Research Article

History Matching of a Channelized Reservoir Using a Serial Denoising Autoencoder Integrated with ES-MDA

Sungil Kim,^{1,2} Baehyun Min^{1,3}, ,^{1,3} Seoyoon Kwon,^{1,3} and Min-gon Chu¹

¹Center for Climate/Environment Change Prediction Research, Ewha Womans University, 52 Ewhayeodae-gil, Seodaemun-gu, Seoul 03760, Republic of Korea

²Petroleum and Marine Research Division, Korea Institute of Geoscience and Mineral Resources, 124 Gwahak-ro, Yuseong-gu, Daejeon 34132, Republic of Korea

³Department of Climate and Energy Systems Engineering, Ewha Womans University, 52 Ewhayeodae-gil, Seodaemun-gu, Seoul 03760, Republic of Korea

Correspondence should be addressed to Baehyun Min; bhmin01@ewha.ac.kr

Received 31 October 2018; Accepted 17 January 2019; Published 16 April 2019

Guest Editor: Sergio Longhitano

Copyright © 2019 Sungil Kim et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

For an ensemble-based history matching of a channelized reservoir, loss of geological plausibility is challenging because of pixel-based manipulation of channel shape and connectivity despite sufficient conditioning to dynamic observations. Regarding the loss as artificial noise, this study designs a serial denoising autoencoder (SDAE) composed of two neural network filters, utilizes this machine learning algorithm for relieving noise effects in the process of ensemble smoother with multiple data assimilation (ES-MDA), and improves the overall history matching performance. As a training dataset of the SDAE, the static reservoir models are realized based on multipoint geostatistics and contaminated with two types of noise: salt and pepper noise and Gaussian noise. The SDAE learns how to eliminate the noise and restore the clean reservoir models. It does this through encoding and decoding processes using the noise realizations as inputs and the original realizations as outputs of the SDAE. The trained SDAE is embedded in the ES-MDA. The posterior reservoir models updated using Kalman gain are imported to the SDAE which then exports the purified prior models of the next assimilation. In this manner, a clear contrast among rock facies parameters during multiple data assimilations is maintained. A case study at a gas reservoir indicates that ES-MDA coupled with the noise remover outperforms a conventional ES-MDA. Improvement in the history matching performance resulting from denoising is also observed for ES-MDA algorithms combined with dimension reduction approaches such as discrete cosine transform, K-singular vector decomposition, and a stacked autoencoder. The results of this study imply that a well-trained SDAE has the potential to be a reliable auxiliary method for enhancing the performance of data assimilation algorithms if the computational cost required for machine learning is affordable.

1. Introduction

In the petroleum industry, history matching is an essential process to calibrate reservoir properties (e.g., facies, permeability, and PVT parameters) by conditioning one or more reservoir models to field observations such as production and seismic data [1]. Ensemble-based data assimilation methods based on Bayes theorem [2–4] have been applied to solve a variety of petroleum engineering problems since the early 2000's [5]. Specifically, ensemble Kalman filter

(EnKF) [2], ensemble smoother (ES) [6], and ensemble smoother with multiple data assimilation (ES-MDA) [7] have been utilized for history matching of geological features such as channels (the subject of this study). Loss of geological characteristics due to pixel-based manipulation of channel features (such as shape and connectivity) is challenging for an ensemble-based history matching of a channelized reservoir. Note, the loss is regarded as noise in this study. Despite sufficient conditioning to field observations during ensemble updates, increase in noise often

causes failure to deliver the geologically plausible reservoir models. This decreases the reliability of history matching results [8]. For this reason, a relevant problem is how to update the reservoir models with consideration for geological plausibility in a practical manner.

Previous studies have improved the performance of ensemble-based history matching by adopting data transformation [9–13]. In general, transformation methods have a substantial energy compaction property that is useful for feature extraction and dimension reduction of parameters and helping to save computational cost in data processing. If essential features are adequately acquired, updating the features can also yield an improved history matching performance over calibrating original parameters. For these reasons, discrete cosine transform (DCT) [14, 15], discrete wavelet transform [1], K-singular value decomposition (K-SVD) [16, 17], and autoencoder (AE) [18] have been employed as ancillary parameterizations of ensemble-based methods. For history matching of channelized reservoirs, DCT has been utilized to preserve channel properties because DCT figures out overall trends and main patterns of channels by using only essential DCT coefficients [19–22]. Updating essential DCT coefficients implies the importance of determining the optimal number of DCT coefficients for preserving channel connectivity and continuity [22]. K-SVD has an advantage of sparse representations of data as weighted linear combinations of prototype realizations. However, it takes preprocessing time to construct a set of prototype realizations called a dictionary. As a remedy, a combination of DCT and iterative K-SVD was proposed to complement the limitations of both methods [23]. Canchumuni et al. [18] coupled AE with ES-MDA for an efficient parameterization and compared its performance with that of ES-MDA coupled with principal component analysis.

Recent advances in machine learning have offered opportunities for using complex meta-heuristic tools based on artificial neural networks (if the tools are well trained at affordable computational cost). In petroleum engineering, examples include production optimization [24–26] and history matching [18, 27–29]. As investigated in [18, 27], AE is a multilayer neural network that learns efficient data coding in an unsupervised manner. This is useful for representation (encoding) of a given dataset followed by reconstruction of the encoded dataset [30, 31]. In image processing and recognition, AE has a capability of denoising through encoding and decoding processes if noise data is input and purified data is output [32]. This type of AE is called denoising autoencoder (DAE) [33, 34].

Taking this capability of DAE into consideration, this study designs a serial denoising autoencoder (SDAE) and integrates the algorithm in the ensemble update of ES-MDA to improve the performance of ensemble-based history matching. The SDAE learns how to eliminate the noise and restore the clean reservoir models through encoding and decoding processes using the noise realizations as inputs and the original realizations as outputs of the SDAE. The trained SDAE imports the posterior reservoir models derived using Kalman gain of ES-MDA for purifying the models and exports the purified models as prior models

for the subsequent assimilation of ES-MDA. The ES-MDA coupled with SDAE is applied to history matching of a channelized gas reservoir. Its performance is compared with that of the conventional ES-MDA. Also, denoising effects are investigated for ES-MDA coupled with dimension reduction methods such as DCT and K-SVD.

2. Methodology

In this study, ES-MDA is the platform to calibrate the reservoir models for history matching. A procedure of ES-MDA is mainly composed of two steps: numerical simulation for the reservoir models and update of reservoir parameters. A brief description of ES-MDA is given in Section 2.1. SDAE purifies noise in the updated reservoir models (Section 2.2). AE (Section 2.2), DCT (Section 2.3), and K-SVD (Section 2.4) are introduced as parameterization techniques. Section 2.5 proposes the ES-MDA algorithm coupled with the SDAE and the parameterization methods.

2.1. ES-MDA. Ensemble-based history matching methods update parameters of the target models using observed data such as production rate and 4D seismic data. For model updates, EnKF utilizes observed data one-time step by one-time step in time sequence. The principle of EnKF might cause inconsistency between the updated static models and dynamic behaviors due to sequential updates without returning to an initial time step [7, 35]. ES updates models using observed data measured at all time steps at once to solve the inconsistency issue [6]. However, history matching performance obtained using ES was often less satisfactory due to the one-time calibration of the reservoir models. ES-MDA is a variant of ES. ES-MDA repeats ES with inflation coefficients for the covariance matrix of observed data measurement error. Therefore, it has advantages in not only history matching performance but also the consistency between static data and dynamic data [35].

For ensemble-based history matching, the equation of model update is as follows:

$$m_i = m_i^b + C_{md}(C_{dd} + \alpha_p C_D)^{-1}(d_i^{\text{unc}} - d_i), \quad \text{for } i = 1, \dots, N_{\text{ens}}, \quad (1)$$

where m is the state vector consisting of reservoir parameters (e.g., facies and permeability), the subscript i means the i th ensemble member, the superscript b means before update in this study, C_{md} is the cross-covariance matrix of m and d , C_{dd} is the autocovariance matrix of d , α_p is the inflation coefficients for C_D (which is the covariance matrix of the observed data measurement error [7]), d is the simulated responses obtained by running a forward simulation, d^{unc} is the observation data perturbed by inflated observed data measurement error, and N_{ens} is the number of ensemble members (i.e., the reservoir models in the ensemble). In equation (1), $C_{md}(C_{dd} + \alpha_p C_D)^{-1}$ is the Kalman gain that is computed with regularization by singular value decomposition (SVD) using 99.9% of the total energy in singular values [7].

Definitions of C_{md} and C_{dd} are as follows:

$$C_{md} = \frac{1}{N_{\text{ens}} - 1} \sum_{i=1}^{N_{\text{ens}}} (m_i - \bar{m})(d_i - \bar{d})^T, \quad (2)$$

$$C_{dd} = \frac{1}{N_{\text{ens}} - 1} \sum_{i=1}^{N_{\text{ens}}} (d_i - \bar{d})(d_i - \bar{d})^T, \quad (3)$$

where $\bar{m}$ is the mean of state vectors and $\bar{d}$ is the mean of dynamic vectors.

The condition for α_p is as follows:

$$\sum_{p=1}^{N^a} \frac{1}{\alpha_p} = 1, \quad (4)$$

where N^a is the number of assimilations in ES-MDA. ES-MDA updates all state vectors N^a times using an inflated covariance matrix of measurement error compared to the single assimilation of ES [7, 35]. In other words, ES has $N^a = 1$ and $\alpha_{p=1} = 1$ because of the single assimilation.

In equation (1), the perturbed observation d^{unc} is computed as follows:

$$d_i^{\text{unc}} = d_i^{\text{obs}} + \sqrt{a_p} C_D^{1/2} z_{d,i}, \quad \text{for } i = 1, \dots, N_{\text{ens}}, \quad (5)$$

where d^{obs} means the original observed data. On the right-hand side of equation (5), the second term is the perturbation term quantifying reservoir uncertainty caused by data measurement, processing, and interpretation. The stochastic feature of C_D is realized by $z_d \sim N(0, I_{N_d})$, where z_d is the random error matrix of observations generated with a mean of zero and a standard deviation of I_{N_d} , where N_d is the number of time steps of observation data.

2.2. Autoencoder, Denoising Autoencoder, and Serial Denoising Autoencoder

2.2.1. Autoencoder. AE is an unsupervised learning neural network that enables encoding given data compactly on a manifold and then decoding the encoded data into the original data space [36]. Here, the manifold refers to the dimension that represents essential features of the original data [33, 37]. As a well-designed manifold is useful for data compression and restoration, AE has been recently utilized as a preprocessing tool for feature extraction of the reservoir models [18, 38]. Figure 1(a) is a schematic diagram of AE that shows compression and reconstruction of a channelized reservoir model composed of two facies: sand channels with high permeability and shale background with low permeability. Throughout this paper, indicators for shale and sand facies are 0 and 1, respectively (see the original reservoir model in Figure 1(a)). As a multilayer neural network, AE typically consists of three types of layers: one input layer, one or more hidden layers, and one output layer. Each layer is composed of interconnected units called neurons or nodes [39]. In Figure 1, orange and peach

circles indicate original and reconstructed data. Dark blue diamonds and purple squares represent encoded and double-encoded coefficients, respectively. Light blue diamonds are reconstructed double-encoded coefficients.

If an original reservoir model m is imported to the input layer, the encoded model h is as follows:

$$h = f_{\text{AE}}(m) = W_{\text{enc}} m + b_{\text{enc}}, \quad (6)$$

where $f_{\text{AE}}(\bullet)$ is the encoder of AE, W_{enc} and b_{enc} are the weight matrix and bias vector for $f_{\text{AE}}(\bullet)$, respectively. The subscript enc refers to encoding. For example, in Figure 1(a), m composed of facies indexes in 5,625 gridblocks is compressed into 2,500 coefficients denoted as h_1 in the hidden layer.

The above encoding process is followed by the decoding process as follows:

$$\hat{m} = g_{\text{AE}}(h) = W_{\text{dec}} h + b_{\text{dec}}, \quad (7)$$

where $\hat{m}$ is the reconstructed reservoir model, $g_{\text{AE}}(\bullet)$ is the decoder of AE, W_{dec} is the weight matrix for $g_{\text{AE}}(\bullet)$, and b_{dec} is the bias vector for $g_{\text{AE}}(\bullet)$. The subscript dec refers to decoding. In Figure 1(a), the encoded coefficients h_1 are reconstructed as $\hat{m}$ in the output layer. In Figure 1(b), the encoded h_1 is encoded again into h_2 . For feature extraction, the number of nodes in hidden layers is smaller than that of the input layer. For data reconstruction, the number of nodes in the output layer is the same as that in the input layer.

Training AE indicates tuning W_{enc} , b_{enc} , W_{dec} , and b_{dec} in the direction of minimizing the dissimilarity between the original model m and the reconstructed model $\hat{m}$. The dissimilarity is quantified as the loss function E given as follows:

$$E = \frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} \sum_{j=1}^{N_{\text{para}}} (m_{ij} - \hat{m}_{ij})^2 + \lambda \Omega_W + \beta \Omega_s, \quad (8)$$

where N_{train} is the number of the reservoir models used for AE training, N_{para} is the number of parameters in each reservoir realization, m_{ij} is the j th model parameter value for the i th model, λ is the coefficient for the L2 regularization, Ω_W is the sum of squared weights, β is the coefficient for the sparsity regularization, and Ω_s is the sparsity of network links between nodes of adjacent layers. Ω_W and Ω_s are given as follows:

$$\Omega_W = \frac{1}{2} \sum_{i=1}^{N_{\text{train}}} \sum_{j=1}^{N_{\text{para}}} w_{ij}^2, \quad (9)$$

$$\Omega_s = \sum_{k=1}^{N_{\text{node}}} \rho \log \left(\frac{\rho}{\hat{\rho}_k} \right) + (1 - \rho) \log \left(\frac{1 - \rho}{1 - \hat{\rho}_k} \right), \quad (10)$$

$$\hat{\rho}_k = \frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} h_k(m_i), \quad (11)$$

FIGURE 1: Two autoencoders (a) and (b) used to construct a stacked autoencoder (c).

where w_{ij} is the weight for a node of the j th parameter of the i th model, N_{node} is the number of nodes in a hidden layer, ρ is a desired value for the average output activation measure, $\hat{\rho}_k$ is the average output activation measure of the k th node in a hidden layer, and $h_k(m_i)$ is an assigned value in that k th node [40].

For further feature extraction, an AE (Figure 1(b)) can be nested in another AE, as shown in Figure 1(c). This nested AE is called a stacked AE (SAE) [33]. In Figure 1(b), the encoded model h_1 composed of 2,500

coefficients is compressed into another encoded model h_2 composed of 465 coefficients. Figure 1(c) is a combination of Figures 1(a) and 1(b). In Figure 1(c), h_2 is expanded and becomes the reconstructed model $\hat{m}$ composed of 5,625 gridblocks via the reconstructed encoded model $\hat{h}_1$ composed of 2,500 coefficients.

2.2.2. Denoising Autoencoder. DAE is an AE trained with noise data as inputs and clean data as outputs. A well-trained DAE is expected to be able to refine reservoir

FIGURE 2: Schematic diagram of manifold learning using a denoising autoencoder (DAE) for geomodels: (a) generating the original models as training outputs; (b) noising the original models stochastically as training inputs; (c) learning of purification from the noise models to the original models; (d) building of a manifold dimension; (e) applying DAE to the noise models obtained from ES-MDA for obtaining the purified models; (f) delivering the purified models to ES-MDA.

realizations updated at each data assimilation and make the realizations preserve clean channel features in terms of shape and connectivity. This denoising process is also called purification in this study. Figure 2 shows a procedure of DAE applied to the purification of a channelized reservoir. For obtaining training data of DAE in this study, the clean original models are generated using a multipoint statistics modelling technique called single normal equation simulation (SNESIM) [41] (Figure 2(a)). Black solid circles in Figure 2(a) and m of Figure 2(b) are the original models which are corrupted stochastically with artificial noise using the conditional probability, $q(\tilde{m}|m)$ (Figure 2(b)). The noise models including $\tilde{m}$ are presented with red balls. All the noise models are marked with red colors. In Figure 2(c), black dotted circles and dashed lines indicate the reconstructed models and training the DAE, respectively. The training of DAE is a process to grasp out a manifold and is displayed as a black curve (Figure 2(d)). The main dimension is reflected to represent the original models and purified reservoir models derived from the corresponding noise models. The reconstructed models of Figure 2(c) would be located nearby the manifold if the training is well designed. Once a trained DAE is obtained, Figures 2(e) and 2(f) show the purification of models by regarding the updated models at each assimilation of ES-MDA as the noise models. A noise model $\tilde{m}$ is reconstructed as $\hat{m}$ through the following process:

$$\hat{m} = g_{\text{DAE}}(f_{\text{DAE}}(\tilde{m})), \quad (12)$$

where $f_{\text{DAE}}(\bullet)$ and $g_{\text{DAE}}(\bullet)$ are the encoder and decoder of DAE, respectively. Note that $\hat{m}$ becomes the prior model m_i^b in equation (1).

2.2.3. Types of Noise. Two noise types are considered artificial noises that might occur unexpectedly during data assimilation: salt and pepper noise (SPN) [42] and Gaussian noise (GN) [43]. SPN can be caused by sharp and sudden disturbances in the image signal. GN is statistical noise having a probability density function equal to that of the Gaussian distribution [43]. Both SPN and GN are typical noise types in digital image recognition and so have been used for DAEs [33].

Figure 3 compares a clean reservoir model, the model corrupted with SPN, and the model corrupted with GN. In Figure 3(a), the clean model consists of two facies values: 0 (shale) and 1 (sand). Because the facies value of a grid cell blurred with SPN is converted either from 0 to 1 or from 1 to 0, SPN makes mosaic effects that might break and disconnect sand channels in the shale background while inducing sparse sand facies in the shale background (Figure 3(b)). Grid cells to be perturbed are randomly selected using a specific ratio over the reservoir domain. The range of GN values is $[-1, 1]$. For preserving the range $[0, 1]$ of given data, the facies value at a grid cell is set as 0 if the value is negative after GN is added. The value is 1 if any facies value added with GN is greater than 1 (Figure 3(c)). In brief, the need to remove two noise types necessitates the design of serial DAEs as an auxiliary module embedded in an ensemble-based data assimilation algorithm.

Figure 4 shows how trained SPN and GN filters denoise a channelized reservoir model updated using ES-MDA. This study executes SPN and GN filters sequentially. Grid cells of the updated model have facies values following the Gaussian distribution, as seen in the histogram of Figure 4(a). Recall that the ideal facies models follow the discrete bimodal distribution (not the Gaussian) before import to the simulator.

FIGURE 3: Three realizations of the (a) original, (b) salt and pepper noise, and (c) Gaussian noise reservoir models.

FIGURE 4: (a) A reservoir realization updated using ES-MDA. (b) A purified model using a salt and pepper noise (SPN) filter. (c) A further purified model using a Gaussian noise (GN) filter on (b).

Using the SPN filter (Figure 4(b)), a purified model yields the histogram following the bimodal distribution. However, it still reveals blurred channel borders. Some grid values are neither 0 (shale) nor 1 (sand). To obtain more distinct channel traits, the GN filter is applied to Figure 4(b) which yields Figure 4(c). After the GN filtering, a cutoff method is carried out to keep every facies value as either 0 or 1 for reservoir simulation [23].

2.2.4. Serial Denoising Autoencoder. This research proposes a serial denoising autoencoder (SDAE) with consideration of the two noise types. Figure 5 describes the operating procedure of the SDAE denoising a channelized reservoir model. Figure 5(a) shows a DAE trained with the reservoir models corrupted with SPN. Figure 5(b) is a trained DAE with GN. The two neural networks are herein called the SPN filter and the GN filter. Orange circles are original data. Red circles correspond to data noise with SPN or

GN. Dark and light green diamonds indicate encoded coefficients in the hidden layers of SPN and GN filters, respectively. Peach colored circles indicate reconstructed data. m , $\tilde{m}$, and $\hat{m}$ represent the original, noise, and reconstructed models, respectively. In Figure 5(c), $\tilde{m}$ is imported to the input layer of the SDAE. The quality of $\tilde{m}$ is improved (in terms of geological plausibility such as channel pattern and connectivity) using the SPN and GN filters. It is expected that the output of the SDAE $\hat{m}_{GN}$ will be similar to the corresponding original model m during training the SDAE. During data assimilation, $\hat{m}_{GN}$ becomes m_i^b in equation (1) after passing the cutoff method.

2.3. Extraction of Geologic Features Using Discrete Cosine Transform. An efficient reduction of the number of parameters can contribute to improving the history matching performance [1, 15, 17]. Discrete cosine transform (DCT) presents finite data points in a sum of coefficients of cosine functions at different frequencies [44]. Figure 6 depicts

FIGURE 5: (a) A denoising autoencoder eliminating salt and pepper noise. (b) A denoising autoencoder eliminating Gaussian noise. (c) A serial denoising autoencoder with the original, noise, and purified reservoir models.

FIGURE 6: Example of discrete cosine transform (DCT) and inverse DCT (IDCT) applied to the reproduction of shale and sand facies of a channelized reservoir.

an example that applies DCT for extracting features of a channelized reservoir model. Figure 6(a) shows an image of physical parameters (e.g., facies) for a channel reservoir. The DCT application to Figure 6(a) yields Figure 6(b) which shows a distribution of DCT coefficients. In Figure 6(b), DCT coefficients are arranged following an order of cosine frequencies: the upper left part is filled with lower (i.e., more essential) frequencies of cosine functions, and the lower right part is filled with higher frequencies of the functions. DCT coefficients in the lower-frequency region (regarded as essential) have higher energy (for representing the channel image) than those in the higher-frequency region. The total number of DCT coefficients is the same as the number of gridblocks: $75 \times 75 = 5,625$. The number of the upper left coefficients is 465 which is equal to the sum $\sum_{i=1}^{30} i$: from 30 DCT coefficients on the 1st row to 1 DCT coefficient on the 30th row. Throughout this study, the number of essential rows is constant at 30. Hence, the number of essential DCT coefficients N_{DCT} becomes 465 if any data assimilation algorithm is combined with DCT. Thus, the data compression ratio is approximately 12.1 ($\approx 5,625/465$). In Figure 6(c), the coefficients in the upper left part within the red dotted triangle are preserved while the other coefficients are assumed as zero (corresponding to the negative infinity in the color scale). Figure 6(d) shows that a channel image reconstructed using the 465 DCT coefficients is similar to the original image shown in Figure 6(a). This reconstruction is referred to as inverse DCT (IDCT). Due to the data compression, the channel borders get blurred in Figure 6(d). Nevertheless, it seems that Figure 6(d) reliably restores the main channel trend of Figure 6(a).

2.4. Construction of Geologic Dictionaries. Sparse coding is the process used to calculate representative coefficients for

the prototype models composing a geologic dictionary [45–47]. The premise of sparse coding is that geomodels are presented with a weighted linear combination of the prototype models [48, 49]. Once a library Y is built with a large number of sample geomodels, K-SVD extracts essential features from Y and then constructs both the dictionary matrix D and its weight matrix X : $Y \cong DX$ [16]. Orthogonal matching pursuit (OMP) aids the decomposition of Y [50, 51].

Figure 7 compares sparse coding to construct geologic dictionaries using K-SVD and OMP in the original facies domain and in the DCT domain. The procedure starts with constructing the library matrix Y (a N_{para} by N_{lib} matrix in Figure 7(c)) which consists of a variety of channel reservoir realizations generated by SNESIM (Figure 7(b)) with a given training image (Figure 7(a)). Herein, N_{para} is the number of parameters in each reservoir realization and N_{lib} is the number of reservoir realizations in Y . In Figure 7(c), N_{para} equals N_{grid} . Applying OMP and K-SVD decomposes Y into D and X (Figure 7(d)). Strictly speaking, the multiplication of D and X produces Y' , which is the reconstructed Y (see the right-hand side of Figure 7(d)). D is a N_{grid} by N_{dict} matrix and X is a N_{dict} by N_{lib} matrix. D and Y' are visualized in Figures 8(a) and 8(b), respectively.

The above procedure to construct D and X can be carried out in the DCT domain as well if each reservoir realization is transformed into DCT coefficients (Figure 7(e)). For this modified sparse coding, Figure 7(f) shows that applying K-SVD and OMP builds D_{DCT} (Figure 8(c)) and Y_{DCT}' (Figure 8(d)) of which the dimensions are N_{DCT} by N_{dict} and N_{dict} by N_{lib} , respectively. It appears that both procedures sufficiently capture the channel connectivity and pattern of the original realizations (compare Figures 8(b) and 8(d) with Figure 7(b)). Also, N_{DCT} is typically smaller than N_{grid} for dimensionality reduction. For this reason, the

FIGURE 7: Construction of sparse geologic dictionaries using K-SVD without and with DCT: (a) training image, (b) generation of the initial channel models (Y) using SNESIM, (c) organization of the models in a matrix, (e) transformation of Y into DCT coefficients, and construction of D and X from Y using K-SVD in (d) facies domain and (f) in DCT domain.

computational cost of sparse coding is reduced more in the DCT domain than in the original grid domain [16, 23]. Furthermore, the previous work by the authors [23] claimed that iterating the modified sparse coding has the potential to improve the overall history matching performance of a channelized reservoir by updating the geologic dictionary in every assimilation of ES-MDA. N_{qual} qualified ensemble members are selected for the efficient update of the geologic dictionary. More details on iterative update of sparse geologic dictionaries can be found in [23].

2.5. Integration of DCT, K-SVD, and SAE in ES-MDA with SDAE. In this study, ten variants of ES-MDA are investigated to analyze the effects of SDAE on history matching performance of a channelized reservoir. Table 1 summarizes state vectors and postprocesses of the ten ES-MDA algorithms. Note, some algorithms are integrated with one or more of the transformation techniques addressed in Section 2. The first to fifth ES-MDA algorithms update the reservoir models without the SDAE. The sixth to tenth ES-MDA algorithms (which correspond to the first to fifth algorithms in numerical order) apply the SDAE as a noise remover to the ensemble update addressed in equation (1).

Figure 9 shows the flowchart of the ten algorithms. First, the N_{lib} (thousands of) reservoir models are realized using SNESIM (Figure 9(a)) for considering various geological scenarios [17, 18, 23]. For the parameterization techniques, the K-SVD uses the whole realization pool for constructing the geologic dictionary (see the left box of Figure 9(a)). The SAE and SDAE utilize some realizations as their training data. The initial ensemble is composed of randomly selected N_{ens} realizations from the pool (see the right box of Figure 9(a)). As shown in Figure 9(b), forward reservoir simulation is run for the initial ensemble and the initial parameters are imported to the ten algorithms (depending on their transformation techniques). For each ES-MDA algorithm, the transformed parameters are updated using the Kalman gain (Figure 9(c)). For example, for the first algorithm, the facies indexes (i.e., 0 for shale and 1 for

sand) are the target parameters of the reservoir models updated using the conventional ES-MDA without any transformation. These original coefficients are transformed into the DCT domain for the second algorithm. The third algorithm updates weight coefficients of K-SVD [23]. The fourth algorithm adjusts weight coefficients of K-SVD in the DCT domain with an iterative update of the dictionary matrix [23]. The fifth algorithm updates coefficients encoded using SAE (as described in Section 2.2.1). It should be clarified that the SAE and SDAE have different purposes. Similar to the DCT, the SAE is used to represent the facies distribution of a reservoir model in a lower dimension. The number of nodes in the hidden layer of the SAE equals the number of representatives. Meanwhile, the SDAE introduced in Section 2.2.4 aims at purifying the reservoir models in each data assimilation.

The updated reservoir parameters are retransformed into the facies domain to figure out the updated reservoir realizations in the physical state (see the left box in Figure 9(d)). Neither 0 nor 1 facies values are changed as 0 or 1 using the cut-off (see the right box in Figure 9(d)). In this study, 0.5 is the threshold facies value distinguishing sand from shale. The ensemble update is repeated until the assimilation count p reaches the number of assimilations N^a (Figure 9(e)). After the final assimilation is complete, well behaviors are predicted through forward simulation for the updated reservoir models (Figure 9(f)).

3. Results and Discussion

The ten ES-MDA algorithms addressed in Table 1 were applied to history matching and production forecasts of a channelized gas reservoir to investigate the efficacy of denoising using the proposed SDAE on ensemble-based reservoir model calibration. Section 3.1 provides the field description and experimental settings for the algorithms. Section 3.2 describes experimental settings for SAE and SDAE. The simulation results of the ten algorithms are compared

FIGURE 8: Reservoir realizations of dictionaries and reconstructed libraries from Figure 7. (a, b) D and Y' of Figure 7(d), respectively. (c, d) D_{DCT} and Y_{DCT}' of Figure 7(f), respectively.

TABLE 1: Comparison of state vectors and postprocesses after each assimilation for ten ES-MDA algorithms investigated in this study.

	State vector m	Postprocess after every assimilation
Algorithm w/o denoising		
ES-MDA	Facies index	
ES-MDA-DCT	DCT coefficients in facies domain	
ES-MDA-K-SVD	Weight matrix X in facies domain	Cutoff
ES-MDA-DCT-i-K-SVD	X in DCT domain	
ES-MDA-SAE	Encoded coefficients in facies domain	
Algorithm w/ denoising		
ES-MDA	Facies index	SPN filter
ES-MDA-DCT	DCT coefficients in facies domain	↓
ES-MDA-K-SVD	Weight matrix X in facies domain	GN filter
ES-MDA-DCT-i-K-SVD	X in DCT domain	↓
ES-MDA-SAE	Encoded coefficients in facies domain	Cutoff

FIGURE 9: Flowchart of ten ES-MDA algorithms conducted in this study.

regarding history-matched and history-predicted production rates (Section 3.3), updated facies distribution (Section 3.4), and error estimation (Section 3.5).

3.1. Field Description. Table 2 summarizes properties of a channelized gas reservoir model applied to the ES-MDA algorithms. This gas reservoir is composed of two facies: sand and shale. Boundaries of the gas reservoir are attached to a numerical aquifer modelled by pore volume multiplication.

Figure 10 shows the training image (Figure 10(a)) and reference model (Figure 10(b)) employed for the ten algorithms. Sixteen vertical gas production wells are set up: eight wells (P1, P4, P6, P7, P9, P12, P14, and P15) are drilled in the sand formation while the other eight wells (P2, P3, P5, P8, P10, P11, P13, and P16) are drilled in the shale formation. Facies information at the well locations

are used as hard data of SNESIM that realizes the reference model and N_{lib} reservoir realizations.

Table 3 describes well coordinates, operating conditions, and simulation period for history matching and forecast. The total simulation period was 7,000 days: 3,500 days for history matching was followed by production forecasts for 3,500 days. Target parameters of history matching were gas production rate and bottomhole pressure (BHP). Water production rate was regarded as the watch parameter (thus excluded from the matching targets).

Table 4 presents the number of the reservoir models and parameters used for the ten algorithms. $N^a = 4$ and $\alpha_p = 4$ according to equation (4) for all the ES-MDA algorithms.

3.2. Experimental Settings for SAE and SDAE. Recall that the SDAE was designed for denoising updated ensemble members, while the SAE was adopted as a feature extraction tool

TABLE 2: Reservoir properties of the channelized gas reservoir.

Parameter	Value
Number of gridblocks in the x -direction (dimensionless)	75
Number of gridblocks in the y -direction (dimensionless)	75
Number of gridblocks in the z -direction (dimensionless)	1
Grid size (ft ³)	200 × 200 × 100
Initial gas saturation (fraction)	0.75
Initial water saturation (fraction)	0.25
Initial reservoir pressure (psia)	3,000
Index of sand facies (dimensionless)	1
Index of shale facies (dimensionless)	0
Permeability of sand facies (md)	300
Permeability of shale facies (md)	0.1

(such as the DCT). All autoencoders were developed using the *trainAutoencoder* toolbox in MATLAB [40].

Table 5 describes experimental settings for the SDAE. As the SDAE was the sequence of SPN and GN filters (Figure 5(c)), the number of hidden nodes in each filter was the same. With the 15% visiting probability, SPN altered the rock facies values of the visited gridblocks either from 0 to 1 or vice versa for each training the reservoir model. The SPN filter was trained with 2,100 noise reservoir models: 700 clean reservoir models were equiprobably noise three times. The number of the reservoir models used for training the GN filter was 2,000. All the training models originated from one clean model. For each training model, GN contaminated all gridblocks with the mean of 0 and standard deviation of 0.35. If a contaminated value of a gridblock was smaller than the minimum facies index of 0, the minimum was assigned to that gridblock. Also, the maximum of 1 was assigned if a value exceeded the maximum.

Table 6 describes hyperparameters used for the SAE. 5,625 gridblocks were represented by 465 node values in the second hidden layer via 2,500 node values in the first hidden layers. Note, the number of SAE coefficients in the second hidden layer is kept equal to the number of DCT coefficients for a fair comparison throughout the study.

3.3. History Matching and Forecasts of Dynamic Data. Figure 11 presents profiles of gas production rate during the 10-year history matching and the following 10-year prediction periods. Figures 11(a)–11(e) are the profiles obtained using the five ES-MDA algorithms uncoupled with SDAE. Figures 11(f)–11(j) are those obtained using the algorithms coupled with SDAE. In each subfigure, the solid grey and light blue lines represent the production behaviors of the initial and final updated ensembles, respectively. The dark blue line corresponds to the mean of the final ensemble. The red line indicates the production profile from the reference model (Figure 10(b)) regarded as actual data. The profiles at the production wells (P1, P4, P9, and P15) located

on the sand formation were presented because these wells near the reservoir boundary were sensitive to the aquifer water influx in this case study.

For all ten ES-MDA algorithms, the updated ensembles decreased the discrepancies between the reference and updated ensemble models compared to the initial ensembles. Furthermore, the comparison of the subfigures indicates the denoising using the SDAE was effective to improve both matching and prediction accuracy during data assimilation. The five ES-MDA algorithms with SDAE yielded better performance (Figures 11(f)–11(j)) than the uncoupled algorithms (Figures 11(a)–11(e)) after the assimilations were complete. For the updated ensembles, reservoir uncertainty was somewhat left at well P15 during the prediction period. This was because the decrease in gas rate due to water breakthrough was hardly observed at this well during the history matching period. As shown in the reference model, the inflow from the numerical aquifer could arrive at well P15 after breaking through wells P12 or P14. This late water breakthrough caused the remaining uncertainty at well P15 despite the satisfactory assimilation results at the other wells. Figure 12 shows well BHP profiles during both periods. Every final ensemble got conditioned to the reference data sufficiently. This yielded the narrow bandwidth of the simulation results including the reference data at most wells. Also, denoising effects caused by the use of the SDAE were captured at well P15.

Figure 13 compares matched and predicted water production rate at the four production wells. Discrepancies between the updated ensemble mean profiles (dark blue lines) and the reference profiles (red lines) remained in the results of the ES-MDA algorithms without SDAE. For example, simulation results were somewhat unmatched to observations at well P1 in Figure 13(a) and at well P4 in Figure 13(b). The SDAE was effective in correcting these discrepancies. When comparing Figures 13(a)–13(e) and corresponding Figures 13(f)–13(j), both matching and prediction accuracy improved due to the coupling of SDAE and ES-MDA. In particular, remarkable improvements due to denoising were captured in prediction at wells P4 and P9. At wells P9 and P15, discrepancies were observed but acceptable considering water rate was used as the watch parameter and not used for history matching.

3.4. Distribution of Facies and Permeability after History Matching. Figure 14 presents the evolution of an ensemble member obtained using the ES-MDA coupled with the SDAE method over four assimilations. The row number indicates the assimilation sequence. The first column presents the ensemble member obtained using ES-MDA (Figure 14(a)). The second column shows the member denoise using the SPN filter (Figure 14(b)). The denoise model was purified again using the GN filter, as shown in the third column (Figure 14(c)). Channel features blurred in Figure 14(a) improved significantly by passing the two filters in sequence. The filtering functionality of the trained SDAE refined the facies value of each gridblock in the ensemble member similar to 0 (i.e., shale) or 1 (i.e., sand). Finally, applying the cutoff

FIGURE 10: Training images and the reference model used for history matching. P1 to P16 indicate well numbers.

TABLE 3: Experimental settings for reservoir simulation.

Parameter	Value
Observed well data	Gas rate and BHP
Maximum well gas production rate (Mscf/day)	15,000
Minimum well BHP (psia)	1,000
Total simulation period (days)	7,000
History matching period (days)	3,500
Prediction period (days)	3,500
Coordinates of well locations in sand facies (dimensionless)	(14, 14), (62, 14), (30, 30), (46, 30), (14, 46), (62, 46), (30, 62), (46, 62)
Coordinates of well locations in shale facies (dimensionless)	(30, 14), (46, 14), (14, 30), (62, 30), (30, 46), (46, 46), (14, 62), (62, 62)

TABLE 4: Number of parameters used for the ten ES-MDA algorithms.

Parameter	Symbol	Value
Number of the static reservoir models used for constructing an initial library matrix Y	N_{lib}	3,000
Number of the static reservoir models used for constructing an initial dictionary matrix D	N_{dict}	1,000
Number of gridblocks in each reservoir model	N_{grid}	5,625
Number of filtered DCT coefficients in each reservoir model	N_{DCT}	465
Number of ensemble members	N_{ens}	100
Number of qualified ensemble members	N_{qual}	20

to the filtered model yielded the prior model of the next assimilation (Figure 14(d)). The cutoff delivered the models only composed of sand and shale facies.

Figure 15 compares the updated permeability distributions obtained using the ten ES-MDA algorithms. The first row of Figure 15 deploys the reference field and the mean of the initial ensemble members. The initial ensemble mean reveals high dissimilarity to the reference in terms of channel

TABLE 5: Experimental settings for the SDAE.

Parameter	Value or function	
Transfer function	Logistic sigmoid function	
Backpropagation algorithm	Levenberg-Marquardt	
	SPN filter	GN filter
Number of nodes in the hidden layer	5,000	5,000
Number of the reservoir models used for training the filter	2,100	2,000

TABLE 6: Experimental settings for the SAE.

Parameter	Value or function
Transfer function	Logistic sigmoid function
Backpropagation algorithm	Levenberg-Marquardt
Number of nodes in the first hidden layer	2,500
Number of nodes in the second hidden layer	465
Number of nodes in the third hidden layer	2,500
Number of the models used for training the SAE	100

connectivity and pattern. The average maps of the updated ensemble members obtained using the ten algorithms are arrayed in the second and third rows. The conventional ES-MDA without SDAE had broken and thus had geologically less plausible channels due to the direct perturbation of gridblock pixels (i.e., facies) (Figure 15(a)). Though coupling DCT with ES-MDA reduced the pixel heterogeneity, the quality of the ensemble mean was less satisfactory. ES-MDA-K-SVD showed better results than the two previous algorithms. However, there was a room for improvement regarding connectivity between wells P9 and P14 (Figure 15(c)). For the ES-MDA-DCT-i-K-SVD, inconsistent channel widths and broken connectivity between wells P1 and P6 were observed (Figure 15(d)). Similar to the ES-MDA-K-SVD, ES-MDA-SAE suffered from the connectivity issue (Figure 15(e)).

FIGURE 11: Profiles of gas production rate at the four wells (P1, P4, P9, and P15) obtained by executing the ten ES-MDA algorithms. The latter five algorithms from (f) to (j) are the algorithms coupled with SDAE.

When comparing the plots on the second and third rows in the same column, the results obtained using the five ES-MDA algorithms with SDAE (Figures 15(f)–15(j)) preserved the main X-shaped channel patterns with consistent channel width, though Figures 15(g)–15(j) had unexpected channel connectivity between wells P9 and P14. In the case of Figures 15(b) and 15(g), it seems that coupling feature extraction techniques (such as DCT and SAE) might deteriorate history matching performance due to data compression of a reservoir realization. The same issue was seen in Figures 15(e) and 15(j). An improvement in the results is expected if an optimal number of essential coefficients and hyperparameters are used for the transformation. In

summary, the above results imply a well-trained machine learning-based noise remover has the potential to preserve geological plausibility of a target reservoir model during ensemble-based history matching.

Figure 16 displays error histograms of facies distributions for the final ensembles. The error equals $N_m/N_{\text{grid}} \times 100(\%)$, where N_m is the number of facies-unmatched gridblocks where the updated facies are different from the facies in the reference model. Histograms are scaled 0 to 50 for the y-axis to make the results more readable. Note, the sum of the frequencies for ensemble members is 100 in each histogram. For the initial ensemble, the range of errors is between 17 and 37% (see the first row of Figure 16). All

FIGURE 12: Profiles of BHP at the four wells (P1, P4, P9, and P15) obtained by executing the ten ES-MDA algorithms. The latter five algorithms from (f) to (j) are the algorithms coupled with SDAE.

ten algorithms decreased errors compared to the initial ensemble. The ES-MDA-SAE had a smaller error range (Figures 16(e) and 16(j)) than the other algorithms uncoupled (Figures 16(a)–16(d)) or coupled with the SDAE (Figures 16(f)–16(i)). In Figures 16(e) and 16(j), histogram bars are cut because of the scale. Embedding the SDAE in the ES-MDA contributed to reducing errors regardless of which transformation technique was combined with the ensemble-based data assimilation algorithm. Note, the results of the quantitative error analysis addressed in Figure 16 do not often correspond to the qualitative analysis

of geological plausibility shown in Figure 15. For example, it appears Figure 15(i) has more similar geological patterns to the reference model in comparison with Figure 15(j). Also, Figure 16(i) had higher error values than Figure 16(j). This incompatibility emphasizes the importance of a multisource data interpretation.

Tables 7 and 8 summarize the discrepancies between observation and simulation results for dynamic data (gas rate, water rate, and BHP) during the history matching and prediction periods, respectively. The discrepancies were calculated for the wells located on the sand channel as the

FIGURE 13: Profiles of water production rate at the four wells (P1, P4, P9, and P15) obtained by executing the ten ES-MDA algorithms. The latter five algorithms from (f) to (j) are the algorithms coupled with SDAE.

cumulative gas production was insignificant at the wells on the shale background.

The quality of the updated ensemble was quantified using the equations as follows:

$$\varepsilon_i = \frac{1}{N_d} \sum_{j=1}^{N_d} \left(\frac{d_j - d_j^{\text{obs}}}{d_j^{\text{obs}}} \right)^2, \quad \text{for } i = 1, \dots, N_{\text{ens}}, \quad (13)$$

$$\mu_\varepsilon = \frac{(1/N_{\text{ens}}) \sum_{i=1}^{N_{\text{ens}}} \varepsilon_i^{\text{upd}}}{(1/N_{\text{ens}}) \sum_{i=1}^{N_{\text{ens}}} \varepsilon_i^{\text{init}}} \times 100(\%), \quad (14)$$

$$\sigma_\varepsilon = \frac{\sqrt{(1/N_{\text{ens}} - 1) \sum_{i=1}^{N_{\text{ens}}} (\varepsilon_i^{\text{upd}} - \overline{\varepsilon_i^{\text{upd}}})^2}}{\sqrt{(1/N_{\text{ens}} - 1) \sum_{i=1}^{N_{\text{ens}}} (\varepsilon_i^{\text{init}} - \overline{\varepsilon_i^{\text{init}}})^2}} \times 100(\%), \quad (15)$$

where $\varepsilon_{i,\text{type}}$ is the error of the i th ensemble member in terms of each dynamic data type, μ_ε is the normalized mean of ε , and σ_ε is the normalized standard deviation of ε . The superscripts upd and init indicate the updated and initial ensemble members, respectively.

The overall ensemble quality was improved using the SDAE as the errors were significantly reduced after SDAE

FIGURE 14: Updated facies distribution results of the first ensemble member in every assimilation by the ES-MDA with SDAE.

activation. The SDAE helped any ES-MDA algorithm reduce μ_ϵ and σ_ϵ . For example, μ_ϵ for the matched gas rate obtained using ES-MDA without and with the SDAE were 20.98% and 5.17%, respectively. σ_ϵ was also reduced from 40.23% to 8.27%. Furthermore, the average values of μ_ϵ and σ_ϵ for the ES-MDA algorithms were decreased for all data types after coupling the SDAE. For the history matching, the average error values of gas rate, water rate, and BHP were improved from 13.44% to 4.95%, from 76.85% to 37.65%, and from 21.04% to 13.14%, respectively. For the prediction, the average error values went from 35.29% to 5.39%, from 6.29% to 6.78%, and from 53.03% to 18.18%, respectively. The ES-MDA-DCT yielded ϵ values greater than 100% for the prediction, which indicated the degradation of the ensemble. This phenomenon claims any less- or unoptimized feature extraction might cause the deterioration of the ensemble quality.

3.5. Computational Costs for the Denoising Process. Table 9 summarizes the computational cost required for executing the transformation and denoising methods embedded in the ES-MDA. The specification of the computer used in this study was Intel (R) Core (TM) i7-8700 CPU at 3.2 GHz with 16 GB RAM. The cost for reservoir simulation was excluded

from Table 9 because each ES-MDA algorithm expended the same cost for the ensemble update. The total number of reservoir simulation runs was 400, which was the product of the ensemble size (100) and the number of assimilations (4). The machine learning-based methods were more expensive than the transformation methods. It took a few seconds for the activation of the DCT to transform and reconstruct reservoir facies in each assimilation. It took approximately 3.6 hours for the K-SVD to construct the library and weight matrices, as addressed in [23]. In contrast, the ES-MDA-DCT-i-K-SVD needed only about one-ninth of the time for the ES-MDA-K-SVD due to the dimension reduction of reservoir parameters by DCT. It took approximately 8.7 hours to train the SAE. The SDAE was the most computationally expensive method. It took 17.9 hours to train the two filters: 7.2 hours for the SPN filter and 10.7 hours for the GN filter. Nonetheless, the overall results addressed in this study highlight the efficacy of the proposed SDAE as a supplementary tool to the data assimilation algorithm for improving the ensemble quality. Once developed, the noise remover could be easily applied to other algorithms without additional learnings. It is also anticipated that the development of computer hardware will enhance the efficacy of soft computing for big data machine learning.

FIGURE 15: Permeability distributions of the ensemble mean maps obtained using the ten ES-MDA algorithms.

FIGURE 16: Error histograms for facies distribution of the final ensembles obtained using the ten ES-MDA algorithms.

TABLE 7: Statistical parameters of history matching errors for gas rate, water rate, and BHP (only for the wells installed in the sand formation). μ_ε and σ_ε refer to the mean and standard deviation of errors, respectively.

	Gas rate		Water rate		BHP	
	μ_ε (%)	σ_ε (%)	μ_ε (%)	σ_ε (%)	μ_ε (%)	σ_ε (%)
ES-MDA without SDAE						
ES-MDA	20.98	40.23	223.13	91.10	24.17	31.73
ES-MDA-DCT	27.74	43.94	41.66	25.55	31.08	32.64
ES-MDA-K-SVD	5.49	13.40	46.73	52.59	13.72	17.32
ES-MDA-DCT-i-K-SVD	2.08	3.66	41.00	19.63	14.96	8.62
ES-MDA-SAE	10.89	12.80	31.70	18.67	21.27	10.69
Average	13.44	22.80	76.85	41.51	21.04	20.20
ES-MDA with SDAE						
ES-MDA	5.17	8.27	39.99	30.40	16.48	21.51
ES-MDA-DCT	4.55	10.08	41.22	31.27	15.37	13.85
ES-MDA-K-SVD	4.60	6.91	53.49	12.48	15.48	12.85
ES-MDA-DCT-i-K-SVD	6.92	8.29	29.34	12.55	9.69	4.82
ES-MDA-SAE	3.50	6.78	24.22	19.82	8.66	15.74
Average	4.95	8.07	37.65	21.30	13.14	13.75

TABLE 8: Statistical parameters of prediction errors for gas rate, water rate, and BHP (only for the wells installed in the sand formation). μ_ε and σ_ε refer to the mean and standard deviation of errors, respectively.

	Gas rate		Water rate		BHP	
	μ_ε (%)	σ_ε (%)	μ_ε (%)	σ_ε (%)	μ_ε (%)	σ_ε (%)
ES-MDA without SDAE						
ES-MDA	48.00	35.27	0.56	0.88	97.46	135.43
ES-MDA-DCT	105.83	136.65	16.03	16.40	88.58	57.90
ES-MDA-K-SVD	6.61	17.23	6.97	8.44	24.93	21.78
ES-MDA-DCT-i-K-SVD	6.44	2.23	2.61	1.92	37.27	37.20
ES-MDA-SAE	9.57	12.58	5.19	2.16	16.90	23.16
Average	35.29	40.79	6.27	5.96	53.03	55.09
ES-MDA with SDAE						
ES-MDA	2.48	3.12	10.28	12.38	19.16	35.29
ES-MDA-DCT	12.98	12.57	1.72	1.66	31.59	14.63
ES-MDA-K-SVD	4.20	1.10	3.77	1.56	13.35	5.15
ES-MDA-DCT-i-K-SVD	4.24	4.19	8.24	3.14	11.22	1.84
ES-MDA-SAE	3.06	2.10	9.89	4.24	15.59	22.39
Average	5.39	4.62	6.78	4.60	18.18	15.86

TABLE 9: Comparison of the computational cost required for executing an auxiliary module embedded in ES-MDA.

Algorithm	Auxiliary module	Computational cost (hour)
ES-MDA	N/A	0.0
ES-MDA-DCT	DCT	Less than 0.1
ES-MDA-K-SVD	K-SVD	3.6
ES-MDA-DCT-i-K-SVD	Iterative K-SVD in DCT domain	0.4
ES-MDA-SAE	SAE	8.7
SDAE	SPN filter + GN filter	$7.2 + 10.7 = 17.9$

4. Conclusions

The SDAE was implemented as the postprocessor of ES-MDA for enhancing the preservation of geological plausibility during ensemble-based history matching. The SDAE is composed of SPN and GN filters facilitating and purifying the updated reservoir models resulting from the smoothing effects. The denoising effects were investigated by comparing the results of the five ES-MDA algorithms coupled with the SDAE and those uncoupled. The application was to history matching of the channelized gas reservoir. The results obtained using ten different algorithms showed the performance difference between the cases with and without the

SDAE in terms of production data matching and geological plausibility. The SDAE showed excellent accuracy in history matching and prediction for gas rate, water rate, and BHP. Executing the SDAE decreased the average matching error by 75% in the ES-MDA results. The SDAE was also efficient for improving the performance of the ES-MDA algorithms combined with the data transformation methods. The improvement in the matching and prediction accuracy of dynamic data resulted from the conservation of geological plausibility achieved using the SDAE. Consequently, the purified models followed the discrete bimodal distribution for mimicking the channelized reservoir while maintaining channel width and connectivity consistently. These results highlight the potential of the machine learning-based noise remover as an efficient auxiliary method that enhances the performance of ensemble-based history matching if the proxy is designed properly at affordable computational cost.

Nomenclature

b_{enc} :	Bias vector of the neural network in an encoder
b_{dec} :	Bias vector of the neural network in a decoder
C_D :	Covariance matrix of observed measurement error
C_{dd} :	Autocovariance matrix of simulation data d
C_{md} :	Cross-covariance matrix of state vector m and simulation data d
d :	Simulation data
d^{obs} :	Observation data
d^{unc} :	Perturbed observation data
$\bar{d}$:	Mean of simulation data
D :	Dictionary matrix
E :	Loss function
$f_{\text{AE}}(\bullet)$:	Encoder function of autoencoder
$f_{\text{DAE}}(\bullet)$:	Encoder function of denoising autoencoder
$g_{\text{AE}}(\bullet)$:	Decoder function of autoencoder
$g_{\text{DAE}}(\bullet)$:	Decoder function of denoising autoencoder
h :	Encoded coefficient
m :	State vector of a reservoir model
$\bar{m}$:	Mean of state vectors
$\hat{m}$:	Noise reservoir model
$\tilde{m}$:	Reconstructed reservoir model
m^{b} :	State vector of a reservoir model before an update
I_{N_d} :	Identity matrix
K :	Number of parameters in a model for training
N :	Number of models for training
N^{a} :	Number of assimilations
N_d :	Number of time steps in observations
N_{DCT} :	Number of essential DCT coefficients
N_{dict} :	Number of reservoir models in the dictionary matrix D
N_{ens} :	Number of ensemble members
N_{grid} :	Number of gridblocks in a reservoir model
N_{lib} :	Number of reservoir models in the library matrix Y
N_{node} :	Number of nodes in a hidden layer
N_{train} :	Number of reservoir models used for AE training
N_{para} :	Number of parameters in a reservoir model
w :	Weight coefficients of a node in a hidden layer
W_{dec} :	Weight matrix of the neural network in a decoder

W_{enc} :	Weight matrix of the neural network in an encoder
x_{kn} :	Each value of model parameters for autoencoder training
X :	Weight matrix
Y :	Library matrix
Y' :	Reconstructed library matrix
z_d :	Random error to observations
α_p :	Inflation coefficients of C_D
β :	Sparsity regularization term in the loss function
ε :	Error
λ :	Coefficient for L2 regularization term
ρ :	Desire average output activation measure
$\hat{\rho}$:	Average output activation measure of a node in a hidden layer
μ :	Mean
Ω_W :	Sum of squared weights in the loss function
Ω_s :	Sparsity of network links between nodes of two layers
σ :	Standard deviation.

Subscripts

AE:	Autoencoder
DAE:	Denoising autoencoder
DCT:	Discrete cosine transform
dec:	Decoder
dict:	Dictionary
enc:	Encoder
ens:	Ensemble
lib:	Library
para:	Parameter
s:	Sparsity
train:	Training
type:	Dynamic data type
W :	Weight.

Superscripts

a:	Assimilation
b:	Before update
dec:	Decoding
enc:	Encoding
init:	Initial
obs:	Observation
upd:	Update.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Disclosure

Part of this study was presented at the 2018 AGU Fall Meeting.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

The authors acknowledge Schlumberger for providing reservoir simulation software. This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2018R1A6A1A08025520) and Global Infrastructure Program through the NRF funded by the Ministry of Science and ICT (NRF-2017K1A3A1A05069660). Dr. Sungil Kim was partially supported by the Basic Research Project of the Korea Institute of Geoscience and Mineral Resources (Project No. GP2017-024).

References

- [1] X. Luo, T. Bhakta, M. Jakobsen, and G. Nævdal, "An ensemble 4D-seismic history-matching framework with sparse representation based on wavelet multiresolution analysis," *SPE Journal*, vol. 22, no. 3, pp. 985–1010, 2017.
- [2] G. Evensen, "Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics," *Journal of Geophysical Research*, vol. 99, no. C5, pp. 10143–10162, 1994.
- [3] G. Nævdal, T. Manneseth, and E. H. Vefring, "Near-well reservoir monitoring through ensemble Kalman filter," in *SPE/DOE Improved Oil Recovery Symposium*, Tulsa, OK, USA, April 2002, SPE-75235-MS.
- [4] F. Bouttier and P. Courtier, *Data Assimilation Concepts and Methods*, Meteorological Training Course Lecture Series, European Centre for Medium-Range Weather Forecasts, 2002.
- [5] D. S. Oliver and Y. Chen, "Recent progress on reservoir history matching: a review," *Computational Geosciences*, vol. 15, no. 1, pp. 185–221, 2011.
- [6] P. J. Van Leeuwen and G. Evensen, "Data assimilation and inverse methods in terms of a probabilistic formulation," *Monthly Weather Review*, vol. 124, no. 12, pp. 2898–2913, 1996.
- [7] A. A. Emerick and A. C. Reynolds, "History matching time-lapse seismic data using the ensemble Kalman filter with multiple data assimilations," *Computational Geosciences*, vol. 16, no. 3, pp. 639–659, 2012.
- [8] K. Park and J. Choe, "Use of ensemble Kalman filter with 3-dimensional reservoir characterization during waterflooding," in *SPE Europec/EAGE Annual Conference and Exhibition*, Vienna, Austria, June 2006, Paper No. 100178.
- [9] R. J. Lorentzen, G. Nævdal, and A. Shafieirad, "Estimating facies fields by use of the ensemble Kalman filter and distance functions—applied to shallow-marine environments," *SPE Journal*, vol. 3, no. 1, pp. 146–158, 2012.
- [10] Y. Shin, H. Jeong, and J. Choe, "Reservoir characterization using an EnKF and a non-parametric approach for highly non-Gaussian permeability fields," *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, vol. 32, no. 16, pp. 1569–1578, 2010.
- [11] K. Lee, S. Jung, and J. Choe, "Ensemble smoother with clustered covariance for 3D channelized reservoirs with geological uncertainty," *Journal of Petroleum Science and Engineering*, vol. 145, pp. 423–435, 2016.
- [12] X. Luo, A. S. Stordal, R. J. Lorentzen, and G. Nævdal, "Iterative ensemble smoother as an approximate solution to a regularized minimum-average-cost problem: theory and applications," *SPE Journal*, vol. 20, no. 5, pp. 962–982, 2015.
- [13] H. Zhou, L. Li, and J. J. Gómez-Hernández, "Characterizing curvilinear features using the localized normal-score ensemble Kalman filter," *Abstract and Applied Analysis*, vol. 2012, Article ID 805707, 18 pages, 2012.
- [14] S. Kim, C. Lee, K. Lee, and J. Choe, "Aquifer characterization of gas reservoirs using ensemble Kalman filter and covariance localization," *Journal of Petroleum Science and Engineering*, vol. 146, pp. 446–456, 2016.
- [15] Y. Zhao, F. Forouzanfar, and A. C. Reynolds, "History matching of multi-facies channelized reservoirs using ES-MDA with common basis DCT," *Computational Geosciences*, vol. 21, no. 5–6, pp. 1343–1364, 2017.
- [16] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [17] F. Sana, K. Katterbauer, T. Y. Al-Naffouri, and I. Hoteit, "Orthogonal matching pursuit for enhanced recovery of sparse geological structures with the ensemble Kalman filter," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 4, pp. 1710–1724, 2016.
- [18] S. A. Canchumuni, A. A. Emerick, and M. A. Pacheco, "Integration of ensemble data assimilation and deep learning for history matching facies models," in *Offshore Technology Conference*, Rio de Janeiro, Brazil, October 2017, OTC-28015-MS.
- [19] B. Jafarpour and D. B. McLaughlin, "Estimating channelized-reservoir permeabilities with the ensemble Kalman filter: the importance of ensemble design," *SPE Journal*, vol. 14, no. 2, pp. 374–388, 2013.
- [20] S. Kim, C. Lee, K. Lee, and J. Choe, "Characterization of channelized gas reservoirs using ensemble Kalman filter with application of discrete cosine transformation," *Energy Exploration & Exploitation*, vol. 34, no. 2, pp. 319–336, 2016.
- [21] S. Kim, C. Lee, K. Lee, and J. Choe, "Characterization of channel oil reservoirs with an aquifer using EnKF, DCT, and PFR," *Energy Exploration & Exploitation*, vol. 34, no. 6, pp. 828–843, 2016.
- [22] S. Kim, H. Jung, K. Lee, and J. Choe, "Initial ensemble design scheme for effective characterization of three-dimensional channel gas reservoirs with an aquifer," *Journal of Energy Resources Technology*, vol. 139, no. 2, article 022911, 2017.
- [23] S. Kim, B. Min, K. Lee, and H. Jeong, "Integration of an iterative update of sparse geologic dictionaries with ES-MDA for history matching of channelized reservoirs," *Geofluids*, vol. 2018, Article ID 1532868, 21 pages, 2018.
- [24] B. H. Min, C. Park, J. M. Kang, H. J. Park, and I. S. Jang, "Optimal well placement based on artificial neural network incorporating the productivity potential," *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, vol. 33, no. 18, pp. 1726–1738, 2011.
- [25] Z. Ma, J. Y. Leung, and S. Zanon, "Integration of artificial intelligence and production data analysis for shale heterogeneity characterization in steam-assisted gravity-drainage reservoirs," *Journal of Petroleum Science and Engineering*, vol. 163, pp. 139–155, 2018.
- [26] I. Jang, S. Oh, Y. Kim, C. Park, and H. Kang, "Well-placement optimisation using sequential artificial neural networks," *Energy Exploration & Exploitation*, vol. 36, no. 3, pp. 433–449, 2017.

- [27] S. Ahn, C. Park, J. Kim, and J. M. Kang, "Data-driven inverse modeling with a pre-trained neural network at heterogeneous channel reservoirs," *Journal of Petroleum Science and Engineering*, vol. 170, pp. 785–796, 2018.
- [28] H. Jeong, A. Y. Sun, J. Lee, and B. Min, "A learning-based data-driven forecast approach for predicting future reservoir performance," *Advances in Water Resources*, vol. 118, pp. 95–109, 2018.
- [29] Y. Liu, W. Sun, and L. J. Durlofsky, "A deep-learning-based geological parameterization for history matching complex models," <https://arxiv.org/abs/1807.02716>.
- [30] M. A. Ranzato, C. Poultney, S. Chopra, and Y. LeCun, "Efficient learning of sparse representations with an energy-based model," in *Advances in Neural Information Processing Systems 19*, pp. 1137–1144, The MIT Press, 2007.
- [31] D. Erhan, Y. Bengio, A. Courville, P. A. Manzagol, and P. Vincent, "Why does unsupervised pre-training help deep learning," *Journal of Machine Learning Research*, vol. 11, pp. 625–660, 2010.
- [32] S. S. Roy, S. I. Hossain, M. A. H. Akhand, and K. Murase, "A robust system for noisy image classification combining denoising autoencoder and convolutional neural network," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 1, pp. 224–235, 2018.
- [33] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. A. Manzagol, "Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.
- [34] L. Chen and W. Y. Deng, "Instance-wise denoising autoencoder for high dimensional data," *Geofluids*, vol. 2016, Article ID 4365372, 13 pages, 2016.
- [35] A. A. Emerick and A. C. Reynolds, "Ensemble smoother with multiple data assimilation," *Computers & Geosciences*, vol. 55, pp. 3–15, 2013.
- [36] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.
- [37] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*, MIT Press, Cambridge, MA, USA, 2006.
- [38] K. Lee, J. Lim, S. Ahn, and J. Kim, "Feature extraction using a deep learning algorithm for uncertainty quantification of channelized reservoirs," *Journal of Petroleum Science and Engineering*, vol. 171, pp. 1007–1022, 2018.
- [39] S. Mohaghegh, "Virtual-intelligence applications in petroleum engineering: part 1—artificial neural networks," *Journal of Petroleum Technology*, vol. 52, no. 9, pp. 64–73, 2013.
- [40] M. H. Beale, M. T. Hagan, and H. B. Demuth, *Deep Learning Toolbox™ Reference*, The MathWorks, Natick, MA, USA, 2018.
- [41] S. Strebelle, "Conditional simulation of complex geological structures using multiple-point statistics," *Mathematical Geology*, vol. 34, no. 1, pp. 1–21, 2002.
- [42] S. Deivalakshmi, S. Sarath, and P. Palanisamy, "Detection and removal of salt and pepper noise in images by improved median filter," in *2011 IEEE Recent Advances in Intelligent Computational Systems*, pp. 363–368, Trivandrum, India, September 2011.
- [43] M. Wang, S. Zheng, X. Li, and X. Qin, "A new image denoising method based on Gaussian filter," in *2014 International Conference on Information Science, Electronics and Electrical Engineering*, pp. 163–167, Sapporo, Japan, April 2014.
- [44] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Transactions on Computers*, vol. C-23, no. 1, pp. 90–93, 1974.
- [45] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. W. Lee, and T. J. Sejnowski, "Dictionary learning algorithms for sparse representation," *Neural Computation*, vol. 15, no. 2, pp. 349–396, 2003.
- [46] M. M. Khaninezhad, B. Jafarpour, and L. Li, "Sparse geologic dictionaries for subsurface flow model calibration: part I. Inversion formulation," *Advances in Water Resources*, vol. 39, pp. 106–121, 2012.
- [47] M. R. Khaninezhad and B. Jafarpour, "Sparse geologic dictionaries for field-scale history matching application," in *SPE Reservoir Simulation Symposium*, Houston, TX, USA, February 2015, SPE-173275-MS.
- [48] L. Li and B. Jafarpour, "Effective solution of nonlinear subsurface flow inverse problems in sparse bases," *Inverse Problems*, vol. 26, no. 10, article 105016, 2010.
- [49] E. Liu and B. Jafarpour, "Learning sparse geologic dictionaries from low-rank representations of facies connectivity for flow model calibration," *Water Resources Research*, vol. 49, no. 10, pp. 7088–7101, 2013.
- [50] T. T. Cai and L. Wang, "Orthogonal matching pursuit for sparse signal recovery with noise," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4680–4688, 2011.
- [51] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.