Correspondence to:

J. Luo,
jian.luo@ce.gatech.edu

Author Contributions:

Conceptualization: Jian Luo
Data curation: Quan Guo
Formal analysis: Quan Guo, Ming Liu
Funding acquisition: Jian Luo
Methodology: Quan Guo, Ming Liu, Jian Luo
Project Administration: Jian Luo
Software: Quan Guo, Ming Liu
Supervision: Jian Luo
Validation: Quan Guo
Visualization: Quan Guo
Writing – original draft: Quan Guo
Writing – review & editing: Jian Luo

Predictive Deep Learning for High-Dimensional Inverse Modeling of Hydraulic Tomography in Gaussian and Non-Gaussian Fields

Quan Guo[1] , **Ming Liu[1]**, and **Jian Luo[1]**

[1]School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA, USA

Abstract Inverse modeling of hydraulic tomography (HT) is computationally expensive for estimating high-dimensional hydrogeologic parameter fields. In this work, we develop a novel method called HT-INV-NN, which combines dimensionality reduction techniques with a predictive deep learning (DL) model to estimate high-dimensional Gaussian and non-Gaussian channel fields. The HT-INV-NN model consists of a predictor that directly learns the inverse process from hydraulic head measurements to latent variables of random fields, and a decoder that generates high-dimensional parameter fields from predicted latent variables. For Gaussian spatially correlated fields, the decoder utilizes principal components derived from spatial covariance, and for non-Gaussian channel fields, a generative adversarial network (GAN) is trained using generated realizations based on a training image (TI). The predictor is a deep neural network calibrated using the reference data obtained from HT forward simulations, which can be implemented in parallel. HT-INV-NN is successfully tested in multiple numerical experiments including steady-state and transient HT for estimating Gaussian fields in 2D and 3D, as well as binary discontinuous or continuous non-Gaussian channel fields. The training process is efficient, and the model structure demonstrates robustness for input data with perturbations. The model performance on multiple validation data sets are satisfactory when compared with other numerical and deep learning methods.

Plain Language Summary This study introduces an innovative approach for solving complex groundwater flow problems. In hydraulic tomography, we aim to estimate hydraulic properties of underground aquifers by analyzing water pressure measurements from wells. Traditionally, this process involves computationally intensive simulations and optimization techniques. We propose a predictive deep learning model called HT-INV-NN, which directly predicts the unknown hydraulic properties using machine learning algorithms. This eliminates the need for time-consuming simulations and optimization, making the inverse modeling process much more efficient. We conducted a series of numerical experiments using both Gaussian and non-Gaussian fields to evaluate the performance of our model. The results demonstrate that HT-INV-NN achieves high accuracy in estimating the hydraulic properties, even in challenging scenarios with complex channel patterns and varying smoothness levels. Our findings highlight the effectiveness and versatility of HT-INV-NN in handling different geostatistical variations and field characteristics. By leveraging the power of predictive deep learning, our approach offers a faster and more accurate solution to inverse modeling in hydraulic tomography. This has significant implications for groundwater management, environmental monitoring, and decision-making processes. Overall, our study presents a promising advancement in hydraulic tomography and contributes to the growing body of research on predictive modeling using deep learning techniques.

1. Introduction

Inverse modeling of hydraulic tomography (HT) estimates spatially distributed fields of hydrogeological parameters, such as hydraulic conductivity, transmissivity and specific storage, using steady or time-dependent hydraulic head measurements in sequential pumping tests (Butler Jr. et al., 1999; Cardiff et al., 2009; Gottlieb & Dietrich, 1995; Huang et al., 2022; Illman et al., 2008, 2010; Liu et al., 2013; Liu et al., 2014; Tosaka et al., 1993; Yeh & Lee, 2007; Yeh & Liu, 2000; Zha et al., 2018; Zhu & Yeh, 2005). The most widely used approach for solving HT inverse problems is geostatistical approach (GA), including quasilinear GA (Fienen et al., 2008; Kitanidis, 1995) and successive linear estimator (SLE) (Yeh et al., 1995) The major bottleneck of GA is that it requires iterative forward model simulations to evaluate the Jacobian matrix, which are computationally

expensive for large-scale, high-dimensional models. Many efforts have been invested to save the computation of GA through reducing the number of forward simulations or accelerating the computational implementation (Ambikasaran et al., 2013; Broyden, 1965; Kitanidis & Lee, 2014; Klein et al., 2017; Nowak & Cirpka, 2004; Nowak et al., 2003; Saibaba et al., 2012; Zhao & Luo, 2021a; Zhao et al., 2022).

Dimensionality reduction is usually used for HT inverse modeling for estimating high-dimensional parameter fields (Lee & Kitanidis, 2014; Zhao & Luo, 2020). Gaussian spatially correlated fields can be represented by a set of latent variables and a linear combination of orthogonal components obtained from Karhunen-Loève Expansions (KLE) or Principal Component Decomposition (PCA). The latent variables essentially quantify the fluctuations of the realization about the mean of the stochastic process along each component. If mean and principal components are known a prior, GA can be reformulated (named RGA) to estimate low-dimensional latent variables instead of high-dimensional parameter fields (Zhao & Luo, 2020). RGA has been further extended to high-dimensional HT inverse problems with biased prior information by iteratively correcting principal components (Zhao & Luo, 2021b). For non-Gaussian fields such as channel fields, the linear dimensionality reduction method discussed above can hardly extract effective low-dimensional parameterizations, unless there are other regularizations. Vo and Durlofsky (2014) proposed optimization PCA (OPCA) to reduce the dimension of binary channel fields. OPCA uses PCA to reduce the dimension but introduces a regularization term to force the estimates to be binary.

Deep learning models have recently been used for groundwater inverse problems. Nonlinear deep-learning (DL) based methods such as variational autoencoders (VAE), convolutional adversarial autoencoder (CAAE), generative adversarial network (GAN) and spatial GAN (SGAN) have been used to obtain low-dimensional latent variables of non-Gaussian binary channel fields (Laloy et al., 2018; Mosser et al., 2017; Zhou et al., 2022). To solve HT inverse problems, Laloy et al. (2018) incorporated VAE or SGAN in the Bayesian framework by applying the Markov Chain Monte Carlo (MCMC) sampling method. In addition, Zhou et al. (2022) combined CAAE with a deep learning based surrogate model and ensemble smoother to solve inverse problem of multiple data assimilation. A hydraulic tomography–physics informed neural network (HT-PINN) was recently developed for inverting large-scale HT for Gaussian parameter fields (Guo, Zhao, et al., 2023). PINN is essentially a deep neural network (DNN) constrained or regularized by governing equations of a physical system (Raissi et al., 2017a, 2017b, 2019). However, trained PINN forward models are not generalizable for different parameter realizations, and for estimating small-scale heterogeneities or non-smooth fields, a large number of collocation points and reference data may be necessary for training a PINN inverse model (Guo, Zhao, et al., 2023; He et al., 2020; He & Tartakovsky, 2021; Tartakovsky et al., 2020; Wang et al., 2021a, 2021b). Instead of identifying low-rank parameterizations, several deep learning methods directly output plausible realizations conditioning on the input reference data. For example, Cui et al. (2022) proposed the GM-ConvCNP model, which embedded direct measurements in a spatial matrix as input and reconstructed channel fields conditioning on these measurements through CNN. Vu and Jardani (2022b) proposed the SegNet-Fracture model, which interpolated HT observations over the domain and used the interpolated map of hydraulic heads as input to estimate channel fields. However, since the underlying patterns of the channel field are not explicitly encoded, such models cannot be used to generate realizations, nor can they be combined with other inverse methods for estimating high-dimensional parameter fields. Huang et al. (2022) proposed using CNNs and positron emission tomography (PET) data to estimate the permeability distribution in 3D subsurface domains. The CNN architecture was designed based on 3D convolution and can effectively capture spatial patterns and relationships within the PET data.

In this study, we aim to explore the feasibility of using deep-learning (DL) predictive models to perform HT inverse modeling for estimating hydraulic conductivity or transmissivity fields through predicting latent variables. The novelty is to implement dimensionality reduction on hydrogeological parameter fields for Gaussian and non-Gaussian fields and use DL predictive models to directly learn the inverse modeling process from measurements to latent variables. Our model is named HT-INV-NN and composed of a parameter field decoder and a latent variable predictor. The prior stochastic process of Gaussian and non-Gaussian fields is encoded as principal components and generators in trained GAN, respectively. The decoder simulates the prior stochastic process and recovers the predicted latent random variables to actual realizations. The HT-INV-NN with a PCA decoder is inspired by RGA and works with a known Gaussian process like most GA applications. The HT-INV-NN with a GAN decoder is inspired by Laloy et al. (2018), but uses a network different from SGAN for dimensionality reduction. The predictor is a fully-connected DNN that takes hydraulic heads monitored from HT as input and predicts latent random variables as output. The data for training the DNN is generated by forward simulations

on random fields generated by the decoder in HT-INV-NN. Therefore, unlike the methods discussed above, the simulations for training the predictor DNN can be implemented independently in parallel, and no additional forward model simulations or surrogate models are needed during the inverse modeling process.

This paper is organized as follows. First, we introduce dimensionality reduction of Gaussian and non-Gaussian random fields and their low-rank approximations. Second, we explain our model architecture and numerical implementation. Then, multiple numerical experiments of HT are presented to demonstrate the applications. Finally, conclusions and discussion are presented.

2. Random Field and Dimensionality Reduction

In this section, we briefly introduce random fields and dimensionality reduction methods we use for modeling. The general observation function describing the relationship between observable variables and unknown parameters is given by Equation 1:

$$\mathbf{y} = \mathrm{f}(\mathbf{s}) + \boldsymbol{\epsilon} \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^{n \times 1}$ represents an observed data vector, $\mathbf{s} \in \mathbb{R}^{m \times 1}$ is an unknown parameter vector, h represents the forward model, and $\boldsymbol{\epsilon} \in \mathbb{R}^{n \times 1}$ represents the white noise vector.

For HT, the function f represents a numerical model that simulates sequential pumping tests. The 3D governing equation for this model is:

$$S_s \frac{\partial h}{\partial t} = -\nabla \cdot (-K \nabla h) + Q \tag{2}$$

where S_s is specific storage, t is time, h is hydraulic head, K is hydraulic conductivity, and Q is the water accumulation/reduction rate in the selected control volume. Our study specifically focuses on the estimation of hydraulic conductivity or 2D transmissivity field by assuming constant specific storage or storativity. Estimating specific storage fields would require additional regularization information, which is often unavailable in field settings or would necessitate making strong assumptions (Zhao & Illman, 2021). Thus, inverse modeling of HT in this study is to monitor the hydraulic heads and estimate the unknown spatial field $\mathbf{s}$, which is hydraulic conductivity or transmissivity field. For high-dimensional or fine-resolution fields of $\mathbf{s}$, dimensionality reduction is implemented to encode $\mathbf{s}$ into low-dimensional variables. The decoding process then allows for the reconstruction of high-dimensional fields from the low-dimensional variables. This decoding process is useful for generating realizations with spatial distribution patterns similar to the original field.

2.1. Gaussian Field and PCA

Gaussian spatially correlated fields are typically described by first- and second-order statistical moments, that is, mean and covariance. We have:

$$E[\mathbf{s}] = \mathbf{X}\beta \tag{3}$$

$$E[\mathbf{s}\mathbf{s}^T] = \mathbf{Q} \tag{4}$$

where $E[\cdot]$ is the expectation operator, $\mathbf{X} \in \mathbb{R}^{m \times p}$ represents the drift of mean, $\beta \in \mathbb{R}^{p \times 1}$ represents the unknown coefficient vector of the drift function, and p is the number of mean drifts. Covariance matrix $\mathbf{Q}$ is symmetric and can be modeled by a two-point geostatistical covariance function. $\mathbf{Q}$ has a low-rank approximation using PCA. Assume λ_i and $\mathbf{v}_i$ is a pair of eigen-value and vector, we sorted them in descending order of λ_i, then, $\mathbf{Q}$ can be approximated by a k rank matrix $\mathbf{Z}_k$ as Equation 5:

$$\mathbf{Q} \approx \mathbf{Z}_k \mathbf{Z}_k^{\mathrm{T}} \tag{5}$$

$\mathbf{Z}_k \in \mathbb{R}^{m \times k}$ is the matrix with the first k scaled principal components at each column: $\mathbf{z}_i = \sqrt{\lambda_i} \mathbf{v}_i$. Thus, the Gaussian process is approximated by a deterministic drifted mean and stochastic fluctuations as:

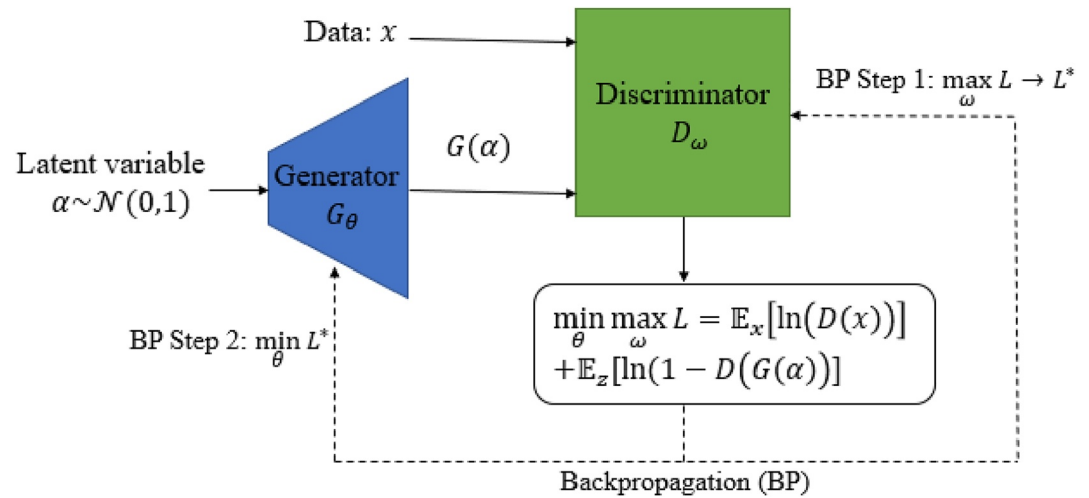$$\mathbf{s} \approx \mathbf{X}\beta + \mathbf{Z}_k \alpha \tag{6}$$

Figure 1. Diagram of GAN with a binary cross-entropy loss function. Generator (G) is a neural network with coefficients θ; Discriminator is a neural network with coefficients ω; x is realization from reference data labeled by one, α is latent variable vector sampled from a normal distribution, generated realization $G(\alpha)$ is labeled by zero. Solid lines represent forward propagation, and dash lines represent backpropagation. The backpropagation is ordered. In the first step, the discriminator D with coefficients ω is optimized to maximize the loss. In the second step, the generator G with coefficients θ is optimized to minimize the loss.

where $\alpha \in \mathbb{R}^{k \times 1} \sim N(\mathbf{0}, \mathbf{I})$ is the latent random variable vector representing the coefficients or projections of the stochastic fluctuations on scaled principal component axes. The inverse problem of estimating $\mathbf{s}$ can be transformed to estimate β and α. $\mathbf{Z}_k$ also serves as a decoder for recovering $\mathbf{s}$ from estimated β and α. In addition, $\mathbf{Z}_k$ can be used to generate realizations by randomly sampling α from N(0,**I**).

2.2. Non-Gaussian Channel Field and GAN

Channel fields mainly refer to connected binary realizations. Typically, the prior information of channel fields is represented by a training image (TI), and each generated realization has spatial correlation similar to the TI. However, the spatial correlation of a channel TI can only be described by multiple-point statistics (MPS) instead of two-point statistics (Strebelle, 2002). Traditional MPS-based simulation is slow, which leads to exploration of deep learning (DL) based methods. In these DL applications, GAN is recognized as one of the most efficient models for generating channel fields. The original GAN was proposed by Goodfellow et al. (2014) as a neural network based generating model, which simulates an unknown stochastic process through training with realizations generated from it. Complete GAN is composed of neural network generator (G) and neural network discriminator (D). Latent random variables are inputs and random realizations are outputs for the generator. The discriminator takes a realization as inputs and provides a decision on whether the realization is from real data (real) or generated (fake). Two networks are trained jointly to minimax a common loss function. Both of them are forced to improve each other until a convergence that G can generate realizations that cannot be distinguished from the reference data, that is, D always predicts that the probability of an input to be real is 50%, whether it is actually the reference data or generated by the generator. Figure 1 shows a GAN with a binary cross-entropy loss function. The converged G is the imitation of the stochastic process of the reference data, which can be used to generate realizations with latent variables $\boldsymbol{\alpha}$:

$$\mathbf{s} = G(\alpha), \quad \alpha \sim p(\alpha) \tag{7}$$

Similar to the Gaussian process, the latent variable vector $\boldsymbol{\alpha}$ is the low dimensional parameterization of the generated realization $\mathbf{s}$, which can be sampled from its prior distribution.

3. Inverse Method

The proposed inverse modeling is composed of a decoder for decoding latent variables to high-dimensional parameter fields and a predictor for estimating latent variable from measurements.
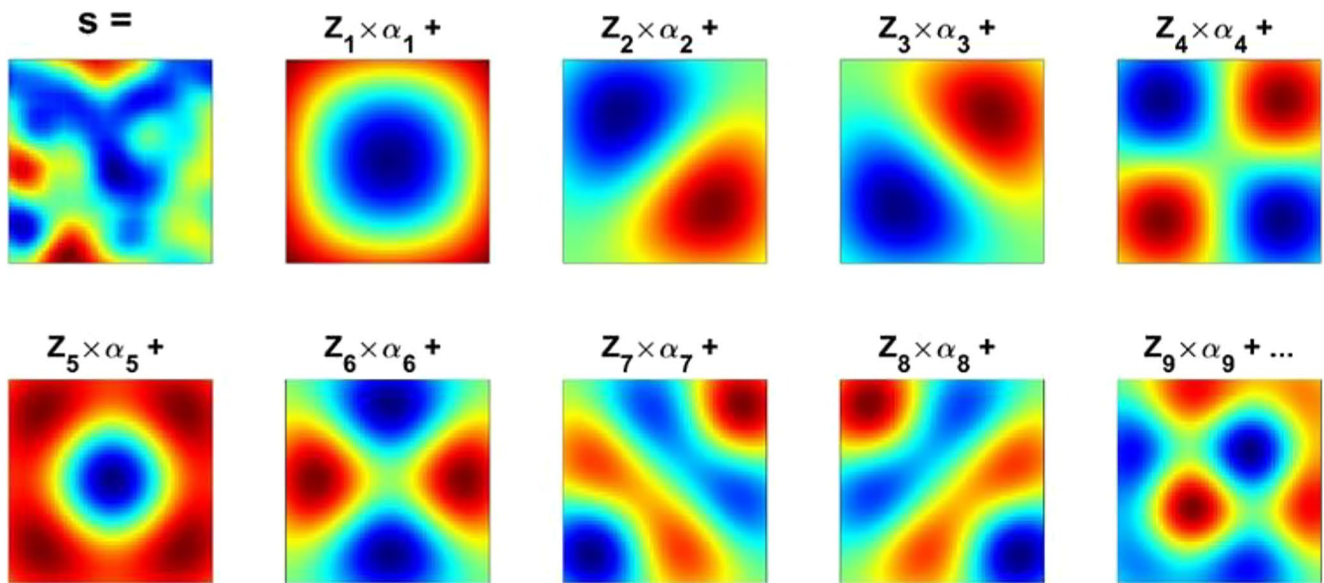
Figure 2. Decomposition of a Gaussian random field on its principal components. The random field has a resolution $1,024 \times 1,024$ and a Gaussian covariance model.

3.1. Decoder Construction

3.1.1. PCA Decoder for Gaussian Field

For high-dimensional Gaussian fields with given covariance, the principal component matrix $\mathbf{Z}_k$ is obtained by singular value decomposition (SVD) of randomly generated realizations (Lee & Kitanidis, 2014). The number or rank of principal components to be retained can be determined by the total variance that needs to be quantified. Figure 2 shows the decomposition of a Gaussian random field on the principal components. By sampling the latent variables α, a random realization can be generated. Similarly, for estimated α, this process is naturally used as a decoder to obtain estimated fine-resolution parameter fields.

3.1.2. GAN Decoder for Non-Gaussian Channel Fields

We consider two types of non-Gaussian channel fields: discontinuous and continuous. The basic channel fields are discontinuous and comprise two facies, representing high and low permeability zones (channel and non-channel). We used a MPS simulation program called *snesim* to generate discontinuous channel fields containing sinuous channels as the reference data for training a GAN (Liu, 2006). The generator in trained GAN is used as the decoder for discontinuous channel fields. The *snesim* program, originally developed by Strebelle (2000) and later improved by Stanford Center for Reservoir Forecasting (SCRF), is publicly available at https://github.com/SCRFpublic/snesim-standalone. Figure 3A1 shows the generated realizations by *snesim*. The resolution of the original TI is 250×250, and the resolution generated is 128×128. For continuous binary fields, they are derived from the discontinuous fields generated by *snesim*, which are regarded as the spatial mean. In order to introduce smaller-scale heterogeneity, a spatial covariance is applied to each field. The heterogeneous channel fields are smoothed using a Gaussian filter to emulate natural channel field, which are relatively continuous between different facies. Figure 3B1 shows samples of continuous channel fields generated through this process. These fields are used as reference data to train a continuous-GAN.

Figure 4a illustrates the structure of the channel field decoder. In contrast to the original GAN architecture, which utilizes fully-connected neural networks (FC-NNs) for both the generator and discriminator, we adopt the deep convolution GAN (DC-GAN) proposed by Radford and Metz (2016) for improved performance. Here is a detailed description of the developed decoder.

The generator G is a convolutional-transpose neural network. The input is a normally distributed latent variable vector α with a dimension of 16. A dense layer maps input variables to hidden variables with a dimension of 2,048. Then, hidden variables are reshaped by a reshape layer to a 3D tensor of the size (8, 8, 32), which represents (height, width, channel). A generator module is designed with a 2D convolution transpose layer, a
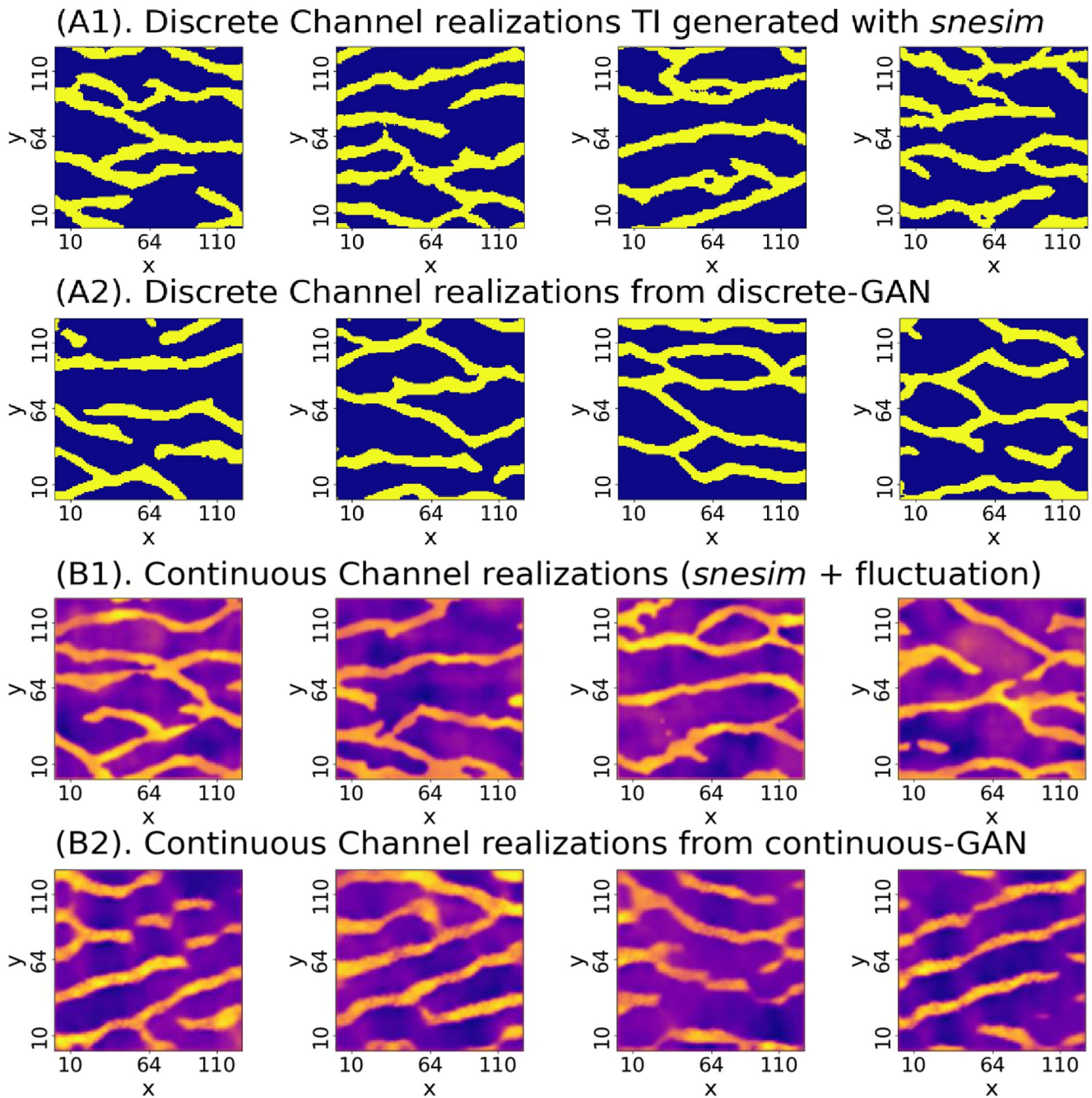
Figure 3. Binary non-Gaussian channel fields. (A1) Discontinuous realizations generated by *snesim* with a TI from Strebelle (2002); (A2) Realizations generated by the discontinuous-GAN trained with discontinuous realizations; (B1) Continuous realizations generated by *snesim*, fluctuated by Gaussian covariance and smoothed by Gaussian filter. (B2) Realizations generated by the continuous-GAN.

leaky ReLU activation layer, and a 2D upsampling layer. In each module, the filter size of the 2D convolution transpose layer is 5×5, the stride is 2 along height and width, the padding at boundaries is the same, and the number of output channels is twice the number of input channels, so the size of input tensors will double on all three dimensions. The Leaky ReLU activation layer is $y = max(y, ay)$, $a = 0.2$. The 2D upsampling layer works on each channel and has a ratio of 2 along height and width, doubling the size of input tensors again on height and width. Thus, the size of input tensors in each module is four times on height and width, and twice on the channel. We use two modules consecutively to show that the size of reshaped tensors changes from (8, 8, 32) to (32, 32, 64) to (128, 128, 128). At the output layer, tensors are downsized from (128, 128, 128) to (128, 128, 1) by a 2D
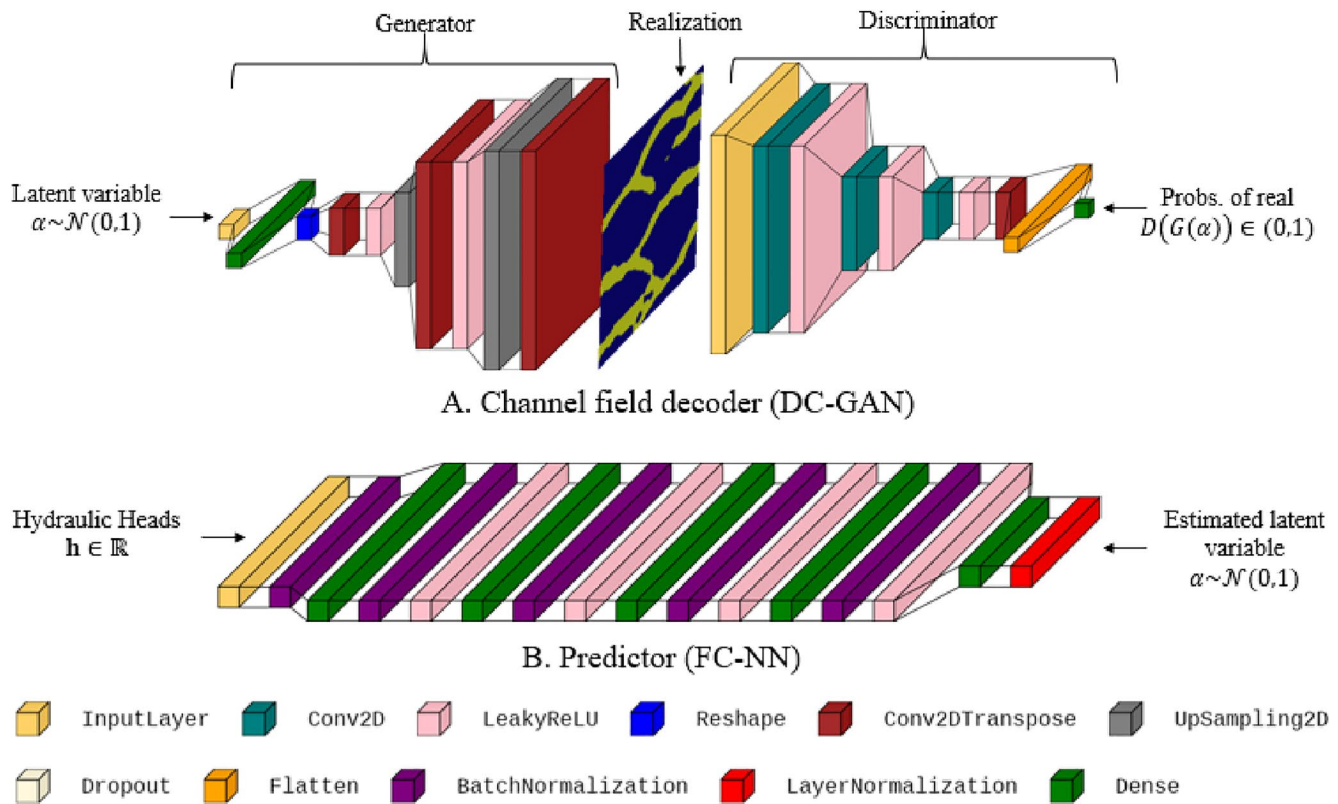
Figure 4. Models of channel field decoder and Predictor. (a) The channel field decoder in structure of deep convolutional generative adversarial network (DC-GAN). (b) Predictor in structure of fully connected neural network (FC-NN). The models are developed with Tensorflow Keras and visualized by visualkeras (Gavrikov, 2020). Legend at bottom shows the label of components in the model. More details can be found at: https://keras.io/api/layers/ and https://github.com/paulgavrikov/visualkeras.

convolution transpose layer with filter size of 5×5, strides of 1, same padding on boundaries, and a hyperbolic ($\tanh$) activation function is used to bound the outputs within $[-1, 1]$.

The discriminator D is a convolutional neural work. The input is a realization of resolution 128×128. We transform it to an input tensor and extend a dimension along the channel, so the size of the input tensor is (128, 128, 1). A discriminator module is designed with a 2D convolution layer and a Leaky ReLU layer. The Leaky ReLU layer is same as the one in the generator module, and the convolution layer has filter size of 5×5, stride of 2 along height and width, and same padding at boundaries. We repeatedly use this module three times, and the number of output channels is 64, 128, 128. The tensor size track by each discriminator module is (128, 128, 1), (64, 64, 64), (32, 32, 128), (16, 16, 128). A dropout layer is used with rate at 0.2. Then, the (16, 16, 128) 3D tensor is flattened to a vector and mapped by a dense layer to dimension of 1. The activation function of the dense layer is *sigmoid*, so the output is bounded within [0, 1], representing the probability of the input realization being real or from real data. The development software employed Tensorflow Keras. For more detailed information about each component in the model, please refer to the official website at https://keras.io/api/layers/.

During training, the binary field values are rescaled to -1 and 1. We used a total of 1,000 realizations, batch size of 100, binary cross-entropy loss function and the Adam optimizer with a learning rate of 0.5. The network was trained for 5,000 epochs. Computation was implemented on Google Colab platform with NVIDIA T4 Tensor Core GPU at backend, hardware computational time is 0.68 hr Figure 5 shows the learning curves of D and G for the discontinuous-GAN, and at specific steps, we sample generated images (in a 3×3 grid) to show the status of G. At an early stage, D obtains a fast loss drop because the fake images generated by the premature G are completely different from real images. After the turning point at the 200th epoch, the training loss of D and G continuously increase and decrease, respectively, as G gradually captures the characteristic of the TI and generates similar images. As G matured after 2,000 epochs, fake images are close to the real images in terms of spatial patterns, and the adversarial learning converges typically with the loss of D being greater and the loss of G being smaller compared with early ages. The network structure and training process of the continuous-GAN are the same as the discontinuous-GAN.
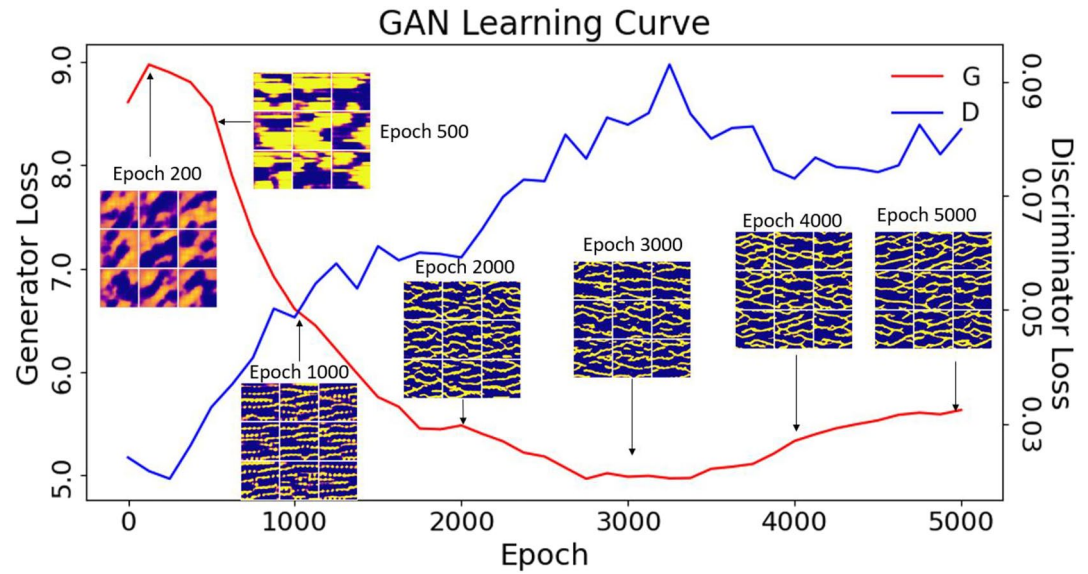
Figure 5. Learning curve of DC-GAN. Red line is the learning curve of the generator G, blue line is the learning curve of the discriminator D. The loss is moving averaged on 100 epochs. Samples generated by G is drawn at 200, 500, 1,000, 2,000, 3,000, 4,000, 5,000 epochs in a 3×3 grid.

After the training process, we utilize the generator G as the decoder to generate new realizations for training the predictor. Figure 3A2 showcases four randomly chosen samples of the generated realizations, which visually appear similar to the samples in the reference data shown in Figure 3A1. Similarly, Figure 3B2 displays four continuous-GAN generations, which exhibit similarities to the samples in Figure 3B1 representing the reference data for continuous channel fields. Additionally, the connectivity of two facies in the generated and reference fields is evaluated with two cluster functions as quantitative metrics. Figures 6A1 and 6A2 display the connectivity of facies 0 (low permeability media) along the horizontal (X) and vertical (Y) directions, respectively. Figures 6B1 and 6B2 display the connectivity of facies 1 (high permeability media, channels). The blue solid lines represent the mean connectivity of 200 randomly chosen reference fields, while the blue dashed lines represent the two standard deviation bounds. The gray lines show the connectivity of the 200 generated realizations by the trained GAN, which predominantly fall within the blue dashed lines. These results suggest that the generator effectively captures the distribution of channel connectivity patterns in the reference data.

3.2. Predictor for Inverse Problem

Most DL models are trained to learn the forward model, that is, $f(\mathbf{s})$ in Equation 1. Our method is to train a predictive model to directly learn the inverse process, that is, $f^{-1}(\mathbf{y})$. The predictor for both Gaussian and channel realizations is a fully connected neural network (FC-NN). Figure 4b shows the predictor structure. The input dimension equals to the number of monitored hydraulic heads. The input layer is followed by a batch normalization layer. The predictor module is designed with a dense layer with batch normalization and a Leaky ReLU activate layer as the basic module. The dimension of hidden variables in the dense layer is 3,000. The Leaky ReLU layer is the same as that in DC-GAN. This module is repeated four times until the output layer. The output layer is a linear layer with layer normalization. The dimension of output variables is equal to the dimension of latent variables. The structures of DC-GAN and FC-NN are independently and empirically determined.

The constructed decoder can be integrated into the pipeline of training and validating the predictor. Figure 7 shows the technical diagram, where the red and blue arrows indicate the training and validation route, respectively. During the training process, we sample latent variables and generate realizations through the decoder. Forward model simulations on these realizations can be implemented in parallel to prepare data as the input (sample features) for the predictor. A loss function of true and estimated latent variables is used to tune coefficients of the predictor through backpropagation. When the predictor is trained, we use another set of sampled latent variables to generate a true field to demonstrate its application for estimating the random field given the measurement data.
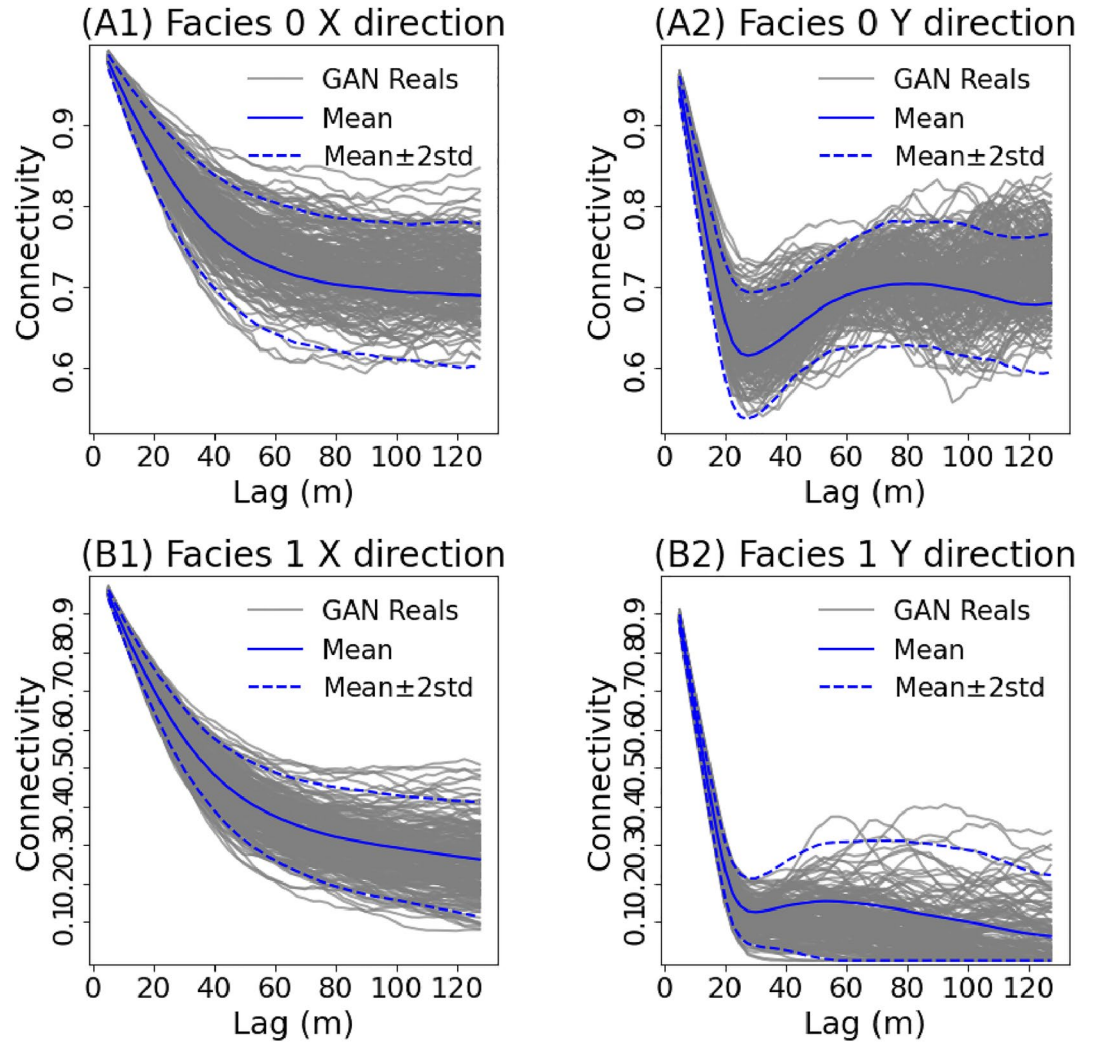
Figure 6. Connectivity evaluation of different facies in discontinuous channel fields. Panels (A1) and (A2) shows the connectivity of facies 0 (low permeability porous media) in X (horizontal) and Y (vertical) directions; (B1) and (B2) shows the connectivity of facies 1 (high permeability channel) in X (horizontal) and Y (vertical) directions. The blue solid and dash lines denote the mean and uncertainty bounds of 200 reference fields generated by *snesim*. Gray lines are 200 realizations generated by the generator *G*.

The realization generation, data preparation, and predictor estimation are the same as the training process. The estimated latent variables are used by the decoder to generate the estimated realization.

4. Numerical Experiments and Results

To test the effectiveness of HT-INV-NN, we conduct multiple numerical experiments of HT to estimate heterogeneous transmissivity or conductivity fields. The experiments can be divided into two sets: Experiments 1, 2, and 3 involved Gaussian random fields with a PCA decoder, while Experiments 4 and 5 focused on binary non-Gaussian channel fields. For the Gaussian fields, the dimension and covariance model of the experimental fields are 2D Gaussian, 2D Exponential, and 3D Gaussian, respectively. In the 2D random fields, 50 principal components are retained, and the domain size is 320 m × 320 m. The boundaries along the *y*-direction are impermeable, while the *x*-direction boundaries are constant-head. In the 3D experiment, the depth is 16 m, with an impermeable bottom and a confined top in the vertical *z*-direction. The storage coefficient is assumed to be 10^{-4} m, and the conductivity (or transmissivity in 2D case) is isotropic. The well network of HT includes 25 wells, indicated by black dots in Figure 8a. These wells are labeled from p1 to p25 and are alternatively used
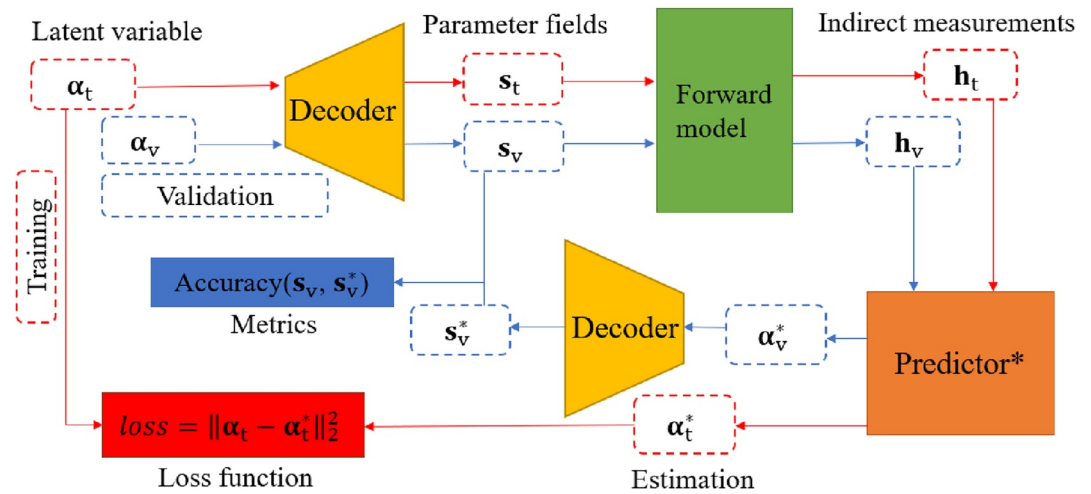
Figure 7. Diagram of training and validating the predictor. α is the latent variable vector. Red lines show the training route. The variables used for training are subscripted by "t". The loss function of training is mean squared error between true and estimated α. Blue lines show the validation route and the variables used for validation are subscripted by "v". The validation metric is map accuracy.

as pumping or monitoring wells. During each pumping test, water is withdrawn from a well at a constant rate of 3.6 m²/hr, and the steady-state or transient hydraulic heads are monitored from the remaining 24 wells. To account for realistic conditions, all data values are corrupted with 5% white noise. Table 1 lists the hydrogeological and geostatistical parameters used in the experiments.
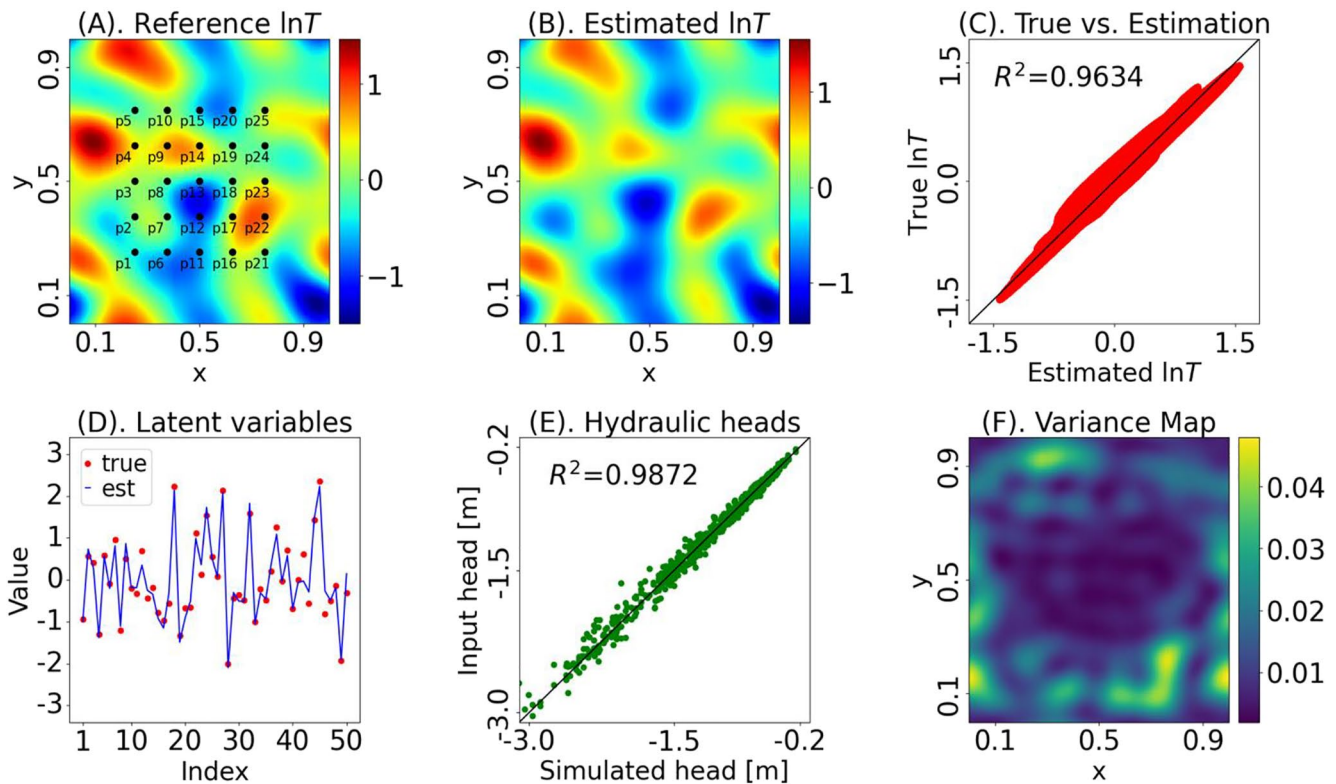

Figure 8. Inverse modeling of Experiment 1: steady-state HT in a Gaussian field with a Gaussian spatial covariance model. (a) A reference $\ln T$ realization with a Gaussian covariance model, and black dots denote the HT well network; (b) The estimated field by HT-INV-NN; (c) Comparison of true and estimated $\ln T$ values; (d) True and estimated latent variables; (e) Comparison of measured hydraulic heads and simulated hydraulic heads using the estimated $\ln T$ field; (f) Variance map of estimations.

Table 1
Hydrogeological and Geostatistical Parameters for HT Experiments With Three Gaussian Fields

Parameter	Experiment 1	Experiment 2	Experiment 3
Domain size, $L_x \times L_y \times L_z$ [m]	320×320	320×320	$320 \times 320 \times 16$
Spatial resolution, $n_x \times n_y \times n_z$	512×512	256×256	$64 \times 64 \times 16$
Geostatistical property of transmissivity T [m²/hr]			
Geometric mean of lnT	0	0	0
Covariance model	Gaussian	Exponential	Gaussian
Variance of lnT, σ^2_{lnT}	1	2	1
Correlation length, $\lambda_x \times \lambda_y \times \lambda_z$ [m]	64×48	64×128	$64 \times 64 \times 6.4$
Hydraulic and boundary conditions			
Pumping type	Steady-state	Transient	Steady-state
y boundaries	Impermeable	Impermeable	Impermeable
x boundaries	$h = 0$ m	$h = 0$ m	$h = 0$ m
Initial Condition	$h = 0$ m	$h = 0$ m	$h = 0$ m
Pumping Rate [m³/hr]	3.6	3.6	3.6

For training the predictor, the number of samples is 4,500, the loss function is mean squared error between true and estimated latent variables, the batch size is 1,000, the number of epochs is 30,000, the optimizer is Adam, and the start learning rate is 10^{-3} and decays to 10^{-4} after 10,000 epochs and 10^{-5} after 20,000 epochs. The computation platform is Google Colab with NVIDIA T4 Tensor Core GPU. For validation, the number of samples is 500, the quantitative metric is map accuracy with a threshold of 15%.

4.1. Gaussian Random Field Inversion

4.1.1. 2D Steady-State HT and Gaussian Covariance

In Experiment 1, we generate transmissivity (T) fields with a resolution 512×512. The Gaussian logarithmic transmissivity field (lnT) has a zero mean and a Gaussian spatial covariance model with a variance of 1. 50 dominant principal components are retained to account for 90% of the total variance. For each steady-state HT, we sequentially conduct pumping tests in each of the 25 wells and collect steady-state hydraulic heads in the other 24 wells, leading to totally 600 indirect measurements. Correspondingly, the input layer of the predictor has 600 input dimensions, the total required training time is 0.92 hr.

Figure 8 shows the results of one experiment. The map accuracy of the estimated lnT field in Figure 8b compared with the true field in Figure 8a is greater than 96%, demonstrating high inverse modeling accuracy of the predictive DL model, Figure 8c shows the estimations and true values. The lnT field is constructed from the estimated latent variables, which also match the true values on the dominant principal component (Figure 8d). In addition, the hydraulic heads simulated with the estimated field reproduce the noised input data with 5% relative residuals. For uncertainty analysis, Monte Carlo simulations are performed by inverse modeling of 20 realizations of measurement data with randomly generated white noises. Figure 8f shows the variance map. The accuracy of all 20 estimations is over 96%, which shows that the predictor is robust to small perturbations on input. Moreover, we use 500 different realizations as the validation data to evaluate the inverse modeling performance. Results are summarized in Table 2. The mean map accuracy of the estimated field compared with the true field is 96%, and the standard deviation is 4%. The mean normalized residual between forward simulations with the estimated transmissivity field and the input hydraulic heads is only 5% and the standard deviation is 1%.

4.1.2. 2D Transient HT and Exponential Covariance

In Experiment 2, we conduct transient HT to collect time-dependent data for estimating a Gaussian field with an exponential covariance model. The field resolution is 256×256, and the variance is 2, which is more heterogeneous and less smooth than the random fields in Experiment 1. We also retain the dominant 50 principal components for dimensionality reduction. In the transient HT, we sequentially perform five pumping tests at p1, p5, p13, p21, and p25, which are located at the four corners and the center of the well network. In each pumping

Table 2
Inverse Modeling Performance on Multiple Validation Data Sets in Numerical Experiments

Experiment	Field type	Accuracy of ln$T$ estimation		Normalized residual of hydraulic heads	
		Mean ± std	Best	Mean ± std	Best
Experiment 1	2D Gaussian	0.96 ± 0.04	0.99	0.05 ± 0.01	0.04
Experiment 2	2D Exponential	0.85 ± 0.06	0.98	0.05 ± 0.01	0.04
Experiment 3	3D Gaussian	0.97 ± 0.02	0.99	0.07 ± 0.01	0.05
Experiment 4	2D Binary Channel	0.85 ± 0.04	0.93	0.27 ± 0.11	0.07
Experiment 5	2D Continuous Channel	0.88 ± 0.06	0.99	0.07 ± 0.03	0.03

test, we monitor hydraulic heads in all the other 24 monitoring wells at the time steps of 2, 4, 6, 8, and 10 (hour), yield a total of 600 measurements. Thus, the predictor has the same input dimension as the steady state experiment and requires the same amount of training time (about 0.92 hr).

Figure 9 shows an example of Experiment 2. The map accuracy is over 90% (Figures 9a and 9b), also demonstrating the effectiveness of the predictive DL model for estimating Gaussian fields with exponential covariance models. The overall inverse performance and hydraulic head predictions (Figures 9c–9e) are not as good as smooth Gaussian fields with Gaussian covariance models in Experimental 1. This is consistent with other inverse methods such as GA because hydraulic head measurements are not sensitive to small-scale heterogeneities (Zhao et al., 2022). In addition, although the number of measurements in Experiment 2 is equal to steady-state HT in Experiment 1, the transient HT actually conducts fewer pumping tests, causing the collected hydraulic heads to be less informative. The performance of HT-INV-NN on data sets of multiple realizations is presented in Table 2. The mean inverse accuracy is 85%, and the standard deviation is 6%. The relative residual between the input and
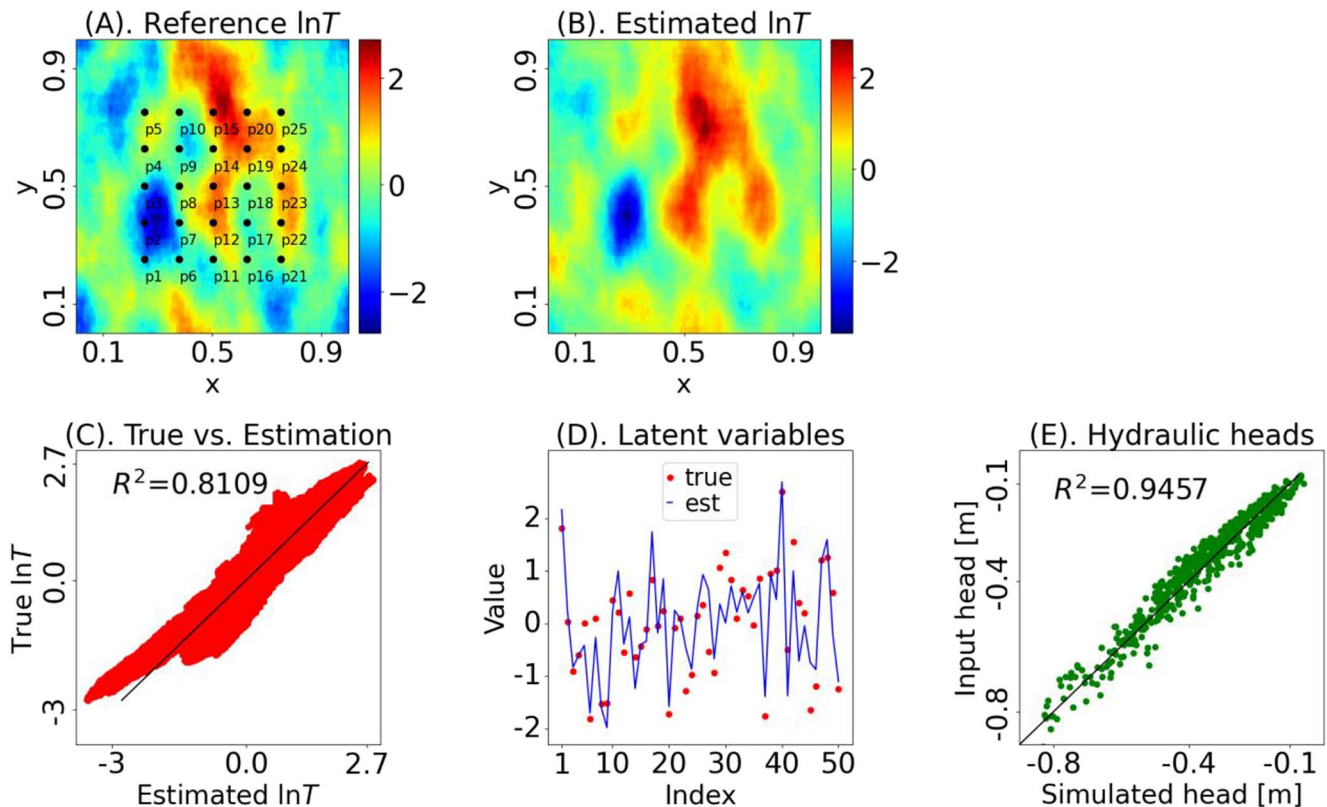

Figure 9. Inverse modeling of Experiment 2: transient HT in a Gaussian field with an exponential spatial covariance model. (a) Reference ln$T$ realization; (b) Estimation from HT-INV-NN; (c) Comparison of true and estimated ln$T$ values; (d) True and estimated latent variables; (e) Input hydraulic heads and simulated hydraulic heads with the estimated ln$T$ field.

simulated hydraulic heads is about 5% with a standard deviation of 1%, which are similar to Experiment 1. This reveals that the inverse problem may be ill-posed, that is, HT in true and estimated fields with a lower map accuracy still provides a similar hydraulic head prediction.

4.1.3. 3D Steady-State HT and Gaussian Covariance

In Experiment 3, we generate conductivity (K) fields at a resolution of $64 \times 64 \times 16$. The Gaussian logarithmic conductivity field ($\ln K$) has a zero mean and a Gaussian spatial covariance model with a variance of 1. The steady-state HT is conducted with pumping tests in each of the 25 wells and steady-state hydraulic heads is collected in the other 24 wells, with hydraulic heads measured in each 1-m-deep layer. This results in a total of 9,600 measurements ($24 \times 25 \times 16$). So, the predictor input dimension is modified to 9,600, containing more tunable coefficients in the input layer, and the required training time is approximately 3.67 hr.

Figure 10 illustrates the impressive performance of 3D HT-INV-NN. Specifically, Figures 10A1 and 10A2 display the reference $\ln K$ field and hydraulic heads map in the pumping test at p1, respectively. Figures 10B1 and B2 show the estimated $\ln K$ field and simulated pumping test p1 based on the estimated field. The map accuracy is over 99%, demonstrating the high accuracy of the model. The input data is also reproduced with a relative residual of 4% and an R^2 coefficient of over 0.96, as shown in Figure 10d. Moreover, Figure 10c depicts the closely aligned predicted and true latent variables. The quantitative metrics of the model performance over 500 validation data are summarized in Table 2. The mean and standard deviation of map accuracy are 97% and 2%, respectively. Additionally, the mean and standard deviation of the relative residual of the simulated hydraulic heads are 5% and 1%, respectively. This performance is comparable to the 2D Gaussian experiment and better than the 2D exponential experiment, indicating that spatial correlation is a more impactful factor when inverse modeling a Gaussian random field with PCA decoder. The smoother and less variant field can be better encoded in PCA components.

4.2. Binary Non-Gaussian Channel Field Inversion

In experiments 4 and 5, we focus on non-Gaussian channel fields with a 2D transmissivity field to explore the estimation performance of HT-INV-NN in the presence of non-Gaussian channel patterns. The latent variable dimension for these experiments is set to 16, and the generated fields have a resolution of 128×128. The domain size, boundaries, steady-state HT setup, and predictor training are consistent with those used in Experiment 1. We conduct two separate experiments to address continuous and discontinuous fields.

4.2.1. Discontinuous Channel Field

In Experiment 4, the discontinuous channel fields are generated using the trained discontinuous-GAN and strictly contain two facies: high transmissivity channels and low transmissivity solid regions. The transmissivity values used are 1 m²/hr for channels and 10^{-4} m²/hr for solid regions. The outputs from the generator G are filtered by a median filter to remove impurities and binarized by a split value of 0. Figure 11 shows one inverse modeling example of Experiment 4. Overall, the predictive DL model shows very good inverse results: the map accuracy between the true and estimated transmissivity field exceeds 90%, and the majority of latent variables are well estimated. The predicted hydraulic heads closely match the observation data. These results demonstrate that HT-INV-NN can predict latent variables for channel fields and DC-GAN can provide satisfactory generation of the original field from the estimated latent variables. We further perform the inverse modeling on multiple validation data sets and the results are summarized in Table 2. For 500 realizations, the mean accuracy is 84% and the standard deviation is 4%; the mean relative residual of hydraulic head prediction is 27% and standard deviation is 11%. Compared with Gaussian experiments in the previous sections, the accuracy of estimated channel fields is slightly lower, and simulated hydraulic heads from estimated channel fields are more biased and uncertain. This shows that the experiments in channel fields are sensitive to small-scale heterogeneity or connectivity patterns, which are more challenging to be characterized by HT experiments. In fact, only given indirect hydraulic head measurements and two-point geostatistics, it is difficult to inversely estimate field connectivity (Han et al., 2022; Hao et al., 2008). Using the nonlinear feature extractor, DC-GAN is possible to recognize the complex spatial distribution patterns in the TI and provide effective dimensionality reduction.
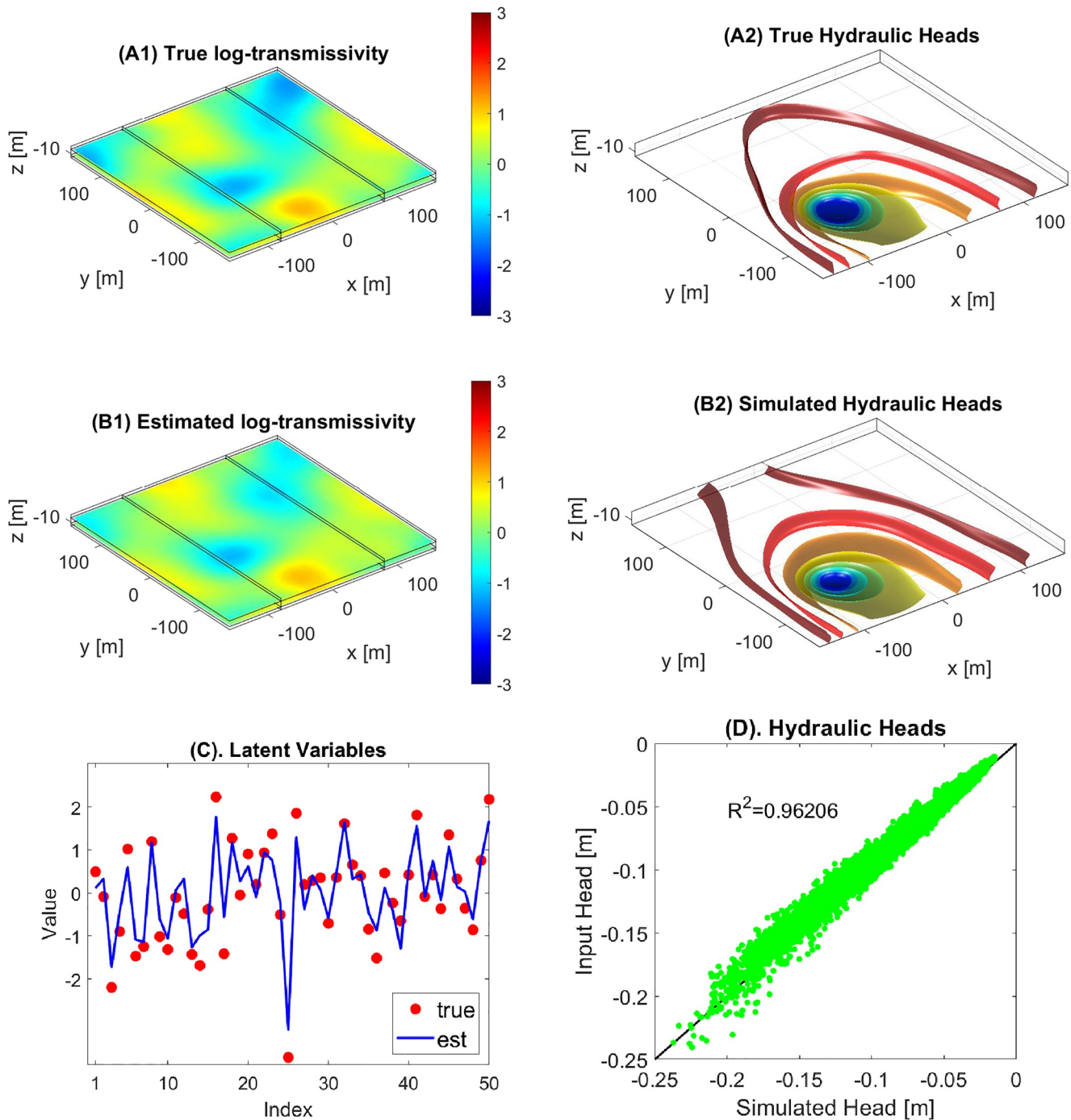
Figure 10. Inverse modeling of Experiment 3: steady-state HT in a 3D Gaussian field with a Gaussian spatial covariance model. (A1) A reference ln$K$ realization with a Gaussian covariance model; (A2) The estimated field by HT-INV-NN; (B1) Simulated hydraulic heads based on the true ln$K$ field; (B2) Simulated hydraulic heads based on the estimated ln$K$ field; (C) True and estimated latent variables; (D) Comparison of measured hydraulic heads and simulated hydraulic heads using the estimated ln$K$ field.

4.2.2. Continuous Channel Field

In Experiment 5, the continuous channel fields are generated using the continuous-GAN. These fields exhibit fluctuated and blurred facies, but still maintain the expected high and low transmissivity characteristics. Figure 12 showcases an example of inverse modeling from Experiment 5, indicating that the HT-INV-NN model produced excellent inverse results. The map accuracy between the estimated (B) and true (A) transmissivity field exceeds
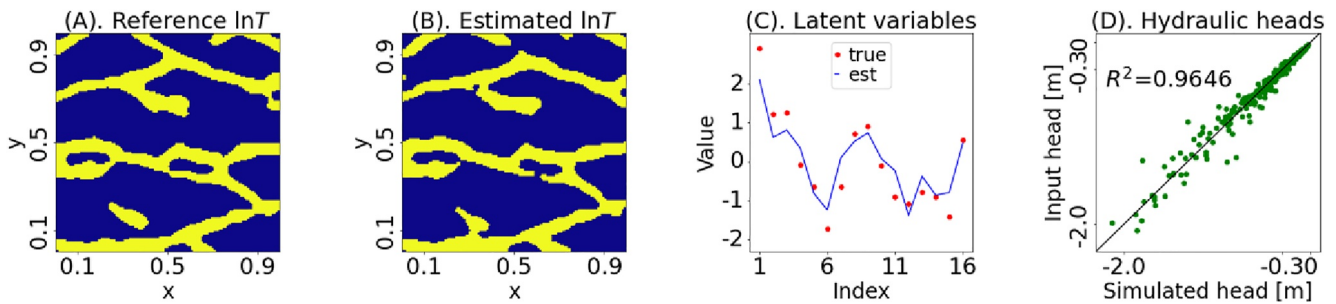
Figure 11. Inverse modeling of Experiment 4: steady-state HT in a binary non-Gaussian channel field. (a) Reference channel ln$T$ realization; (b) Estimated field from HT-INV-NN; (c) True and estimated latent variables; (d) Input hydraulic heads and simulated hydraulic heads with the estimated ln$T$ field.

90%, with a majority of the latent variables accurately estimated (C), and the predicted hydraulic heads matching the observed data (D). These results clearly demonstrate the ability of HT-INV-NN to predict latent variables, while DC-GAN successfully generates satisfactory continuous channel fields from the estimated latent variables. We perform inverse modeling on the entire validation data set, and the results are summarized in Table 2. For 500 realizations, the mean accuracy is 88%, with a standard deviation of 6%. The mean relative residual of hydraulic head prediction is 7%, with a standard deviation of 3%. The accuracy of estimated channel fields is slightly higher compared to the binary discontinuous channel experiments in the previous sections, and the simulated hydraulic heads from the estimated channel fields exhibit higher precision. This improvement can be attributed to the fact that continuous channel fields produce smoother and more predictable transitions of hydraulic heads compared to the discontinuous binary fields.

4.3. Model Comparison

In this section, we compare HT-INV-NN with other two inverse approaches: UPCIA and MCMC + GAN. The UPCIA method, proposed by Zhao et al. (2022), is a promising numerical technique for high-dimensional inverse modeling of groundwater flow. It utilizes dimensionality reduction and upscaling techniques based on principal component analysis (PCA) to construct an effective model that can capture the essential features of high-dimensional random fields. The effective model is then used in a Bayesian gradient-based inverse framework to estimate the latent variables of the groundwater flow system. The effectiveness of the UPCIA approach has been demonstrated through numerical experiments involving 3D Gaussian random fields, where it achieved high accuracy with a relatively low computational cost. However, its effectiveness has not been tested in binary fields, which exhibit sensitivity to small-scale heterogeneities. MCMC + GAN proposed by Laloy et al. (2018) combines DC-GAN and Markov chain Monte Carlo (MCMC) sampling for inverse modeling of binary channel fields. The trained DC-GAN model is used as a prior for the inverse problem, and MCMC is employed to sample from the posterior distribution of the unknown parameters. The results show that the DC-GAN + MCMC approach can accurately estimate the spatial distribution of high and low permeability media in binary channel fields. However, it requires a significant computational cost due to the large number of MCMC iterations needed for convergence.
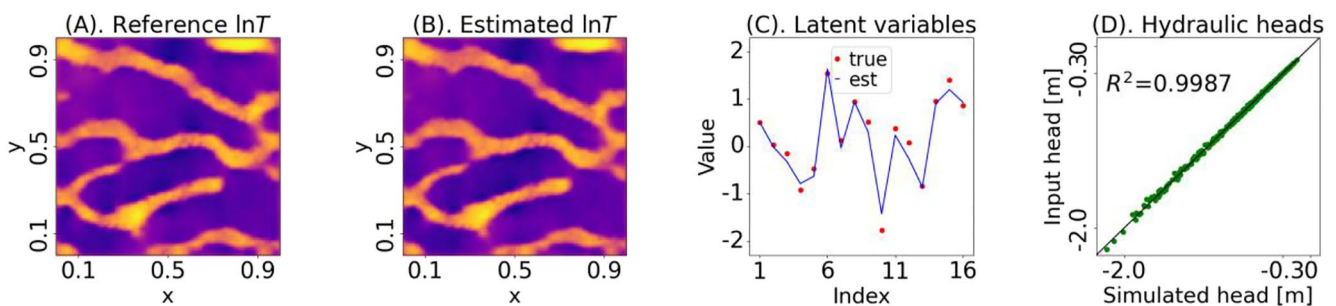

Figure 12. Inverse modeling of Experiment 5: steady-state HT in a continuous non-Gaussian channel field. (a) Reference channel ln$T$ realization; (b) Estimated field from HT-INV-NN; (c) True and estimated latent variables; (d) Input hydraulic heads and simulated hydraulic heads with the estimated ln$T$ field.

Table 3
Inverse Modeling Performance of Different Models on HT Experiment

Model	Field type	Resolution	Accuracy	Number of wells	Runtime
UPCIA [a]	3D Gaussian	$256 \times 256 \times 16$	96%	16	307 s per inverse case
PCA + HT-INV-NN	3D Gaussian	$64 \times 64 \times 16$	97%	25	3.67 hr training + 2 s for 500 inverse cases
GAN + MCMC [b]	2D Binary Channel	128×128	84%	49	48,400 MCMC iterations per inverse case
GAN + HT-INV-NN	2D Binary Channel	128×128	85%	25	0.92 hr training + 2 s for 500 inverse cases

Note. Metrics of UPCIA and GAN + MCMC are extracted or derived from references [a] Zhao et al. (2022) and [b] Laloy et al. (2018).

Table 3 compares the inverse modeling performance of four different models on HT experiments with varying field types, resolutions, numbers of wells, and runtimes. The UPCIA model performs inverse modeling on a 3D Gaussian field with a resolution of $256 \times 256 \times 16$ and achieves an accuracy of 96% using data from 16 wells with a runtime of 307 s per inverse case. The PCA + HT-INV-NN model uses a 3D Gaussian field with a resolution of $64 \times 64 \times 16$ and achieves higher accuracy of 97% using data from 25 wells. It requires a longer training time of 3.67 hr, but only 2 s for 500 inverse cases. The GAN + MCMC model and the GAN + HT-INV-NN model both use 2D binary channel fields with a resolution of 128×128 and achieve accuracies of 84% and 85%, respectively. The GAN + MCMC model required data from 49 wells and approximately 48,400 MCMC iterations per inverse case, which likely takes much longer than 1 hr (a physical runtime was not provided). While the GAN + HT-INV-NN model uses data from 25 wells and requires a training time of 0.92 hr and only 2 s for the 500 inverse cases. Overall, the HT-INV-NN model demonstrates the highest accuracy with a moderate training time and a minimum inverse estimation runtime. The training is one-time cost which can be marginalized to a large number of predictions. To further improve performance, it is possible to rearrange the pumping well networks to reduce the number of hydraulic heads used as inputs for the HT-INV-NN model, while maintaining equivalent estimation accuracy. This adjustment would decrease the number of tunable coefficients in the input layer of the predictor, resulting in shorter training times. However, the optimal experimental design falls outside the scope of the current study.

5. Conclusion

In this work, we develop HT-INV-NN, a deep learning-based inverse model, for high-dimensional HT inverse problems. HT-INV-NN consists of a decoder and a predictor. The decoder utilizes prior geostatistical information to convert latent variables into high-dimensional realizations. For Gaussian fields, the decoder uses principal components derived from two-point spatial covariance models. For non-Gaussian channel fields, the decoder is implemented as a generator of a GAN trained on reference channel realizations using a TI. With the constructed decoder, the model can generate or recover Gaussian or channel fields based on latent variables. The predictor in HT-INV-NN is a DNN model that directly learns the inverse process from measurements to latent variables, rather than a forward model. In our experiments, we implement steady-state and transient HT forward models to collect hydraulic heads in a well network, which serve as the inputs for the predictor. The training data can be generated independently and in parallel, resulting in an efficient training process. Compared to DNN surrogate models for forward models, the predictive model has fewer hyperparameters to tune and exhibits great robustness to small perturbations in input data.

The main advantage of HT-INV-NN lies in its ability to directly predict latent variables without the need for additional optimization or simulation of forward models during the inverse modeling process. This eliminates the computational burden associated with iterative methods or surrogate models. Although the predictor needs a large set of training data generated by forward models, the total number of training cases is manageable, especially considering that the training is one-time cost. After training, the computational time required for inverse modeling as well as forward propagation of the predictor is minimal. The effectiveness of the designed GAN decoder and DNN predictor in HT-INV-NN has been demonstrated across different types of random fields. The model exhibits great versatility and can handle various scenarios, including Gaussian fields with different

smoothness levels and binary non-Gaussian channel fields with discontinuous and continuous patterns. This versatility highlights the robustness and adaptability of HT-INV-NN to different geostatistical variations and field characteristics.

It is important to note that HT-INV-NN relies on a known stochastic process, such as spatial correlations or a TI, to generate reference data. As a result, the model is limited to solving inverse problems within the same stochastic process. This limitation is common among most deep learning-based methods, regardless of the generalization capabilities of the model architecture. It is challenging to develop a single deep learning model that is capable of estimating both Gaussian fields and non-Gaussian fields, or even Gaussian fields with different structural parameters (Jardani et al., 2022; Vu & Jardani, 2022a, 2022b). As additional characterization information becomes available, it is convenient to update the encoder and decoder components of HT-INV-NN to ensure that the estimates conform to the newly identified distribution patterns. Recent advancements in gradient-based models with self-correction mechanisms have shown promise in addressing biased prior geostatistical information, offering potential solutions to overcome this limitation (Zhao & Luo, 2021b). In our future work, we plan to enhance HT-INV-NN by developing more generalizable decoders that do not rely heavily on accurate geostatistical prior information. This will allow the model to handle a broader range of stochastic processes and improve its applicability to various field types. Furthermore, we are actively exploring the extension of HT-INV-NN to other groundwater applications, such as tracer testing. This involves training a DNN predictor specifically tailored to these applications and integrating it with the GAN decoder. By expanding the capabilities of HT-INV-NN to different hydrogeological scenarios, we aim to provide a more comprehensive and versatile tool for groundwater characterization and management.

Data Availability Statement

Data sets, computational notebooks, and saved models that support the findings of this study are available in Guo, Liu, & Luo (2023). Open source software including *snesim* program (Strebelle, 2002), Tensorflow (Abadi et al., 2015), Matplotlib (Hunter, 2007), and visualkeras (Gavrikov, 2020) are used to generate data sets, models, and figures.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems [Software]. Tensorflow. https://www.tensorflow.org/

Ambikasaran, S., Li, J. Y., Kitanidis, P. K., & Darve, E. (2013). Large-scale stochastic linear inversion using hierarchical matrices. *Computers & Geosciences*, *17*(6), 913–927. https://doi.org/10.1007/s10596-013-9364-0

Broyden, C. G. (1965). A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation*, *19*(92), 577–593. https://doi.org/10.1090/s0025-5718-1965-0198670-6

Butler, J. J., Jr., McElwee, C. D., & Bohling, G. C. (1999). Pumping tests in networks of multilevel sampling wells: Motivation and methodology. *Water Resources Research*, *35*(11), 3553–3560. https://doi.org/10.1029/1999WR900231

Cardiff, M., Barrash, W., Kitanidis, P. K., Malama, B., Revil, A., Straface, S., & Rizzo, E. (2009). A potential-based inversion of unconfined steady-state hydraulic tomography. *Ground Water*, *47*(2), 259–270. https://doi.org/10.1111/j.1745-6584.2008.00541.x

Cui, Z., Chen, Q., & Liu, G. (2022). Characterization of subsurface hydrogeological structures with convolutional conditional neural processes on limited training data. *Water Resources Research*, *58*(12), e2022WR033161. https://doi.org/10.1029/2022WR033161

Fienen, M. N., Clemo, T., & Kitanidis, P. K. (2008). An interactive Bayesian geostatistical inverse protocol for hydraulic tomography. *Water Resources Research*, *44*(12), W00B01. https://doi.org/10.1029/2007wr006730

Gavrikov, P. (2020). *visualkeras*. GitHub repository. Retrieved from https://github.com/paulgavrikov/visualkeras

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozairy, S., et al. (2014). Generative adversarial networks. *arXiv*. https://doi.org/10.48550/arXiv.1406.2661

Gottlieb, J., & Dietrich, P. (1995). Identification of the permeability distribution in soil by hydraulic tomography. *Inverse Problems*, *11*(2), 353–360. https://doi.org/10.1088/0266-5611/11/2/005

Guo, Q., Liu, M., & Luo, J. (2023). Predictive deep learning for high-dimensional inverse modeling of hydraulic tomography in Gaussian and non-Gaussian fields [Data]. Zenodo. https://doi.org/10.5281/zenodo.7992543

Guo, Q., Zhao, Y., Lu, C., & Luo, J. (2023). High-dimensional inverse modeling of hydraulic tomography by physics informed neural network (HT-PINN). *Journal of Hydrology*, *616*, 128828. https://doi.org/10.1016/j.jhydrol.2022.128828

Han, Z., Kang, X., Wu, J., & Shi, X. (2022). Characterization of the non-Gaussian hydraulic conductivity field via deep learning-based inversion of hydraulic-head and self-potential data. *Journal of Hydrology*, *610*, 127830. https://doi.org/10.1016/j.jhydrol.2022.127830

Hao, Y., Yeh, T. C., Xiang, J., Illman, W. A., Ando, K., Hsu, K. C., & Lee, C. H. (2008). Hydraulic tomography for detecting fracture zone connectivity. *Ground Water*, *46*(2), 183–192. https://doi.org/10.1111/j.1745-6584.2007.00388.x

He, Q., Barajas-Solano, D., Tartakovsky, G., & Tartakovsky, A. M. (2020). Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport. *Advances in Water Resources*, *141*, 103610. https://doi.org/10.1016/j.advwatres.2020.103610

He, Q., & Tartakovsky, A. M. (2021). Physics-informed neural network method for forward and backward advection-dispersion equations. *Water Resources Research*, *57*(7), e2020WR029479. https://doi.org/10.1029/2020WR029479

Huang, Z., Kurotori, T., Pini, R., Benson, S. M., & Zahasky, C. (2022). Three-dimensional permeability inversion using convolutional neural networks and positron emission tomography. *Water Resources Research*, *58*(3), e2021WR031554. https://doi.org/10.1029/2021WR031554

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

Illman, W. A., Zhu, J., Craig, A. J., & Yin, D. (2010). Comparison of aquifer characterization approaches through steady state groundwater model validation: A controlled laboratory sandbox study. *Water Resources Research*, *46*(4), W04502. https://doi.org/10.1029/2009WR007745

Illman, W. A., Craig, A. J., & Liu, X. (2008). Practical issues in imaging hydraulic conductivity through hydraulic tomography. *Ground Water*, *46*(1), 120–132. https://doi.org/10.1111/j.1745-6584.2007.00374.x

Jardani, A., Vu, T. M., & Fischer, P. (2022). Use of convolutional neural networks with encoder-decoder structure for predicting the inverse operator in hydraulic tomography. *Journal of Hydrology*, *604*, 127233. https://doi.org/10.1016/j.jhydrol.2021.127233

Kitanidis, P., & Lee, J. (2014). Principal component geostatistical approach for large-dimensional inverse problems. *Water Resources Research*, *50*(7), 5428–5443. https://doi.org/10.1002/2013WR014630

Kitanidis, P. K. (1995). Quasi-linear geostatistical theory for inversing. *Water Resources Research*, *31*(10), 2411–2419. https://doi.org/10.1029/95WR01945

Klein, O., Cirpka, O. A., Bastian, P., & Ippisch, O. (2017). Efficient geostatistical inversion of transient groundwater flow using preconditioned nonlinear conjugate gradients. *Advances in Water Resources*, *102*, 161–177. https://doi.org/10.1016/j.advwatres.2016.12.006

Laloy, E., Hérault, R., Jacques, D., & Linde, N. (2018). Training-image based geostatistical inversion using a spatial generative adversarial neural network. *Water Resources Research*, *54*(1), 381–406. https://doi.org/10.1002/2017WR022148

Lee, J., & Kitanidis, P. (2014). Large scale hydraulic tomography and joint inversion of head and tracer data using the Principal Component Geostatistical Approach (PCGA). *Water Resources Research*, *50*(7), 5410–5427. https://doi.org/10.1002/2014WR015483

Liu, X., Zhou, Q., Birkholzer, J., & Illman, W. A. (2013). Geostatistical reduced-order models in underdetermined inverse problems. *Water Resources Research*, *49*(10), 6587–6600. https://doi.org/10.1002/wrcr.20489

Liu, X., Zhou, Q., Kitanidis, P. K., & Birkholzer, J. T. (2014). Fast iterative implementation of large-scale nonlinear geostatistical inverse modeling. *Water Resources Research*, *50*(1), 198–207. https://doi.org/10.1002/2012WR013241

Liu, Y. (2006). Using the Snesim program for multiple-point statistical simulation. *Computers & Geosciences*, *32*(10), 1544–1563. https://doi.org/10.1016/j.cageo.2006.02.008

Mosser, L., Dubrule, O., & Blunt, M. J. (2017). Reconstruction of three-dimensional porous media using generative adversarial neural networks. *Physical Review E*, *96*(4), 043309. https://doi.org/10.1103/PhysRevE.96.043309

Nowak, W., & Cirpka, O. A. (2004). A modified Levenberg–Marquardt algorithm for quasi-linear geostatistical inversing. *Advances in Water Resources*, *27*(7), 737–750. https://doi.org/10.1016/j.advwatres.2004.03.004

Nowak, W., Tenkleve, S., & Cirpka, O. A. (2003). Efficient computation of linearized cross-covariance and auto-covariance matrices of interdependent quantities. *Mathematical Geology*, *35*(1), 53–66. https://doi.org/10.1023/A:1022365112368

Radford, A., & Metz, L. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv*. https://doi.org/10.48550/arXiv.1511.06434

Raissi, M., Perdikaris, P., & Karniadakis, G. (2017a). Physics informed deep learning (Part I): Data-driven solutions of nonlinear partial differential equations. arXiv, abs/1711.10561. https://doi.org/10.48550/arXiv.1711.10561

Raissi, M., Perdikaris, P., & Karniadakis, G. (2017b). Physics informed deep learning (Part II): Data-driven discovery of nonlinear partial differential equations. arXiv, abs/1711.10566. https://doi.org/10.48550/arXiv.1711.10566

Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, *378*, 686–707. https://doi.org/10.1016/j.jcp.2018.10.045

Saibaba, A. K., Ambikasaran, S., Li, J. Y., Kitanidis, P. K., & Darve, E. F. (2012). Application of hierarchical matrices to linear inverse problems in geostatistics. *Oil and Gas Science and Technology*, *67*(5), 857–875. https://doi.org/10.2516/ogst/2012064

Strebelle, S. (2000). *Sequential simulation drawing structures from training images*. Ph.D. Dissertation (p. 187). Stanford University.

Strebelle, S. (2002). Conditional simulation of complex geological structures using multiple-point statistics. *Mathematical Geology*, *34*(1), 1–21. https://doi.org/10.1023/A:1014009426274

Tartakovsky, A. M., Marrero, C. O., Perdikaris, P., Tartakovsky, G. D., & Barajas-Solano, D. (2020). Physics-informed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems. *Water Resources Research*, *56*(5), e2019WR026731. https://doi.org/10.1029/2019WR026731

Tosaka, H., Masumoto, K., & Kojima, K. (1993). *Hydropulse tomography for identifying 3-D permeability distribution*. American Nuclear Society, Inc.

Vo, H., & Durlofsky, L. (2014). A new differentiable parameterization based on principal component analysis for the low-dimensional representation of complex geological models. *Mathematical Geosciences*, *46*(7), 775–813. https://doi.org/10.1007/s11004-014-9541-2

Vu, M. T., & Jardani, A. (2022a). Mapping of hydraulic transmissivity field from inversion of tracer test data using convolutional neural networks. CNN-2T. *Journal of Hydrology*, *606*, 127443. https://doi.org/10.1016/j.jhydrol.2022.127443

Vu, M. T., & Jardani, A. (2022b). Mapping discrete fracture networks using inversion of hydraulic tomography data with convolutional neural network: SegNet-Fracture. *Journal of Hydrology*, *609*, 127752. https://doi.org/10.1016/j.jhydrol.2022.127752

Wang, N., Chang, H., & Zhang, D. (2021a). Deep-learning-based inverse modeling approaches: A subsurface flow example. *Journal of Geophysical Research: Solid Earth*, *126*(2), e2020JB020549. https://doi.org/10.1029/2020JB020549

Wang, N., Chang, H., & Zhang, D. (2021b). Theory-guided Auto-Encoder for surrogate construction and inverse modeling. *Computer Methods in Applied Mechanics and Engineering*, *385*, 114037. https://doi.org/10.1016/j.cma.2021.114037

Yeh, T., Jin, M., & Hanna, S. (1995). An iterative stochastic inverse method: Conditional effective transmissivity and hydraulic head fields.

Yeh, T., & Liu, S. (2000). Hydraulic tomography: Development of a new aquifer test method. *Water Resources Research*, *36*(8), 2095–2105. https://doi.org/10.1029/2000WR900114

Yeh, T. C., & Lee, C. H. (2007). Time to change the way we collect and analyze data for aquifer characterization. *Ground Water*, *45*(2), 116–118. https://doi.org/10.1111/j.1745-6584.2006.00292.x

Zha, Y., Yeh, T.-C. J., Illman, W. A., Zeng, W., Zhang, Y., Sun, F., & Shi, L. (2018). A reduced-order successive linear estimator for geostatistical inversion and its application in hydraulic tomography. *Water Resources Research*, *54*(3), 1616–1632. https://doi.org/10.1002/2017WR021884

Zhao, Y., Guo, Q., Lu, C., & Luo, J. (2022). High-dimensional groundwater flow inverse modeling by upscaled effective model on principal components. *Water Resources Research*, *58*(7), e2022WR032610. https://doi.org/10.1029/2022WR032610

Zhao, Y., & Luo, J. (2020). Reformulation of Bayesian geostatistical approach on principal components. *Water Resources Research*, *56*(4), e2019WR026732. https://doi.org/10.1029/2019WR026732

Zhao, Y., & Luo, J. (2021a). A Quasi-Newton reformulated geostatistical approach on reduced dimensions for large-dimensional inverse problems. *Water Resources Research*, *57*(1), e2020WR028399. https://doi.org/10.1029/2020WR028399

Zhao, Y., & Luo, J. (2021b). Bayesian inverse modeling of large-scale spatial fields on iteratively corrected principal components. *Advances in Water Resources*, *151*, 103913. https://doi.org/10.1016/j.advwatres.2021.103913

Zhao, Z., & Illman, W. A. (2021). On the importance of considering specific storage heterogeneity in hydraulic tomography: Laboratory sandbox and synthetic studies. *Journal of Hydrology*, *593*, 125874. https://doi.org/10.1016/j.jhydrol.2020.125874

Zhou, Z., Zabaras, N., & Tartakovsky, D. M. (2022). Deep learning for simultaneous inference of hydraulic and transport properties. *Water Resources Research*, *58*(10), e2021WR031438. https://doi.org/10.1029/2021WR031438

Zhu, J., & Yeh, T.-C. J. (2005). Characterization of aquifer heterogeneity using transient hydraulic tomography. *Water Resources Research*, *41*(7), W07028. https://doi.org/10.1029/2004WR003790