Correlation-driven direct sampling method for geostatistical simulation and training image evaluation

Chen Zuo, Zhibin Pan,[*] Zhaoqi Gao, and Jinghuai Gao

School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China

(Received 16 October 2018; revised manuscript received 29 April 2019; published 29 May 2019)

Multiple-point geostatistics (MPS) is a competitive algorithm that produces a set of geologically realistic models. Viewing a training image (TI) as a prior model, MPS extracts patterns from the TI and reproduces patterns which are compatible with the hard data (HD). However, two challenges within the MPS framework are the geologically complex simulation and the TI evaluation. With the objective to achieve a high-quality simulation, we explore a way to address these two issues. First, correlation-driven direct sampling (CDS) is proposed to realize geostatistical simulation. We introduce the correlation-driven distance as a measure of similarity between two patterns. The weights in our distance measurement are derived by the concepts of the ellipse, correlation coefficient, Gaussian function, and affine transformation. Second, we fulfill TI evaluation on the basis of the consistency between TI and HD. Inspired by CDS, the minimum correlation-driven distance (MCD) is proposed to improve the evaluation accuracy. We suggest a conditioning pattern extraction history strategy to speed up the evaluation program. Third, the local consistency is presented to address nonstationarity. The program automatically divides the simulation domain into several subareas. A two-dimensional (2D) channelized reservoir image and a three-dimensional (3D) rock image are used to validate our proposed method. In comparison with previous methods, CDS yields better simulation quality. The further applications include a set of 2D TI evaluations and a 3D simulation domain segmentation. MCD exhibits sensible evaluation accuracy, excellent computational efficiency, and the ability to deal with nonstationarity.

DOI: 10.1103/PhysRevE.99.053310

I. INTRODUCTION

Models of porous media are basic materials in a wide range of geological applications. Prior to carrying out a spatial analysis, it is necessary to obtain realistic models which describe the geological phenomenon of interest. Aiming at creating a set of accurate models, the multiple-point geostatistics (MPS) algorithm is presented. The core idea of MPS is to extract patterns from a training image (TI) and reproduce patterns which are compatible with local hard data (HD) in the simulation domain. The first MPS program, extended normal equation simulation (ENESIM), was suggested in 1993 [1]. There are three main innovations in this program. (1) The concept of multiple-point statistics becomes a valuable alternative to two-point statistics, such as covariance or the variogram [2,3], (2) the program views a training image (TI) as an explicitly prior model to express the geological phenomenon, (3) the use of nonparameter statistics. However, ENESIM is not frequently used in practice due to its high computational burden. In 2002, Strebelle proposed a prominent MPS framework that is referred to as single normal equation simulation (SNESIM) [4]. With the purpose of tackling the speed issue, a dynamic data structure called a search tree is applied to store patterns. In the simulation procedure, SNESIM checks the search tree to find patterns which agree with HD. The value of an unknown point is determined by the occurrences of compatible patterns. In addition, a multigrid strategy is reported to observe the TI at different scales. Consequently, SNESIM significantly improves the computational efficiency as well as simulation quality. Because points are separately simulated, SNESIM is classified as a point-based method.

Since SNESIM, the point-based MPS program has attracted considerable attention. Strebelle and Cavelius investigate the speed and memory issues in SNESIM [5]. The main areas of progress contain an intermediary subgrid strategy, templates which preferentially include previously simulated points, and a template size optimization technique. Straubhaar *et al.* designed a program called the improved parallel multiple-point algorithm (IMPALA) [6]. Instead of search trees, the list provides a way to reduce memory demands and realize parallelization. In 2013, IMPALA was further modified [7]. The strengths from the search tree and list are combined. With the objective to speed up simulation, Zuo *et al.* developed the database structure and proposed vector quantization multiple-point statistics (VQMPS) [8]. Using vector quantization as a clustering operation, VQMPS compresses training patterns and creates a set of centroids. A tree-structure codebook is applied to organize centroids and generate a database of small size. A refined database has a positive effect on saving running time. In 2010, Mariethoz *et al.* suggested a framework named direct sampling (DS) [9]. In order to overcome the memory limitation, DS randomly samples the TI to find desired patterns. The searching procedure is repeatedly performed until the similarity between a training pattern and local HD is closer than a predefined threshold. For the categorical variables, DS applies the Hamming distance to compare two patterns. By contrast, Euclidean distance is a

[*] zbpan@mail.xjtu.edu.cn

commonly used criterion for the continuous variables. Based on a variety of applications, a practical guide to perform DS is reported by Meerschman *et al.* [10].

Based on the explanations above, it is clear that the TI and HD are fundamental materials within the MPS framework. On the one hand, the physical properties and spatial characteristics of the geological phenomenon are expressed by the TI. On the other hand, HD are directly sampled from reality. The simulation programs not only reproduce the information in the TI but also honor HD. Therefore, the need for TI evaluation and selection arises. Prior to carrying out a geostatistical modeling program, a necessary step is to select a suitable TI from a set of candidates. If the TI is compatible with HD, it is easy for MPS to create geologically realistic models. Conversely, the inconsistency between TI and HD is challenging to the geostatistical simulation.

Inspired by the DS framework, Pérez *et al.* propose a method to verify the high-order consistency of TI with HD [11]. At first, a relative compatibility measurement is suggested. Next, they develop an absolute compatibility measurement. The match pattern proportion (MP) is presented as an indicator. In 2017, Feng *et al.* introduced a TI evaluation and selection method called minimum data event distance (MDevD) [12]. With the aim of improving accuracy, a weighted distance is applied to compare two patterns. Then, the method analyzes the mean and variance of a collection of MDevDs. The most suitable TI is chosen according to these statistical properties.

As mentioned above, the distance between two patterns plays a central role in point-based MPS methods and two TI evaluation methods. These methods either use a uniform distance or assign a weight to each point in accordance with its relative locations. For example, the known points in VQMPS, original DS, and MP have the same influence in the similarity calculation. By contrast, SNESIM and IMPALA apply a pruning strategy. The furthest point will be discarded if there is no compatible pattern in the database. However, these designs are not always sensible because of anisotropy. Spatial characteristics along different directions may not be the same. It is reasonable to pay attention to the directions involving intensive connection and correlation. In addition, the previous TI evaluation methods are time demanding due to an exhaustive extraction strategy. This means that the programs have to perform a searching procedure for each conditioning pattern. Moreover, MP and MDevD only compute the global consistency between TI and HD. The nonstationary simulation domain is not addressed.

In Ref. [8], our group modified the database structure and saved simulation time. In this paper, we focus on the accuracy and explore a way to implement geostatistical simulation as well as TI evaluation. First, a correlation-driven direct sampling (CDS) is proposed to realize high-quality simulation. We present a correlation-driven distance as a generalization of previous distance measurement. The concepts of the ellipse, correlation coefficient, Gaussian function, and affine transformation are applied to conduct weight computation. Second, we fulfill TI evaluation on the basis of consistency between TI and HD. Inspired by CDS, the minimum correlation-driven distance (MCD) is proposed to improve evaluation accuracy. A conditioning pattern extraction history strategy is introduced to improve computational efficiency at the expense of memory consumption. Third, the local consistency is suggested to deal with nonstationarity. Our method automatically divides the simulation domain into several areas.

An application of two-dimensional (2D) channelized reservoir simulation is conducted to validate our modeling method. Compared with previous DS, CDS yields better simulation quality. Next, we carry out a three-dimensional (3D) rock simulation from a 2D slice. The spatial characteristics of a simulation realization agree well with the target. Further applications include a set of 2D TI evaluations and a 3D rock segmentation. MCD exhibits sensible evaluation accuracy, excellent computational efficiency, and the ability to solve nonstationarity.

The rest of this paper is organized as follows. Section II provides related work such as detailed procedures within DS, MP, and MDevD. Our proposed algorithm is presented in Sec. III. Section IV shows some applications. Finally, conclusions are drawn in Sec. V.

II. RELATED WORK

A. Direct sampling framework

As a sequential simulation framework, the key idea of DS is to directly sample the TI in order to find a compatible pattern. A pattern is also referred to as a data event in MPS literature. There are two basic steps within the DS framework: (1) the conditioning pattern extraction, and (2) a searching procedure. Figure 1 explains a DS point simulation.

As the first step, the DS program visits an unknown point p_i and creates a conditioning pattern. For example, point p_1 in Fig. 1(a) was checked. In this paper, let p denote a point in the simulation domain and q denote a point in the TI. In the neighborhood of p_1, DS applied a flexible template to collect conditioning data. Viewing p_1 as the center at the template, N informed points are found in a close-to-far fashion. In this case, we set $N = 3$ and gathered points $p_{1,1}$, $p_{1,2}$, and $p_{1,3}$. These three points constituted a conditioning pattern $P_N(p_1) = \{Z(p_{1,1}), Z(p_{1,2}), Z(p_{1,3})\} = \{0, 0, 1\}$, where $Z(p_{1,n})$ denotes the value of point $p_{1,n}$. Moreover, the relative coordinates of these three points create a lag vector. Suppose that the x axis extends to the right and the y axis extends upwards. The lag vector of p_1 was $L_N(p_1) = \{(x_1, y_1), (x_2, y_2), (x_3, y_3)\} = \{(0, 1), (1, 1), (-1, -2)\}$, where (x_n, y_n) denotes the relative coordinates of the nth point in the template.

After the conditioning pattern extraction, a searching procedure begins. DS randomly visits a point in the TI. As shown in Fig. 1(b), point q_1 was tested. A training pattern $P_N(q_1) = \{1, 0, 1\}$ was created on the basis of $L_N(p_1)$. In the original DS, the Hamming distance is used to compare two patterns. The Hamming distance between $P_N(p_1)$ and $P_N(q_1)$ is defined as

$$D_H[P_N(p_1), P_N(q_1)] = \frac{1}{N}\sum_{n=1}^{N} Z_n = \frac{1}{3}(1+0+0) \approx 0.33, \tag{1}$$

where

$$Z_n = \begin{cases} 0 & \text{if } Z(p_{1,n}) = Z(q_{1,n}) \\ 1 & \text{otherwise} \end{cases}. \tag{2}$$

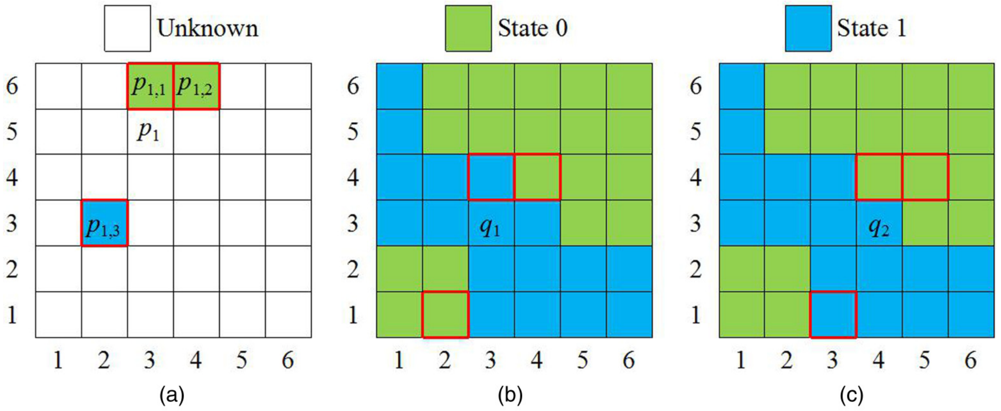


FIG. 1. A point simulation within the DS framework. (a) A conditioning pattern in the simulation domain of size 6 × 6. (b,c) Two training patterns in a TI of size 6 × 6.

It is clear that $P_N(q_1)$ is not compatible with $P_N(p_1)$. The DS program has to repeatedly sample the TI until it finds a compatible pattern. As Fig. 1(c) displays, point q_2 was found. The pattern $P_N(q_2)$ was in complete agreement with $P_N(p_1)$. Consequently, point q_2 was determined as the result of this point simulation. The value of point q_2 was pasted to point p_1.

Based on the previous description, two parameters are introduced to enhance DS. As the first parameter, the threshold t is used to define compatible patterns. With the purpose of extracting high-order statistics, the algorithm increases the value of N. It is time demanding to find a training pattern that is fully compatible with the conditioning pattern. Therefore, the DS program accepts a pattern whose distance with the conditioning pattern is smaller than t. The second parameter is the fraction f, which controls the search scope in the TI. For example, $f = 0.5$ indicates that DS only checks 50% of the points in the TI. If a compatible pattern is not found, the training pattern with the minimum distance becomes the result. The DS procedure and applications are elaborated upon in Ref. [9].

In Eq. (1), each point in the template has the same influence on the distance measurement. That is, the weight of each point is fixed and unchangeable. However, a uniform template is not sensible in practice. The key idea of MPS is to predict the value of an unknown point according to the conditioning points. This process can be viewed as a classification task. According to several conditions (known points in a template), the algorithm identifies which categories (facies) an instance (an unknown point or center in the template) belongs to. The correlation between conditions and the instance plays a central role in the classification program. If the conditioning point is strongly correlated with the center, it is helpful for MPS to simulate a point. On the contrary, the points which are less correlated with the estimation point not only have a weak influence on point simulation but also bring a computational burden. Therefore, the simulation method should pay attention to the points with high correlation.

Based on the explanation above, the weighted Hamming distance is used to improve DS. For clarification, we refer to the DS program involving weighted Hamming distance as weighted direct sampling (WDS). It is intuitive that points which are close to the estimation point exhibit strong correlations. Therefore, the relative distance to the estimation point is adopted to calculate weights. Suppose that the TI and the simulation domain are two-dimensional. The size of a template is N and the relative coordinates of the nth point in the template are (x_n, y_n). Let w_n denote the weight of point (x_n, y_n) and δ denote the order of the power function. The weighted Hamming distance is defined as

$$D_W[P_N(p), P_N(q)] = \frac{\sum_{n=1}^{N} w_n Z_n}{\sum_{n=1}^{N} w_n}, \tag{3}$$

where

$$w_n = \left(x_n^2 + y_n^2\right)^{-\delta}. \tag{4}$$

B. TI evaluation methods

Pérez *et al.* designed a method that measures the consistency between TI and HD [11]. Suppose that J training images and a hard data set H are provided. Similar to DS, the first step of TI evaluation is to visit a point in HD and create a conditioning pattern. As Fig. 2 shows, point p_2 was checked and pattern $P_N(p_2)$ was generated. In this case, N was set as 6. Afterward, the approach samples a training image T_j to find compatible patterns. Aiming at storing the result of the searching procedure, an indicator $Y_{i,j}$ is introduced. $Y_{i,j}$ takes the value of 1 if a compatible pattern of $P_N(p_i)$ can be found in T_j, considering the threshold t. Otherwise, the value of $Y_{i,j}$ becomes 0. Let I denote the total number of points in H. It should be noted that the TI evaluation method visits the known points as well as the unknown points. For instance, the value of I was $7 \times 7 = 49$ in Fig. 2. The matching pattern proportion (MP) can be expressed as

$$m_j^{\text{MP}} = \frac{\sum_{i=1}^{I} Y_{i,j}}{I}. \tag{5}$$

It is obvious that the consistency between T_j and H is strong if m_j^{MP} approaches 1. Conversely, T_j does not match H when the value of m_j^{MP} is close to 0.

With the intention of improving evaluation accuracy, Feng *et al.* reported the minimum data event distance (MDevD) [12]. The core idea of MDevD is to use the weighted distance,

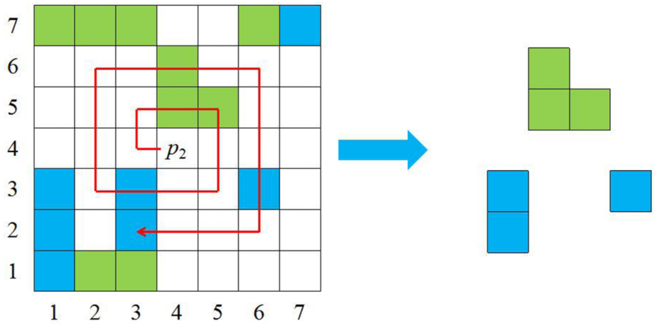


FIG. 2. An HD image of size 7 × 7 and a conditioning pattern.

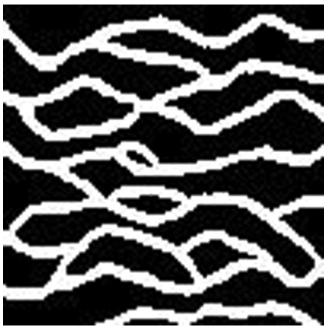


FIG. 3. A channelized reservoir image.

which is introduced in Eqs. (3) and (4). MDevD consists of three main steps. First, the program visits a point p_2 in H and creates a conditioning pattern $P_N(p_2)$. This operation is illustrated in Fig. 2. Second, T_j is exhaustively scanned. Using Eqs. (3) and (4), MDevD finds a training pattern whose distance with $P_N(p_2)$ is the smallest. This distance is called the minimum data event distance $m_j^{\text{MDevD}}(p_2)$. The preceding two procedures are repeatedly performed until each point in H is tested. Third, the program obtains a set of $m_j^{\text{MDevD}}(p_i)$. The statistical properties, such as mean and variance, become indicators. For example, the mean value m_j^{MDevD} of $m_j^{\text{MDevD}}(p_i)$ can be expressed as

$$m_j^{\text{MDevD}} = \frac{1}{I}\sum_{i=1}^{I} m_j^{\text{MDevD}}(p_i). \tag{6}$$

It is clear that the smaller m_j^{MDevD}, the more compatibility exists between T_j and H.

According to the explanations above, the limitations of MP and MDevD can be summarized as follows:

(1) The weights in distance measurement are fixed and inflexible for various TIs. In Eq. (1), each point has the same influence. Equation (4) calculates weights using relative coordinates. However, the anisotropy is an ignored issue. The spatial characteristic of the TI is not considered in the weight computation.

(2) The previous TI evaluation methods are time consuming because of an exhaustive extraction strategy. The program has to perform a searching procedure for each conditioning pattern. This mechanism brings a heavy computational burden.

(3) The nonstationarity is not dealt with. In general, there are several phenomena in a nonstationary simulation domain. A common way to address nonstationarity is to divide the simulation domain into several areas. It is helpful to use several TIs to inspire the simulation program. However, MP and MDevD evaluate the global consistency between TI and HD. In other words, HD are processed as a whole. Candidate TIs are ranked according to their compatibility. This evaluation result is only suitable for the stationary simulation.

III. PROPOSED METHODOLOGY

With the objective to improve simulation accuracy, we explore a way to implement geostatistical modeling and TI evaluation. First, a correlation-driven direct sampling (CDS) is proposed. We introduce the correlation-driven distance as a measure of similarity between two patterns. The previous weight computation method in Eq. (4) becomes a special case of our method. Second, we improve the evaluation accuracy as well as computational efficiency. Inspired by CDS, the minimum correlation-driven distance (MCD) is advised to calculate the consistency between TI and HD. We suggest the conditioning pattern extraction history strategy to save running time at the cost of memory consumption. Third, the local consistency is introduced to solve nonstationarity. On the basis of evaluation results, our program automatically divides the simulation domain into several areas.

A. Principle of the correlation-driven direct sampling

We gradually construct components of the proposed distance measurement, through an ongoing refinement of our formulations. In Eq. (4), weights are calculated based on a power function of a circle. There are three underlying assumptions behind this design. First, the correlation decreases with the increasing distance. Second, the stronger the correlation with the center, the greater weight a point has. Third, correlations have the same tendency along different directions. However, these assumptions are not always true because of anisotropy. As an example, a channelized reservoir image is shown in Fig. 3. This image has been widely used as the TI in the MPS literature [5,6,8,9]. Evidently, channels (white areas) are more connective along the horizontal direction (x axis) than the vertical direction (y axis). In other words, the horizontal direction exhibits more correlation. Thus, it is reasonable to assign more weights along the horizontal direction than another. However, weight computation in Eq. (4) does not satisfy this target.

As a generalization of a circle, the concept of ellipse plays a key role in our proposed weight computation. The shapes and parameters of a circle and an ellipse are shown in Figs. 4(a) and 4(b), respectively. Let a and b denote two shape parameters of the ellipse. Accordingly, the weight formulation shown in Eq. (4) is modified to

$$w_n = \left(\frac{x_n^2}{a^2} + \frac{y_n^2}{b^2}\right)^{-\delta}. \tag{7}$$

In particular, Eq. (7) is equivalent to Eq. (4) when the values of a and b are 1. Based on Eq. (7), our program is capable of assigning greater weights to a certain axis.

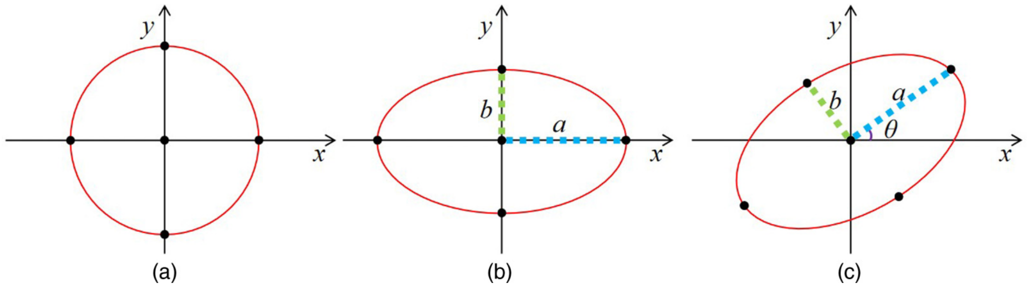


FIG. 4. The shape and parameters of three geometries: (a) a circle; (b) an ellipse; (c) a rotated ellipse.

A necessary step of our method is to determine the parameters of an ellipse. As in the explanations above, the correlation is the central concept in the distance measurement. Therefore, we adopt the correlation coefficient to specify the values of a and b. The correlation coefficient is a numerical measure of correlation between two variables. Let q_3 denote a point under consideration and q_c denote the center in a template. Suppose that relative coordinates of q_3 to q_c are (x_3, y_3). The correlation coefficient $R(q_c, q_3)$ between these two points is defined as

$$R(q_c, q_3) = \frac{\sum_{k=1}^{K} [Z(q_{c,k}) - \overline{Z(q_c)}][Z(q_{3,k}) - \overline{Z(q_3)}]}{\sqrt{\sum_{k=1}^{K} [Z(q_{c,k}) - \overline{Z(q_c)}]^2}\sqrt{\sum_{k=1}^{K} [Z(q_{3,k}) - \overline{Z(q_3)}]^2}}, \tag{8}$$

where $(q_{c,k}, q_{3,k})$ composes the kth point pair in the TI. The function $Z(q_{3,k})$ denotes the value of point $q_{3,k}$ and $\overline{Z(q_3)}$ expresses the mean value of $Z(q_3)$. K is the number of pairs in the TI.

An example in Fig. 5(a) is used to explain how to compute $R(q_3, q_c)$. Suppose that $(x_3, y_3) = (1, -2)$ in this case. The value of K was set as 20. Our program scans the TI and checks each available-point pair. Based on these point pairs, we obtained that $\overline{Z(q_c)} = 0.45$ and $\overline{Z(q_3)} = 0.65$. Then, the correlation coefficient is computed according to Eq. (8). Consequently, $R(q_c, q_3)$ was 0.2423 in this case.

A noticeable phenomenon is that the correlation coefficients are symmetrical with respect to the center of a template. Suppose that the coordinates of point q_4 are $(x_4, y_4) = (-1, 2)$, which are symmetrical with q_3. As Fig. 5(b) shows, the correlation coefficient $R(q_c, q_4)$ is computed via collecting every point pair. The correlation computation method checks the same point pairs as the operation in calculating $R(q_c, q_3)$. Consequently, $R(q_c, q_4)$ is always equal to $R(q_c, q_3)$. This phenomenon helps us to reduce the computational burden in the following paragraphs.

Using Eq. (8), the correlation coefficient between point q_n and the center q_c in the template is calculated. As an example, we compute the correlation coefficient $R(q_c, q_n)$ according to Fig. 3. Suppose that the relative coordinates of q_n are (1, 0), (2, 0), …, (28, 0), respectively. These values represent the correlation function along the horizontal direction in Fig. 3. In a similar manner, we obtain the correlation function along the vertical direction. These two functions are drawn in Fig. 6. There are four phenomena.

(1) These two functions start at 1. This means that a variable is perfectly correlated with itself.

(2) The correlation coefficients decrease with increasing distances because the points become less and less correlated.

(3) The functions start to fluctuate around zero when the distance is larger than a certain value. Moreover, the fluctuation of the vertical direction is more intensive. The reason for this phenomenon is the geologic cyclicity [13]. The repetitive and cyclic variations of channels along the vertical direction have an impact on the behavior of correlation functions.

(4) The anisotropy is intensive. It is clear that there is a considerable difference between two correlation functions. The correlation function rapidly decreases along the vertical direction while the correlation function varies smoothly along the horizontal direction.

On the basis of these phenomena, parameters a and b in Eq. (7) are computed as follows. Because DS collects the

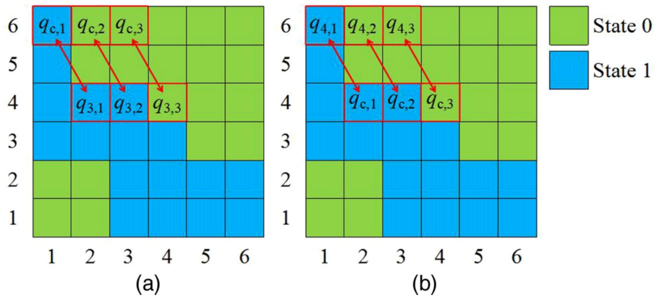


FIG. 5. The correlation coefficient computation: (a) the pairs (p_c, p_3); (b) the pairs (p_c, p_4).

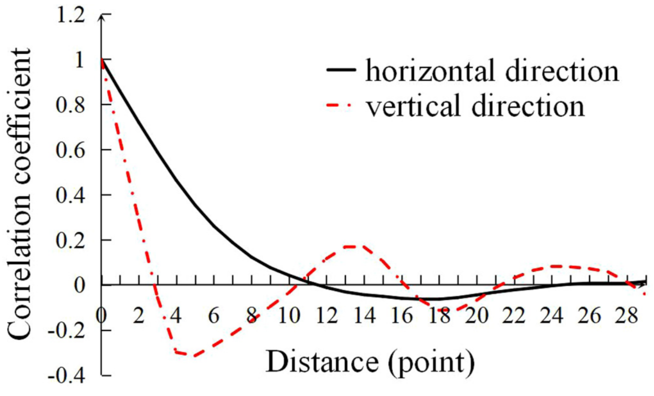


FIG. 6. Correlation coefficient functions along the horizontal direction and the vertical direction in Fig. 3.

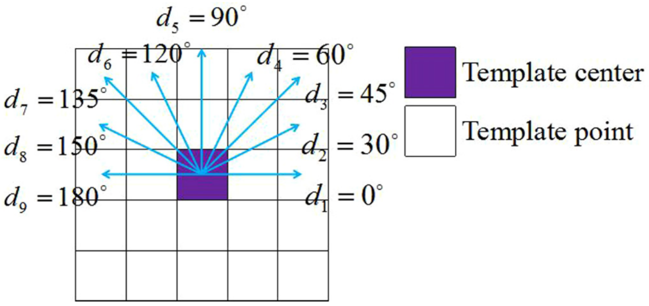


FIG. 7. Candidate directions of a rotated ellipse.

conditioning points in a close-to-far fashion, we only consider points which are close to the estimation point. Starting from 0, the value of a is set as the distance when the correlation coefficient function along the horizontal direction is first smaller than zero. In a similar manner, we derive the value of b. As an example in Fig. 6, the values of a and b were 12 and 3, respectively.

An important property of correlation coefficients is that their values are always in the range of $[-1, 1]$. Considering the practical significance, the desired range of weights is $[0, 1]$. However, the power function used in Eq. (7) cannot guarantee each weight is inside this range. It is not proper to approximate correlation coefficients by means of a power function. Therefore, we apply the Gaussian function as a valuable alternative. Considering its characteristics, the Gaussian function is widely used in the context of computer science and statistics. Thus, we modify Eq. (7) to

$$w_n = e^{-(\frac{x_n^2}{a^2}+\frac{y_n^2}{b^2})}. \tag{9}$$

In this paper, we refer to the direction where correlations decrease on the lowest slope as the major correlation direction. It is intuitive that the major correlation direction of Fig. 3 is the horizontal direction. Thus, the major axis of the ellipse is located along the x axis. However, one defect of Eq. (9) is that the major axis is along either the x axis or y axis. In practice, there are a large number of TIs whose major correlation directions are not limited to the horizon or vertical. With the purpose of generalizing our method, the affine transformation is used to rotate the ellipse. A rotated ellipse is shown in Fig. 4(c). Let θ denote an angle between the major axis and the x axis. Equation (9) is modified to

$$w_n = e^{-[\frac{(x_n\cos\theta+y_n\sin\theta)^2}{a^2}+\frac{(-x_n\sin\theta+y_n\cos\theta)^2}{b^2}]}. \tag{10}$$

In particular, it is digital images that we process. This means that the coordinates (x_n, y_n) are always integers. It is time demanding to assign redundant values to θ. As Fig. 7 displays, we employed nine values of θ varying from 0° to 180°. Along the dth direction, our program calculates a correlation function. When the value of the correlation function is first smaller than zero, the current length r_d is recorded. After checking each direction, our method obtains a set of r_d. The maximum value $r_{\max}$ is assigned to a and its corresponding direction determines the value of θ. Afterward, b becomes the only unknown item in Eq. (10). As Fig. 8 depicts, the values of r_x and r_y help us. r_x and r_y are the distances r_d along the x axis and y axis, respectively. Solving one of two of the following formulations can obtain the value of b:

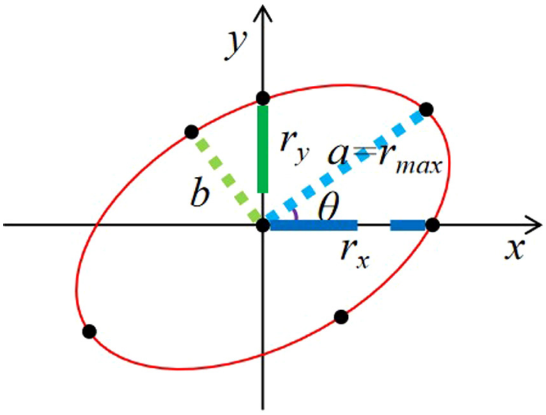


FIG. 8. Parameters of a rotated ellipse.

$$\frac{(r_x\cos 0^\circ + 0\sin 0^\circ)^2}{a^2} + \frac{(-r_x\sin 0^\circ + 0\cos 0^\circ)^2}{b^2} = 1, \tag{11}$$

$$\frac{(0\cos 90^\circ + r_y\sin 90^\circ)^2}{a^2} + \frac{(0\sin 90^\circ + r_y\cos 90^\circ)^2}{b^2} = 1. \tag{12}$$

In the preceding paragraphs, we apply the concepts of the ellipse, correlation coefficient, Gaussian function, and affine transformation to develop the weight computation. In the following, the last modification in our proposed distance is presented. As explained in Fig. 5, correlation coefficients are symmetrical. Accordingly, we partition the template into four quadrants. This operation is shown in Fig. 9. One ellipse parameter set (a_1, b_1, θ_1) is applied to calculate the weights of points in the first and third quadrants. The weights in the second and fourth quadrants are derived by another parameter set (a_2, b_2, θ_2). Consequently, the correlation-driven distance and corresponding weight computation can be expressed as

$$D_W[P_n(p), P_n(q)] = \frac{\sum_{n=1}^{N} w_n Z_n}{\sum_{n=1}^{N} w_n}, \tag{13}$$

$$w_n = w(x_n, y_n) = \begin{cases} e^{-[\frac{(x_n\cos\theta_1+y_n\sin\theta_1)^2}{a_1^2}+\frac{(-x_n\sin\theta_1+y_n\cos\theta_1)^2}{b_1^2}]} & \text{if } (x_n, y_n) \in \text{the first or third quadrant} \\ e^{-[\frac{(x_n\cos\theta_2+y_n\sin\theta_2)^2}{a_2^2}+\frac{(-x_n\sin\theta_2+y_n\cos\theta_2)^2}{b_2^2}]} & \text{otherwise} \end{cases}. \tag{14}$$

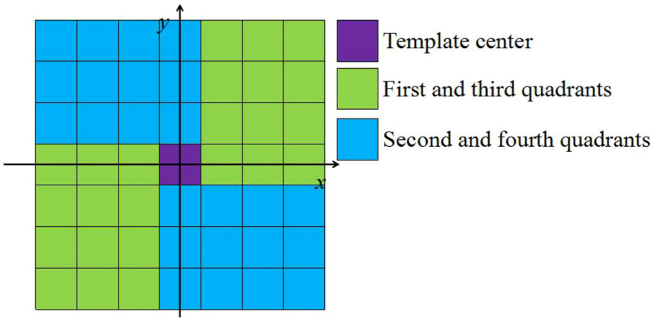


FIG. 9. A template is classified into four quadrants.

In summary, our developments consist of five aspects. First, the concept of ellipse becomes a generalization of the circle in the previous weight computation. Second, we adopt the correlation coefficient to determine parameters of the ellipse. Third, the Gaussian function replaces the power function in the previous method. Fourth, the ellipse is rotated via affine transformation. Fifth, we establish two ellipses to derive weights because of the symmetry. The detailed steps to compute the parameters (a_1, b_1, θ_1) are listed below.

(1) Input a TI.

(2) Draw the correlation coefficient function in five directions d_1, d_2, d_3, d_4, d_5. We refer to the direction where correlations decrease on the lowest slope as the major correlation direction d_{major}.

(3) The angle corresponding to d_{major} determines the value of θ_1. The value of a_1 is the distance between the center point q_c and a point q_n, whose correlation coefficient $R(q_c, q_n)$ is first smaller than zero along d_{major}.

(4) The value of b_1 is calculated by Eqs. (11) or (12).

(5) The weights in the first and third quadrants are computed based on Eq. (14).

In a similar manner, the values of (a_2, b_2, θ_2) are specified. Thus, we obtained the weight of each point in the template. Because the correlation coefficient plays a central role in the weight computation, we refer to the proposed distance measurement as the correlation-driven distance. The key advantage is that our distance is data driven. Weights are computed based on the intrinsic characteristic of the TI. Each TI has specific weights. As a contrast, the previous weight computations are fixed and inflexible. Therefore, the correlation-driven distance becomes a better similarity criterion to compare two patterns.

Because we focus on developing distance measurement, the proposed correlation-driven distance can be straightforwardly integrated into the DS framework. We present the correlation-driven direct sampling (CDS) to realize geostatistical simulation. The detailed steps of CDS are in the following.

(1) Input a TI to express the geological phenomenon of interest.

(2) Apply the correlation-driven distance to calculate weights.

(3) Visit an unknown point in the simulation domain. A conditioning pattern is created according to the surrounding N known points.

(4) Sample the TI to find a compatible training pattern. Considering the threshold t, the correlation-driven distance is applied to compare a conditioning pattern and a training pattern. The fraction f controls the search scope. The center of a compatible pattern is assigned to the unknown point.

(5) Repeat steps 3 and 4 until there is no uninformed point in the simulation domain. CDS outputs the domain as the simulation realization.

B. Minimum correlation-driven distance

The correlation-driven distance is a powerful tool to compare two patterns. Inspired by CDS and MDevD, we introduce a method called minimum correlation-driven distance (MCD) to realize TI evaluation and selection. Our proposed distance measurement is straightforwardly integrated into the MDevD framework. Although the evaluation accuracy is improved by the correlation-driven distance, the speed issue in MP and MDevD is challenging. For each conditioning pattern, these two methods perform an exhaustive search in the TI to find a compatible pattern. Aiming at accelerating TI evaluation, we propose a strategy named the conditioning pattern extraction history. The previous searching results provide a valuable guide to the subsequent pattern extraction.

The principle of conditioning pattern extraction history strategy is shown in Fig. 10. Prior to extracting a pattern, we assign an index counter to each known point in the simulation domain. As Fig. 10(a) shows, numbers appearing in the hard data point are their indices. Next, the program performs a conditioning pattern extraction procedure. A point in the simulation domain is visited. For example, we checked point p_2 in Fig. 10(a). The hard data points are collected in a close-to-far fashion. Assuming that N was 6 in this case, a conditioning pattern $P_N(p_2) = \{0, 0, 1, 1, 0, 1\}$ was produced. In addition, we created an index vector $C_N(p_2) = \{6, 7, 8, 10, 11, 13\}$. These indices are sorted in ascending order. With the aim of finding a compatible pattern, the TI evaluation program conducts a searching procedure in the TI. As Fig. 10(c) shows, a fully compatible pattern $P_N(q_5)$ was found. The full compatibility means the distance between two patterns is zero.

Afterward, the program visits another point in the simulation domain. As shown in Fig. 10(d), the point p_3 was checked and the conditioning pattern $P_N(p_3)$ was created. Meanwhile, the method produces an index vector $C_N(p_3) = \{6, 7, 8, 10, 11, 13\}$. A worthwhile phenomenon is that $C_N(p_2)$ is the same as $C_N(p_3)$. These six hard data points constitute not only $P_N(p_2)$ but also $P_N(p_3)$. Because the distance between $P_N(p_2)$ and $P_N(q_5)$ is zero, $P_N(p_3)$ is in full agreement with $P_N(q_5)$. It is unnecessary to carry out a searching procedure to find a compatible pattern of $P_N(p_3)$. Furthermore, it should be noted that full compatibility plays an important role. The reason is that each Z_n in Eqs. (1), (3), and (13) is equal to 0 when $P_N(p_2)$ is fully compatible with $P_N(q_5)$. Regardless of the weight criterion, $P_N(p_3)$ is always the same as $P_N(q_5)$.

Based on the illustration mentioned above, the conditioning pattern extraction history is suggested to speed up the TI evaluation. The core idea is to store the fully compatible

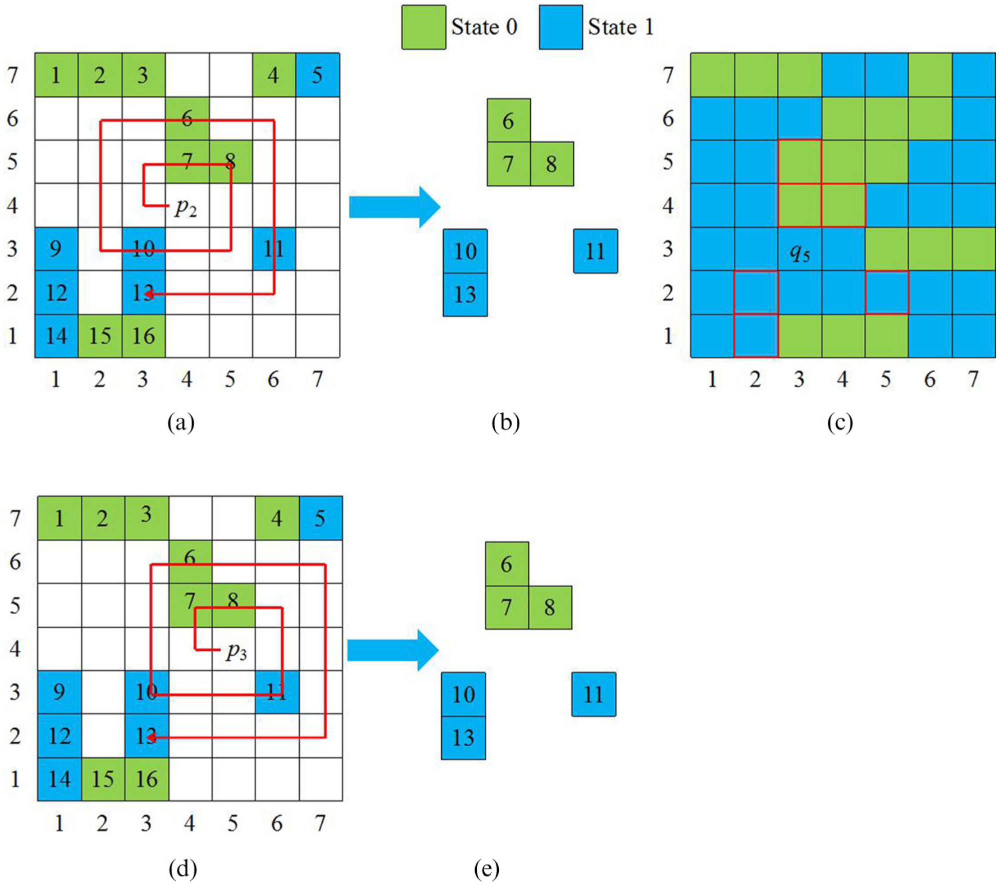


FIG. 10. The principle of the conditioning pattern extraction history strategy: (a) the first point visited by the program; (b) the first conditioning pattern; (c) the searching result; (d) the second point visited by the program; (e) the second conditioning pattern.

patterns in computer memory. After creating a conditioning pattern, the program checks the memory. If the conditioning pattern exists, it is confirmed that this pattern has a compatible pattern in the TI. It is not necessary to perform a searching procedure. Otherwise, a searching procedure in the TI is conducted.

With the objective to evaluate and select the TI, we propose the minimum correlation-driven distance (MCD). The strengths from the correlation-driven distance and the conditioning pattern extraction history are combined together. These two developments have a positive effect on evaluation accuracy as well as computational efficiency. Suppose that a hard data image H and a set of training images T_j are provided. I and J represent the total number of points in H and the number of candidate TIs, respectively. Let $m_j^{\mathrm{MCD}}(p_i)$ denote the minimum correlation-driven distance between a conditioning pattern $P_N(p_i)$ and training patterns in T_j. The detailed steps of MCD are as follows.

(1) Carry out the correlation-driven distance to compute the weights on the basis of T_j.

(2) Assign an index to each informed point in H.

(3) Construct a list to store index vectors.

(4) Visit a point p_i in H and collect N known points in a close-to-far fashion. These N points constitute a conditioning pattern $P_N(p_i) = \{Z(p_{i,1}),\ Z(p_{i,2}),\ \ldots,\ Z(p_{i,N})\}$; a lag vector $L_N(p_i) = \{(x_1, y_1),\ (x_2, y_2),\ \ldots,\ (x_N, y_N)\}$; and an index vector $C_N(p_i) = \{c_{i,1},\ c_{i,2},\ \ldots,\ c_{i,N}\}$, where $c_{i,n}$ is an operation that finds the index of point $p_{i,n}$.

(5) Check the list to find $C_N(p_i)$. If $C_N(p_i)$ exists in the list, the program confirms that $m_j^{\mathrm{MCD}}(p_i) = 0$ and goes to step 7; otherwise, the program performs step 6.

(6) Repeatedly sample TI_j until our program finds a pattern whose correlation-driven distance with $P_N(p_i)$ is the smallest. This operation can be expressed as

$$m_j^{\mathrm{MCD}}(p_i) = \underset{q \in TI_j}{\arg}\ \min D_w[P_N(p_i),\ P_N(q)]. \tag{15}$$

If the value of $m_j^{\mathrm{MCD}}(p_i)$ is 0, $C_N(p_i)$ is added to the list.

(7) Record $m_j^{\mathrm{MCD}}(p_i)$.

(8) Repeatedly perform steps 4–7 until each point in the simulation domain is visited. The program yields a set of MCDs. The mean intensity of these MCDs expresses the consistency between T_j and H. The mean operation is defined as

$$m_j^{\mathrm{MCD}} = m^{\mathrm{MCD}}(H, T_j) = \frac{1}{I}\sum_{i=1}^{I} m_j^{\mathrm{MCD}}(p_i). \tag{16}$$

Similar to Eq. (6), it is clear that a small value of $m^{\mathrm{MCD}}(\mathrm{HD}, \mathrm{TI}_j)$ indicates a high consistency between H and T_j. In addition, our proposed method outputs an MCD image of the same size as H. The value of $m_j^{\mathrm{MCD}}(p_i)$ becomes the intensity of point p_i in the MCD image. It should be noticed that TI evaluation methods focus on calculating the consistency between TI and HD. The value of p_i in the simulation domain does not change. The purpose of the MCD image is to help us to observe the spatial characteristics of the consistency.

C. Simulation domain segmentation

Nonstationarity is a common scenario in the geostatistical simulation. In general, a nonstationary simulation means there are a variety of geological phenomena in one simulation domain. A widely used strategy to handle nonstationarity is to carry out the simulation domain segmentation. The simulation domain is partitioned into several stationary subareas. Each area expresses a certain spatial characteristic. In such case, it is helpful to inspire the simulation program by a set of TIs. However, the previous TI evaluation programs compute the global consistency between TI and HD. That is, the simulation domain is processed as a whole. Candidate TIs are ranked according to their scores. This result does not have a positive effect on solving the nonstationarity.

In this section, we use minimum correlation-driven distance to address nonstationarity. The concept of local consistency is introduced to complete the simulation domain segmentation. Suppose that there are J TIs and a hard data image H. At first, we perform MCD with respect to each TI. The calculation results are saved in computer memory. Afterward, the simulation domain is uniformly partitioned into S subareas. Suppose that the hard data image (HD) is two-dimensional. The x axis is divided into S_x equal-length bins according to the x coordinate. In a similar manner, we partition the y axis into S_y bins. As a result, the two-dimensional HD is divided $S_x \times S_y = S$ subareas.

For a point p_i in the sth subarea H_s of hard data H, our method retrieves $m_j^{\mathrm{MCD}}(p_i)$ from memory. After checking each point in H_s, we obtain a set of distances. Accordingly, the mean of these distances represents the local consistency between T_j and H_s. The mean distance is defined as

$$m_{s,j}^{\mathrm{MCD}} = m^{\mathrm{MCD}}(H_s, \mathrm{TI}_j) = \frac{1}{I_s}\sum_{i=1}^{I_s} m_j^{\mathrm{MCD}}(p_i), \qquad (17)$$

where p_i is the ith point in H_s and I_s is the number of points in H_s.

Carrying out the preceding procedure with respect to each TI, the program obtains a distance vector $(m_{s,1}^{\mathrm{MCD}}, m_{s,2}^{\mathrm{MCD}}, ..., m_{s,J}^{\mathrm{MCD}})$. Suppose that $m_{s,\min}^{\mathrm{MCD}}$ is the minimum item in the vector. It is intuitive that the image corresponding to $m_{s,\min}^{\mathrm{MCD}}$ is the most suitable TI for H_s. Moreover, it is possible that the number of minimum items is more than 1. For this situation, the corresponding TIs are all proper for H_s.

The procedure described above is repeatedly conducted until each subarea is dealt with. Therefore, we acquire the local consistency between each subarea and T_j. The program assigns a specified TI for each subarea in accordance with the value of $m_{s,j}^{\mathrm{MCD}}$. As a result, the nonstationary simulation domain is automatically partitioned into S subareas. Based on this segmentation, the simulation program can reproduce various geological phenomena in one simulation domain.

TABLE I. The parameters of CDS in the 2D channel simulation.

Parameters	Values
a_1	12
b_1	3
θ_1	0
a_2	12
b_2	3
θ_2	180

IV. APPLICATIONS OF THE METHODOLOGY

A. 2D application with a binary stationary training image

As the first application, a two-dimensional channelized reservoir image of size 101 × 101 is applied to validate our method. As Fig. 3 shows, many MPS programs use this image as a TI [5,6,8,9]. Our method was implemented in JAVA and performed on a 3.2-GHz Windows computer. In this application, we adopted an unconditional simulation strategy whose sampling rate is 0.5%. Moreover, the simulation domain is of the same size as the TI. Thus, a simulation domain contains $101 \times 101 \times 0.5\% \approx 51$ known points before the simulation.

As the first step, the correlation-driven distance is applied to derive weights from the intrinsic characteristic of the TI. The parameters in Eq. (14) are shown in Table I. Based on the proposed distance, CDS is carried out. With the goal of ensuring high quality, we set $N = 30$, $t = 0$, and $f = 1$ within the DS framework. In other words, we conducted an exhaustive search. The TI is repeatedly sampled until the program finds a fully compatible pattern. If no such pattern is found, the program receives the pattern with the minimum distance. Fifty simulation realizations are produced and the first two are displayed in Figs. 11(a) and 11(b). Furthermore, DS and weighted DS were separately implemented to simulate 50 realizations. The values of N, t, and f are the same as those of CDS. The order of power function in WDS is set as 1. The first two realizations of DS and WDS are shown in Figs. 11(c)–11(f), respectively.

Prior to assessing realizations, the weights used by DS, WDS, and our proposed CDS are shown in Fig. 12. We only draw the weights of the 48 closest points in a template of size 7 × 7. Two decimal places are applied. In Fig. 12(c), weights along the horizontal direction decrease more smoothly than weights along the vertical direction. This structure agrees with the spatial characteristics of the TI. In comparison, DS and WDS use predefined and inflexible weights. The anisotropy is not considered by these two methods.

According to the visual comparison, it is clear that there are noise points in realizations simulated by DS. By contrast, Figs. 11(a), 11(b), 11(e), and 11(f) exhibit better performance. This visual comparison reveals that weighted distance measurements are able to preserve the spatial characteristic of the TI.

An analysis of distance (ANODI) is an important method to rank TI-based simulation algorithms [14,15]. From the geostatistical point of view, there are two kinds of variability in the simulation realizations: spatial uncertainty and pattern reproduction. The former represents the distance between two

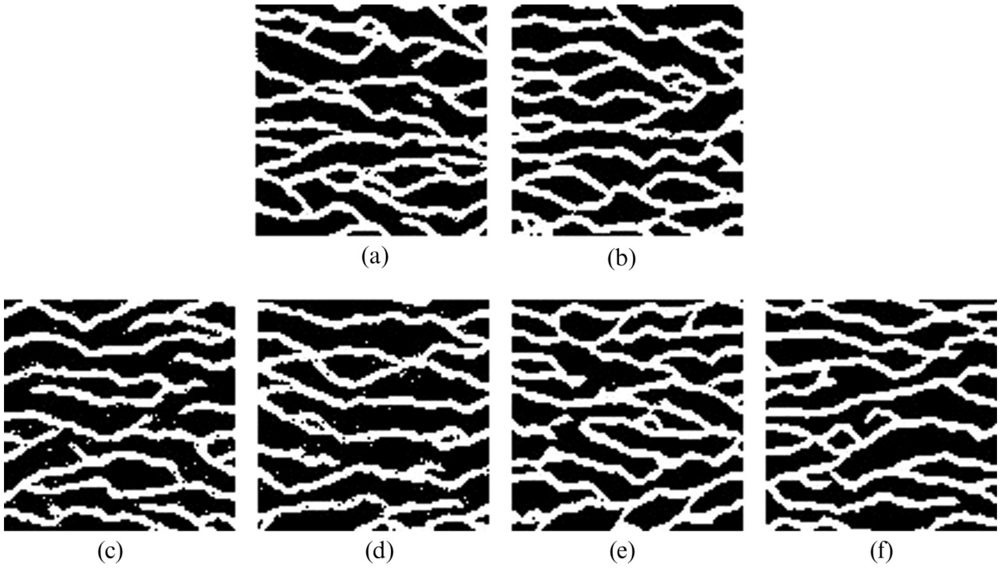


FIG. 11. The simulation realizations: (a,b) the first two realizations produced by the proposed CDS; (c,d) the first two realizations produced by DS; (e,f) the first two realizations produced by WDS.

simulation realizations while the latter expresses the distance between a simulation realization and the TI. A prominent simulation program should enlarge the spatial uncertainty as well as improve pattern reproduction.

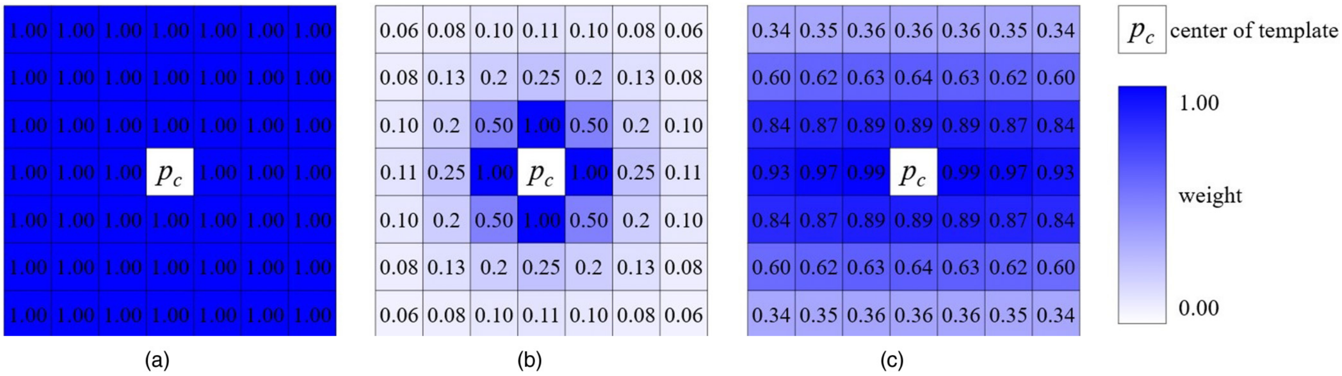
In this paper, we adopted a multiple-point histogram (MPH) scheme within ANODI to compare CDS and existing DS. Suppose that two image sets A and B are input. As the name implies, the core idea of ANODI is to compute the distance between two images. Let A_1 and A_2 denote the first two images in the set A. As the first step, MPH defines a template to extract patterns. According to Ref. [14], a template of 4 × 4 is applied. Because there are two states in the realization, the maximum number of possible patterns is $2^{4\times 4} = 65\,536$. Second, two images are scanned. The first image is characterized by a frequency distribution $F(A_1) = \{F(A_1, P_1),\ F(A_1, P_2), \ldots,\ F(A_1, P_{65\,536})\}$, with $F(A_1, P_1)$ expressing the frequency that pattern appears in A_1. In a similar manner, the method yields the distribution $F(A_2)$. Third, MPH applies Jensen-Shannon divergences to compare $F(A_1)$ and $F(A_2)$. The distance between two distributions can be expressed as

$$D_{\mathrm{JS}}[F(A_1), F(A_2)] = \frac{1}{2}\sum_{v=1}^{65\,536} F(A_1, P_v)\log\frac{F(A_1, P_v)}{F(A_2, P_v)} + \frac{1}{2}\sum_{v=1}^{65\,536} F(A_2, P_v)\log\frac{F(A_2, P_v)}{F(A_1, P_v)}. \quad (18)$$

With the intention of observing an image at different resolutions, MPH adopts a multigrid strategy [4,5]. Let g denote the grid index and G denote the coarsest grid. The distance between A_1 and A_2 is defined as

$$D_{\mathrm{JS}}(A_1, A_2) = \sum_{g=0}^{G}\frac{1}{2^g} \times D_{\mathrm{JS},g}[F(A_1),\ F(A_2)]. \quad (19)$$

Using Eq. (19), we calculate the distance between two images. As mentioned above, the spatial uncertainty represents the average distance between two realizations while the pattern reproduction describes the average distance within the



(a)

1.00	1.00	1.00	1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00	1.00	1.00	1.00
1.00	1.00	1.00	p_c	1.00	1.00	1.00
1.00	1.00	1.00	1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00	1.00	1.00	1.00

(b)

0.06	0.08	0.10	0.11	0.10	0.08	0.06
0.08	0.13	0.2	0.25	0.2	0.13	0.08
0.10	0.2	0.50	1.00	0.50	0.2	0.10
0.11	0.25	1.00	p_c	1.00	0.25	0.11
0.10	0.2	0.50	1.00	0.50	0.2	0.10
0.08	0.13	0.2	0.25	0.2	0.13	0.08
0.06	0.08	0.10	0.11	0.10	0.08	0.06

(c)

0.34	0.35	0.36	0.36	0.36	0.35	0.34
0.60	0.62	0.63	0.64	0.63	0.62	0.60
0.84	0.87	0.89	0.89	0.89	0.87	0.84
0.93	0.97	0.99	p_c	0.99	0.97	0.93
0.84	0.87	0.89	0.89	0.89	0.87	0.84
0.60	0.62	0.63	0.64	0.63	0.62	0.60
0.34	0.35	0.36	0.36	0.36	0.35	0.34

FIG. 12. The weights used by three simulation methods in the 2D channel application: (a) weights in DS; (b) weights in WDS; (c) weights in CDS.

TABLE II. Comparisons between CDS and two previous DS programs.

Method to create A	$r_{A,B}^{\text{between}}$	$r_{A,B}^{\text{within}}$	$r_{A,B}^{\text{overall}}$
DS	1.17:1	1.27:1	0.91:1
WDS	0.85:1	0.95:1	0.88:1

TI and each realization. Let M denote the number of images in the image set A. These two kinds of variability can be expressed as

$$D_A^{\text{between}} = \frac{1}{M(M-1)} \sum_{m}^{M} \sum_{m'}^{M} D_{\text{JS}}(A_m, A_{m'}), \tag{20}$$

$$D_A^{\text{within}} = \frac{1}{L} \sum_{l}^{L} D_{\text{JS}}(A_l, \text{TI}). \tag{21}$$

As the last step, MPH outputs three ratios $r_{A,B}^{\text{between}}$, $r_{A,B}^{\text{within}}$, and $r_{A,B}^{\text{overall}}$ to rank two image sets A and B. A smaller value of $r_{A,B}^{\text{between}}$ reveals that set A has a smaller uncertainty than set B. A greater value of $r_{A,B}^{\text{within}}$ implies that A has a weaker pattern reproductive ability than B. The last ratio summarizes the above two aspects. As a result, $r_{A,B}^{\text{overall}} < 1$ indicates that set outperforms set A. in terms of the simulation quality. These three ratios are defined as

$$r_{A,B}^{\text{between}} = \frac{D_A^{\text{between}}}{D_B^{\text{between}}}, \tag{22}$$

$$r_{A,B}^{\text{within}} = \frac{D_A^{\text{within}}}{D_B^{\text{within}}}, \tag{23}$$

$$r_{A,B}^{\text{overall}} = \frac{r_{A,B}^{\text{between}}}{r_{A,B}^{\text{within}}}. \tag{24}$$

Based on the explanations mentioned above, the realizations-created CDS is independently compared with the realizations simulated by DS and WDS. Using CDS realizations as set B, the ranking results are shown in Table II. It is obvious that our proposed CDS yields the best simulation quality because two ratios $r_{A,B}^{\text{overall}}$ are smaller than 1. This 2D channel simulation application indicates that the correlation-driven distance is a more effective measurement to compare two patterns.

B. 3D rock model simulated from a 2D slice

In petroleum engineering, a three-dimensional rock model is a basic material. However, it is expensive to directly generate a high-resolution 3D model in some scenarios. Aiming at handling the absence of 3D models, the 3D simulation based on a high-resolution 2D slice is presented [16–18]. In this section, we applied CDS to simulate a 3D rock model from one 2D slice. Our method is straightforwardly integrated into the framework elaborated upon in Ref. [17].

Provided by Ref. [19], a 2D sandstone sample slice of size 128 × 128 is shown in Fig. 13. The white areas represent the pore while the black areas express the other materials. This image is produced by the computed tomography (CT) technique with a resolution of 10 μm. In our method, the TI is analyzed in advance to derive weights. Next, CDS is repeatedly performed to realize a point simulation until each point has its value. The simulation domain is of size 128 × 128 × 128. We utilized an unconditional simulation strategy in this application and the sampling rate is 1%. We carried out CDS under the condition of $N = 35$, $t = 0$, and $f = 1$. The reason is that Ref. [18] employs this group in a 3D simulation application. Moreover, Ref. [10] recommends that $N \geqslant 30$, $t \leqslant 0.2$, and $f \geqslant 0.5$. Figures 14(a)–14(c) express a simulation realization from various views. In exterior and cross view, green areas represent the pore while black areas represent the other materials. In the perspective image, the pore is depicted by gray and the others are transparent.

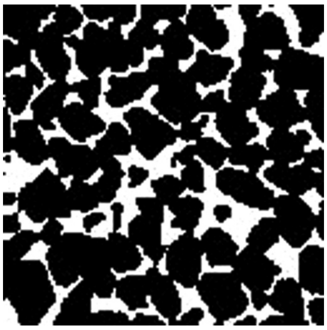


FIG. 13. A sandstone rock slice.

In comparison, DS and WDS were also applied to achieve this simulation task. The three parameters N, t, and f are identical to the parameters in CDS. We set the order of power function in WDS as 1. The realizations created by these two methods are shown in Figs. 14(d)–14(i). With the aim of validating our method, the 3D CT image of this sandstone sample is shown in Figs. 14(j)–14(l). According to the visual comparison, the pore in the simulation realization has similar spatial distributions and geometrical characteristics to the CT sample.

Besides the visual observation, there are a variety of ways to assess the simulation quality. In particular, the autocorrelation function (ACF) and lineal path function (LPF) are two widely used descriptors [19–21]. These two characterization functions separately measure the variability and continuity of a 3D rock model. The ACF is defined as

$$R(l) = \frac{\langle [I(u) - \varphi][I(l+u) - \varphi] \rangle}{\varphi - \varphi^2}, \tag{25}$$

where u is any point in the model. $I(u)$ is an indicator function such that $I(u) = 1$ if u lies within a pore and $I(u) = 0$ otherwise. The angular bracket denotes a mean operation. φ denotes the porosity and can be expressed as $\varphi = \langle I(u) \rangle$.

It is worth noting that ACF can be viewed as an independent test in this case. The main reason is that CDS only employs the concept of correlation coefficient as a static and preprocessing procedure. We concentrate on the training image and do not perform any correlation analysis on the simulation domain. By contrast, the correlation function within simulated annealing (SA) [22,23] framework is a dynamic and real-time procedure. The pore pixels are repeatedly exchanged until a target two-point correlation function is achieved.

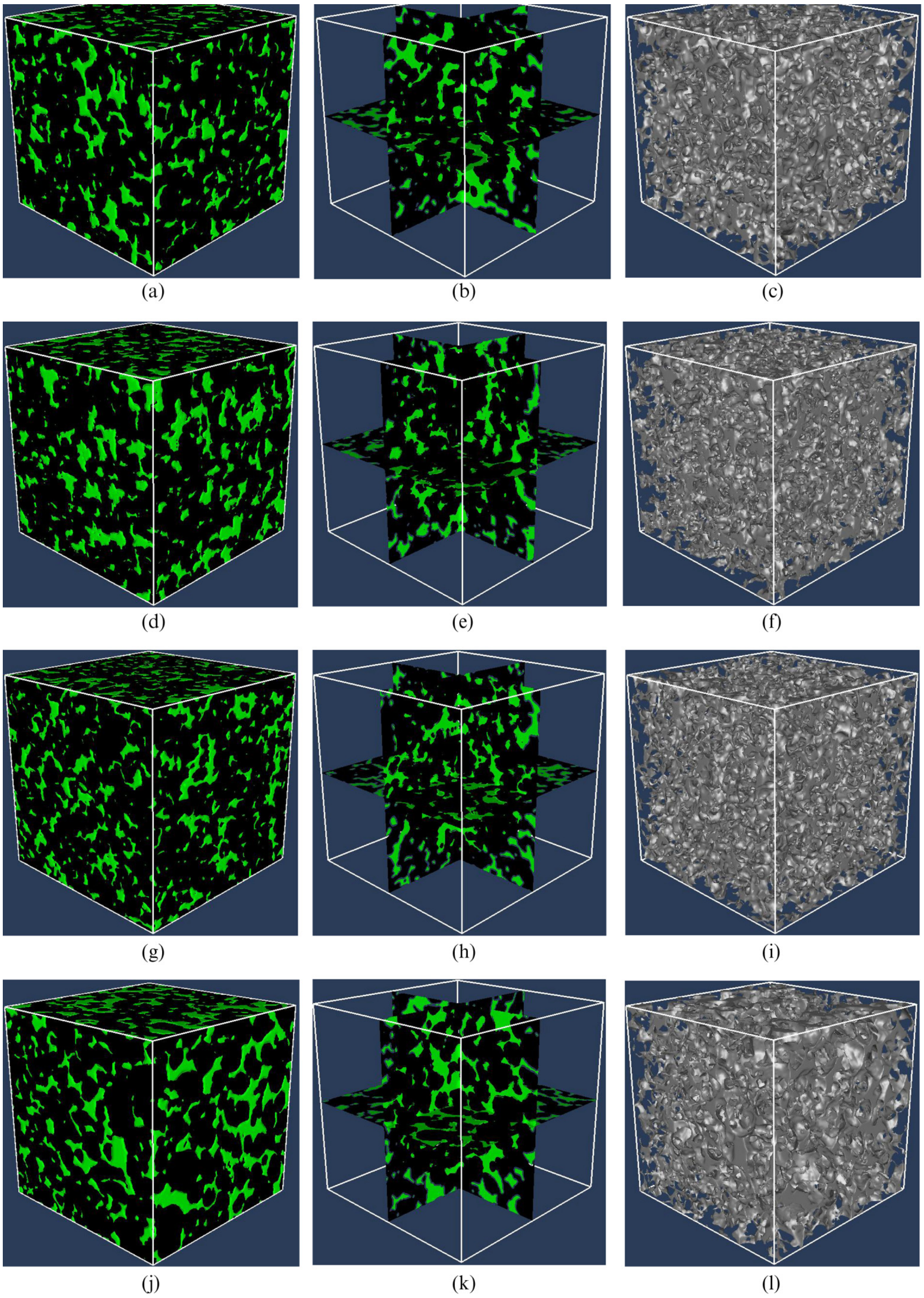


FIG. 14. Three 3D rock models and a CT sample: (a) 3D exterior view of the CDS realization; (b) a cross section of the CDS realization; (c) the perspective image of the CDS realization; (d) 3D exterior view of the DS realization; (e) a cross section of the DS realization; (f) the perspective image of the DS realization; (g) 3D exterior view of the WDS realization; (h) a cross section of the WDS realization; (i) the perspective image of the WDS realization; (j) 3D exterior view of the CT sample; (k) a cross section of the CT sample; (l) the perspective image of the CT sample.

The LPF, which focuses on the continuity of pore, is defined as

$$L(l) = \frac{\text{Prob}[I(u) = 1,\ I(u+1) = 1, \dots,\ I(u+l) = 1]}{\phi}, \tag{26}$$

where u is an outset of a segment of length l. The LPF describes the probability that a straight line of length l is entirely in the pore.

ACFs and LPFs of the TI, the CT sample, and three realizations are displayed in Fig. 15. It is obvious that the statistical characteristics of the CDS realization agree well with the CT sample. In comparison, WDS realization does

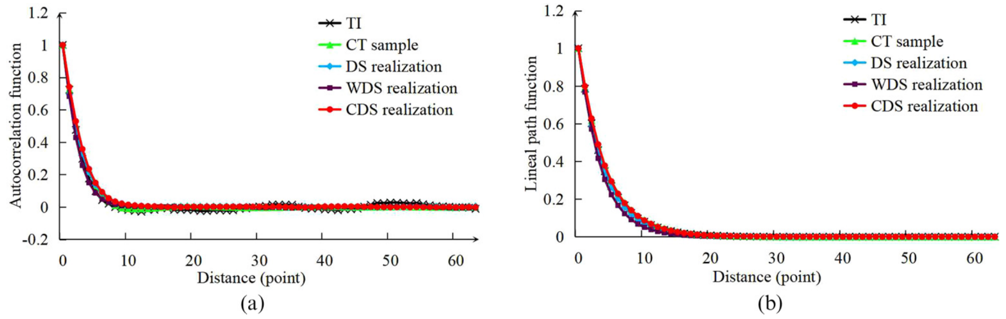


FIG. 15. Comparisons of statistical characteristics for TI, CT sample, and three simulation realizations: (a) the ACFs; (b) The LPFs.

not perform well in terms of LPF. The shape of the WDS curve is lower than others in Fig. 15(b). The reason for this phenomenon is that weights in WDS rapidly decrease with increasing distance. The long-range structure is not accurately preserved.

In addition, we further evaluate the realizations by means of the local porosity theory (LPT). LPT is a frequently used method to characterize the pore space in a 3D rock model [24,25]. In this work, we conduct local porosity distribution as well as local percolation probability to test our method. The key idea of LPT is to scan the 3D model via a measurement cell of length l. Assuming that the cell visits a point in a 3D model, the local porosity $\phi(u,l)$ is defined as

$$\phi(u,l) = \frac{V_1(u,l)}{V(u,l)}, \tag{27}$$

where $V(u,r)$ denotes the volume of the cell, and $V_1(u,r)$ is u the pore space of the cell.

Visiting each available point in a 3D model, the method generates the frequency distribution of $\phi(u,l)$. The local porosity distribution $\mu(\phi,l)$ can be expressed as

$$\mu(\phi,l) = \frac{1}{U}\sum_{u=1}^{U}\delta[\phi - \phi(u,l)], \tag{28}$$

where U denotes the number of cells, and $\delta(\cdots)$ represents the Dirac delta function.

Based on a measurement cell, local percolation probability is a descriptor that focuses on the connectivity of the pore space. Given a cell whose center is u and length is l, an indicator called the connectivity function is such that $\Lambda_d(u,l) = 1$ if there is a path through the pore space in this cell along the d direction and $\Lambda_d(u,l) = 0$ otherwise. Here d represents a certain direction and can be x, y, and z in a 3D model. In particular, $\Lambda_3(u,l) = 1$ reveals that the paths can be found along all three directions. In order to substantially assess the quality, we set $d = 3$ in the following. According

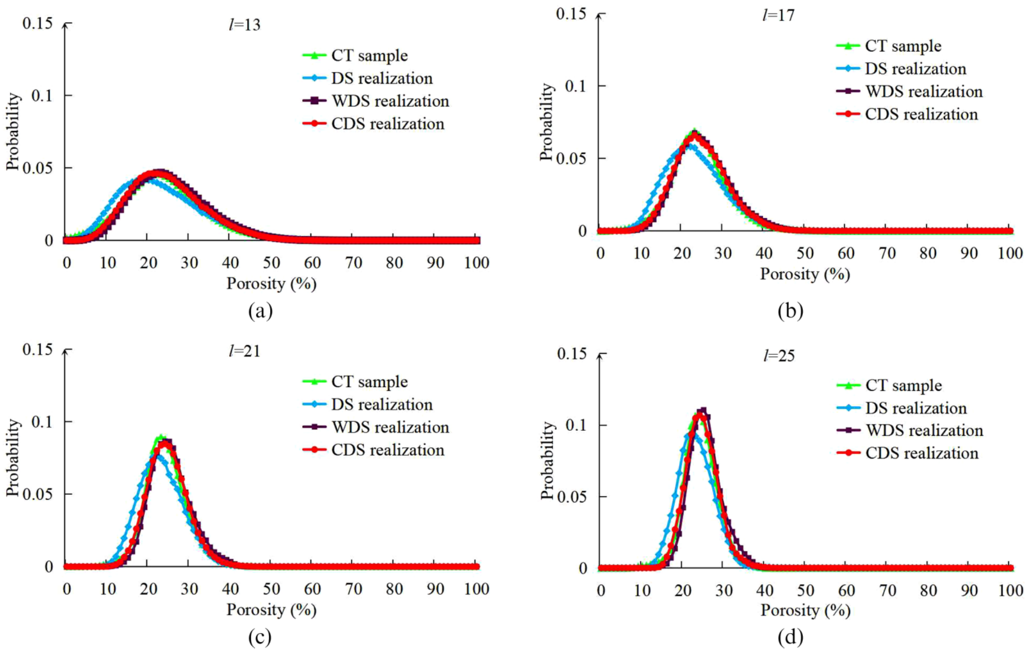


FIG. 16. Comparisons of local porosity distribution for CT sample and three simulation realizations: (a) $l = 13$; (b) $l = 17$; (c) $l = 21$; (d) $l = 25$.

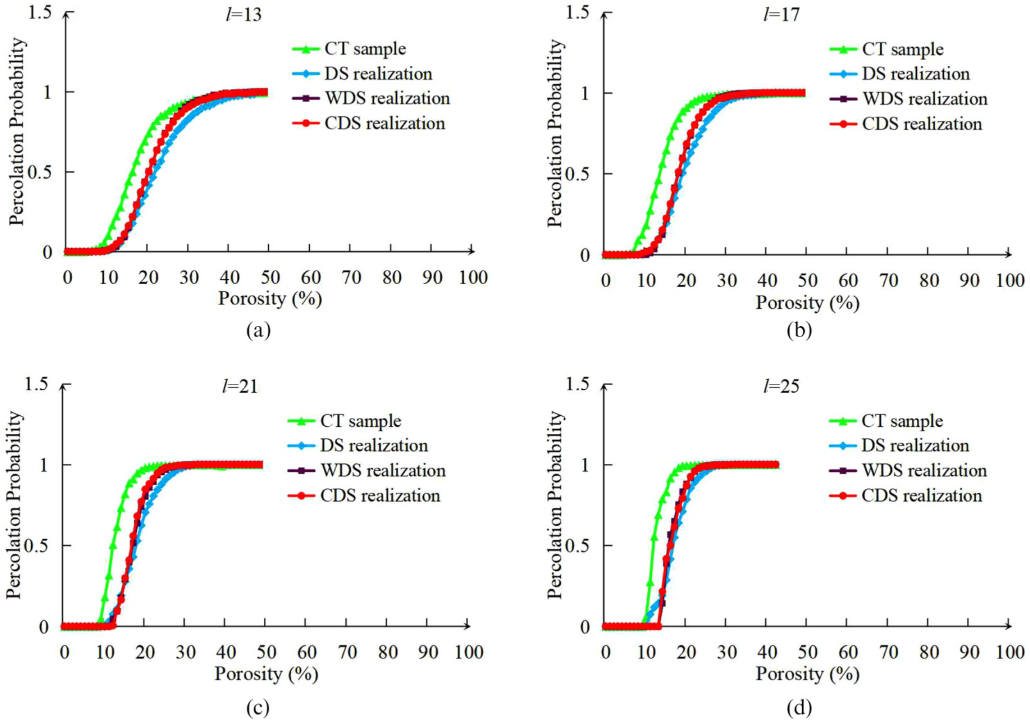


FIG. 17. Comparisons of local percolation probability for CT sample and three simulation realizations: (a) $l = 13$; (b) $l = 17$; (c) $l = 21$; (d) $l = 25$.

to the explanations above, the local percolation probability is defined as

$$\lambda_c(\phi, l) = \frac{\sum_u \Lambda_3(u, l)\delta_{\phi,\phi(u,l)}}{\sum_u \delta_{\phi,\phi(u,l)}}, \tag{29}$$

$$\delta_{\phi,\phi(u,l)} = \begin{cases} 1 & \text{if } \phi = \phi(u, l) \\ 0 & \text{otherwise} \end{cases}, \tag{30}$$

where $\delta_{\phi,\phi(u,l)}$ is the Kronecker delta. The detailed descriptions about LPT can be found in Refs. [24,25].

Evidently, the length l has a considerable impact on LPT measurement. With the aim of extensively comparing, we apply a set of lengths to check the realizations. Because the size of realizations is 128 × 128 × 128, the value of l is separately set as 13, 17, 21, and 25. The local porosity distributions of four 3D models with different lengths are shown in Fig. 16. It is clear that the local porosity distribution curve of DS realization is not in agreement with the CT sample. By contrast, the local porosity distributions corresponding to WDS and the proposed CDS are close to the CT sample. Figure 17 displays comparisons with four models in terms of local percolation probability. The results reveal that WDS and our CDS realizations have better percolation behaviors than DS realization. However, there is still a difference between each MPS model and the CT sample. The descriptions mentioned above imply that the weighted distance significantly improves simulation quality.

As the last criterion, the flow characteristic of the 3D rock model is applied to check our proposed CDS method. In this work, we used the pore network model to calculate the relative permeability curve during a two-phase flow simulation.

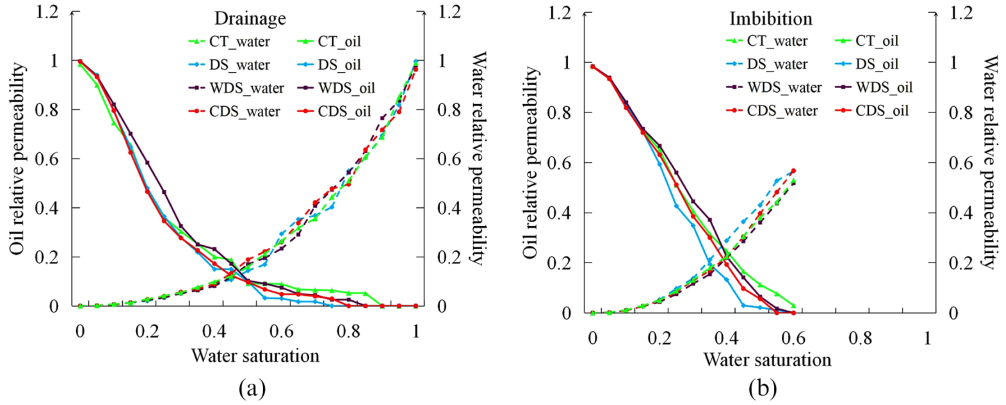


FIG. 18. Relative permeability curves of the CT sample and three realizations: (a) drainage and (b) imbibition.

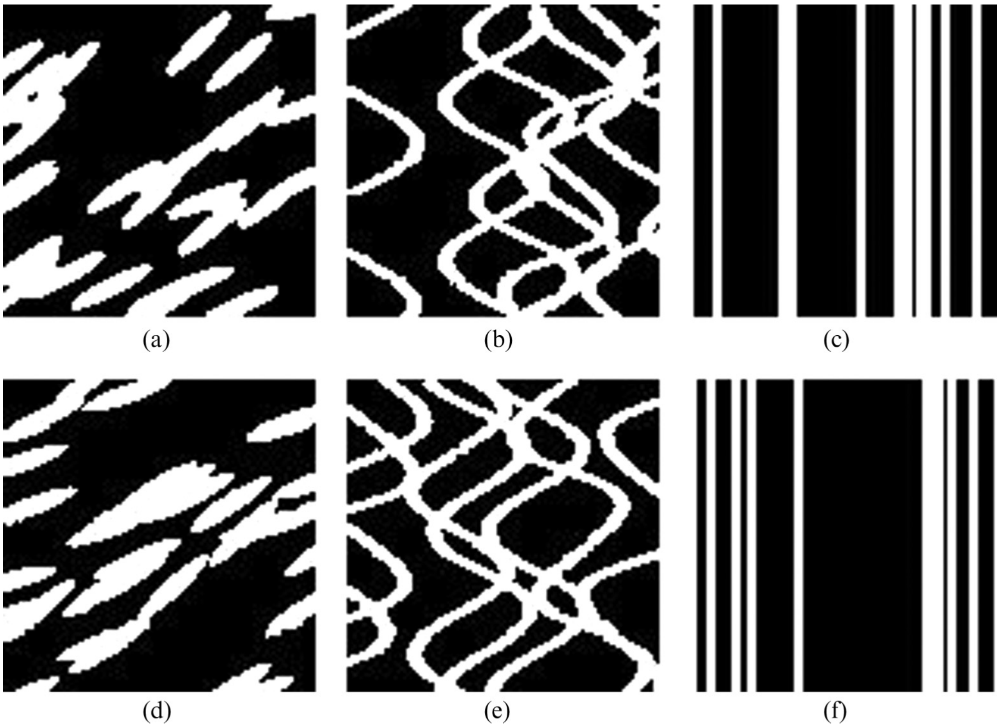


FIG. 19. Six images created by the TIGENERATOR tool: (a–c) training images; (d–f) images used to produce the hard data images.

As the first step, the maximal ball algorithm [26] is conducted to regularize and categorize the pore space into two classes: pore and throat. The pores are viewed as the fluid storage while the throats become bridges between two pore bodies. Next, we carried out the two-phase flow simulation based on two flooding procedures. Assuming that our rock model is in a water-oil system, the primary flooding is the drainage process. In other words, a wetting phase (water) is gradually replaced by a nonwetting phase (oil). In this application, we set the densities of water and oil as 900 and $100\,\mathrm{kg/m^3}$, respectively. The water-oil surface tension is taken to be $3 \times 10^2\,\mathrm{N/m}$. The water and oil viscosities are separately specified as 1.05×10^3 and $1.0 \times 10^3\,\mathrm{kg/ms}$. Furthermore, one important assumption is that the rock is intensively water wetting during the drainage. The receding contact angle and the advancing contact angle are both equal to 0°. As the secondary flooding, the imbibition process is implemented to imitate oil production. We change the receding contact angle to 50° and the advancing contact angle to 60°. The two-phase flow procedure is elaborated in Refs. [19,20].

According to the procedures and parameters discussed above, the flow characteristic of the proposed CDS realization is evaluated. In comparison, we conduct the pore network model using the CT sample, and the DS realization, as well as the WDS realization. The relative permeability curves are shown in Fig. 18. There is agreement between the CT sample and our CDS realization. This provides strong evidence that the flow characteristic of our realization matches with the target.

In summary, we employ not only structural descriptors but also transport property to validate our method. Two statistical characteristic functions, local porosity theory, and the pore network model are applied to complete extensive comparisons. The results indicate that our proposed CDS is capable of producing a 3D rock model from a single 2D image. The spatial characteristics of the 2D TI are well preserved and reproduced in the 3D simulation domain.

C. 2D training image evaluations

In this section, the experimental materials shown in Refs. [11,12] were used to validate the minimum correlation-driven distance method. Considering three different parameter sets, TIGENERATOR tool [27] was adopted to generate TIs. For each parameter set, two images of size 100 × 100 are separately created. The first image is viewed as the TI while the second one is used to produce the hard data. These images are shown in Fig. 19.

The HD image is created by a random sampling strategy. In other words, we randomly sampled an image until the number of informed points in HD exceeds I_{sample}. Here I_{sample} is a predefined threshold. With the objective to extensively test MCD, the values of I_{sample} were 1091, 222, and 80, respectively. These values are provided by Ref. [11]. Figure 20 displays these HD images. In HD images, the black and white points are sampled from the corresponding image. The unknown points are expressed by the gray area. According to the sampling strategy, it is clear that the most suitable TI for Figs. 20(a), 20(d), and 20(g) is Fig. 19(a). Thus, the TI evaluation method should select Fig. 19(a) from three candidate TIs. In a similar manner, it is proper to select Fig. 19(b) for Figs. 20(b), 20(e), and 20(h). Moreover, Fig. 19(c) is the appropriate TI for Figs. 20(c), 20(f), and 20(i).

Aiming at checking the performance of MCD, we implemented two versions as (1) MCD without the conditioning pattern extraction history strategy, and (2) MCD with the

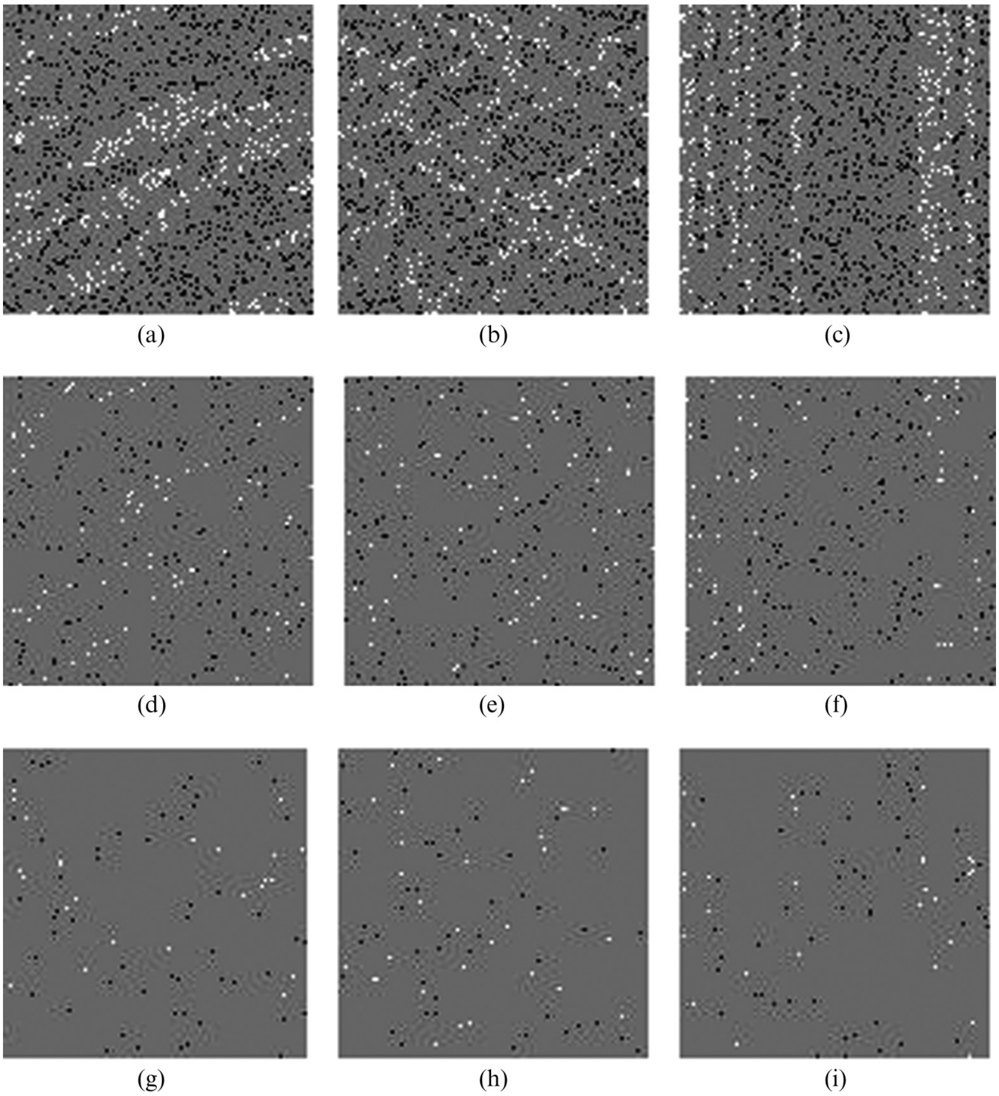


FIG. 20. Hard data images: (a) HD sampled from Fig. 19(a) with $I_{\text{sample}} = 1091$; (b) HD sampled from Fig. 19(b) with $I_{\text{sample}} = 1091$; (c) HD sampled from Fig. 19(c) with $I_{\text{sample}} = 1091$; (d) HD sampled from Fig. 19(a) with $I_{\text{sample}} = 222$; (e) HD sampled from Fig. 19(b) with $I_{\text{sample}} = 222$; (f) HD sampled from Fig. 19(c) with $I_{\text{sample}} = 222$; (g) HD sampled from Fig. 19(a) with $I_{\text{sample}} = 80$; (h) HD sampled from Fig. 19(b) with $I_{\text{sample}} = 80$; (i) HD sampled from Fig. 19(c) with $I_{\text{sample}} = 80$.

conditioning pattern extraction history strategy. The correlation-driven distance is applied in these two programs to compare two patterns. Accordingly, these two methods have the same effectiveness but different efficiency.

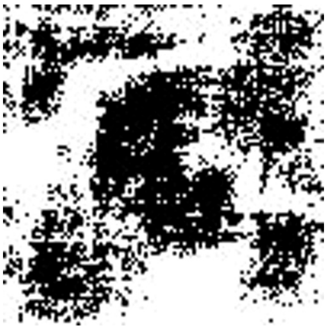


FIG. 21. The result of the conditioning pattern extraction history.

As an example, the consistency between Figs. 19(a) and 20(d) is evaluated. In this application, we set $N = 30$. Our

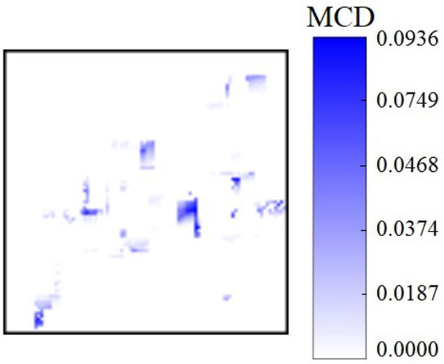


FIG. 22. The MCD output image using Figs. 20(d) and 19(a).

TABLE III. The MCD mean distance between an HD and a TI.

HD images	Figure 19(a)	Figure 19(b)	Figure 19(c)
Figure 20(a)	**0.018**	0.048	0.167
Figure 20(b)	0.080	**0.043**	0.187
Figure 20(c)	0.105	0.087	**0.013**
Figure 20(d)	**0.001**	0.005	0.064
Figure 20(e)	0.011	**0.003**	0.048
Figure 20(f)	0.015	0.009	**0.003**
Figure 20(g)	**0.000**	3.6×10^{-6}	0.018
Figure 20(h)	7.2×10^{-5}	$\mathbf{1.7 \times 10^{-5}}$	0.007
Figure 20(i)	0.003	2×10^{-4}	**0.000**

computer configuration is reported in Sec. IV A. At first, we implement the first version of MCD. The computer takes 6 s without the conditioning pattern extraction history. In comparison, 4 s are required by the conditioning pattern extraction strategy. Our program only checks 42.6% of the points via the searching procedure in the TI. Figure 21 displays the computations which are eliminated by the proposed strategy. The black area represents the points whose conditioning pattern exists in the history list. The points that require a searching procedure are expressed by the white area. This conceptual application indicates that the conditioning pattern extraction history strategy reduces the computational burden.

Using the steps explained in Sec. III B, the consistency between Figs. 20(d) and 19(a) is evaluated. The value of m^{MCD} [Fig. 20(a), Fig. 19(a)] is 0.001. This implies that the consistency between these two images is high. Furthermore, the MCD image is shown in Fig. 22. In this image, the intensity of a point corresponds to its MCD distance. This MCD image helps us to observe the spatial characteristic of the consistency. It is obvious that most of the distances are below 0.0374. Consequently, we confirm that the training image Fig. 19(a) is compatible with the hard data set Fig. 20(d).

We used MCD to compute the consistency between each HD and each TI. The resulting MCD mean distances are shown in Table III. The minimum distance in each row is emphasized by the bold. Our method recommends the TI with the smallest distance. Given an HD, it is evident that MCD precisely selects a TI from the candidate set. The selected TIs are consistent with the targets stated at the beginning of Sec. IV C. In this TI evaluation application, MCD exhibits an excellent performance in terms of efficiency as well as accuracy.

D. Rock simulation domain segmentation

As mentioned in Sec. IV B, it is expensive to directly obtain a 3D high-resolution rock image of large size in some applications. In this section, we apply a low-resolution CT device whose resolution is 10 μm to scan a rock sample. As shown in Fig. 23(a), a rock model of size 200 × 250 × 24 is generated. The pore is depicted by the gray while the rest of the material is not drawn. Figures 23(b)–23(d) display three rock slices whose layer indices are 7, 12, and 18, respectively. The black area represents the pore and the white area represents other materials. It should be noticed that the borderline is not part of these images.

Next, an upsampling operation is applied to each slice in Fig. 23(a). As a result, 24 slices of size 400 × 500 are obtained. For example, the operation separately processes Figs. 23(b)–23(d). The output images are shown in Fig. 24. The pore and other materials are represented by black and white points, respectively. The gray area denotes the unknown

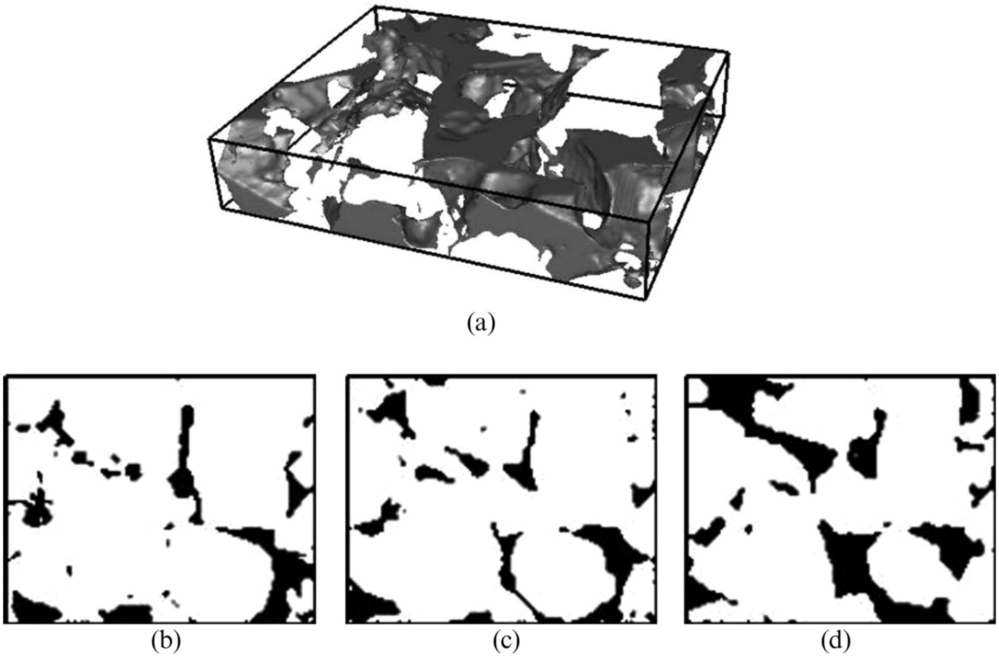


FIG. 23. A three-dimensional low-resolution rock image of resolution 10 μm: (a) the perspective image; (b) the seventh slice; (c) the 12th slice; (d) the 18th slice.

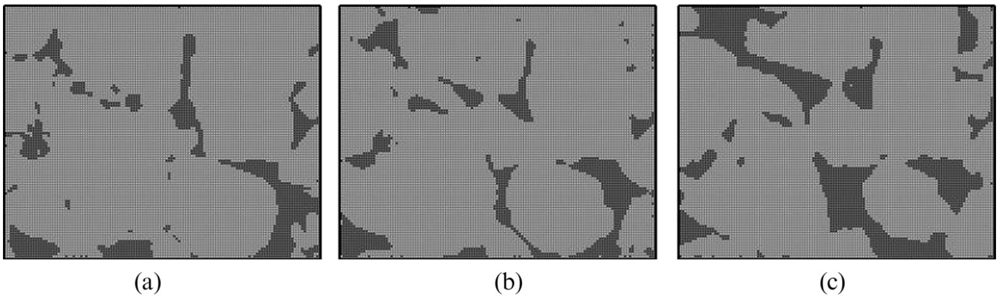


FIG. 24. The hard data slices: (a) the seventh slice; (b) the 12th slice; (c) the 18th slice.

points. This upsampling operation allows us to obtain a set of 2D HD images. Accordingly, we use H to denote the group of 2D HD images.

Afterward, we apply a CT scanner of resolution 5 μm to create TIs. Due to the limitation of devices, only three 2D high-resolution images are produced. These three images are displayed in Fig. 25. The pore is depicted by the black and the other materials are drawn in white.

We apply MCD to evaluate the consistencies between HD and three TIs. In this case, we used H to denote the set of 2D hard data slices. Prior to performing the local consistency, the global consistency is computed. In other words, we propose 24 HD slices as a whole. Three MCD distances are m^{MCD} [H, Fig. 25(a)] = 0.0037, m^{MCD}[H, Fig. 25(b)] = 0.0036, and m^{MCD}[H, Fig. 25(c)] = 0.0037, respectively. According to these resulting distances, each TI has a high compatibility with the hard data image set. The TI selection program cannot reasonably rank TIs on the basis of global consistency.

Then, the concept of local consistency is used. We adopted MCD to compute compatibility between each 2D HD slice and each TI. The results are shown in Fig. 26. It is clear that the spatial characteristic of HD significantly changes with the indices. For the slices whose indices are smaller than 8, Fig. 25(a) is a suitable TI. As a contrast, Fig. 25(b) is appropriate for the images whose indices are in the range [9,17]. For the rest of the slices, our method recommends Fig. 25(c) to express the geological phenomenon. The explanation described above reveals that nonstationarity in this 3D HD is intensive. The local consistency provides a practical guide to observe and express this nonstationary phenomenon.

In addition, we further divided each slice into more subareas. With the aim of extensively comparing, MCD applies a set of S to partition the slice. We use Fig. 24(a) as an example. This image is uniformly partitioned into 4, 16, 64, 256, and 1024 subareas, respectively. We apply the mean distance between each subarea and each TI as a measure of local consistency. The resulting segmentation is shown in Fig. 27. The significance of each color is described in Fig. 27(f).

As shown in Fig. 26, MCD advises Fig. 25(a) as training images with respect to Fig. 24(a). However, the local consistency does not agree with this suggestion. As Fig. 27(a) shows, Fig. 24(a) is divided into four subareas. The consistency between these four subareas and three TIs is independently evaluated. Our method suggests two top areas should be simulated by Figs. 25(b) and 25(c) while the bottom area is supposed to be created by Fig. 25(a). This result implies that patterns appearing in Figs. 25(b) and 25(c) are helpful for the simulation task. With the purpose of better describing the rock model, MCD advises that the strengths from three candidate TIs should be combined together. By contrast, the previous TI evaluation methods only rank candidate TIs. The unselected images are directly discarded. As a result, the information in the candidate set is not effectively used.

With the increasing S, the diversity of subareas becomes more intensive. Each subarea has its own spatial characteristic. It is worth noting that a large value of S reduces the size of the subarea. Because the core idea of MPS is to collect a set of points at a time, a subarea of small size is not beneficial to the simulation program. Consequently, we use $S = 256$ as a suitable parameter.

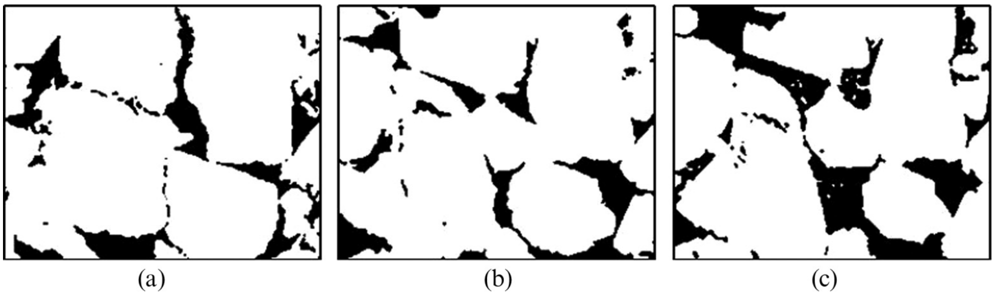


FIG. 25. Three two-dimensional high-resolution rock images of resolution 5 μm.

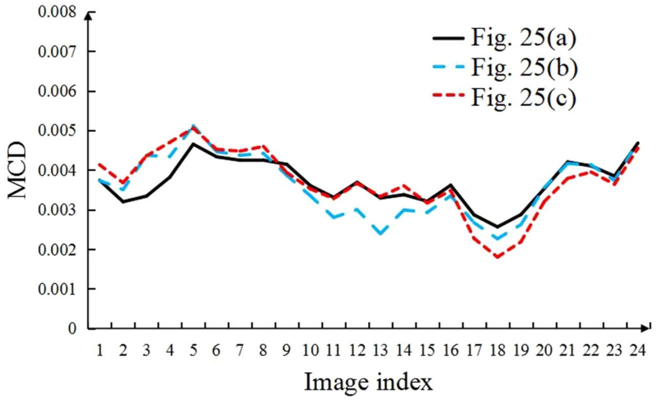


FIG. 26. MCD mean distances between each slice and three TIs.

In summary, this conceptual experiment indicates that the local consistency is an effective tool to describe nonstationarity. Although the differences among global consistencies are not significant, we observe the nonstationarity in this 3D rock model by means of local consistency. Our method automatically partitions the simulation domain into several subareas.

V. CONCLUSION

We explored a way to realize geostatistical simulation as well as training image (TI) evaluation in this paper. First, the correlation-driven direct sampling (CDS) is presented. We apply the concepts of the ellipse, correlation coefficient, Gaussian function, and affine transformation to conduct weight computations. As a generalization of previous distance measurement, the correlation-driven distance can better compare two patterns. Second, we fulfill TI evaluation according to the consistency between TI and hard data (HD). Inspired by CDS, minimum correlation-driven distance (MCD) is introduced to improve evaluation accuracy. The conditioning pattern extraction history is suggested to save running time at the cost of memory consumption. Third, nonstationarity is dealt with by local consistency. Our proposed method automatically partitions the simulation domain into several subareas.

We tested the performance of the proposed program via a 2D channel simulation and a 3D rock simulation. The results are quantitatively assessed by the analysis of distance (ANODI), autocorrelation function (ACF), lineal path function (LPF), and local porosity theory (LPT) and the pore network model. Compared with two previous DS methods, CDS exhibits better performance in terms of simulation quality. The properties of a simulation realization agree well with reality. These experimental results indicate that our proposed correlation-driven distance is a competitive measurement to compare patterns. Further applications contain a set of 2D TI evaluations and a 3D rock segmentation. MCD exhibits sensible evaluation accuracy, excellent computational efficiency, and the ability to address nonstationarity. The TIs selected by MCD are the same as the target. A nonstationary simulation domain is observed by the local consistency. Considering their efficiency and accuracy, we believe that the proposed method of CDS and MCD may potentially have a wide range of practical applications.

ACKNOWLEDGMENTS

This work was supported in part by the Major Programs of the National Natural Science Foundation of China under Grants No. 41390450 and No. 41390454. National Key R & D Program of the Ministry of Science and Technology of China

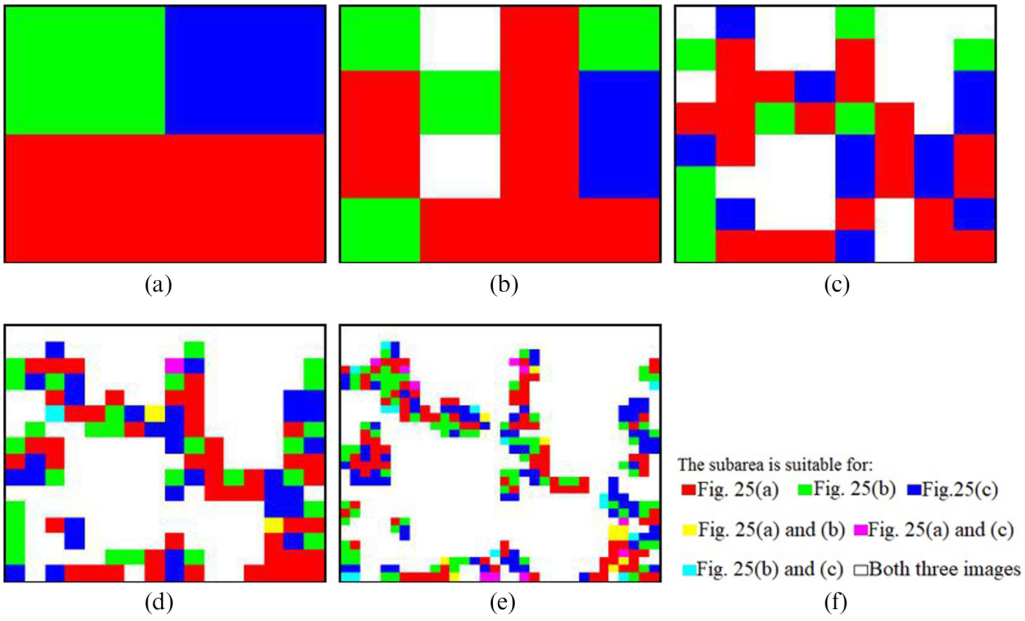

(a)
(b)
(c)
(d)
(e)
The subarea is suitable for:
🟥Fig. 25(a) 🟩Fig. 25(b) 🟦Fig.25(c)
🟨Fig. 25(a) and (b) 🟪Fig. 25(a) and (c)
🟦Fig. 25(b) and (c) □Both three images
(f)

FIG. 27. The senenth slice is divided into the different amounts of subareas: (a) four subareas; (b) 16 subareas; (c) 64 subareas; (d) 256 subareas; (e) 1024 subareas; (f) Representative colors.

(Grants No. 2018YFC1504202 and No. 2018YFC0603500). The authors would like to acknowledge constructive comments from the anonymous referees that prompted us to improve the paper. We are grateful to Mrs. Chengcheng Liang for helpful discussions, and we also thank Ms. Tingrong Zhang for providing some useful materials in 3D rock applications.

[1] F. B. Guardiano and R. M. Srivastava, in *Geostatistics Tróia'92*, edited by A. Soares (Kluwer Academic, Dordrecht, 1993), Vol. 1, p. 133.
[2] C. V. Deutsch and A. G. Journel, *GSLIB: Geostatistical Software Library and User's Guide* (Oxford University, New York, 1992).
[3] P. Goovaerts, *Geostatistics for Natural Resources Evaluation* (Oxford University, Oxford, 1997).
[4] S. Strebelle, Math. Geol. **34**, 1 (2002).
[5] S. Strebelle and C. Cavelius, Math. Geosci. **46**, 171 (2013).
[6] J. Straubhaar, P. Renard, G. Mariethoz, R. Froidevaux, and O. Besson, Math. Geosci. **43**, 305 (2011).
[7] J. Straubhaar, A. Walgenwitz, and P. Renard, Math. Geosci. **45**, 131 (2013).
[8] C. Zuo, Z. B. Pan, and H. Liang, Phys. Rev. E **97**, 033302 (2018).
[9] G. Mariethoz, P. Renard, and J. Straubhaar, Water Resour. Res. **46**, W11536 (2010).
[10] E. Meerschman, G. Piro, G. Mariethoz, J. Straubhaar, M. V. Meirvenne, and P. Renard, Comput. Geosci. **52**, 307 (2013).
[11] C. Pérez, G. Mariethoz, and J. M. Ortizab, Comput. Geosci. **70**, 190 (2014).
[12] W. Feng, S. Wu, Y. Yin, J. Zhang, and K. Zhang, Comput. Geosci. **104**, 35 (2017).
[13] E. Gringarten and C. V. Deutsch, Math. Geol. **33**, 507 (2001).
[14] X. Tan, P. Tahmasebi, and J. Caers, Math. Geosci. **46**, 149 (2014).
[15] P. Tahmasebi and M. Sahimi, Water Resour. Res. **52**, 2074 (2016).
[16] A. Comunian, P. Renard, and J. Straubhaar, Comput. Geosci. **40**, 49 (2012).
[17] G. Mariethoz and P. Renard, Math. Geosci. **42**, 245 (2010).
[18] N. Gueting, J. Caers, A. Comunian, J. Vanderborght, and A. Englert, Math. Geosci. **50**, 53 (2018).
[19] M. L. Gao, X. H. He, Q. Z. Teng, C. Zuo, and D. D. Chen, Phys. Rev. E **91**, 013308 (2015).
[20] M. L. Gao, Q. Z. Teng, X. H. He, C. Zuo, and Z. J. Li, Phys. Rev. E **93**, 012140 (2016).
[21] B. Lu and S. Torquato, Phys. Rev. A **45**, 922 (1992).
[22] C. L. Y. Yeong and S. Torquato, Phys. Rev. E **57**, 495 (1998).
[23] C. L. Y. Yeong and S. Torquato, Phys. Rev. E **58**, 224 (1998).
[24] F. D. E. Latief, B. Biswal, U. Fauzi, and R. Hilfer, Phys. A **389**, 1607 (2010).
[25] Y. Wang, J. Y. Arns, S. S. Rahman, and C. H. Arns, Phys. Rev. E **98**, 043310 (2018).
[26] H. Dong and M. J. Blunt, Phys. Rev. E **80**, 036307 (2009).
[27] A. Maharaja, Comput. Geosci. **34**, 1753 (2008).