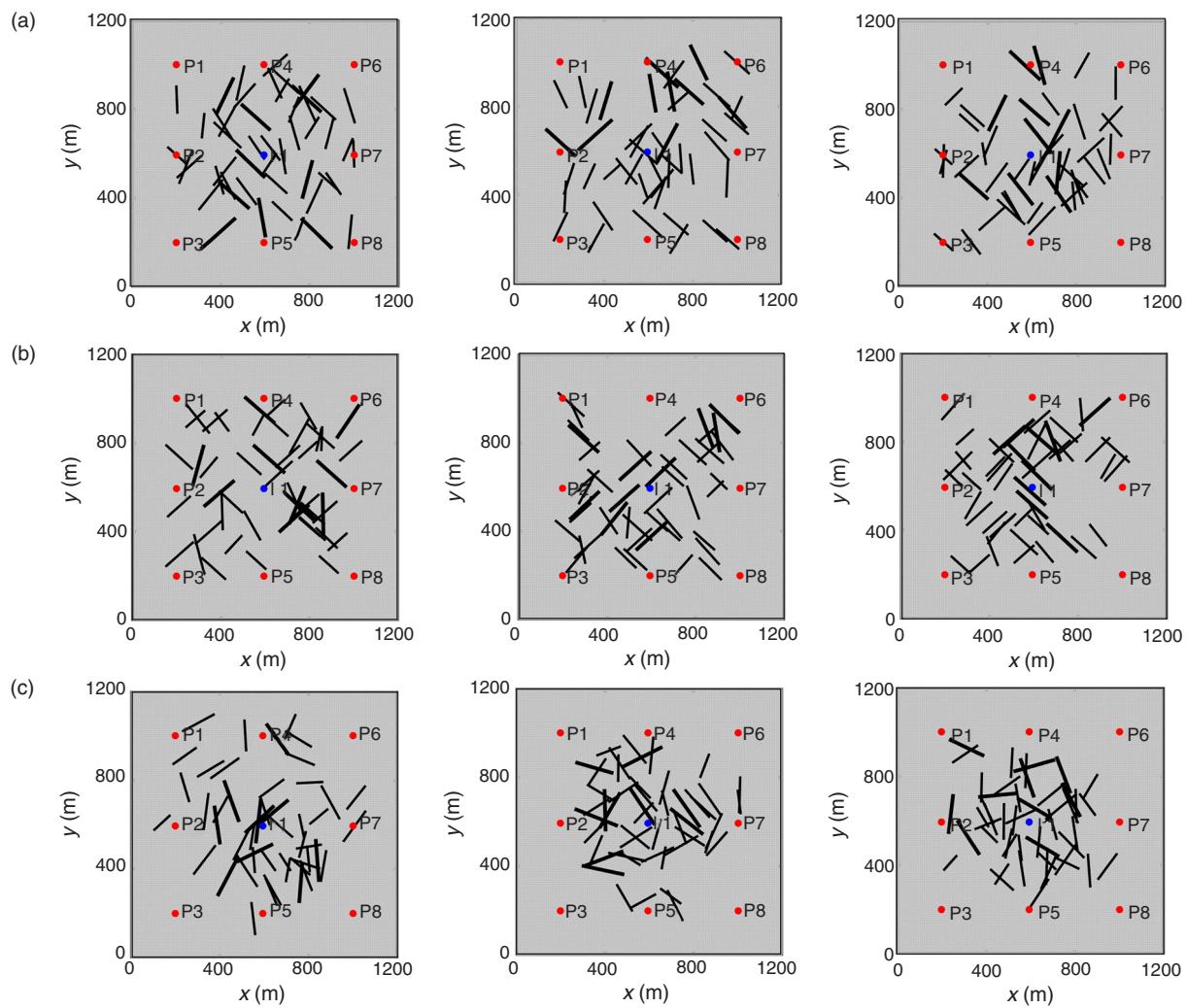


Fig. 17—Fracture networks updated after one data assimilation (a) using ES-MDA directly, (b) DSAE, and (c) PCA for Case 2.

Figs. 18, 19, and 20 show history-match results using ES-MDA directly, DSAE, and PCA, respectively. The results are similar among these methods. This means that although fracture networks after updating are unrealistic, the data-matching results are satisfactory using ES-MDA.

Fig. 21 shows the fracture-probability map. The first row shows some initial fracture networks and the second row shows some history-matched fracture networks. We can see that some fractures of initial models using PCA are unrealistic because of the reconstruction performance in Fig. 21c. For using DSAE, initial fractures are more similar to initial fractures using ES-MDA directly in Figs. 21a and 21b. The history-matched models using ES-MDA directly or PCA are far from the reference model in Figs. 21d and 21f. Fractures of history-matched models using DSAE are mostly vertical and the distribution features are more consistent with the reference model in Fig. 21e.

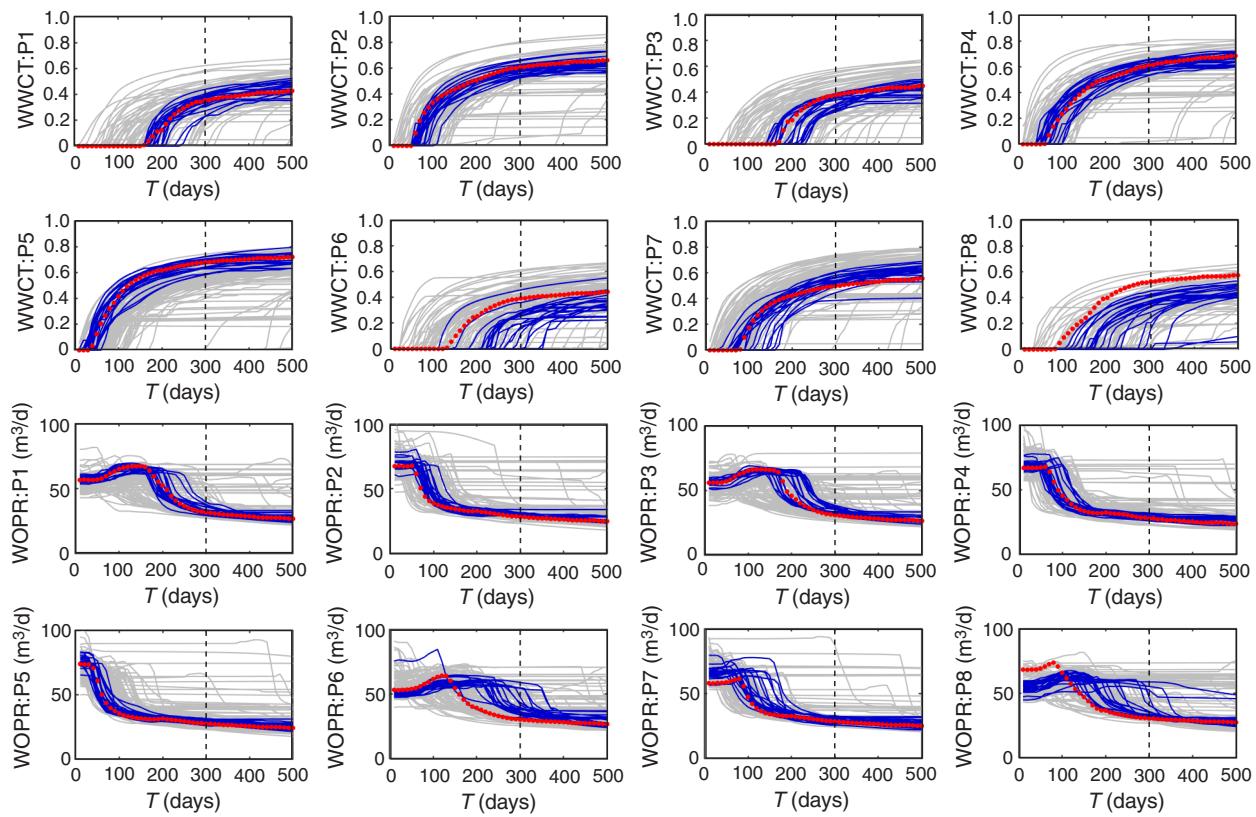


Fig. 18—History-match results using ES-MDA directly for Case 2. The first and second rows show the water cut (WWCT) and the third and fourth rows show the oil-production rate (WOPR). The gray represents the simulated data of the initial models, the blue represents the simulated data of the history-matched models, and the red dots represents the observation data.

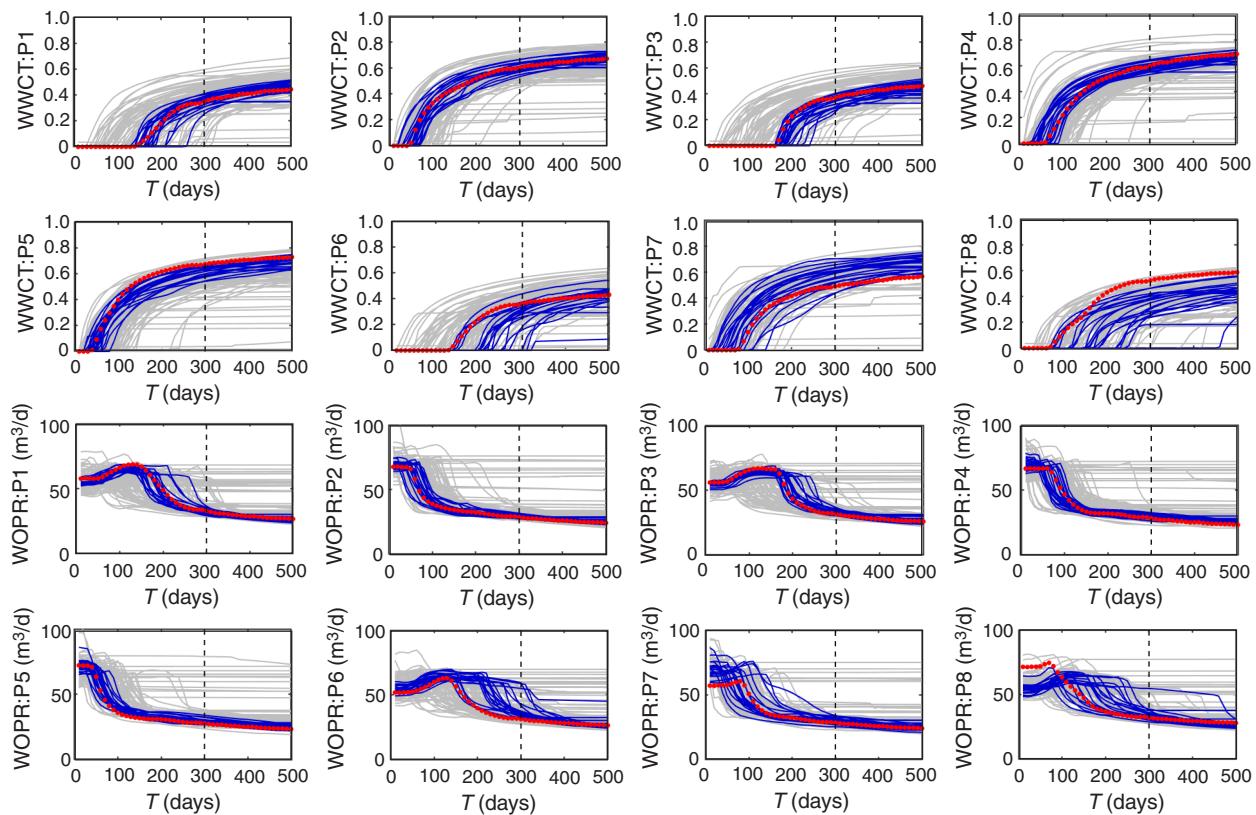


Fig. 19—History-match results using DSAE for Case 2. The first and second rows show the water cut (WWCT) and the third and fourth rows show the oil-production rate (WOPR). The gray represents the simulated data of the initial models, the blue represents the simulated data of the history-matched models, and the red dots represents the observation data.

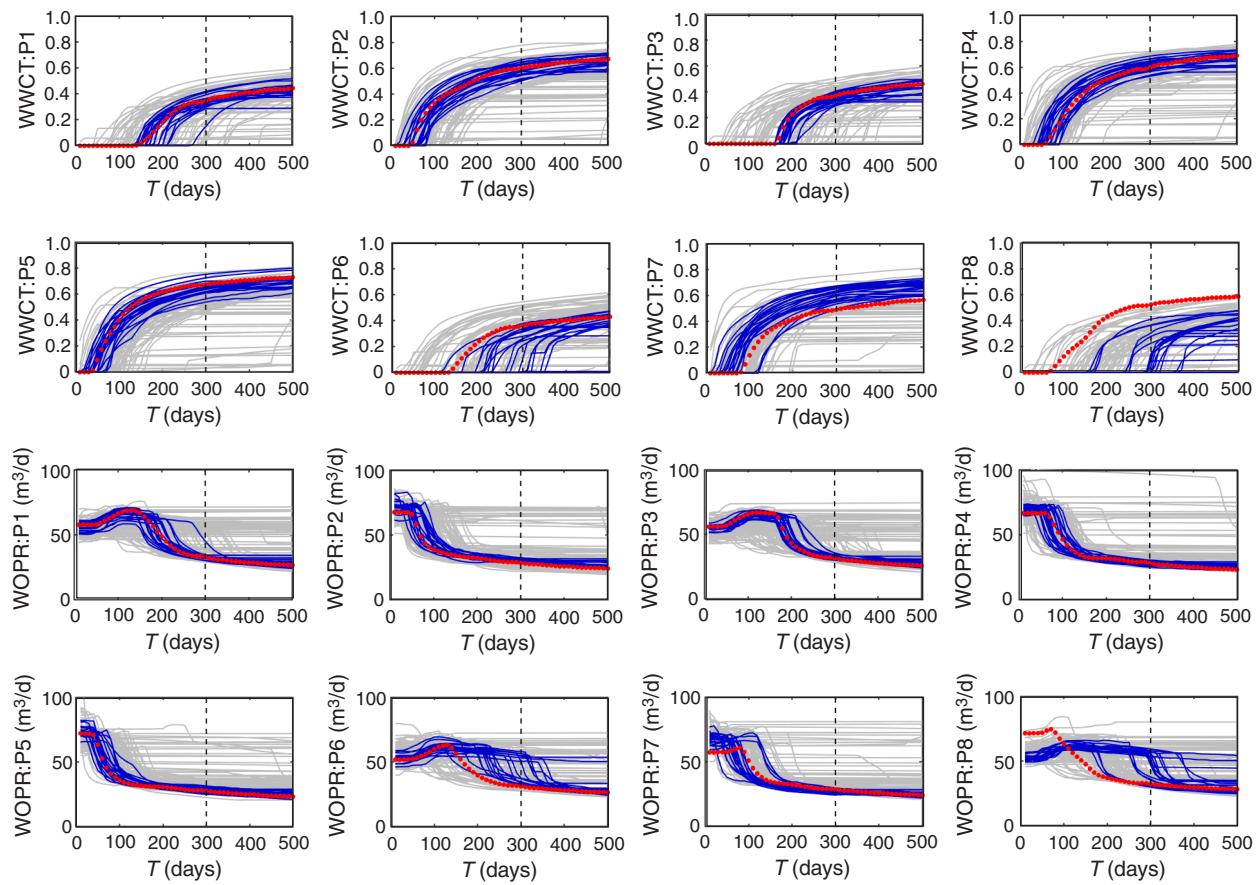


Fig. 20—History-match results using PCA for Case 2. The first and second rows show the water cut (WWCT) and the third and fourth rows show the oil-production rate (WOPR). The gray represents the simulated data of the initial models, the blue represents the simulated data of the history-matched models, and the red dots represents the observation data.

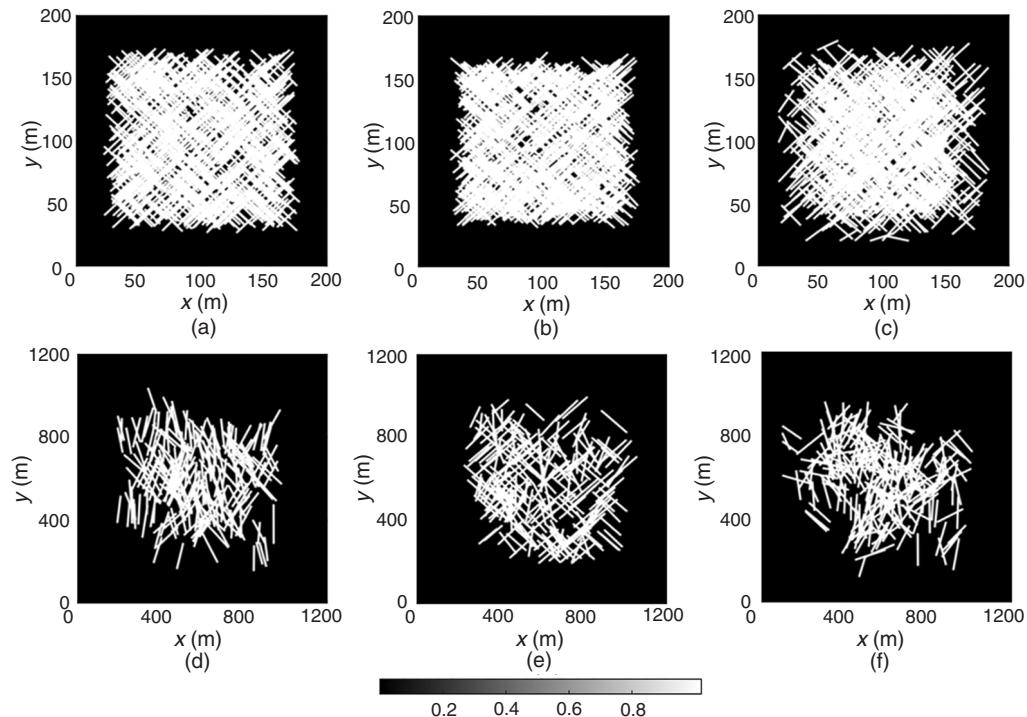


Fig. 21—The fracture-probability maps for Case 2: (a) initial models using ES-MDA directly; (b) initial models using DSAE; (c) initial models using PCA; (d) history-matched models using ES-MDA directly; (e) history-matched models using DSAE; (f) history-matched models using PCA.

We compared oil-saturation maps of history-matched models in **Fig. 22**. The mean-squared errors using ES-MDA directly, DSAE, and PCA are 0.0633, 0.0585, and 0.0602, respectively. The oil-saturation map of the history-matched model by DSAE is the closest to the reference model. The oil-saturation maps using ES-MDA directly or PCA are not similar to the reference model. The results of Case 2 show that using ES-MDA or PCA to history match fractures that have a certain distribution will result in unrealistic fracture networks during the update, and the DSAE could preserve distribution features of fractures and perform better.

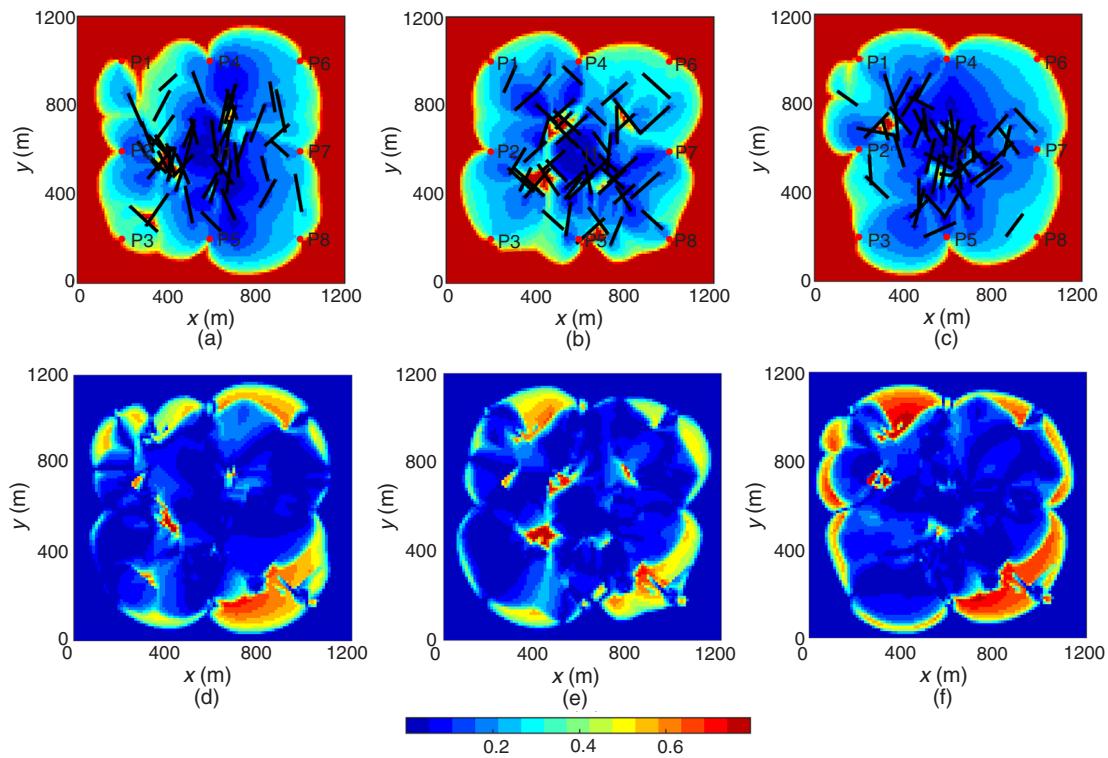


Fig. 22—Comparison of the oil-saturation maps for Case 2. The first row shows the oil-saturation map at the 300th day of the history-matched model: (a) using ES-MDA directly; (b) using DSAE; (c) using PCA. The second row shows the difference of oil-saturation maps with the reference model: (d) using ES-MDA directly; (e) using DSAE; (f) using PCA.

Case 3: Field Application of an Outcrop-Based Model. For verifying the effectiveness of the DSAE for high-dimensional problems, we apply this method to a more complex model with 10 large-scale fractures and 300 small-scale fractures. The dimension of this case is up to 1,240. The 10 large-scale fractures are selected from an outcrop example in a quarry in southeast France (Bisdom et al. 2014). Assuming that the ranges of parameters are known, we generated the small-scale fractures using the different distribution functions. We used the same model settings as in Case 2. For the 10 large-scale fractures, we calculated the parameters of the reference model using the outcrop data and then generated the range of parameters. **Table 2** shows the true value of the reference model and the parameter range for the large-scale fractures. Next, we set the parameters of small-scale fractures. The total number of small-scale fractures for the reference model is 250. For the initial models, the range of the small-scale-fracture number is 200 to 300. For the reference model, the proportions of the fracture number in four regions are set to 0.1, 0.3, 0.3, and 0.1 respectively. We generated the initial proportion of the regions using four random numbers between 0.1 and 1. The length of the small-scale fractures follows the log-normal distribution, and the mean and standard deviation are set to 4 and 0.3, respectively, for the reference model. The initial ranges of the mean and standard deviation are 3.8 to 4.3 and 0.2 to 0.4, respectively. The orientation of small-scale fractures follows the normal distribution, which is evenly divided into two sets. For Set 1, the mean and standard deviation for the reference model are 45 and 30, respectively. The initial ranges of the mean and standard deviation are 35 to 55 and 20 to 40, respectively. For Set 2, the mean and standard deviation for the reference model are 135 and 30, respectively. The initial ranges of the mean and standard deviation are 125 to 145 and 20 to 40. Moreover, we randomly generated the midpoint coordinates of small-scale fractures for each region using the uniform distribution. **Table 3** presents the parameters of the small-scale fractures. **Fig. 23a** shows the fracture network, the well pattern, and the grids. With the EDFM simulator, we can obtain the residual oil saturation at the 300th day shown in **Fig. 23b**.

The maximum fracture number of the fracture network is 300, which brings about a 1,240-dimension inverse problem. Using 1,240 dimensions, we generated 4,000 samples for training the DSAE. The number of small-scale fractures is changing between 200 and 300, and the dimension is fixed by mending zero in the matrix. Considering the high-dimensional data, we stacked four autoencoders to efficiently reduce the dimension. The number of nodes for each layer is set empirically. The DSAE consists of an encoder with layers of (1,240)-1,000-600-300-100 and a symmetric decoder. We tested fracture networks generated by DSAE using random latent variables in **Fig. 24**. **Fig. 24a** shows original fracture networks and **Fig. 24b** shows fracture networks generated by DSAE using random latent variables. The distribution features for both large-scale and small-scale fractures generated by DSAE are similar to original fracture networks. The results of fracture networks generated by DSAE demonstrate the feasibility of the DSAE in history matching.

The codes of the hidden layer of the DSAE are taken as variables for the ES-MDA algorithm. The dimension of parameters changes from 1,240 to 100 through using the DSAE. The ensemble size is set to 100 and the number of iterations is set to 10. **Fig. 25** presents the results of the objective function using DSAE. As the number of iterations increases, the objective function has a significant decline. When the number of iterations reaches seven, the objective function converges.

Fracture	Length		Orientation		x-Coordinate		y-Coordinate	
	True Value	Initial Range	True Value	Initial Range	True Value	Initial Range	True Value	Initial Range
1	796.20	650–900	90.10	70–120	289.80	200–350	664.80	550–800
2	301.65	200–400	2.32	0–30	371.60	300–450	1,044.00	900–1,050
3	274.11	150–400	165.93	140–180	907.05	850–1,000	918.00	850–1,000
4	266.73	150–400	167.19	140–180	711.85	650–800	626.95	600–700
5	263.81	150–400	26.66	10–50	406.80	350–500	833.55	700–900
6	261.16	150–350	91.21	70–120	512.65	400–600	847.65	700–900
7	251.47	150–350	18.27	0–30	633.60	550–700	992.80	900–1,050
8	247.42	150–350	30.74	10–40	611.05	550–700	782.80	600–850
9	247.05	150–350	164.52	140–180	756.55	650–850	1,043.35	900–1,050
10	223.24	150–350	29.98	20–60	300.00	200–400	629.05	550–700

Table 2—The reference-model parameters for the large-scale fractures and the initial range.

	Parameters		True Value	Initial Range
	Fracture number	All regions	250	200–300
Proportion of the fracture number	Region 1	0.1	0–1	
	Region 2	0.3	0–1	
	Region 3	0.3	0–1	
	Region 4	0.1	0–1	
Length	Mean	4	3.8–4.3	
	Standard deviation	0.3	0.2–0.4	
Orientation	Mean (Set 1)	45	35–55	
	Standard deviation (Set 1)	30	20–40	
	Mean (Set 2)	135	125–145	
	Standard deviation (Set 2)	30	20–40	

Table 3—The reference-model parameters for small-scale fractures and the initial range.

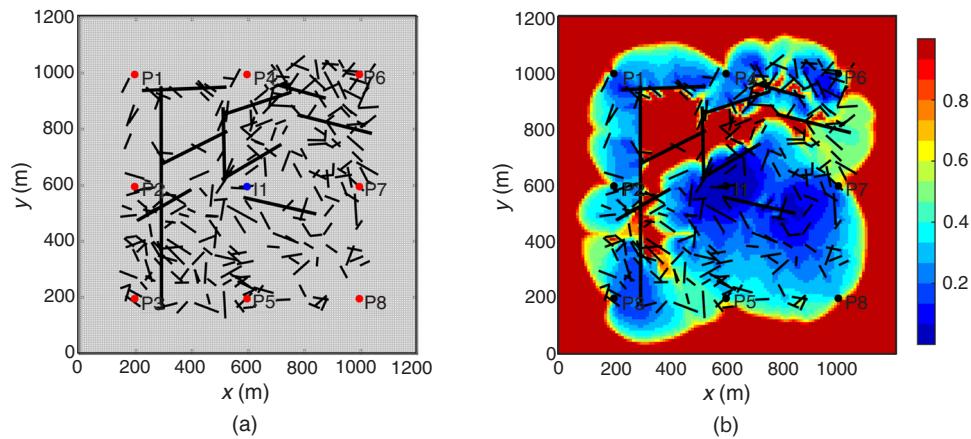


Fig. 23—The reference model for Case 3: (a) the fracture network; (b) the oil-saturation map at the 300th day.

Fig. 26 shows history-match results and prediction results using DSAE. For the initial ensemble, the simulated data including WWCT and WOPR have a wide distribution range, which is far from the observed data of the reference model. After 10 iterations, the simulated data as a whole are close to the observed data for both methods. We obtain satisfactory data-matching results using the low-dimensional variables.

Fig. 27 shows the fracture-probability map of initial models and history-matched models. The distribution of fracture networks of initial models has great uncertainties. After 10 iterations, the distribution range of fracture networks of history-matched models becomes small. Large-scale fractures of history-matched models almost overlap and are similar to the reference model. Small-scale fractures of history-matched models concentrate on the upper-right part of the region shown in Fig. 27b. However, the distribution of most small-scale fractures of the reference model concentrates on the lower-left and upper-right parts of the region. This difference

between small-scale fractures of the reference model and small-scale fractures of the history-matched model is because small-scale fractures do not have a big influence on the fluid flow compared with large-scale fractures. Fig. 28 shows the comparison of the oil-saturation map for Case 3. Fig. 28a shows the initial model, which is far from the reference model. Fig. 28b shows the oil-saturation map of the history-matched model and Fig. 28c shows the reference model. The shape of the oil-saturation map of the history-matched model is similar to the reference model. The oil saturation near production wells and the injector well nearly match the oil saturation of the reference model. However, the oil saturation near a production well, Well P6, has some differences. We can also see the influence on the data-matching results in Fig. 26. The simulated data of history-matched models for Well P6 have a larger range compared with the simulated data of other wells. The oil saturations of some other regions of the model also do not have a good match. Because of the limitation of the observed data and the multiplicity of the inverse problem, we could not match the oil saturation in the whole region. The results of Case 3 demonstrate that the DSAE could learn distribution features of the complex multiscale fracture network and it could be applied to history match the naturally fractured reservoirs.

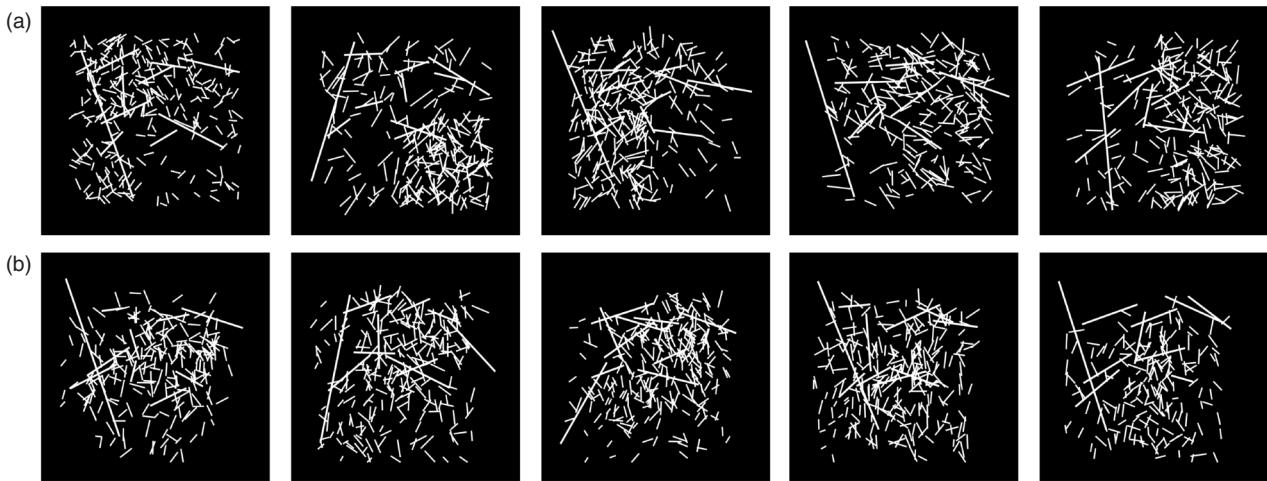


Fig. 24—Comparison of (a) original fracture networks, (b) fracture networks generated by DSAE for Case 3.

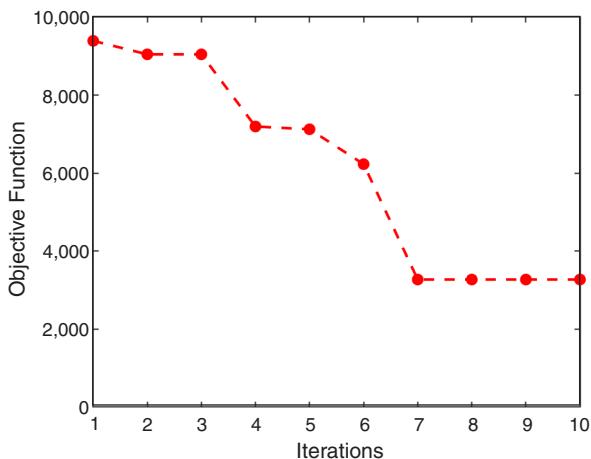


Fig. 25—Convergence behavior of the objective function for Case 3.

Discussion. In this work, we used three cases to demonstrate the efficiency of our approach for history matching fractured reservoirs. Case 1 is a quite simple reservoir model with five large-scale fractures. We compared the performance of using ES-MDA directly, DSAE, and PCA. The results show that using ES-MDA directly to solve this low-dimensional problem can obtain the best history-matched model and using DSAE obtains similar history-match results compared with using PCA. Case 2 has 50 vertical fractures and the dimension is 200. After training, the DSAE could generate more realistic fracture networks compared with using PCA. Note that fracture networks after updating using ES-MDA directly or PCA are far from samples and could not preserve the vertical properties. The results show that if the fracture parameters have a certain distribution, using DSAE could preserve the distribution features during the updates and obtain the best history-match results. Case 3 is a more complex multiscale fracture-network model. The dimension is up to 1,240. We use the DSAE to reduce the dimension to 100. The simulated data match the observed data well. The results of Case 3 demonstrate that our approach is suitable for the inversion of the high-dimensional, complex, and multiscale fracture network.

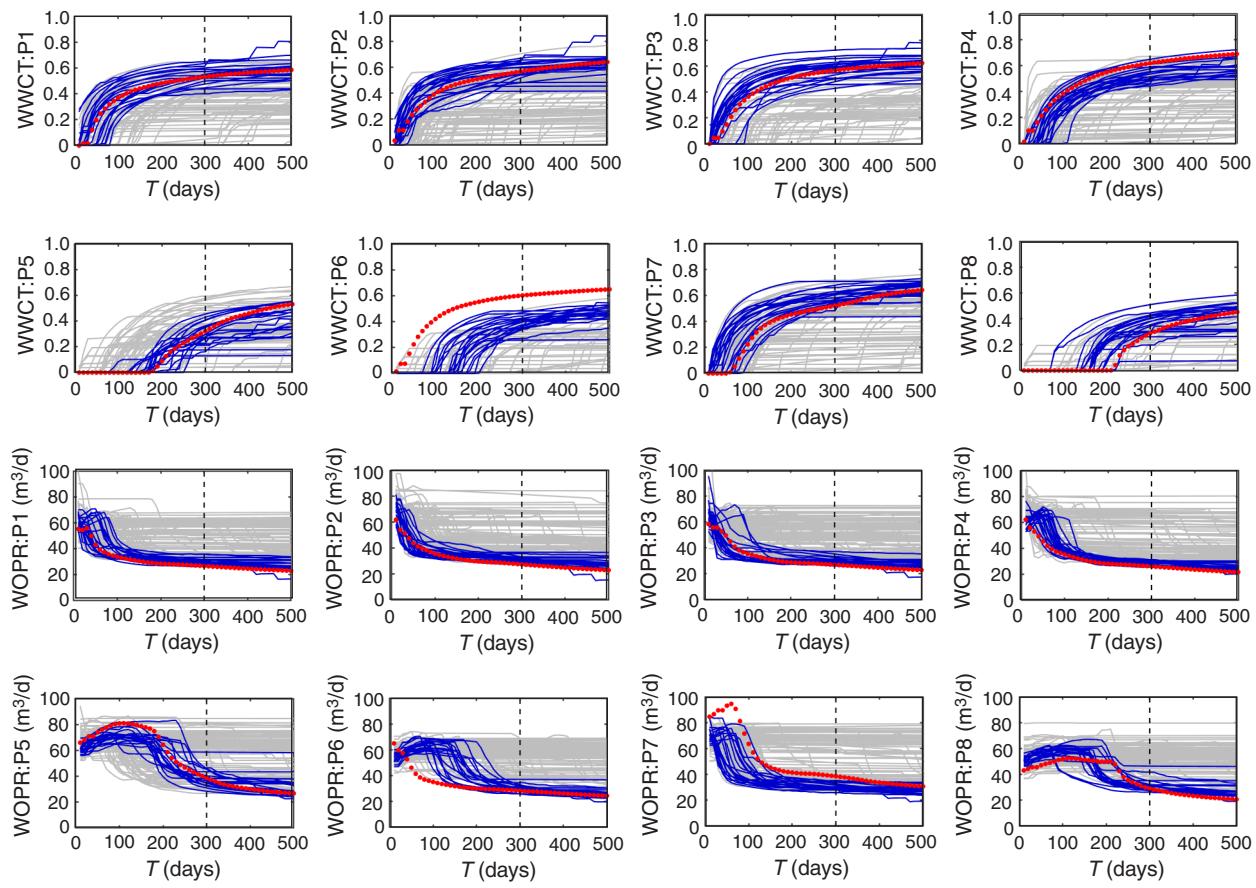


Fig. 26—History-match results and prediction results by DSAE for Case 3. The first and second rows show the water cut (WWCT) and the third and fourth rows show the oil-production rate (WOPR). The gray represents the simulated data of initial models, the blue represents the simulated data of history-matched models, and the red dots represent the observation data.

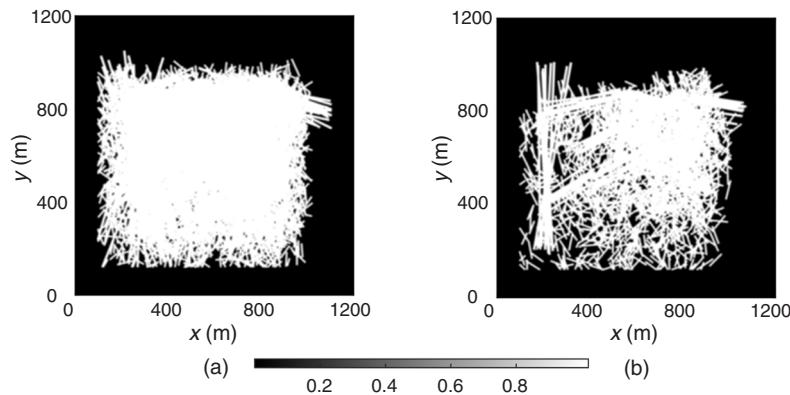


Fig. 27—The fracture-probability maps for Case 3: (a) initial models by DSAE; (b) history-matched models by DSAE.

The previous work by Ma et al. (2020) applied different characterization methods for large-scale and small-scale fractures. We develop this method using the consistent characterization method for the fracture network and a dimensionality-reduction method. The results show the feasibility and efficiency of our approach. It provides an alternative scheme to history matching the complex multiscale fractured reservoirs.

However, there are still some limitations. First, training a DSAE for dimension reduction is time consuming, which needs to adjust the parameters constantly, such as the number of samples, hidden layers, hidden nodes, and so forth. There are not explicit answers for identifying these parameters. We still need a great amount of research to improve the applicability of this autoencoder for history matching. Second, the inversion of fractures usually has multiple solutions because of the complexity of the structure and the limitations of production data. Whether we obtain a true model requires various analysis for the results. Third, in case studies, the small-scale fractures have small effects on the fluid flow. In some cases, if the number of small-scale fractures is large enough, it will have a significant influence on production. We also need to perform research for the small-scale fractures in history matching. Finally, the 2D numerical simulation for the fractured reservoir is a simplification, which is not realistic for the actual fractured reservoirs. It is necessary to improve the 2D numerical simulation or develop the 3D numerical simulation.

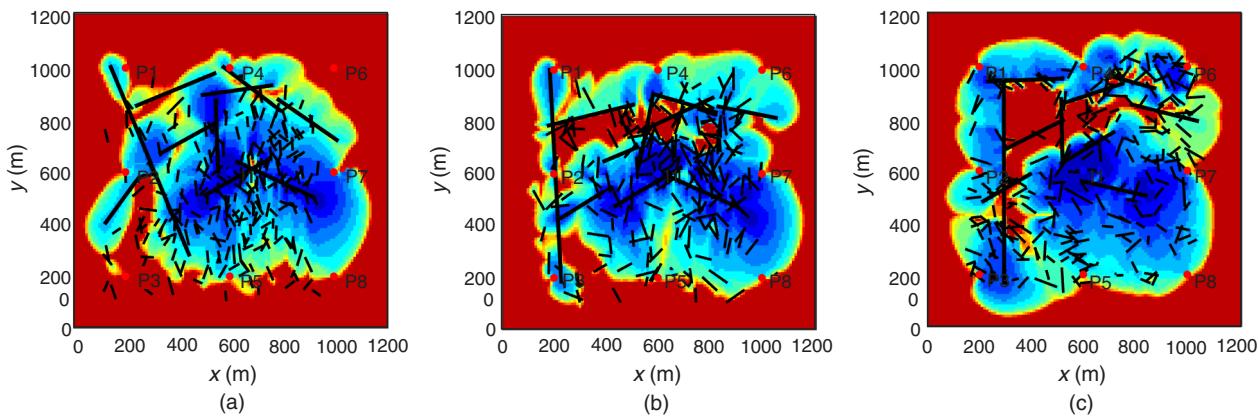


Fig. 28—Comparison of the oil-saturation maps for Case 3: (a) the initial model by DSAE; (b) the history-matched model by DSAE; (c) the reference model.

Conclusions

In this paper, we propose a characterization method for the fracture network and a dimensionality-reduction method using the DSAE. Three synthetic cases show the advantages of the characterization method integrated with DSAE for history matching of fracture networks. In terms of simple fracture networks, the history-match results by DSAE, PCA, and no-dimensionality-reduction method are similar. However, the updated fracture networks during history matching by PCA or no-dimensionality-reduction method become unrealistic when fracture networks have complex distribution characteristics. In comparison, DSAE can preserve the distribution characteristics of fractures during history matching and obtain a better history-match result. Our proposed method provides some inspiration for history matching of actual naturally fractured reservoirs. There are still some limitations in our method, such as the 2D characterization method for fractures and the elapsed time of training DSAE. Thus, it is critical to develop an efficient characterization method in 3D space for fractures and to improve the applicability of DSAE for actual naturally fractured reservoirs.

Nomenclature

- $b^{(1)}$ = bias of the encoder, vector
- $b^{(2)}$ = bias of the decoder, vector
- c_a = coefficient of the cubic law, $\in R$
- C = number of the hidden layers, $\in R$
- C_D^n = covariance matrix of measurement errors of the n th data assimilation, $\in R^{N_d \times N_d}$
- C_{DD}^n = auto-covariance matrix of predicted data of the n th data assimilation, $\in R^{N_d \times N_d}$
- C_{MD}^n = cross-covariance matrix between parameters and predicted data of the n th data assimilation, $\in R^{N_m \times N_d}$
- d_{obs} = observed data, $\in R^{N_d}$
- $d_j^{n,f}$ = predicted data of the j th member of the n th data assimilation, $\in R^{N_d}$
- $d_{uc,j}^n$ = observed data with a noise of the j th member of the n th data assimilation, $\in R^{N_d}$
- D = number of hidden nodes, $\in R$
- E = cost function of the autoencoder
- f_1 = function of the log-normal distribution
- f_2 = function of the normal distribution
- h = aperture of the fracture, m, $\in R$
- k_f = permeability of the fracture, md, $\in R$
- KL = Kullback-Leibler divergence
- L = length of the fracture
- L_{li} = length of the i th large-scale fracture, m, $\in R$
- $L_{l\max}$ = maximum length of large-scale fractures, m, $\in R$
- $L_{l\min}$ = minimum length of large-scale fractures, m, $\in R$
- L_{si} = length of small-scale fractures of the i th sample, vector, m
- m = model parameters
- $m_{j,a}^{n,a}$ = analytical model parameters of the j th member of the n th data assimilation, $\in R^{N_m}$
- $m_j^{n,f}$ = forecast model parameters of the j th member of the n th data assimilation, $\in R^{N_m}$
- n_{si} = number of small-scale fractures of the j th sample, $\in R$
- $n_{s\max}$ = maximum number of small-scale fractures, $\in R$
- $n_{s\min}$ = minimum number of small-scale fractures, $\in R$
- N = number of training samples, dimensionless, $\in R$
- $p(d_{obs}|m)$ = likelihood function
- $p(m)$ = priori PDF
- $p(m|d_{obs})$ = posteriori PDF
- r_j = random number between 0.1 and 1
- $W^{(1)}, W^{(2)}$ = weight matrix of the encoder and decoder, vector
- x_0 = x -coordinate of the midpoint, m
- $x_{0\max}$ = maximum value of x -coordinate of the midpoint for large-scale fractures, m
- $x_{0\min}$ = minimum value of x -coordinate of the midpoint for large-scale fractures, m
- x_{0si} = x -coordinate of the midpoint of small-scale fractures of the i th sample, m
- x_1, x_2 = x -coordinate of the endpoints, m

- y_0 = y -coordinate of the midpoint, m
 $y_{0\max}$ = maximum value of y -coordinate of the midpoint for large-scale fractures, m
 $y_{0\min}$ = minimum value of y -coordinate of the midpoint for large-scale fractures, m
 y_{0si} = y -coordinate of the midpoint of small-scale fractures of the i th sample, m
 y_1, y_2 = y -coordinate of the endpoints, m
 Z_d = random values from the normal distribution
 α_i = coefficient of the noise for the observed data, $\in R$
 β = coefficient of the sparse regularization, $\in R$
 $\delta^{(1)}, \delta^{(2)}$ = activation function of the encoder and decoder for the autoencoder
 θ = orientation of fractures, degrees
 θ_{li} = orientation of large-scale fractures of the i th sample, degrees
 $\theta_{l\max}$ = maximum orientation of large-scale fractures, degrees
 $\theta_{l\min}$ = minimum orientation of large-scale fractures, degrees
 θ_{si} = orientation of small-scale fractures of the i th sample, degrees
 λ = coefficient of the L_2 regularization, $\in R$
 μ_{Li} = mean value of the length of small-scale fractures of the i th sample, $\in R$
 $\mu_{L\max}$ = maximum mean value of the length for small-scale fractures of the i th sample, $\in R$
 $\mu_{L\min}$ = minimum mean value of the length for small-scale fractures of the i th sample, $\in R$
 $\mu_{\theta\max}$ = maximum mean value of the orientation for small-scale fractures of the i th sample, $\in R$
 $\mu_{\theta\min}$ = minimum mean value of the orientation for small-scale fractures of the i th sample, $\in R$
 ρ = desired value of the hidden unit, $\in R$
 $\hat{\rho}$ = average value of the hidden unit, $\in R$
 σ_{Li} = standard deviation of the length of small-scale fractures of the i th sample, $\in R$
 $\sigma_{L\max}$ = maximum standard deviation of the length of small-scale fractures, $\in R$
 $\sigma_{L\min}$ = minimum standard deviation of the length of small-scale fractures, $\in R$
 $\sigma_{\theta\max}$ = maximum standard deviation of the orientation of small-scale fractures, $\in R$
 $\sigma_{\theta\min}$ = minimum standard deviation of the orientation of small-scale fractures, $\in R$
 ϕ_i = model parameters of the i th sample, vector
 Ω_s = sparse regularization
 Ω_w = L_2 regularization
 D_j = proportion of the fracture number of the j th region between zero and unity
 \mathcal{H} = hidden layer
 \mathcal{X} = input of the autoencoder
 \mathcal{X}' = output of the autoencoder

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant Nos. 51722406, 51874335, and 51674280), the Shandong Provincial Natural Science Foundation (Grant Nos. JQ201808 and ZR2019JQ21), the Fundamental Research Funds for the Central Universities (Grant No. 18CX02097A), the Major Scientific and Technological Projects of the China National Petroleum Corporation (Grant No. ZD2019-183-008), the Science and Technology Support Plan for Youth Innovation of University in Shandong Province (Grant No. 2019KJH002), the National Science and Technology Major Project of China (Grant No. 2016ZX05025001-006), and the 111 Project (Grant No. B08028).

References

- Abdi, H. and Williams, L. J. 2010. Principal Component Analysis. *Wiley Interdiscip Rev Comput Stat* **2** (4): 433–459. <https://doi.org/10.1002/wics.101>.
- Bisdom, K., Gauthier, B. D. M., Bertotti, G. et al. 2014. Calibrating Discrete Fracture-Network Models with a Carbonate Three-Dimensional Outcrop Fracture Network: Implications for Naturally Fractured Reservoir Modeling. *AAPG Bull.* **98** (7): 1351–1376. <https://doi.org/10.1306/02031413060>.
- Canchumuni, S. A., Emerick, A. A., and Pacheco, M. A. 2017. Integration of Ensemble Data Assimilation and Deep Learning for History Matching Facies Models. Paper presented at the OTC Brasil, Rio de Janeiro, Brazil, 24–26 October. OTC-28015-MS. <https://doi.org/10.4043/28015-MS>.
- Chen, C., Gao, G., Honorio, J. et al. 2014. Integration of Principal-Component-Analysis and Streamline Information for the History Matching of Channeled Reservoirs. Paper presented at the SPE Annual Technical Conference and Exhibition, Amsterdam, The Netherlands, 27–29 October. SPE-170636-MS. <https://doi.org/10.2118/170636-MS>.
- Emerick, A. A. and Reynolds, A. C. 2012. History Matching Time-Lapse Seismic Data Using the Ensemble Kalman Filter with Multiple Data Assimilations. *Computat Geosci* **16** (3): 639–659. <https://doi.org/10.1007/s10596-012-9275-5>.
- Emerick, A. A. and Reynolds, A. C. 2013a. Ensemble Smoother with Multiple Data Assimilation. *Comput Geosci* **55** (June): 3–15. <https://doi.org/10.1016/j.cageo.2012.03.011>.
- Emerick, A. A. and Reynolds, A. C. 2013b. Investigation of the Sampling Performance of Ensemble-Based Methods with a Simple Reservoir Model. *Computat Geosci* **17** (2): 325–350. <https://doi.org/10.1007/s10596-012-9333-z>.
- Evensen, G. 1994. Sequential Data Assimilation with Nonlinear Quasi-Geostrophic Model Using Monte Carlo Methods To Forecast Error Statistics. *J Geophys Res Oceans* **99** (C5): 10143–10162. <https://doi.org/10.1029/94JC00572>.
- Evensen, G. and Van Leeuwen, P. J. 2000. An Ensemble Kalman Smoother for Nonlinear Dynamics. *Mon Weather Rev* **128** (6): 1852–1867. [https://doi.org/10.1175/1520-0493\(2000\)128<1852:AEKSFN>2.0.CO;2](https://doi.org/10.1175/1520-0493(2000)128<1852:AEKSFN>2.0.CO;2).
- Fung, L. S. K. and Collins, D. A. 1991. An Evaluation of the Improved Dual Porosity Model for the Simulation of Gravity Effects in Naturally Fractured Reservoirs. *J Can Pet Technol* **30** (3): 61–68. PETSOC-91-03-06. <https://doi.org/10.2118/91-03-06>.
- Goodfellow, I., Bengio, Y., and Courville, A. 2016. *Deep Learning*. Cambridge, Massachusetts, USA: The MIT Press.
- Karimi-Fard, M., Durlofsky, L. J., and Aziz, K. 2003. An Efficient Discrete Fracture Model Applicable for General Purpose Reservoir Simulators. Paper presented at the SPE Reservoir Simulation Symposium, Houston, Texas, USA, 3–5 February. SPE-79699-MS. <https://doi.org/10.2118/79699-MS>.
- Klimczak, C., Schultz, R. A., Parashar, R. et al. 2010. Cubic Law with Aperture-Length Correlation: Implications for Network Scale Fluid Flow. *Hydrogeol J* **18** (4): 851–862. <https://doi.org/10.1007/s10040-009-0572-6>.

- Li, L. and Lee, S. H. 2008. Efficient Field-Scale Simulation of Black Oil in a Naturally Fractured Reservoir Through Discrete Fracture Networks and Homogenized Media. *SPE Res Eval & Eng* **11** (4): 750–758. SPE-103901-PA. <https://doi.org/10.2118/103901-PA>.
- Lie, K.-A., Krogstad, S., Ligaarden, I. S. et al. 2012. Open-Source MATLAB Implementation of Consistent Discretisations on Complex Grids. *Computat Geosci* **16** (2): 297–322. <https://doi.org/10.1007/s10596-011-9244-4>.
- Liu, M. and Grana, D. 2019. Seismic History Matching in the Low-Dimensional Model and Data Space Using Deep Convolutional Auto-Encoder. *SEG Technical Program Expanded Abstracts 2019* 3324–3328. <https://doi.org/10.1190/segam2019-3216322.1>.
- Lu, L. and Zhang, D. 2015. Assisted History Matching for Fractured Reservoirs by Use of Hough-Transform-Based Parameterization. *SPE J.* **20** (5): 942–961. SPE-176024-PA. <https://doi.org/10.2118/176024-PA>.
- Ma, X., Al-Harbi, M., Datta-Gupta, A. et al. 2008. An Efficient Two-Stage Sampling Method for Uncertainty Quantification in History Matching Geological Models. *SPE J.* **13** (1): 77–87. SPE-102476-PA. <https://doi.org/10.2118/102476-PA>.
- Ma, X., Zhang, K., Yao, C. et al. 2020. Multiscale-Network Structure Inversion of Fractured Media Based on a Hierarchical-Parameterization and Data-Driven Evolutionary-Optimization Method. *SPE J.* **25** (5): 2729–2748. SPE-201237-PA. <https://doi.org/10.2118/201237-PA>.
- Menshawy, A. 2018. *Deep Learning by Example: A Hands-On Guide To Implementing Advanced Machine Learning Algorithms and Neural Networks*. Birmingham, UK: Packt Publishing Ltd.
- Moinfar, A., Varavei, A., Sepehrnoori, K. et al. 2013. 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>.
- Narr, W., Schechter, D. S., and Thompson, L. B. 2006. *Naturally Fractured Reservoir Characterization*. Richardson, Texas, USA: Society of Petroleum Engineers.
- Ng, A. 2011. Sparse Autoencoder. CS294A Unsupervised Deep Learning Lecture Notes, Stanford University, Stanford, California, USA.
- Oliver, D. S., and Chen, Y. 2011. Recent Progress on Reservoir History Matching: A Review. *Computational Geosciences* **15** (1): 185–221. 10.1007/s10596-010-9194-2.
- Oliver, D. S., Cunha, L. B., and Reynolds, A. C. 1997. Markov Chain Monte Carlo Methods for Conditioning a Permeability Field To Pressure Data. *Math Geol* **29** (1): 61–91. <https://doi.org/10.1007/bf02769620>.
- Ping, J. and Zhang, D. 2013. History Matching of Fracture Distributions by Ensemble Kalman Filter Combined with Vector Based Level Set Parameterization. *J Pet Sci Eng* **108** (August): 288–303. <https://doi.org/10.1016/j.petrol.2013.04.018>.
- Robert, C. and Casella, G. 2013. *Monte Carlo Statistical Methods*. New York City, New York, USA: Springer Texts in Statistics, Springer Science+ Business Media.
- Romero, C. and Carter, J. 2001. Using Genetic Algorithms for Reservoir Characterisation. *J Pet Sci Eng* **31** (2–4): 113–123. [https://doi.org/10.1016/S0920-4105\(01\)00124-3](https://doi.org/10.1016/S0920-4105(01)00124-3).
- Sarma, P., Durlofsky, L. J., Aziz, K. et al. 2007. A New Approach To Automatic History Matching Using Kernel PCA. Paper presented at the SPE Reservoir Simulation Symposium, Houston, Texas, USA, 26–28 February. SPE-106176-MS. <https://doi.org/10.2118/106176-MS>.
- Schulze-Riegert, R., Axmann, J., Haase, O. et al. 2002. Evolutionary Algorithms Applied To History Matching of Complex Reservoirs. *SPE Res Eval & Eng* **5** (2): 163–173. SPE-77301-PA. <https://doi.org/10.2118/77301-PA>.
- Shah, S., Møyner, O., Tene, M. et al. 2016. The Multiscale Restriction Smoothed Basis Method for Fractured Porous Media (F-MsRSB). *J Comput Phys* **318** (1 August): 36–57. <https://doi.org/10.1016/j.jcp.2016.05.001>.
- Warren, J. E. and Root, P. J. 1963. The Behavior of Naturally Fractured Reservoirs. *SPE J.* **3** (3): 245–255. SPE-426-PA. <https://doi.org/10.2118/426-PA>.
- Witherspoon, P. A., Wang, J. S., Iwai, K. et al. 1980. Validity of Cubic Law for Fluid Flow in a Deformable Rock Fracture. *Water Resour Res* **16** (6): 1016–1024. <https://doi.org/10.1029/WR016i006p01016>.
- Xia, Y., ZhaoQin, H., June, Y. et al. 2018. An Efficient Numerical Hybrid Model for Multiphase Flow in Deformable Fractured-Shale Reservoirs. *SPE J.* **23** (4): 1412–1437. SPE-191122-PA. <https://doi.org/10.2118/191122-PA>.
- Zadhesive, J., Jalali, S. M. E., and Ramezanzadeh, A. 2014. Estimation of Joint Trace Length Probability Distribution Function in Igneous, Sedimentary, and Metamorphic Rocks. *Arab J Geosci* **7** (6): 2353–2361. <https://doi.org/10.1007/s12517-013-0861-1>.
- Zhang, K., Ma, X., Li, Y. et al. 2018. Parameter Prediction of Hydraulic Fracture for Tight Reservoir Based on Micro-Seismic and History Matching. *Fractals* **26** (2): 1840009. <https://doi.org/10.1142/s0218348x18400091>.
- Zhang, K., Ranran, L., and Wensheng, Z. 2014. Multi-Parameter Gradient-Free Automatic History Matching Method. *Journal of China University of Petroleum (Edition of Natural Science)* **38** (5): 109–115. <https://doi.org/10.3969/j.issn.1673-5005.2014.05.015>.
- Zhang, K., Zhang, X., Zhang, L. et al. 2017. Assisted History Matching for the Inversion of Fractures Based on Discrete Fracture-Matrix Model with Different Combinations of Inversion Parameters. *Computat Geosci* **21** (5–6): 1365–1383. <https://doi.org/10.1007/s10596-017-9690-8>.
- Zhang, L., Cui, C., Ma, X. et al. 2019a. A Fractal Discrete Fracture Network Model for History Matching of Naturally Fractured Reservoirs. *Fractals* **27** (1): 1940008. <https://doi.org/10.1142/s0218348x19400085>.
- Zhang, L., Wang, S., Zhang, K. et al. 2019b. Cooperative Artificial Bee Colony Algorithm with Multiple Populations for Interval Multiobjective Optimization Problems. *IEEE Trans Fuzzy Syst* **27** (5): 1052–1065. <https://doi.org/10.1109/tfuzz.2018.2872125>.
- Zhao, H., Xu, L., Guo, Z. et al. 2020. Flow-Path Tracking Strategy in a Data-Driven Interwell Numerical Simulation Model for Waterflooding History Matching and Performance Prediction with Infill Wells. *SPE J.* **25** (2): 1007–1025. SPE-199361-PA. <https://doi.org/10.2118/199361-PA>.