energies

MDPI

Article

Empowering Reservoir Optimization with AI: Deep Learning Surrogates for Intelligent Control Under Variable Well Conditions

Hu Huang [1], Bin Gong [2,*], Zhengkai Lan [3] and Jinghua Yang [2]

[1] School of Mathematics and Physics, Wuhan Institute of Technology, Wuhan 430205, China; huanghu_cug@126.com
[2] School of Resources, China University of Geosciences (Wuhan), Wuhan 430074, China; yang2202210119_cug@126.com
[3] Marginal Oil and Gas Development Technology Research Institute, Xi'an Shiyou University, Xi'an 710065, China; kaimo666@xsyu.edu.cn
* Correspondence: gongbin@tracyenergy.cn

Abstract

The advancement of Industry 5.0 hinges on the deep integration of artificial intelligence (AI) with domain expertise to foster sustainable industrial development. This study proposes a deep learning-based surrogate modeling framework that integrates reservoir production requirements with AI technologies, providing intelligent decision support for production optimization and enhanced efficiency. To evaluate AI's effectiveness in complex industrial scenarios, we conduct an integrated analysis encompassing model construction, dynamic prediction, and production optimization using a real-world oilfield case. This oilfield features a dynamically increasing number of wells and requires dynamic adjustments to injection–production relationships. To address this challenge, we enhance the Embed-to-Control model by improving the nonlinear representation capability within its decoder structure. Subsequently, we construct a high-fidelity dataset containing 300 samples for model training and testing. The experimental results demonstrate that the proposed improved model achieves a high accuracy in predicting key state variables (pressure and saturation) and oil production. Regarding computational efficiency, a single model run requires only approximately 17.3 s, achieving an over 200× speedup relative to traditional numerical simulators. Finally, we coupled the trained surrogate model with the particle swarm optimization algorithm to optimize the injection well control strategy. The optimized scheme increases daily oil production by 13.84%, boosting economic benefits. This study demonstrates a practical technological pathway to accelerate the oil and gas industry's transition toward Industry 5.0.

Keywords: reservoir simulation; deep learning; embedded control; production optimization; proxy model; multi-well injection optimization; Industry 5.0

Check for updates

Academic Editor: Reza Rezaee

Received: 19 December 2025
Revised: 14 January 2026
Accepted: 28 January 2026
Published: 10 February 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

In recent years, the global manufacturing industry has been undergoing a profound transformation with the advent of Industry 5.0. Unlike its predecessor, Industry 4.0, which focused primarily on automation, digitalization, and the integration of cyber–physical systems, Industry 5.0 emphasizes the collaboration between humans and intelligent systems, aiming to create more resilient, sustainable, and human-centric manufacturing processes. This new paradigm leverages advanced technologies, such as artificial intelligence (AI), big

data analytics, the Internet of Things (IoT), and robotics, not only to enhance productivity and efficiency but also to enable flexible, adaptive, and personalized solutions across various sectors of the manufacturing industry Van Erp, T et al. (2024) [1].

As a critical component of the manufacturing sector, the oil and gas industry is also embracing the principles of Industry 5.0 to address the increasing complexity and challenges in reservoir management and production optimization. By integrating AI-driven approaches with traditional engineering expertise, the industry is moving towards intelligent decision-making and autonomous operations, which are essential for achieving higher efficiency, sustainability, and competitiveness in the era of smart manufacturing Zhang G.Y. et al. (2024) [2]; Zhao B. et al. (2025) [3]. In this context, integrating artificial intelligence and deep learning methods has become a highly promising direction in the field of reservoir engineering. These alternative models and intelligent optimization algorithms developed based on artificial intelligence technology can quickly and accurately predict the production dynamics of oil and gas reservoirs, enhance intelligent decision-making, and thereby achieve rapid improvements in productivity and production efficiency. This will accelerate the progress of industrialization 5.0 in the petroleum industry.

In the development of a surrogate model, Zhang X.D. et al. (2018) [4] generated artificial logging curves based on existing logging curves from a machine learning method, which achieved good application results. However, these purely data-driven methods have inherent limitations in terms of physical consistency and extrapolation. Subsequently, the research moved towards more complex dynamic predictions and geological characterizations. Zhu Y. and Zabaras N. (2018) [5] constructed a model for predicting two-dimensional fluid pressure and velocity using Bayesian and Dense Block methods. Mo et al. (2019) [6] used the Dense Block model to solve the problem of pressure and saturation changes when CO_2 was stored underground, but this model has a poor extrapolation effect on time steps. In order to enhance the accuracy of geological characterization, Liu Y. et al. (2019, 2010) [7,8] proposed the CNN-PCA model to reduce the geological dimension. By reducing the dimension of input geological maps with a convolutional neural network, the model overcomes the shortcoming that PCA cannot fully reflect geological characteristics. However, this method was only tested in river flows and lacks an analysis of its actual effectiveness. Zhu Y. et al. (2019) [9] solved the steady-state flow problem in random homogeneous media with a convolutional neural network. Based on big data and machine learning, Jia D. et al. (2020) [10] put forward a set of optimization methods for a fine water injection scheme driven by big data, which made cumulative oil production increase by 7.2%. Shahkarami A. and Mohaghegh S. (2020) [11] used artificial intelligence and machine learning techniques to develop an intelligent surrogate model for reservoir history fitting, sensitivity analysis, and uncertainty assessment. This model has great advantages in computing speed, time consumption, and cost. Kim J. et al. (2020) [12] sought the optimal well location design with convolution. These efforts collectively demonstrate the significant advantages of AI methods in enhancing computational efficiency and decision-making speed. However, the models employed are relatively simple, and their practical application value is limited.

With the development of deep learning technology, the research focus has shifted to utilizing more powerful network structures to achieve efficient end-to-end simulation and prediction. Tang M. et al. (2020) [13] simulated the changes in pressure and saturation distribution of reservoirs of different permeabilities over time with the R-U-Net model and calculated well responses using the Peaceman formula. The R-U-Net model takes 2 s to predict the state of a reservoir at different moments. However, its accuracy is highly dependent on the completeness of the training data. Wang H.L. et al. (2020) [14] predicted the production of some oilfields in the ultra-high water cut period by using a deep learning method effectively. Zhong Z. et al. (2020) [15] used a generative adversarial neural network

to simulate the change in oil field saturation over time in waterflood development. But this model shows significant errors in capturing the abrupt change in water production wells, reflecting the difficulties of generative models in restoring dynamic details. Gao Z.K. et al. (2021) [16] used a deep learning method based on complex convolutional networks to characterize the gas–liquid flow. Kuang L. et al. (2021) [17] summarized and prospectively reviewed the research of artificial intelligence in the field of oil and gas exploration. Ma C.J. (2021) [18] used an LSTM model to simulate oil production and obtained good results. Chaki S. et al. (2021) [19] predicted variations in field information (pressure or saturation) and the production of oil reservoirs with a U-Net network, saving time and costs. Cornelio J. et al. (2021) [20] transferred a model trained on one oilfield's data to another oilfield through transfer learning, thus reducing the data requirements and the workload of building new models. Based on the U-Net, Zhong Z. et al. (2021) [21] proposed generating an antagonistic neural network (Co-Gan) by coupling to predict the changes in reservoir pressure and fluid saturation in the process of water injection in heterogeneous reservoirs. The inputs of the model include the reservoir static properties (permeability), injection rate, and time to be predicted, and the outputs include the dynamic state of the reservoir corresponding to the predicted time. Alakeely A. and Horne R. (2022) [22] constructed surrogate models with historical production data using the generative deep learning (GDL) method to predict the production of new wells. Relying on more than 40 million sets of historical dynamic monitoring data covering different types of oil wells, Wang X. et al. (2022) [23] prepared oil well condition diagnosis sample sets covering 37 types of working conditions in 5 major categories. On this basis, the convolutional neural network algorithm was selected, and the intelligent diagnosis technology was constructed by using the new generation of artificial intelligence technology of "big data + artificial intelligence". Using this technology, they completed condition diagnosis 5 million times in the field, with an accuracy of 90% and timely alarm pushing. Yang Y. (2022) [24] combined GNN with seepage physical process information to establish a graph neural network model suitable for the injection and production balance training of a well pattern. Huang H. et al. (2023) [25] also used GNN for reservoir simulation and optimization with varying well controls.

In summary, the intelligent surrogate models examined in current research exhibit a pronounced path dependence in their development: their core architectures are largely derived from computer vision (e.g., CNNs, U-Net, GANs), emphasizing the learning of spatiotemporal patterns from data. Yet they generally fail to rigorously embed the reservoir governing equations—capturing the essence of fluid flow—as hard physical constraints, and consequently their predictions often lack robust physical consistency and display a limited extrapolative capability. Meanwhile, most reported advances are grounded in simplified, small-scale theoretical models, and their efficiency and accuracy remain insufficiently validated in real industrial contexts characterized by complex geology, long-term production, large-scale well grids, and comprehensive oilfield management. This highlights a substantial disconnect between theoretical research and large-scale practical application.

In the application of the flow equation, many processing methods based on deep learning models are quite different from those of reduced order models (ROMs) based on partial differential equations and numerical discretization. In the study of reduced-order models, Watter M. et al. (2015) [26] proposed an embedded control framework (E2C) to predict the evolution of the system state with sensor data (images) and control conditions. Jin Z.L. et al. (2020) [27] constructed an end-to-end model based on E2C to realize the prediction of production with the changing control conditions of wells in the reservoir. Firstly, the model uses an AE encoder to reduce the dimension of the input data and extract features; then, according to the control conditions of wells, the fully connected converter is used to obtain new features; finally, the distributions of saturation and pressure

at the new time are obtained by the decoder. Although the E2C model proposed by Jin et al. works well in simple reservoirs, it is poor in accuracy in large-scale reservoirs and cannot give production information directly. Qin F. et al. (2022) [28] used the E2C model to simulate the gas reservoir problems with small wells. Huang H. et al. (2023) [29] optimized the E2C model proposed by Jin to improve its performance and added the prediction of production evolution to it. In the optimized model, skip connection is used to improve the accuracy of the decoder, while the converter is modified to directly calculate production information. Although the model has been verified in some reservoirs, the reservoirs used in the verification were small, short in production period, and simple in well production scheme.

In this work, by leveraging actual production data from a complex oil field, we utilized an Embed-to-Control (E2C) model to predict reservoir pressure, saturation, and well response dynamics. Subsequently, we employ the trained surrogate model to optimize water injection volumes for enhanced oil production. For the first time, a fully integrated deep learning workflow combining model construction, training, and optimization decision-making is used within a realistic, complex reservoir setting that contains frequent injection-to-production conversions. Furthermore, we innovatively enhanced the E2C architecture by optimizing the production calculation network using residual connections, specifically overcoming original network limitations in accurately simulating intricate well control dynamics like injection–production conversion. The application of this optimized surrogate model integrated with optimization algorithms demonstrably improved daily oil production by 13.84%, significantly boosting operational efficiency. Finally, this work exemplifies the surrogate model's high practical value in Industry 5.0, providing a validated pathway for AI-driven intelligent decision-making and efficiency gains within industrial production systems.

2. Theoretical Basis

Based on the mass conservation law of each component, and combined with Darcy's law, the equation governing immiscible oil–water flow is derived as follows:

$$\frac{\partial}{\partial t}(\phi\rho_j S_j) - \nabla\cdot\left[\rho_j\lambda_j k\left(\nabla p_j - \rho_j g\nabla D\right)\right] + q_j^w = 0. \quad (1)$$

In Equation (1), the subscript j ($j = o, w$ for oil and water, respectively) denotes the phase of the fluid, ϕ and k denote porosity and permeability, respectively, λ_j denotes the interaction between the rock and fluid, p_j and S_j represent pressure and saturation, respectively, $]\rho_j$ represents density, q_j^w is the source/sink terms for the well w, g represents gravitational acceleration, and D represents depth. The variable $x_t = \left[p_t^T, S_t^T\right] \in R^{2N}$ denotes the state vector of the calculated area at a specific time step t, N is the number of grids in the model, $u_{t+1} \in R^{n_w}$ denotes the control conditions of the wells from time t to time t + 1, and n_w indicates the number of all wells in the system. Solving the governing Equation (1) with a fully implicit finite volume method, the following equation is obtained:

$$\mathrm{h}^{\mathrm{t+1}} = \mathrm{h}(\mathrm{x}_{\mathrm{t+1}}, \mathrm{x}_{\mathrm{t}}, \mathrm{u}_{\mathrm{t+1}}) = 0. \quad (2)$$

where h is a function, the subscript t indicates the current moment, and t + 1 indicates the next moment. The full-order discrete nonlinear system defined by Equation (2) is usually solved using the Newton method, so each iteration needs to construct a sparse Jacobian matrix of $2N \times 2N$ dimensions, and the solution of these matrices is highly time-consuming, so how to reduce the dimensions of high-dimensional systems has become a major research direction.

He J. and Durlofsky L.J. (2014, 2015) [30,31] proposed a method combining orthogonal decomposition and trajectory piecewise linearity (POD-TPWL) to solve the above equation. In this method, Equation (2) is first Taylor expanded to give the following:

$$g^{t+1} = 0 \approx g^{i+1} + \frac{\partial g^{i+1}}{\partial x_{i+1}}(x_{t+1} - x_{i+1}) + \frac{\partial g^{i+1}}{\partial x_i}(x_t - x_i) + \frac{\partial g^{i+1}}{\partial u_{i+1}}(u_{t+1} - u_{i+1}). \quad (3)$$

The subscript i + 1 represents some known solutions (which can be regarded as a training dataset used to determine some parameters), and t + 1 represents a new example to be calculated. For details on the POD-TPWL's implementation, see He J. and Durlofsky L.J. (2014, 2015) [30,31].

By means of orthogonal decomposition, the high-dimensional state matrix x can be represented by the low-dimensional variable $\xi \in R^{l_\xi}$, namely

$$x = \Phi\xi. \quad (4)$$

where l_ξ denotes the dimension of the lower dimensional space, satisfying $l_\xi \ll N$. Because Φ is orthogonal, it is deduced that $\xi = \Phi^T x$, and thus the mutual transformation between high-dimensional space and low-dimensional space is realized.

In the POD-TPWL method, by using Newton iteration, Equation (3) can be expressed as follows:

$$J_{i+1}x_{t+1} = J_{i+1}x_{i+1} - [A_{i+1}(x_t - x_i) + B_{i+1}(u_{t+1} - u_{i+1})]. \quad (5)$$

where $J_{i+1} = \frac{\partial g^{i+1}}{\partial x_{i+1}} \in R^{2N\times 2N}$, $A_{i+1} = \frac{\partial g^{i+1}}{\partial x_i} \in R^{2N\times 2N}$, and $B_{i+1} = \frac{\partial g^{i+1}}{\partial x_{i+1}} \in R^{2N\times n_w}$. Substituting Equation (4) into Equation (5), we obtain

$$\hat{\xi}_{t+1} = \xi_{i+1} - \left(J^r_{i+1}\right)^{-1}\left[A^r_{i+1}(\xi_t - \xi_i) + B^r_{i+1}(u_{t+1} - u_{i+1})\right]. \quad (6)$$

where $J^r_{i+1} = (\Psi_{i+1})^T J_{i+1}\Phi$, $A^r_{i+1} = (\Psi_{i+1})^T A_{i+1}\Phi$, and $B^r_{i+1} = (\Psi_{i+1})^T B_{i+1}\Phi$, Ψ_{i+1} is a constraint matrix. Since ξ_t is also unknown in the iterative calculation, it is replaced by $\hat{\xi}_t$ approximately, and the final result can be expressed as follows:

$$\hat{\xi}_{t+1} - \xi_{i+1} = \left[-\left(J^r_{i+1}\right)^{-1}A^r_{i+1}\left(\hat{\xi}_t - \xi_i\right)\right] + \left[-\left(J^r_{i+1}\right)^{-1}B^r_{i+1}(u_{t+1} - u_{i+1})\right]. \quad (7)$$

Suppose $\hat{Z}_{t+1} = \hat{\xi}_{t+1} - \xi_{i+1}$, $\hat{Z}_t = \hat{\xi}_t - \xi_i$, $A_t = -\left(J^r_{i+1}\right)^{-1}A^r_{i+1}$, and $B_t = -\left(J^r_{i+1}\right)^{-1}B^r_{i+1}$; then, Equation (7) can be expressed as follows:

$$\hat{Z}_{t+1} = A_t\hat{Z}_t + B_t U_{t+1}. \quad (8)$$

where $\hat{Z}_t$ represents the characteristic of the low-dimensional space at the current moment, $\hat{Z}_{t+1}$ represents the characteristic of the low-dimensional space at the new moment, and U_{t+1} represents the control conditions of the well under the current production state; A_t and B_t are two matrixes related to the current characteristic and time interval. Based on Equation (8), the features at different moments in the low-dimensional space can be iterated, and then the solution in the high-dimensional space can be obtained through the inverse transformation of Equation (4).

It can be seen from the above introduction that POD-TPW generally includes three steps: the projection from a high-dimensional space to a low-dimensional subspace; a dynamic linear approximation in the low-dimensional subspace; and mapping from the low-dimensional space to the high-dimensional space. Each of these steps involves lots of matrix calculations. In order to simplify the calculation method and improve the calculation

rate, deep learning is used to replace the matrix solution in each step, and finally the E2C model is generated.

3. Scheme Design

The purpose of a surrogate model is to quickly predict production dynamics based on given inputs. In the construction of the surrogate model, we first need to establish several physical models under different input conditions and then calculate these physical models using a numerical simulation method to extract the required production dynamic data; finally, based on these datasets, the deep learning method is used to obtain the mapping relationship between the input parameters and the production dynamics. The deep learning model can automatically detect useful features from the data and obtain the mapping from input to output. The calculation process can be expressed as follows:

$$y \approx \hat{y} = \hat{f}(x;\theta). \tag{9}$$

where $\hat{f}$ is the surrogate model, x is the input variable, y is the output (pressure, saturation, well production, etc.), and θ is the deep neural network parameter, which needs to be determined in training.

3.1. Model Design

Based on deep learning, the embedded control framework (E2C) was used to predict the pressure, saturation, and well production information in the process of reservoir development in this study. The model was first proposed by Watter et al. and successfully applied to the robot control system. Jin et al. extended the E2C model in 2020 and applied it to reservoir simulation. However, the accuracy of the model is low in actual oilfields. Huang et al. modified the model in 2023 and improved its accuracy in actual oilfields. But the oilfield used to verify the model then had a short production duration and a small number of wells, and even these wells occupied a large proportion of the video memory. In this study, the E2C model proposed by Huang et al. was optimized so that it can complete the development prediction of large-scale complex oilfields with a smaller occupancy rate of the video memory.

The E2C model mainly uses the convolutional neural network to extract features and calculate fields. The convolutional neural network is composed of one or more convolutional layers and active layers, and the specific calculation method is as follows:

$$F_l(x) = \sigma\left(W_l^T \times F_{l-1}(x) + b\right). \tag{10}$$

where σ represents the activation function, F represents the output, x represents the input, and l represents the layers of the neural network. W and b denote weights in the convolution calculation and need to be determined using an optimization algorithm in the training process.

In the proper orthogonal decomposition–piecewise linearization of trajectory (POD-TPWL) principle, solving the reservoir control equation under different well control conditions mainly includes three steps: down-sampling the original space through orthogonal decomposition, transforming the down-sampled features into features at a new moment through linear transformation, and transforming the features at the new moment into the original space through the inverse transformation of orthogonal decomposition. Similarly to the solving process of POD-TPWL, the E2C model is mainly divided into three parts: (1) an encoder that projects data from a high-dimensional space to a low-dimensional space; (2) a converter that performs feature conversion in the low-dimensional space; and (3) a decoder that performs well production performance prediction and pressure and saturation

calculation from the converted low-dimensional space. The specific structure is shown in Figure 1. In the surrogate model, the encoder is used to down-sample the current feature, the converter is used to evolve the time, and the decoder is used to recover a new feature in the spatial dimension.

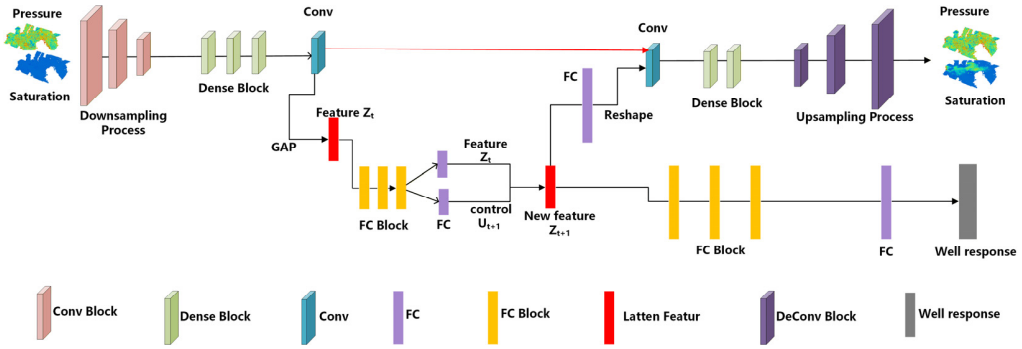


Figure 1. Flow chart of overall calculation of E2C model.

It can be seen from Figure 1 that the inputs of the surrogate model are the pressure distribution, saturation distribution, and well control conditions at the current moment (t), and the outputs are the pressure, saturation, and well production information at the next moment ($t+1$). The figure uses color-coding to differentiate module types: pink denotes the convolutional block (Conv Block), light green the dense block (Dense Block), cyan-blue the convolution operator (Conv), light purple the fully connected layer (FC), yellow the fully connected block (FC Block), red the latent features (e.g., Z_t and Z_{t+1}), dark purple the deconvolutional block (DeConv Block), and gray the well-response output (Well Response). The long red lines represent skip connections from the encoder to the decoder, which inject high-resolution detail and thereby enhance reconstruction quality.

In the encoder, three convolutional blocks—each comprising a convolutional layer, batch normalization, and PReLU (denoted as a Conv Block in Figure 1)—are first applied with a stride of 2 and a $3 \times 3 \times 3$ kernel to extract features from the input information. Then, the Dense Block is used to extract features in the low-dimensional space. The Dense Block is a special convolution structure which still uses convolution operation, but the output is composed of the result of this calculation and the input. Please see the paper of Huang G. et al. [32] in 2017 for more details of the calculation of the Dense Block. Then, a Conv operation is used to fuse the channels to obtain new features, E_t. Finally, global average pooling (GAP) and full connection (FC) are used to obtain the features, Z_t, for subsequent calculations.

In the converter (from Z_t to Z_{t+1}), the feature is updated in the same manner as in Equation (8), where

$$A_t = W_A h_\psi^{trans}(Z_t, \Delta t) + b_A. \tag{11}$$

$$B_t = W_B h_\psi^{trans}(Z_t, \Delta t) + b_B. \tag{12}$$

The variables W_A, b_A, W_B, b_B are the weights to be trained, and h_ψ^{trans} denotes the converter. According to Equations (11) and (12), in the converter, the variable h_ψ^{trans} is first calculated from the feature Z_t using three fully connected blocks (FC Blocks). Similarly to the convolution block, all fully connected blocks consist of "FC + BN + PReLU". The matrices A_t and B_t are then obtained using two different FCs, respectively. Finally, according to Equation (8), the characteristic Z_{t+1} of the next moment is generated with the control conditions U_{t+1} and characteristic Z_t of the well as inputs.

In the decoder, two groups of network structures are used to obtain information on the distributions of pressure and saturation and well production, respectively. First, on the

basis of Z_{t+1}, the production information of the well is calculated using one FC, one FC residual block, one FC Block, and C (the bottom part of the decoder, Z_{t+1} in Figure 1). As the oil field in this study has wells larger in number and complexity, three fully connected layers and residuals are used in the production calculation to increase the nonlinearity of the well production evolution. Then, based on the same Z_{t+1}, the pressure and saturation distributions at the new moment are obtained through a series of operations (the upper right part of Z_{t+1} in Figure 1). In the evolution of pressure and saturation, FC is used first to expand the features and rearrange them according to the dimension of E_t. Since the spatial information is obtained through feature rearrangement, the accuracy is not that high. Therefore, the skip connection is used to rearrange features to improve their accuracy (red arrows in Figure 1). One Conv and two Dense Blocks are then used to compute the features, and three DeConv blocks are used to up-sample the spatial dimensions. Finally, the pressure and saturation distributions at the new moment are obtained by reorganizing the channels through a Conv.

In the above structure, the parameter correction activation function (ParametricRectified Linear Unit, PReLU) is used to make nonlinear changes to the model, and the calculation method is as follows:

$$PReLU(X) = \begin{cases} x, & x \geq 0 \\ ax, & x < 0 \end{cases}. \tag{13}$$

It can be seen from the above calculation that, when the input variable $x < 0$, the gradient of PReLU is a, thus avoiding the phenomenon that the gradient is 0 on a negative number and solving the problem that the gradient is not updated. Slope a can also be learned in model training to reduce the trouble caused by artificial settings.

3.2. Design of Loss Function and Training Hyperparameters

In the model training, the total loss function is composed of pressure loss, saturation loss, and well production loss. The calculation formulas of the losses are Equations (14)–(16).

$$L_{t+1}^{\mu} = \frac{1}{N}\sum_{i=1}^{N} \parallel \mu_i^{t+1} - \hat{\mu}_i^{t+1} \parallel_2^2, \tag{14}$$

$$L_{t+1}^{\varepsilon} = \frac{1}{n_b}\sum_{i=1}^{n_b} \parallel \epsilon_i^{t+1} - \hat{\epsilon}_i^{t+1} \parallel, \tag{15}$$

$$L_{t+1} = \lambda_{\mu} L_{t+1}^{\mu} + \lambda_{\varepsilon} L_{t+1}^{\varepsilon}. \tag{16}$$

In the above equations, the variables with a "hat" represent predicted values, and the variables without a "hat" represent true values, where $\mu = s, p$ represents the saturation and pressure, respectively, and $\epsilon = o, w$ represents the oil production rate and water production rate of the well, respectively. The variable N indicates the number of all grids and n_b indicates the total number of production wells. The variable L_{t+1} represents the final loss and λ represents the weight.

4. Test and Analysis of the Model

Taking a typical low-permeability water drive reservoir in Northwest China as an example, the performance of the E2C model has been analyzed using three aspects: the example background, the analysis of model results, and optimization and tests.

4.1. Design of the Test Example

The reservoir model has 1,312,830 (503 × 435 × 6) grids and 478 wells. The permeability distribution in the X direction and some properties of the reservoir are shown in Figure 2 and Table 1.

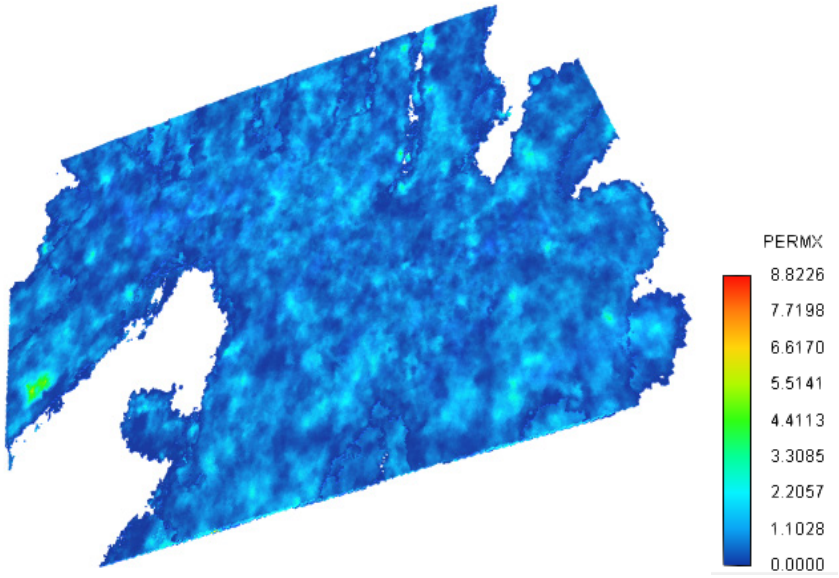


Figure 2. Permeability of this block in X direction.

Table 1. Property distributions of the reservoir in the block.

Attribute	Minimum Value	Maximum Value	Average Value
Permeability X (MD)	0.000	8.821	0.564
Permeability Y (MD)	0.000	15.962	0.665
Permeability Z (MD)	0.000	1.163	0.072
Porosity	0.005	0.310	0.103

The development period of the block is from 1 December 2004 to 1 April 2019, and the control conditions of the well were revised once a month. The control condition of the injection wells was a constant daily water injection rate, and the condition of the production wells was a constant daily liquid production rate. As time went on, new wells were added, and the number of production wells changed accordingly, so the flow in the reservoir became more and more complex, adding difficulty to the simulation of the surrogate model.

For this reservoir, 300 training datasets were first generated by the simulator, of which 270 were used to train the model and 30 were used to test the model. In the prediction stage, the trained model was loaded first. Then, the pressure and saturation of the zero month (initial time) were taken as inputs, and the distributions of pressure, saturation, and production of the first month were obtained according to the control conditions of the first month. Then, the pressure and saturation of the first month obtained from E2C were used as inputs, and the pressure, saturation, and dynamic response of the wells in the second month were obtained according to the control conditions of the second month. Finally, the pressure, saturation, and well production-related information at all the time steps were obtained by continuously calling the model according to the above method. The specific calculation process is shown in Figure 3.

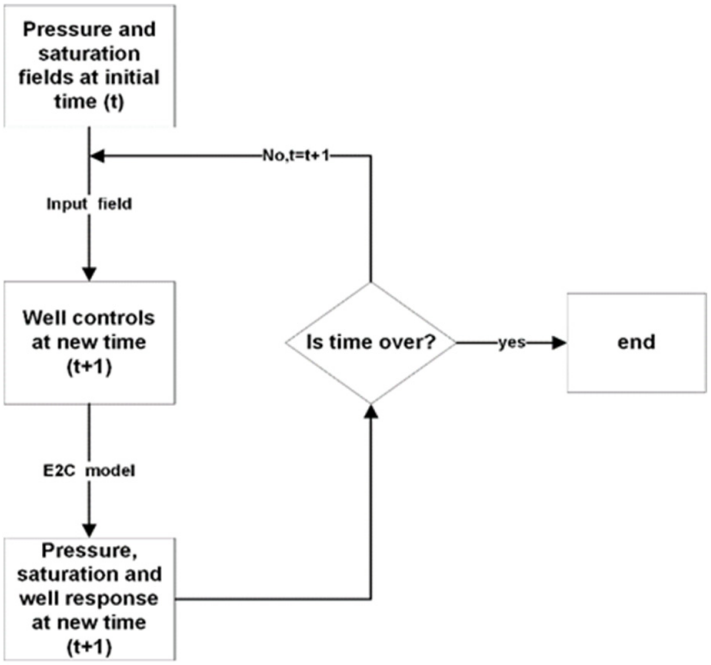


Figure 3. Schematic diagram of the E2C prediction process.

Based on the loss obtained from Equation (16), the optimal hyperparameters determined using the random grid search algorithm were as follows: a learning rate of 2×10^{-4}, the optimizer Adam, and the learning rate decaying once every 10 epochs at a decay coefficient of 0.6, pressure loss weight λ_s of 1, saturation loss weight λ_p of 1, weight of oil production loss λ_o of 20, and weight of water production loss λ_w of 10. The model was trained by setting it to the above hyperparameters.

4.2. Analysis of Model Test Results

Firstly, based on the pressure and saturation on 1 December 2004, changes in the reservoir at each month in the 172 months were predicted in turn according to the control conditions in the corresponding months. Then, the performance of the E2C model was analyzed.

Based on the examples randomly selected from the test set, the pressure and saturation changes in the 30th, 60th, 90th, 120th, 150th, and 170th months were simulated. The results are shown in Figures 4 and 5.

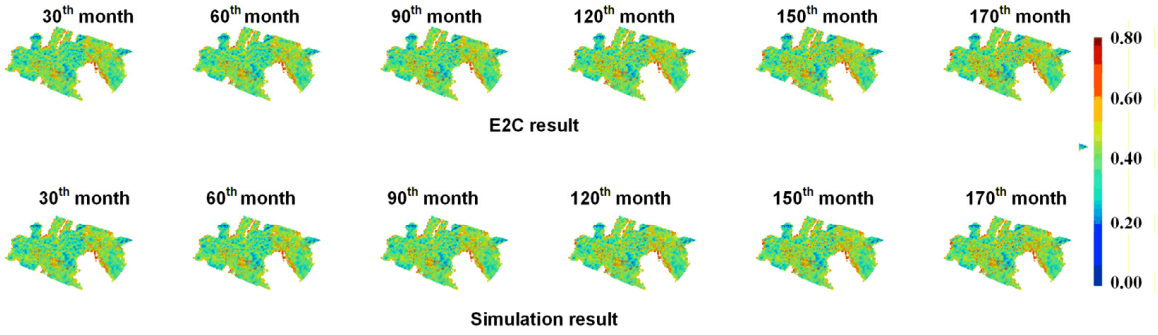


Figure 4. Variation trend of saturation with time obtained using E2C and numerical simulator (in the first row are the E2C simulation results, and in the second row are the simulation results of the simulator).

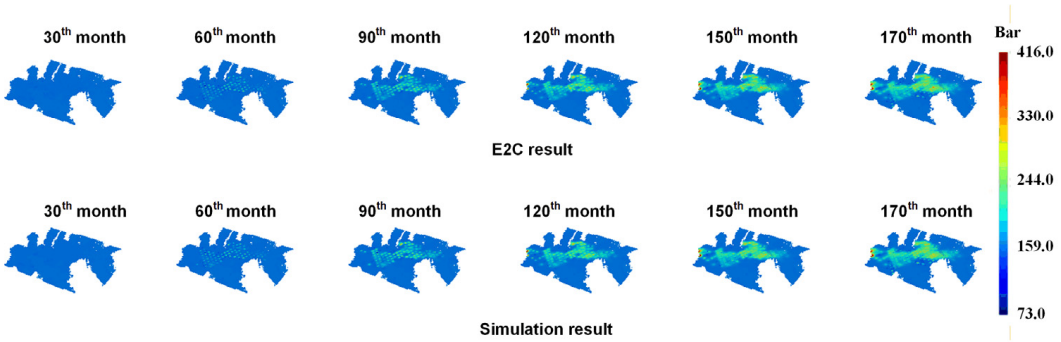


Figure 5. Variation trend of pressure with time obtained using E2C and simulator (in the first row are the E2C simulation results, and in the second row are the simulation result of the simulator).

Figures 4 and 5 show the simulation results of the E2C model and the numerical simulator for the randomly selected examples, respectively, where the E2C simulation results are in the first row and the simulation results of the numerical simulator are in the second row. It can be seen from Figures 4 and 5 that the results of the E2C model are basically consistent with the results obtained using the numerical simulator in terms of the predicted pressure and saturation.

In order to quantitatively analyze the accuracy of the pressure and saturation prediction of the E2C model from a numerical point of view, the concept of average relative error is introduced, and the specific calculation process is shown in Equation (17):

$$\begin{aligned} P_{t+1}^{re} &= \frac{1}{N}\sum_{x=1}^{n_x}\sum_{y=1}^{n_y}\sum_{z=1}^{n_z}\frac{\|P_{t+1}^{x,y,z}-\hat{P}_{t+1}^{x,y,z}\|}{P_{t+1}^{x,y,z}+\epsilon} \\ S_{t+1}^{re} &= \frac{1}{N}\sum_{x=1}^{n_x}\sum_{y=1}^{n_y}\sum_{z=1}^{n_z}\frac{\|S_{t+1}^{x,y,z}-\hat{S}_{t+1}^{x,y,z}\|}{S_{t+1}^{x,y,z}+\epsilon} \end{aligned}. \tag{17}$$

where P represents the pressure, S represents the saturation, superscripts x, y, z represent the coordinates of the grid points, n_x, n_y, n_z represent the dimensions of the grid, N represents the number of test datasets, and the superscript re represents error. The subscript $t+1$ indicates time, and $\varepsilon = 1 \times 10^3$ is used to avoid the case where the denominator is 0. The pressure and saturation errors calculated using Equation (17) are shown in Figure 6. It can be seen from Figure 6 that the saturation and pressure predicted by the E2C model have maximum relative errors of less than 1% and less than 2%, respectively. Apparently, the E2C model has a higher accuracy in predicting pressure and saturation.

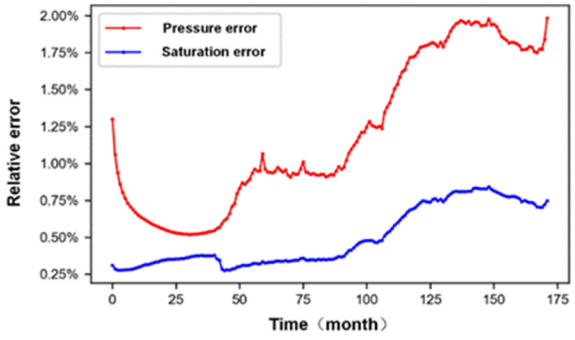


Figure 6. Curve of relative errors of pressures and saturations predicted by E2C simulation with time.

Next, four wells were randomly selected and their daily production rates over time were analyzed, and the results are shown in Figures 7 and 8. It can be seen from them that the production obtained using the E2C model is basically consistent with the simulation result of the numerical simulator. For well B151, although the daily oil and water productions predicted using E2C at some moments are smaller than the results of the

numerical simulator (the peak value of the red line of B151 is lower than the blue line in the figure), with a very low production, this well has little impact on the total production of the reservoir.

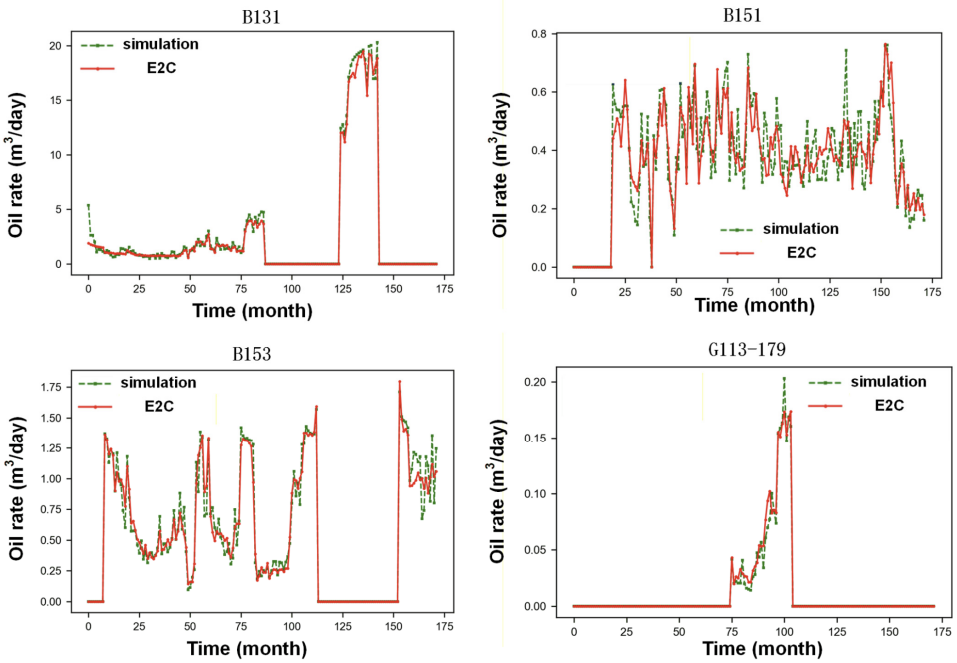


Figure 7. Variations in daily oil production of some wells with time predicted by different models (the red solid line represents the results of the E2C model proposed in this paper, and the green dotted line represents the results of the simulator).

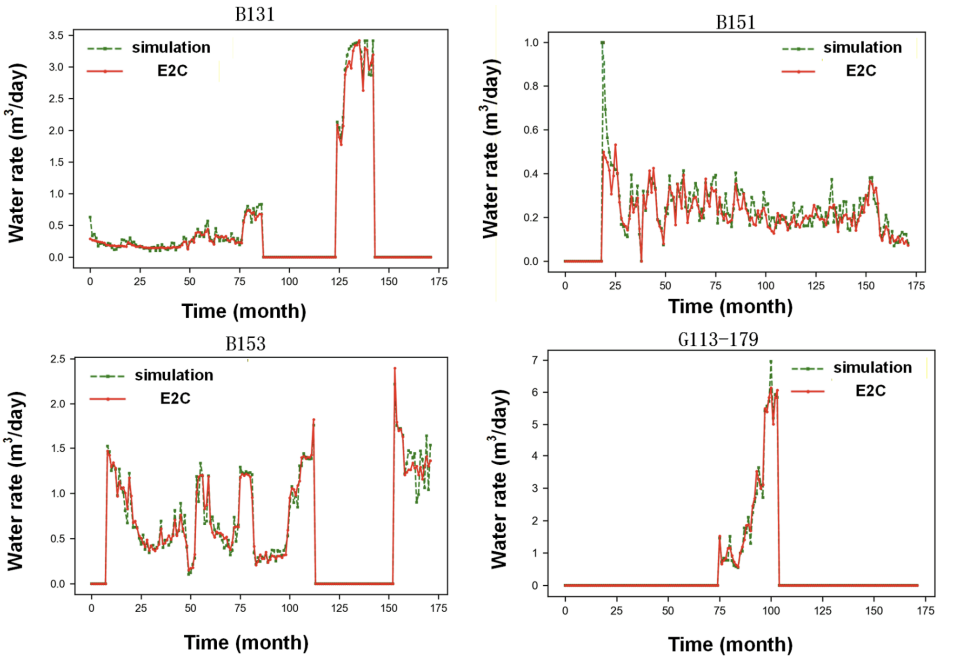


Figure 8. Variations in daily water production of some wells with time predicted by different models (the red solid line represents the results of the E2C model proposed in this paper, and the green dotted line represents the results of the simulator).

To analyze the accuracy of the daily production of the reservoir, the daily production of the reservoir at different moments was calculated using Equation (18). The specific calculation method is as follows:

$$TQ_t^r = \sum_{i=0}^{n_w} Q_{it}^r \tag{18}$$

where $r = o, w$ represents oil and water, respectively, t represents the moment, i represents the well, n_w represents the total number of wells, Q represents the daily production of each well, and TQ represents the daily production of the reservoir. According to the above equation, the daily reservoir productions at different moments were obtained using the simulator and E2C, and the results are shown in Figures 9 and 10. It can be seen from the two figures that all data points are distributed around the 45° line, which proves that the E2C model has a higher consistency and reliable accuracy in predicting the daily production of the reservoir compared with the numerical simulator.

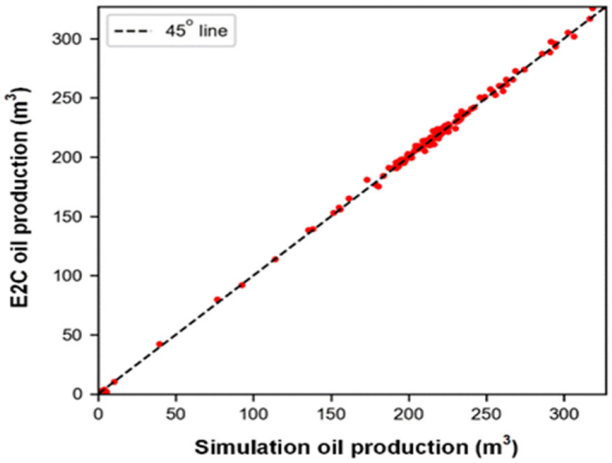


Figure 9. Comparison of daily oil production of the reservoir from E2C simulation and that from the numerical simulator (the red circles represent the total oil production of the oilfield at a certain point in time as obtained by the surrogate model).

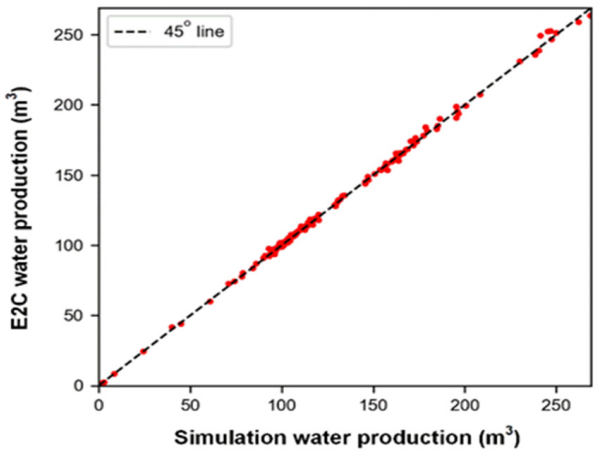


Figure 10. Comparison of the daily water production of the reservoir from the E2C simulation and that from the numerical simulator (the red circles represent the total water production of the oilfield at a certain point in time as obtained by the surrogate model).

In order to avoid the contingency of random sampling, the relative errors of all test sets were counted and analyzed. As the daily production of some wells is 0 at some moments,

in order to reduce the impact of the productions with a denominator of 0 on the error, the error calculation method used in this paper is as follows:

$$AE^r = \frac{1}{T}\frac{1}{n_w}\sum_{t=1}^{T}\sum_{i=1}^{n_w}\left|Q_{it}^r - \hat{Q}_{it}^r\right| \tag{19}$$

$$AP^r = \frac{1}{T}\frac{1}{n_w}\sum_{t=1}^{T}\sum_{i=1}^{n_w}\left|Q_{it}^{ir}\right| \tag{20}$$

$$RE^r = \frac{AE^r}{AP^r} \tag{21}$$

In Equations (19)–(21), the superscripts $r = o, w$ represent oil and water, respectively, t represents the moment, T represents the number of all time steps, i represents the well, n_w represents the number of wells, Q represents the daily production of the well, AE represents the absolute error of the average daily production in each example, AP represents the average daily production in each example, and RE represents the relative error of the average daily production in each example.

According to the above equation, the relative error distributions of all test examples were counted, and the results are shown in Figure 11. It can be seen from this figure that the relative error of the average daily production of the reservoir is about 13%. This is mainly because the daily production of some test wells is low, and the denominator is too small.

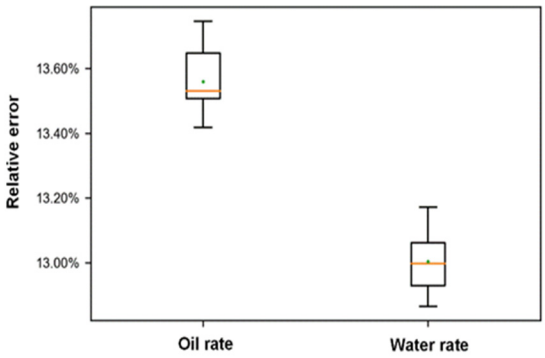


Figure 11. Statistics on relative errors of daily oil and water production of the wells simulated using E2C (the green dots represent the average value, and the orange line represents the median).

Finally, the running time of the numerical simulator was compared with that of E2C. The test cases were simulated using the simulator on an Intel Xeon (R) W-3275M dual CPU (50 cores) node, and each took approximately 4080 s. In comparison, on the RTX1080 GPU node with 11 GB video memory, the E2C model completed the calculation of 30 test cases in 521 s, which was 17.3 s per case on average. Clearly, compared with the traditional simulator, the E2C model is about 200 times faster in computing speed. Although there is a difference between the GPU and CPU, the simulation speed of the simulator did not show any significant improvement using the GPU.

To better demonstrate the effectiveness of the model proposed in this paper, we used the same data to conduct a comparison with the E2C model proposed by Huang H et al. (2023) [29] The results are presented in Appendix A.

4.3. Analysis of Model Optimization Results

The E2C model trained in 3.2 was used to optimize the water injection conditions of the reservoir. In the process of model construction, some historical data were needed. In order to use as much data as possible and illustrate the use of a surrogate model in optimization, only the last month was analyzed in this work. The goal of optimization in this part was to improve the daily oil production of the reservoir in the last month

by adjusting the daily water injection rate of the water injection wells. According to the production history, a total of 97 injection wells were put into operation in the last month, so the number of independent variables to be optimized was $N_{\text{injector}} = 97$. The initial water injection rates of the 97 injection wells are shown in Figure 12. In this study, the liquid production rate of each production well was fixed at the initial rate, the daily water injection volume of the reservoir was 2136.63 m^3, and the daily injection volume of each well ranged from 0 to 28 m^3.

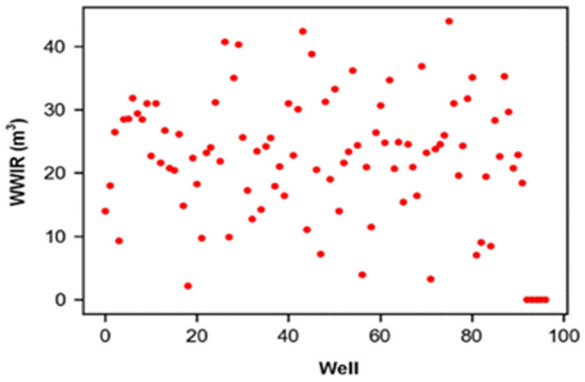


Figure 12. Distribution of initial water injection rates of the injection wells.

The particle swarm optimization (PSO) algorithm is usually used to update independent decision variables and find the optimal solution. The quality of the optimal solution found using PSO usually depends on the number of particles. Making use of the speedup provided by the E2C model, the PSO algorithm used in this study contained 200 particles and 100 iteration steps to find the optimal solution within a wider distribution. The relationship between the daily oil production of the reservoir and the iteration number is shown in Figure 13. It can be seen from the figure that, in the first 30 iterations, the daily oil production of the reservoir increases rapidly and then tends to be stable.

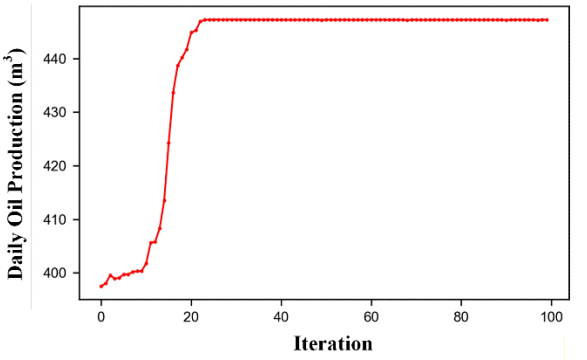


Figure 13. Variation in daily oil production of the reservoir with the number of iterations.

Figure 14 shows the difference between the optimal daily water injection volume and the initial daily water injection volume, where the black dotted line is the 45° line, representing that the results are completely consistent. It can be seen from the figure that the optimal daily water injection volume of each well deviates from the 45° line, and in some wells the water injection volume can reach the set maximum boundary. Therefore, there is a big difference between the optimal daily water injection volume and the initial daily water injection volume, which also sets higher requirements on the generalization ability of the surrogate model.

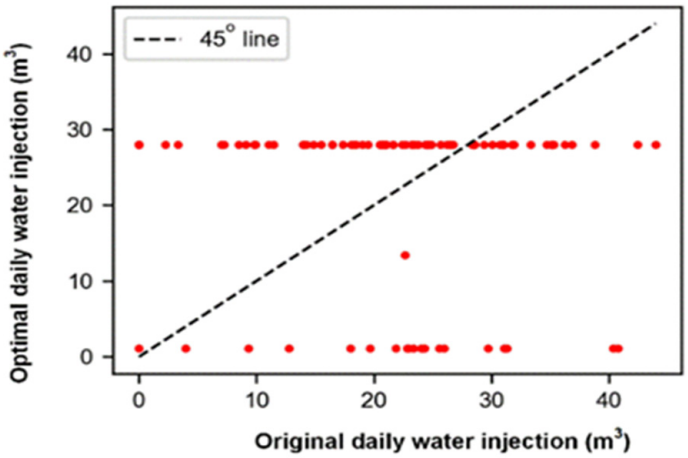


Figure 14. Distribution of initial daily water injection volume and optimal daily water injection volume (The horizontal axis represents the original daily injection volume, while the vertical axis represents the optimized daily injection volume. The red dots indicate the daily water injection volume during optimization when the original daily water injection volume is a certain value).

Figure 15 shows the change in daily oil production of the reservoir before and after the optimization. It can be seen from the figure that the daily oil production of the reservoir at the last time step is 393.09 m^3 under the initial daily water injection volume, while the optimal daily oil production of the reservoir obtained using E2C + PSO is 477.78 m^3. Through optimization, the daily oil production of the reservoir increases by 13.84%. In addition, due to the high calculation efficiency of the E2C model, it only took about 5 h to complete 100 iterations of 200 particles. Apparently, the model proposed in this study can quickly complete the injection–production optimization analysis of the reservoir.

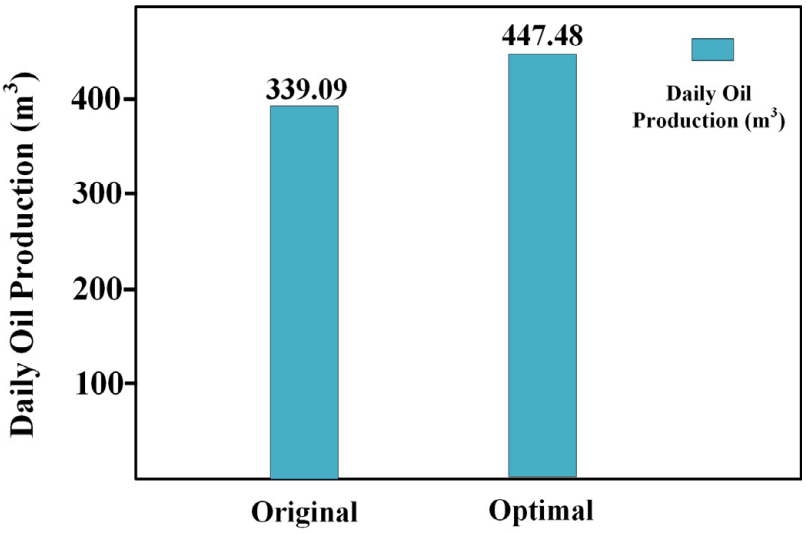


Figure 15. Comparison of oil production of the reservoir before and after optimization.

In order to better illustrate the reliability of the above optimization results, the optimized daily water injection volume was input into the simulator, the daily oil production of all wells at the last time step was simulated, and the results are shown in Figure 16. It can be seen from the figure that the daily oil production of most wells is near the 45° line, so the above optimization results are credible.

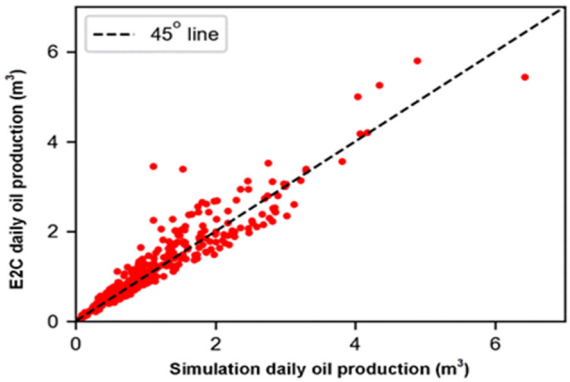


Figure 16. Comparison of daily oil production before and after the optimization (The horizontal axis represents the original oil production information, while the vertical axis represents the optimized oil production information. The red dots indicate the optimized oil production when the original oil production is at a certain value).

This study serves as a proof-of-concept, aimed at demonstrating the feasibility and scalability of the E2C surrogate model under high-dimensional controls and complex, coupled physics.

In practice, optimization is a multi-objective, multi-constraint problem that must simultaneously balance cumulative production, economic performance, energy consumption, and operating costs; incorporate operational windows, facility and pipeline capacity, progressive well pattern expansion, injection–production coordination, the duration of stable production periods, and shut-in/restart scheduling; and account for risks arising from geological uncertainty and data noise. These factors substantially increase the problem dimensionality and algorithmic complexity, which lie beyond the scope of this paper and constitute a distinct research area.

To address these challenges, our future work will prioritize (1) developing a graph neural network (GNN)-based surrogate model that accommodates dynamic well locations, enabling the reuse and transfer of learned representations under varying well counts and evolving well patterns; (2) integrating uncertainty quantification and robust/stochastic optimization to enhance stability and reliability under field conditions; (3) establishing a multi-stage rolling optimization framework and coupling it with surface facility models for long-horizon validation; and (4) conducting systematic evaluation on synthetic and field datasets, reporting standardized error metrics and ablation studies. These directions will be central to our subsequent research, enabling an assessment of the method's practicality under conditions that more closely reflect engineering practice.

5. Conclusions

Based on the E2C structure, the development process of a typical low-permeability water drive reservoir block in Northwest China was simulated in this study. Through the study, it is found that the E2C model proposed can effectively simulate the changes in pressure, saturation, daily oil production, and daily water production with time in the process of oilfield development. The numerical analysis shows that the maximum relative errors of pressure and saturation from E2C are less than 2%. In addition, the trained model was used to optimize the daily water injection volume of the last time step of the water injection wells; as a result, the daily oil production of the reservoir increased by 13.84%. In terms of the simulation time, based on the same hardware facilities, the model proposed in this study is about 200 times faster than the conventional numerical simulator. Therefore, it can be seen from the study that the surrogate model based on deep learning can optimize the reservoir quickly and accurately, thus improving the productivity

and economic profits of the reservoir. This work exemplifies the surrogate model's high practical value in Industry 5.0, providing a validated pathway for AI-driven intelligent decision-making and efficiency gains within industrial production systems.

The future research will cover the following aspects: at present, the time interval of the E2C model for prediction and training must be the same; how to break this limitation and improve the flexibility of the model is worth researching. In addition, after the current E2C model is trained, its well locations are fixed, which limits the application of E2C. Therefore, how to optimize the structure of the model to make it applicable to more scenarios is also of great significance.

Author Contributions: Conceptualization, H.H., B.G., Z.L. and J.Y.; Software, Z.L.; Validation, H.H., Z.L. and J.Y.; Formal analysis, H.H. and J.Y.; Investigation, H.H., B.G. and J.Y.; Resources, H.H., Z.L. and J.Y.; Data curation, H.H. and B.G.; Writing—original draft, H.H.; Writing—review and editing, H.H. and J.Y.; Visualization, H.H., B.G. and J.Y.; Supervision, B.G. and Z.L.; Project administration, B.G. and Z.L.; Funding acquisition, H.H. and B.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China (Grant No. 42502125).

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Acknowledgments: We would like to thank TRACY ENERGY TECHNOLOGIES for providing the computational and data resources used in this study.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. The Comparison with the E2C Model Described by Huang H. (2023) [29]

To more intuitively and reproducibly demonstrate the effectiveness of the proposed optimization, we conducted a stringently controlled benchmarking experiment. Using the training dataset and loss function developed in this work as a unified benchmark, we faithfully reimplemented the Huang H. (2023) [29] model and evaluated its performance on an identical test set under the same data partitioning, preprocessing, and evaluation pipeline. To minimize confounding factors, we harmonized external training hyperparameters across models—including optimizer choice, learning rate scheduling, batch size, and training epochs/early stopping—while retaining each model's canonical architecture and intrinsic optimization design. The dataset comprised 300 well control samples, each consisting of 172 time steps, and the train/test split matched that employed in the main text.

Using Equations (19)–(21), we statistically characterized the distribution of relative errors; the results are reported in Table A1. The benchmark model Huang H. (2023) [29] yields mean relative errors of 21.3% for oil production and 28.1% for water production. With the proposed optimization, these averages decrease to 13.5% (oil) and 13.1% (water), corresponding to reductions of approximately 36.6% and 53.4%, respectively. These findings underscore that the improvements conferred by the optimization are robust and broadly generalizable.

Table A1. Relativate error comparison with different model.

	Average Errors for Oil	Average Errors for Water
E2C described by Huang H. (2023) [29]	13.5%	13.1%
E2C this paper	21.3%	28.1%

References

1. Van Erp, T.; Carvalho, N.G.P.; Gerolamo, M.C.; Gonçalves, R.; Rytter, N.G.M.; Gladysz, B. Industry 5.0: A new strategy framework for sustainability management and beyond. *J. Clean. Prod.* **2024**, *461*, 142271. [CrossRef]
2. Zhang, G.Y.; Lin, C.Y.; Wang, Z.Z.; Ren, L.H.; Zhang, X.G.; Qu, K.; Zhang, X.B. Hybrid knowledge-driven and data-driven intelligent reservoir characterization and its research progress. *Prog. Geophys.* **2024**, *39*, 119–140. [CrossRef]
3. Zhao, B.; Yao, Y.; Xiao, Z.; Wei, Y.; Liu, H.; Huang, J.; Tian, Y.-P.; Wang, J. Identification Method of Remaining Oil Potential Area Based on Deep Learning. *J. Energy Resour. Technol. Part B Subsurf. Energy Carbon Capture* **2025**, *1*, 031001. [CrossRef]
4. Zhang, X.D.; Chen, Y.T.; Meng, P. Synthetic well logs generation via Recurrent Neural Networks. *Pet. Explor. Dev.* **2018**, *45*, 629–639. [CrossRef]
5. Zhu, Y.; Zabaras, N. Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *J. Comput. Phys.* **2018**, *366*, 415–447. [CrossRef]
6. Mo, S.; Zhu, Y.; Zabaras, N.; Shi, X.; Wu, J. Deep convolutional encoder-decoder networks for uncertainty quantification of dynamic multiphase flow in heterogeneous media. *Water Resour. Res.* **2019**, *55*, 703–728. [CrossRef]
7. Liu, Y.; Sun, W.; Durlofsky, L.J. A deep-learning-based geological parameterization for history matching complex models. *Math. Geosci.* **2019**, *51*, 725–766. [CrossRef]
8. Liu, Y.; Durlofsky, L.J. 3D CNN-PCA: A deep-learning-based parameterization for complex geomodels. *Comput. Geosci.* **2021**, *148*, 104676. [CrossRef]
9. Zhu, Y.; Zabaras, N.; Koutsourelakis, P.-S.; Perdikaris, P. Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. *J. Comput. Phys.* **2019**, *394*, 56–81. [CrossRef]
10. Jia, D.; Liu, H.; Zhang, J.; Gong, B.; Pei, X.; Wang, Q.; Yang, Q. Data-driven optimization for fine water injection in a mature oil field. *Pet. Explor. Dev.* **2020**, *47*, 629–636. (In Chinese) [CrossRef]
11. Shahkarami, A.; Mohaghegh, S. Applications of smart proxies for subsurface modeling. *Pet. Explor. Dev.* **2020**, *47*, 372–382. (In Chinese) [CrossRef]
12. Kim, J.; Yang, H.; Choe, J. Robust optimization of the locations and types of multiple wells using CNN based proxy models. *J. Pet. Sci. Eng.* **2020**, *193*, 107424. [CrossRef]
13. Tang, M.; Liu, Y.; Durlofsky, L.J. A deep-learning-based surrogate model for data assimilation in dynamic subsurface flow problems. *J. Comput. Phys.* **2020**, *413*, 109456. [CrossRef]
14. Wang, H.L.; Mu, L.X.; Shi, F.G.; Gong, H.E. Production prediction at ultra-high water cut stage via Recurrent Neural Network. *Pet. Explor. Dev.* **2020**, *47*, 1009–1015. (In Chinese) [CrossRef]
15. Zhong, Z.; Sun, A.Y.; Wang, Y.; Ren, B. Predicting field production rates for waterflooding using a machine learning-based proxy model. *J. Pet. Sci. Eng.* **2020**, *194*, 107574. [CrossRef]
16. Gao, Z.K.; Liu, M.-X.; Dang, W.-D.; Cai, Q. A novel complex network-based deep learning method for characterizing gas–liquid two-phase flow. *Pet. Sci.* **2021**, *18*, 259–268. (In Chinese) [CrossRef]
17. Kuang, L.; Liu, H.; Ren, Y.; Luo, K.; Shi, M.; Su, J.; Li, X. Application and development trend of artificial intelligence in petroleum exploration and development. *Pet. Explor. Dev.* **2021**, *48*, 1–14. (In Chinese) [CrossRef]
18. Ma, C.J. Reservoir production prediction method based on recurrent neural network. *Inn. Mong. Petrochem. Ind.* **2021**, *47*, 108–112. (In Chinese)
19. Chaki, S.; Zagayevskiy, Y.; Wong, T. A Proxy Flow Modelling Workflow to Estimate Gridded Dynamic Properties and Well Production Rates by Deep Learning Algorithms. In Proceedings of the SPE/IATMI Asia Pacific Oil & Gas Conference and Exhibition, Online, 12–14 October 2021. [CrossRef]
20. Cornelio, J.; Razak, S.M.; Jahandideh, A.; Jafarpour, B.; Cho, Y.; Liu, H.-H.; Vaidya, R. Investigating Transfer Learning for Characterization and Performance Prediction in Unconventional Reservoirs. In Proceedings of the SPE Middle East Oil & Gas Show and Conference, Manama, Bahrain, 28 November–1 December 2021. [CrossRef]
21. Zhong, Z.; Sun, A.Y.; Ren, B.; Wang, Y. A deep-learning-based approach for reservoir production forecast under uncertainty. *SPE J.* **2021**, *26*, 1314–1340. [CrossRef]
22. Alakeely, A.; Horne, R. Simulating Oil and Water Production in Reservoirs with Generative Deep Learning. *SPE Reserv. Eval. Eng.* **2022**, *25*, 751–773. [CrossRef]
23. Wang, X.; Yang, Y.Z.; He, Y.F.; Wang, Z.; Gong, X.J. Research and application of intelligent diagnosis technology of oil well working conditions based on deep learning. *Pet. Geol. Recovery Effic.* **2022**, *29*, 181–189. (In Chinese) [CrossRef]
24. Yang, Y. Application progress of exploration and development big data and artificial intelligence technology in Shengli Oilfiel. *Pet. Geol. Recovery Effic.* **2022**, *29*, 1–10. (In Chinese) [CrossRef]
25. Huang, H.; Gong, B.; Liu, Y.; Sun, W. Deep-learning based proxy modeling for fast and accurate simulation in realistic 3D reservoir with varying well controls. *Geoenergy Sci. Eng.* **2023**, *222*, 211418. [CrossRef]
26. Watter, M.; Springenberg, J.; Boedecker, J.; Riedmiller, M. Embed to control: A locally linear latent dynamics model for control from raw images. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1–9.

27. Jin, Z.L.; Liu, Y.; Durlofsky, L.J. Deep-learning-based surrogate model for reservoir simulation with time-varying well controls. *J. Pet. Sci. Eng.* **2020**, *192*, 107273. [CrossRef]
28. Qin, F.; Yan, Z.H.; Tang, S.L.; Luo, R.Q.; Gong, B. Application of deep learning-based surrogate model in 3D simulation of actual gas reservoir. *Pet. Geol. Recovery Effic.* **2022**, *29*, 152–159. (In Chinese) [CrossRef]
29. Huang, H.; Gong, B.; Sun, W. A Deep-Learning-Based Graph Neural Network-Long-Short-Term Memory Model for Reservoir Simulation and Optimization With Varying Well Controls. *SPE J.* **2023**, *28*, 2898–2916. [CrossRef]
30. He, J.; Durlofsky, L.J. Reduced-Order Modeling for Compositional Simulation by Use of Trajectory Piecewise Linearization. *SPE J.* **2014**, *19*, 858–872. [CrossRef]
31. He, J.; Durlofsky, L.J. Constraint reduction procedures for reduced-order subsurface flow models based on POD–TPWL. *Int. J. Numer. Methods Eng.* **2015**, *103*, 1–30. [CrossRef]
32. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21 July–26 July 2017; pp. 4700–4708. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.