Petroleum Science 22 (2025) 3745–3759

Contents lists available at ScienceDirect

Petroleum Science

journal homepage: www.keaipublishing.com/en/journals/petroleum-science

Original Paper

Integrated optimization of reservoir production and layer configurations using relational and regression machine learning models

Qin-Yang Dai [a,b], Li-Ming Zhang [a,b,*], Kai Zhang [a,b,c], Hao Hao [d], Guo-Dong Chen [e], Xia Yan [a,b], Pi-Yang Liu [c], Bao-Bin Zhang [a,b], Chen-Yang Wang [a,b]

[a] State Key Laboratory of Deep Oil and Gas, China University of Petroleum (East China), Qingdao, 266580, Shandong, China
[b] School of Petroleum Engineering, China University of Petroleum (East China), Qingdao, 266580, Shandong, China
[c] School of Civil Engineering, Qingdao University of Technology, Qingdao, 266520, Shandong, China
[d] Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai, 200240, China
[e] Department of Earth Sciences, The University of Hong Kong, 999077, Hong Kong, China

ARTICLE INFO

Article history:
Received 10 September 2024
Received in revised form
6 June 2025
Accepted 7 June 2025
Available online 31 July 2025

Edited by Yan-Hua Sun

Keywords:
Surrogate model
Reservoir management
Evolutionary algorithm
Joint optimization
Layer configuration
Production optimization
Relational learning

ABSTRACT

This study introduces a novel approach to addressing the challenges of high-dimensional variables and strong nonlinearity in reservoir production and layer configuration optimization. For the first time, relational machine learning models are applied in reservoir development optimization. Traditional regression-based models often struggle in complex scenarios, but the proposed relational and regression-based composite differential evolution (RRCODE) method combines a Gaussian naive Bayes relational model with a radial basis function network regression model. This integration effectively captures complex relationships in the optimization process, improving both accuracy and convergence speed. Experimental tests on a multi-layer multi-channel reservoir model, the Egg reservoir model, and a real-field reservoir model (the S reservoir) demonstrate that RRCODE significantly reduces water injection and production volumes while increasing economic returns and cumulative oil recovery. Moreover, the surrogate models employed in RRCODE exhibit lightweight characteristics with low computational overhead. These results highlight RRCODE's superior performance in the integrated optimization of reservoir production and layer configurations, offering more efficient and economically viable solutions for oilfield development.
© 2025 The Authors. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co. Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

The production strategies and layer control schemes of injection and production wells are critical factors influencing the displacement efficiency and overall economic performance of reservoir development. By optimizing these schemes, it is possible to effectively control the underground flow field, thereby maximizing cumulative oil production or net present value (NPV) throughout the development process (Desbordes et al., 2022; Du et al., 2023; Kim and Durlofsky, 2021; Wang Z.Z. et al., 2022, 2023; Xu et al., 2023). In recent years, advanced optimization methods have gained considerable attention in reservoir development scheme design, promoting intelligent decision-making and enhancing economic performance (An et al., 2022; Kim and Durlofsky, 2023; Volkov and Bellout, 2017; Zhang et al., 2021). Traditional intelligent optimization methods typically rely on coupling reservoir numerical simulators with evolutionary algorithms, such as genetic algorithms (Emerick et al., 2009), differential evolution (Nwankwor et al., 2013), and particle swarm optimization (Onwunalu and Durlofsky, 2010). While these methods have achieved some success, they require extensive reservoir simulations for evaluation, resulting in high computational costs. This challenge has driven researchers to explore surrogate models that approximate reservoir responses (Dai et al., 2023; Golzari et al., 2015; Liu and Reynolds, 2021; Ma et al., 2021, 2022; Wang et al., 2024; Wang L. et al., 2023), giving rise to surrogate-assisted evolutionary algorithms (SAEAs), which have

* Corresponding author.
E-mail address: zhangliming@upc.edu.cn (L.-M. Zhang).
Peer review under the responsibility of China University of Petroleum (Beijing).

https://doi.org/10.1016/j.petsci.2025.06.001
1995-8226/© 2025 The Authors. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co. Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

been systematically reviewed and demonstrated across multiple domains (Jin et al., 2019).

In the field of reservoir development optimization, traditional machine learning surrogate models typically employ methods such as kriging models (Chen et al., 2020a), radial basis function networks (Chen et al., 2022), support vector machines (Guo and Reynolds, 2018), among others, to construct regression models. These models use individual reservoir development schemes as input parameters to predict specific outcomes, such as cumulative oil production or net present value, and are effectively combined with evolutionary algorithms to reduce reliance on direct numerical simulation evaluations during the optimization process (Gu et al., 2021; Zhao et al., 2020a, 2020b). However, the inherent nonlinearity and high dimensionality of reservoir development optimization problems often limit the performance of these regression models. Although regression models can provide reasonable accuracy and precision in simpler scenarios, their performance tends to degrade in more complex optimization tasks. Moreover, since evolutionary algorithms like differential evolution and particle swarm optimization rely on evaluating and selecting candidate solutions, a decline in the accuracy of regression model evaluations can adversely impact the overall optimization results.

Compared to regression models, relational models (Hao and Zhou, 2023; Hao et al., 2020, 2022) focus on the relative performance between different samples rather than predicting their specific objective function values. This approach is highly compatible with the nature of evolutionary algorithms, which fundamentally operate by comparing and selecting potential candidate solutions to determine the optimal outcome, thereby driving the optimization process forward. Relational models offer several advantages, including the ability to generate larger training datasets, reduced reliance on extreme samples, and enhanced model stability and generalization capabilities (Hao et al., 2020). These characteristics make relational machine learning models particularly well-suited for addressing the high-dimensional and complex optimization problems typical in reservoir management.

Although relational models have significant potential as surrogate models for reservoir development optimization, to the best of our knowledge, they have not yet been applied in this field. Current research in this domain still primarily relies on regression-based surrogate models, with a focus on optimizing either production schemes alone or jointly with well placement (Xue et al., 2020, 2022; Zhao et al., 2020c). Layer configuration, which is a critical factor affecting reservoir development performance, has been studied less frequently, and research on the integrated optimization of production and layer schemes is even more scarce. This is partly due to the added complexity that integrated optimization of production and layer configurations entails, and partly because adjustments to production strategies are typically more frequent and straightforward compared to layer configurations. However, the impact of production schemes and layer configurations on reservoir displacement efficiency is inherently interrelated, making it essential and necessary to consider their integrated optimization. The interaction between these factors requires more advanced optimization algorithms capable of handling such complexity.

To address the aforementioned challenges, this paper introduces relational machine learning models into reservoir development optimization and proposes an innovative composite differential evolution optimization method (RRCODE) that combines relational and regression models. By leveraging the comparative nature of relational models and the predictive capabilities of regression models, RRCODE provides more accurate and computationally efficient solutions for the integrated optimization of reservoir production and layer configuration. To the best of our knowledge, this is the first application of relational models in the field of reservoir development optimization. The remainder of this paper is organized as follows: Section 2 presents the problem of integrated optimization of production schemes and layer configurations. Section 3 details the proposed RRCODE method and related works. Section 4 provides experimental results and discussion on two multi-layer reservoir models and a real reservoir model, and Section 5 concludes with a summary and discussion.

2. Problem statement

As two critical factors determining the effectiveness of reservoir development, the production strategy directly affects the lateral displacement efficiency, while the layer configuration schemes of each well control the vertical distribution of the flow field (Chang et al., 2020; Dong et al., 2023). These two factors interact during the development process, collectively influencing the dynamic changes in the subsurface flow field and the overall economic returns. Coupling the optimization of production schemes with layer configuration allows for a more comprehensive regulation of the reservoir's flow field, thereby maximizing the overall profitability of the development plan. Compared to optimizing production or layer schemes independently, this integrated optimization scheme can more effectively guide key decision-making in oilfield development.

2.1. Integrated optimization problem and variable design

The integrated optimization of production and layer configurations involves designing production schemes for m wells (each well has n layers with opening and closing options) across T timesteps. Additionally, before production begins at the first timestep, the status (open or closed) of each layer for every well should be determined. In this problem model, each potential integrated optimization scheme is represented by a sample $\boldsymbol{x}$, with the dimensionality of the sample variables denoted as d, which can be calculated as follows:

$$d = m \times (n + T) \tag{1}$$

The first $m \times T$ dimensions of the sample $\boldsymbol{x}$ represent the opening and closing status of the m injection and production wells over T timesteps, where 1 indicates the layer is open, and 0 indicates the layer is closed. The remaining $m \times T$ dimensions represent the production scheme for the m wells across T timesteps.

The sample $\boldsymbol{x}$ is drawn from the decision space of the problem model. The decision range for the optimization variables, [$\boldsymbol{lb}$, $\boldsymbol{ub}$], is determined by the practical constraints of the oilfield, including the lower and upper injection or production limits of each well and the number of layer configurations to be considered. The entire integrated optimization problem can be formulated as the task of finding the optimal sample $\boldsymbol{x}$ that maximizes the objective function within the decision space:

$$\text{Maximize} F = f(\boldsymbol{x}), \quad \boldsymbol{x} \in [\boldsymbol{lb}, \boldsymbol{ub}] \tag{2}$$

where F represents the optimization objective, while $\boldsymbol{lb}$ and $\boldsymbol{ub}$ are the lower and upper bounds of the sample $\boldsymbol{x}$, with each dimension corresponding to those in $\boldsymbol{x}$.

2.2. Optimization objective function

Net present value (NPV) is a key decision-making factor in the design of oilfield development schemes and is widely used in reservoir development optimization problems (Yan et al., 2024; Zhang et al., 2024). Therefore, in this study, NPV is chosen as the objective function F for the integrated optimization of production and layer configurations, and its calculation formula is as follows:

$$f(\boldsymbol{x}) = \mathrm{NPV}(\boldsymbol{x}) = \sum_{t=1}^{T} \Delta t \frac{1}{(1+b)^p}\left(r_\mathrm{o}Q_{\mathrm{o},t} - r_\mathrm{w}Q_{\mathrm{w},t} - r_\mathrm{i}Q_{\mathrm{i},t}\right) \tag{3}$$

where Δt represents the length of the t-th timestep; p is the cumulative timestep; b is the annual discount rate; and r_o, r_w, and r_i represent the oil price, water production cost, and water injection cost, respectively; $Q_{\mathrm{o},t}$, $Q_{\mathrm{w},t}$, and $Q_{\mathrm{i},t}$ denote the oil production rate, water production rate, and water injection rate during the t-th timestep, respectively.

3. Methodology

The essence of evolutionary computation lies in generating a certain number of candidate solutions within the decision space through a series of operations, then evaluating the relative merits of these candidates and selecting the better ones for the next iteration. This process drives the optimization towards a global optimum. Conventional machine learning-assisted evolutionary optimization methods primarily rely on regression-based machine learning models, which predict the objective function values of candidate solutions (i.e., the integrated production and layer configuration schemes) to replace the reservoir numerical simulator. This allows comparison and selection between samples, accelerating the convergence of the optimization process. However, the accuracy of regression models is crucial—any inaccuracies in the model's predictions can significantly impair the overall optimization performance. In contrast, relational machine learning models do not predict specific objective function values for the candidates but instead focus on evaluating the relative superiority between samples. This approach aligns more closely with the fundamental nature of evolutionary algorithms, which are inherently based on comparison and selection. By introducing relational models as a complement to regression models, the dependency on the latter can be reduced, thereby mitigating the uncertainties in the optimization process and improving the robustness and accuracy of the search.

3.1. Radial basis function network model

The radial basis function network (RBFN) is a regression model widely used for high-dimensional and nonlinear problems (Broomhead and Lowe, 1988; Park and Sandberg, 1993). The fundamental idea behind the RBFN is to map input samples into a feature space, where the samples are combined in a weighted linear manner using radial basis functions to approximate the target function values. For an input sample $\boldsymbol{x}$, its predicted value $\hat{f}(\boldsymbol{x})$ can be expressed as follows:

$$\hat{f}(\boldsymbol{x}) = \sum\nolimits_{i=1}^{N} w_i \phi(\|\boldsymbol{x} - c_i\|) \tag{4}$$

where w_i represents the weight parameters; c_i represents the centers of the radial basis functions; $\phi(\boldsymbol{x})$ is the radial basis function, which is typically expressed as a Gaussian function, with the following formulation:

$$\phi(r) = \exp\left(-\frac{r^2}{2\sigma^2}\right) \tag{5}$$

where σ represents the width parameter. Due to its strong approximation capabilities and relatively simple structure, the radial basis function network (RBFN) is widely used as a regression-based machine learning model in reservoir development optimization algorithms.

3.2. Gaussian naive Bayes model

Gaussian naive Bayes (GNB) is a simple, efficient, and widely used classification algorithm with low computational complexity (Zhang, 2004). It assumes that each feature follows a Gaussian distribution and uses this assumption to calculate the conditional probability of the sample's features for different categories. For each class C_k and feature x^j in the j-th dimension of the sample, the conditional probability density function is expressed as

$$P\left(x^j|C_k\right) = \frac{1}{\sqrt{2\pi\sigma_{C_k}^2}} \exp\left(-\frac{\left(x^j - \mu_{C_k}\right)^2}{2\sigma_{C_k}^2}\right) \tag{6}$$

where μ_{C_k} and σ_{C_k} represent the mean and standard deviation of the samples in class C_k for the j-th dimension, respectively; $P(\boldsymbol{x}|C_k)$ is the likelihood function, which is the product of the conditional probabilities of the sample $\boldsymbol{x}$ across all dimensions, and can be expressed as

$$P(\boldsymbol{x}|C_k) = \prod\nolimits_{j=1}^{d} P\left(x^j|C_k\right) \tag{7}$$

According to Bayes' theorem, the posterior probability $P(C_k|\boldsymbol{x})$ can be calculated using the prior probability $P(C_k)$ and the likelihood function $P(\boldsymbol{x}|C_k)$ as follows:

$$P(C_k|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|C_k)\cdot P(C_k)}{P(\boldsymbol{x})} \tag{8}$$

Thus, the class with the highest posterior probability can be selected as the predicted class C_pre for the sample $\boldsymbol{x}$, as follows:

$$C_\mathrm{pre} = \underset{k}{\mathrm{argmax}} P(C_k|\boldsymbol{x}) \tag{9}$$

In this study, the Gaussian naive Bayes model is used as part of the relational surrogate model. It captures key relational features in the optimization of integrated production and layer configuration schemes by learning the relative superiority between sample pairs. This approach allows the model to better adapt to the rapid comparison of sample quality, which is crucial in evolutionary algorithms.

3.3. Relational machine learning surrogate model

The relational machine learning surrogate model works by forming a relational pair from two candidate solutions as training input and directly predicting the relative superiority between the two solutions (Hao et al., 2020). Unlike traditional regression models, this model does not rely on the absolute values of the samples, but rather learns the comparative results between candidate solutions (i.e., determining whether one candidate solution is superior to the other). The key advantage of this approach is that it better aligns with the essential needs of evolutionary algorithms, which rely on comparisons to select superior individuals for the next generation.

For the integrated production and layer configuration scheme of reservoir development, a relational pair $[\boldsymbol{x}_\mathrm{p}, \boldsymbol{x}_\mathrm{q}]$ must first be constructed as the input sample for the relational model. Here, $\boldsymbol{x}_\mathrm{p}$ and $\boldsymbol{x}_\mathrm{q}$ are development schemes that have been evaluated by the reservoir numerical simulator, i.e., $\boldsymbol{x}_\mathrm{p}$, $\boldsymbol{x}_\mathrm{q} \in \boldsymbol{D}$, where $\boldsymbol{D}$ represents the dataset of samples with actual evaluations. By comparing their net present values $f(\boldsymbol{x}_\mathrm{p})$ and $f(\boldsymbol{x}_\mathrm{q})$, the label l for the relational pair $[\boldsymbol{x}_\mathrm{p}, \boldsymbol{x}_\mathrm{q}]$ can be obtained and is defined as follow:

$$l = \begin{cases} +1, & \text{if } f(\boldsymbol{x}_\mathrm{p}) \geq f(\boldsymbol{x}_\mathrm{q}) \\ -1, & \text{otherwise} \end{cases}, \quad \boldsymbol{x}_\mathrm{p}, \boldsymbol{x}_\mathrm{q} \in \boldsymbol{D} \tag{10}$$

If the net present value of the development scheme represented by $\boldsymbol{x}_\mathrm{p}$ is higher than that of $\boldsymbol{x}_\mathrm{q}$, the label l for the relational pair $[\boldsymbol{x}_\mathrm{p}, \boldsymbol{x}_\mathrm{q}]$ is defined as $+1$. Otherwise, the label l is defined as -1. Using the relational pairs to form a training dataset, a Gaussian naive Bayes model is trained to predict whether the label for each new relational pair is $+1$ or -1, thereby enabling the model to assess the relative superiority of any two schemes.

Compared to traditional regression models, relational surrogate models build a much larger training set by using relational pairs (if n_r samples are selected, the number of training samples becomes $n_\mathrm{r} \times (n_\mathrm{r} - 1)$. This significantly increases the number of samples available for learning. Expanding the training set in this way enhances the generalization ability of the relational model and makes it more robust when handling complex, high-dimensional optimization problems.

3.4. Composite differential evolution algorithm

The composite differential evolution (CoDE) algorithm (Wang et al., 2011) is a population-based global optimization method that has demonstrated significant performance across various fields. By combining different mutation strategies and control parameter settings, CoDE enhances the diversity and robustness of traditional evolutionary algorithms, thereby improving their performance in solving complex optimization problems.

3.4.1. Construction of strategy and parameter candidate pools

The strategy candidate pool includes three strategies: rand/1/bin, rand/2/bin, and current-to-best/1 (Das and Suganthan, 2011). The rand/1/bin strategy is suitable for most optimization problems and provides a good balance between exploration and exploitation; the rand/2/bin strategy enhances the intensity of mutation operations, which helps in escaping local optima and improving global search ability; the current-to-best strategy exploits information from the current best individual, accelerating convergence during the later stages of evolution.

rand/1/bin:

$$\boldsymbol{v}_{i,1} = \boldsymbol{x}_{r_1} + F\cdot(\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3}) \tag{11}$$

rand/2/bin:

$$\boldsymbol{v}_{i,2} = \boldsymbol{x}_{r_1} + F\cdot(\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3}) + F\cdot(\boldsymbol{x}_{r_4} - \boldsymbol{x}_{r_5}) \tag{12}$$

current-to-best/1:

$$\boldsymbol{v}_{i,3} = \boldsymbol{x}_i + F\cdot(\boldsymbol{x}_\mathrm{best} - \boldsymbol{x}_i) + F\cdot(\boldsymbol{x}_{r_1} - \boldsymbol{x}_{r_2}) \tag{13}$$

where $\boldsymbol{x}_i \in \boldsymbol{D}$, $\boldsymbol{x}_i$ is the i-th individual in the current parent population; r_1, r_2, r_3, r_4, $r_5 \in [1, N]$ are randomly selected sample indices from 1 to N, with N representing the total number of parent samples; $\boldsymbol{x}_\mathrm{best}$ is the current best-performing individual; $\boldsymbol{v}_{i,1}$, $\boldsymbol{v}_{i,2}$, $\boldsymbol{v}_{i,3}$ are the mutated individuals corresponding to $\boldsymbol{x}_i$, generated by different mutation strategies; and F is the scaling factor.

For different mutation strategies and crossover operations, the scaling factor F and the crossover control factor C_r are randomly selected during each generation's mutation process from the parameter candidate pools, which are composed of the following three parameter ranges: (0.1, 1.0), (0.9, 1.0), and (0.2, 0.8).

These combinations are adopted from the reference (Wang et al., 2011), which validated their effectiveness across a wide range of test functions. Specifically, the pair (1.0, 0.1) provides strong mutation with minimal crossover, encouraging solution diversity; the pair (1.0, 0.9) promotes active parameter mixing through crossover, facilitating rapid exploration in the global search space; the pair (0.8, 0.2) favors stability in exploitation while maintaining moderate mutation strength. This combination of strategies and parameter pools offers good robustness and enables the algorithm to adaptively balance global and local exploration.

3.4.2. Crossover and selection operations

Based on the composite mutation strategy, each individual in the evolutionary process combines different strategies and parameters from the strategy and parameter candidate pools to generate three mutated individuals: $\boldsymbol{v}_{i,1}$, $\boldsymbol{v}_{i,2}$, $\boldsymbol{v}_{i,3}$. These mutated individuals then undergo crossover operations with the current individual across different dimensions, resulting in three trial individuals: $\boldsymbol{u}_{i,1}$, $\boldsymbol{u}_{i,2}$, $\boldsymbol{u}_{i,3}$. The trial individual with the highest objective function value is selected as the offspring $\boldsymbol{x}_i'$ of the current individual. In this study, a Gaussian naive Bayes model based on relational comparisons is used to evaluate the relative superiority between the trial individuals, selecting the best $\boldsymbol{x}_i'$ among the remaining candidates. Eqs. (14) and (15) represent the crossover and selection operations, respectively.

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{if } \mathrm{rand} > C_\mathrm{r} \text{ or } j = j_\mathrm{rand} \\ x_{i,j}, & \text{otherwise} \end{cases} \tag{14}$$

$$\boldsymbol{x}_i' = \max(\boldsymbol{u}_{i,1}, \boldsymbol{u}_{i,2}, \boldsymbol{u}_{i,3}) \tag{15}$$

In the equation, $u_{i,j}$ represents the j-th dimension of the trial individual $\boldsymbol{u}_i$ after the mutation and crossover operations; rand is a randomly generated number from the interval [0, 1), and j_rand is a randomly chosen index that ensures at least one dimension of the trial individual $\boldsymbol{u}_i$ participates in the crossover operation.

3.5. Composite differential evolution optimization framework coupling relational and regression models

In the integrated optimization of production and layer configurations, n_init candidate schemes are first generated in the decision space, which are then evaluated using the reservoir numerical simulator, with the results stored in the dataset $\boldsymbol{D} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_{n_\mathrm{init}}]$. Next, the top n_r samples are selected from dataset $\boldsymbol{D}$ and paired to form a total of $n_\mathrm{r} \times (n_\mathrm{r} - 1)$ relational sample pairs, which are used as the training set for the relational surrogate model based on Gaussian naive Bayes.

The top n_p samples from dataset $\boldsymbol{D}$ are selected as the parent population for the composite differential evolution. Mutation strategies and evolutionary hyperparameters from the strategy and parameter candidate pools are combined to generate three trial individuals $\boldsymbol{u}_{i,1}$, $\boldsymbol{u}_{i,2}$, $\boldsymbol{u}_{i,3}$ for each parent sample $\boldsymbol{x}_i$. Using the relational scoring mechanism, the relational surrogate model sequentially compares the relative superiority of the three trial individuals. That is, for each relational pair of trial individuals, the superior individual is scored, and the scores are accumulated during the comparison process. Once the comparisons are complete, the trial individual $\boldsymbol{x}_i'$ with the highest score is selected as the offspring for the corresponding parent.

After obtaining the offspring population through the relational surrogate-assisted differential evolution, a radial basis function network model is trained using n_near samples near the current best

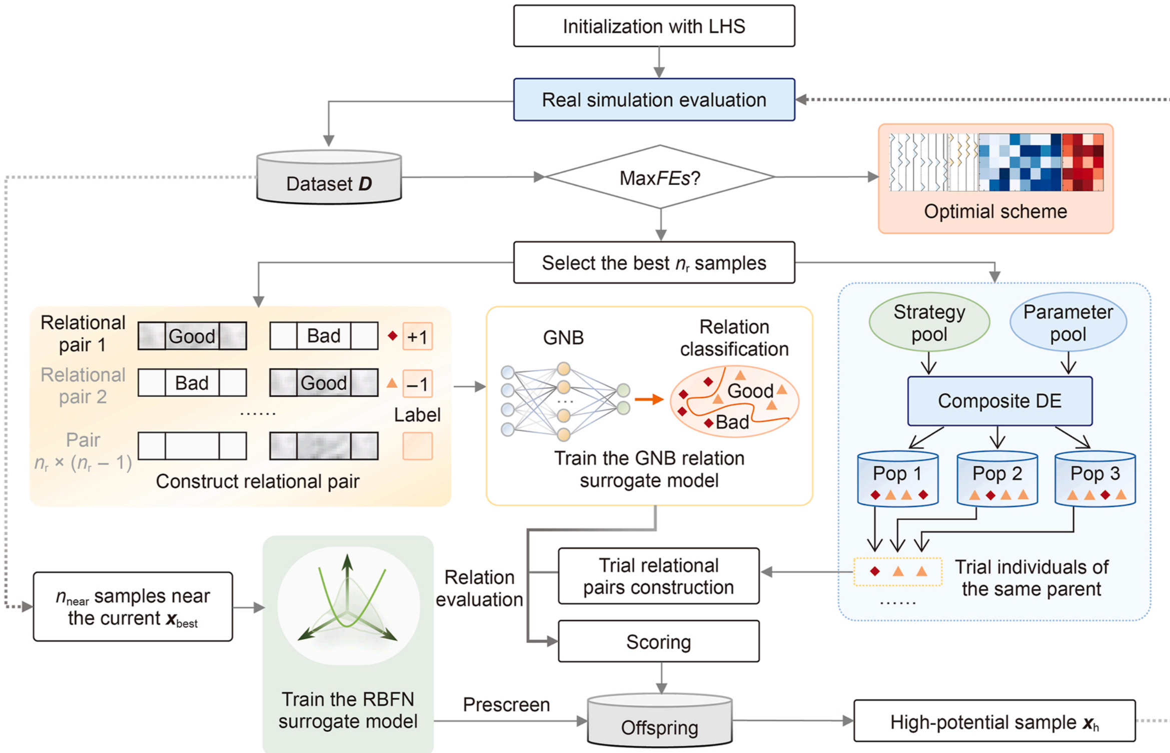

Fig. 1. Workflow of the composite differential evolution optimization algorithm based on relational and regression models (RRCODE).

sample $\boldsymbol{x}_{\text{best}}$. This model is then used to prescreen the offspring individuals, identifying high-potential sample $\boldsymbol{x}_{\text{h}}$, which are subsequently evaluated using the reservoir numerical simulator and updated in dataset $\boldsymbol{D}$. The RBFN surrogate model is retrained at each iteration based on the current best solutions and their neighboring samples, ensuring up-to-date predictive accuracy and effective local exploration. Finally, the optimization process continues until the maximum number Max*FEs* of evaluations is reached, at which point the optimal solution is output. The workflow is illustrated in Fig. 1.

4. Experimental results and discussion

To validate the effectiveness of the proposed relational and regression-based composite differential evolution optimization framework (RRCODE), it was compared with the global and local surrogate-assisted model optimization algorithm (GLSADE) (Chen et al., 2020b), which has shown outstanding performance in reservoir production optimization, the well-established and widely used surrogate-assisted differential evolution algorithm (SADE), and the classical differential evolution algorithm (DE). Each algorithm was applied to optimize the integrated production and layer configuration schemes on both a designed multi-layer multi-channel reservoir model and the Egg reservoir model. To account for the potential impact of the initial population distribution on the optimization performance, Latin hypercube sampling was used to initialize all algorithms, generating and evaluating n_{init} initial schemes as the shared initial population. To ensure fairness in comparison, all algorithms (RRCODE, GLSADE, SADE, and DE) were allocated the same maximum number Max*FEs* (1000) of high-fidelity real evaluations in each optimization task, thus eliminating any bias caused by varying evaluation budgets. The setup details for the four compared algorithms are summarized in Table 1 for clarity and reproducibility.

4.1. Multi-layer multi-channel reservoir case study

The multi-layer multi-channel model is a reservoir model with regular boundaries and multiple high-permeability channels, as shown in Fig. 2. The model contains 9 production wells and 4 injection wells, with a grid size of 101 × 101 in the horizontal plane and 3 effective layers in the vertical direction. The permeability field for each layer is shown in Fig. 3. Each well requires optimized configuration for the opening and closing of the three layers in the vertical direction, involving 39 decision variables. The production scheme optimization for each well is conducted over 10 timesteps, with each timestep representing 365 days, resulting in 130 decision variables. The injection wells operate under a constant liquid

Table 1
Setup details of the compared algorithms.

Algorithm	RRCODE		GLSADE		SADE	DE
Surrogate model	GNB (relation) + RBFN (regression)		RBFN (global + local)		RBFN	None
Initial population size n_{init}	300		300		300	300
Surrogate training parameters	Top samples of relational model n_r	300	Global	All	All	None
	Neighbor samples for RBFN n_{near}	300	Local	300		
Maximum number of real evaluations	1000		1000		1000	1000

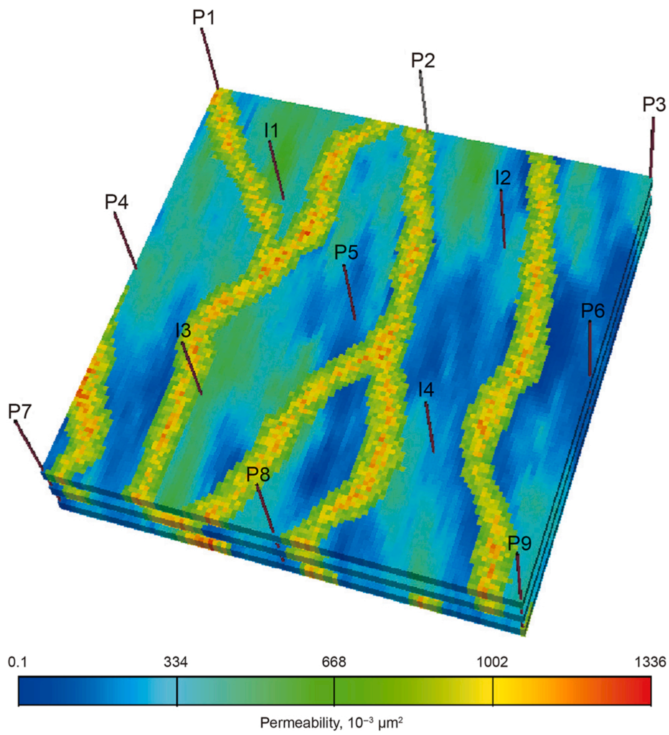

Fig. 2. Multi-layer multi-channel model and well placement distribution.

injection mode, with the injection rate ranging from 500 to 3000 stb/day. The production wells are controlled by a bottom-hole pressure mode, with the bottom-hole pressure limits set between 4800 and 5800 psi. The total number of decision variables for the integrated optimization scheme is 169. In terms of economic parameters, the oil price is set at 80 USD/stb, while the water production and injection costs are set at 3 and 2 USD/stb, respectively. The annual discount rate is set to 0.

The four algorithms were each run independently five times, and the average convergence curves for the multi-layer multi-channel model are shown in Fig. 4. As seen, the performance of the RRCODE algorithm is significantly better than that of the other three algorithms. With an increasing number of real evaluations, RRCODE demonstrates a faster convergence trend. This indicates that RRCODE is more efficient in optimizing this model, effectively identifying and utilizing advantageous high-permeability channels to maximize development benefits.

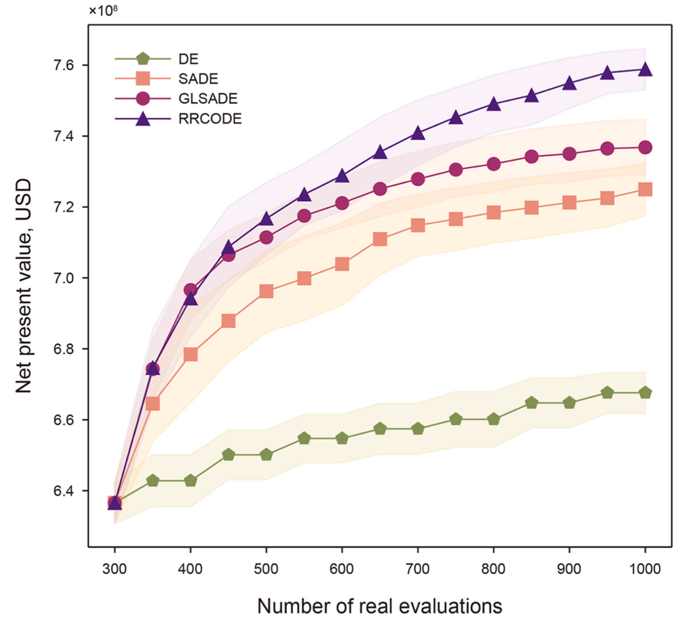

Fig. 4. Optimization curves of NPV for different algorithms in multi-layer multi-channel reservoir model.

The optimal integrated production and layer configuration schemes obtained by each algorithm are shown in Fig. 5. The solution distribution of RRCODE is relatively more uniform; in particular, the bottom-hole pressure distribution of the production wells is more reasonable, resulting in smaller pressure fluctuations for each production well across different timesteps, while also avoiding the negative impacts of excessive water injection or ultra-high-pressure production. In contrast, the GLSADE and SADE schemes exhibit extreme water injection rates and production pressure distributions in certain injection and production wells, such as I1, I4, P4, and P6 in the GLSADE scheme, and I1 and P5 in the SADE scheme, leading to imbalanced development. The DE scheme, on the other hand, displays a more conservative strategy for layer opening and closing, with many wells having most layers closed, potentially leading to lower displacement efficiency.

Fig. 6 illustrates the changes in cumulative oil production, cumulative water injection, cumulative water production, and water cut. The difference in cumulative oil production between the schemes obtained by RRCODE and GLSADE is minimal. However, RRCODE demonstrates a significant advantage in terms of water

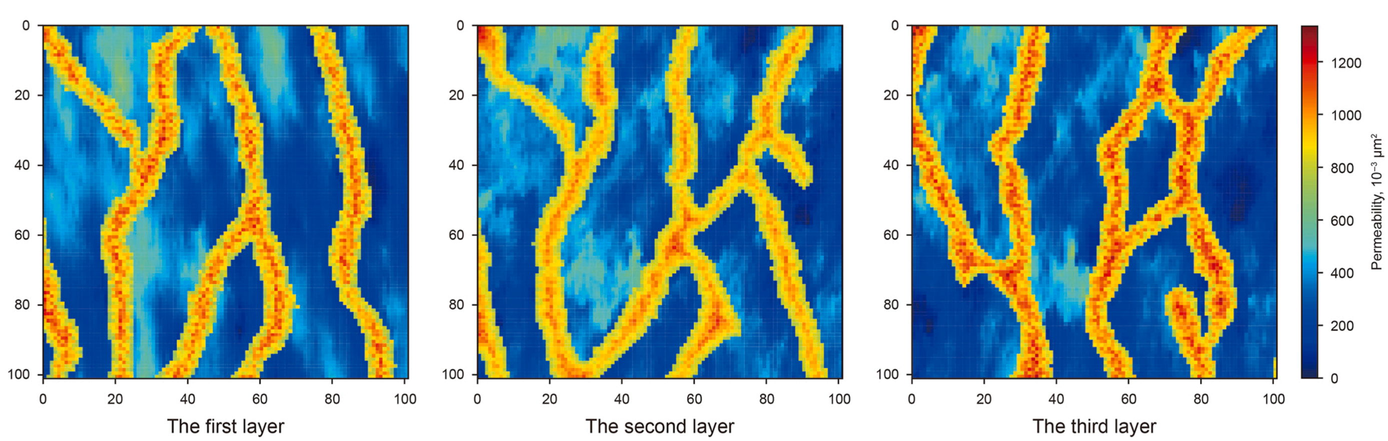

Fig. 3. Permeability field of each layer in the multi-layer multi-channel model.

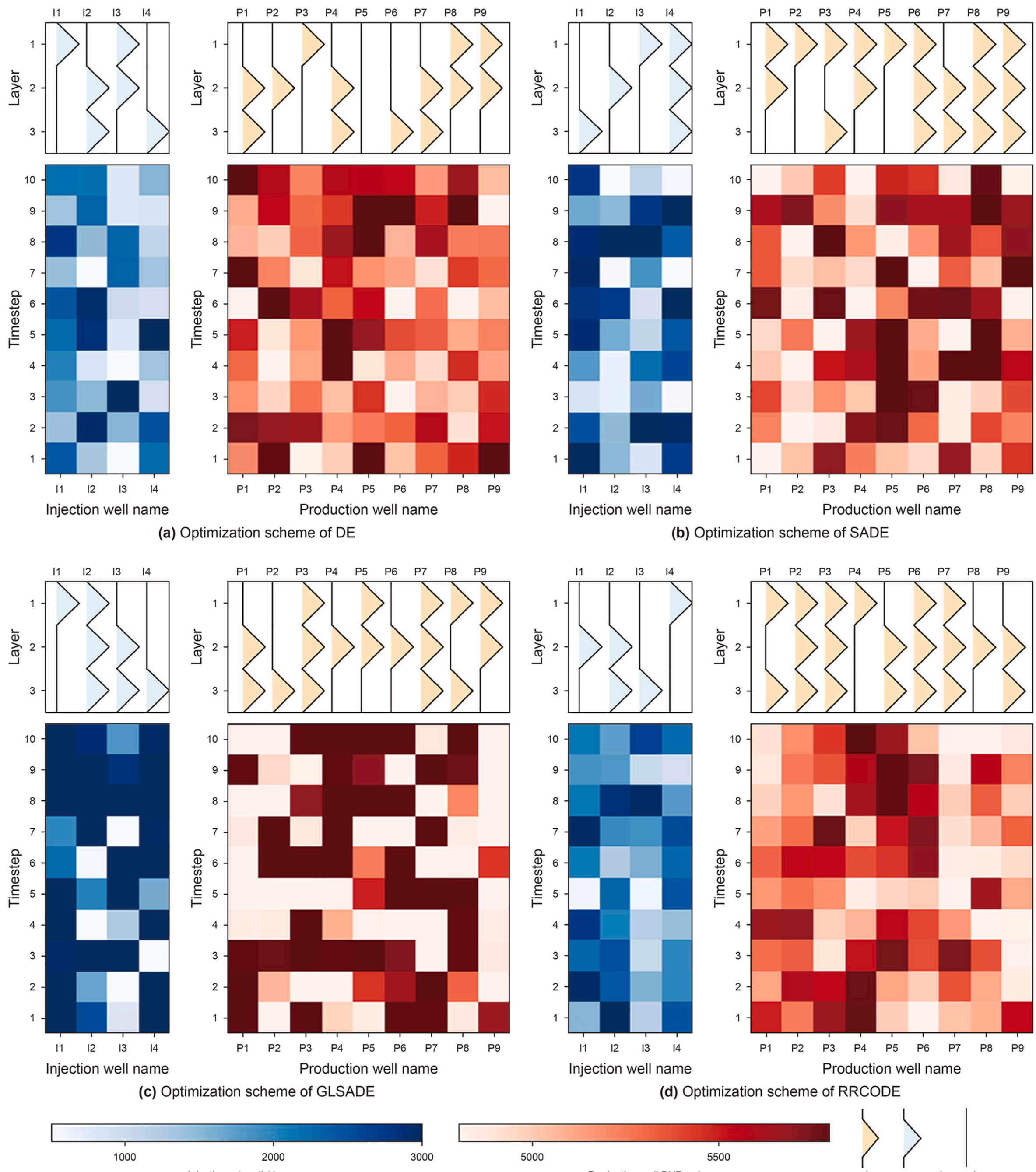

Fig. 5. Production and layer development schemes optimized by different algorithms in multi-layer multi-channel reservoir model.

injection and water production, achieving similar oil production with a lower water injection rate. This is crucial for reducing water injection and production costs and improving economic efficiency. Additionally, the water cut in the RRCODE scheme is lower than that of GLSADE and SADE, indicating that RRCODE can better control water breakthrough, thereby enhancing the sustainability of its development scheme.

4.2. Egg reservoir case study

The Egg reservoir model is a commonly used test model for optimizing reservoir development schemes, consisting of 25,200 grid blocks, 4 production wells, and 8 injection wells, shown as Fig. 7. It is widely used for comparative studies of various optimization algorithms (Feng et al., 2022; Wang J.L. et al., 2023; Zhong et al., 2022). The model is divided into 7 layers in the vertical direction, with the permeability fields of each layer shown in Fig. 8. Based on the distribution characteristics of the permeability field, the second and third layers, as well as the fifth and sixth layers, exhibit similar permeability distributions. Therefore, when designing the layer opening and closing schemes, the second and third layers, as well as the fifth and sixth layers, are combined into a single segment with the same opening and closing strategy. As a result, the number of decision variables for vertical layer

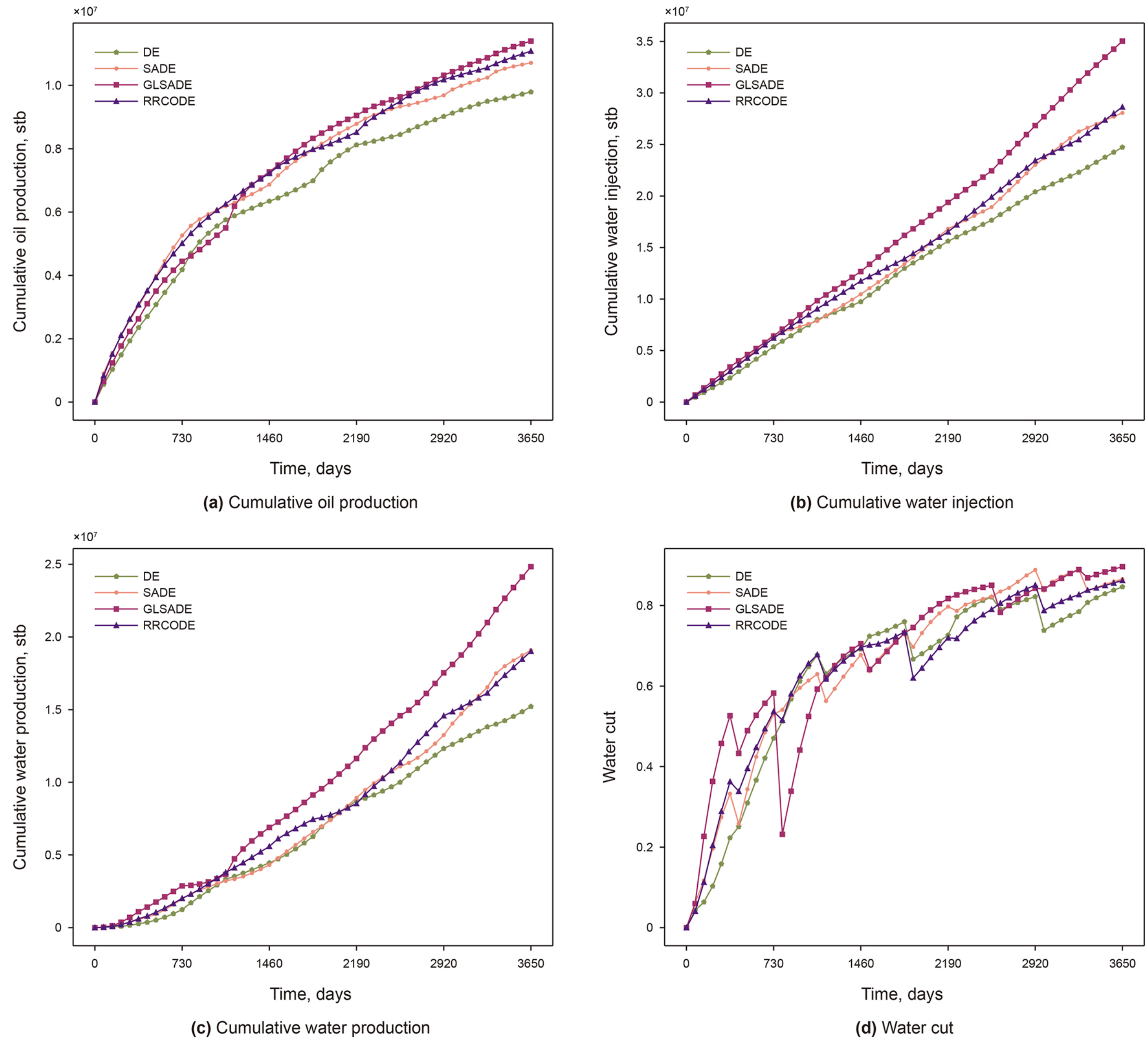

Fig. 6. Cumulative oil production, cumulative water injection, cumulative water production, and water cut curves for optimized schemes by different algorithms in multi-layer multi-channel reservoir model.

configuration of each well is 5, and the total number of decision variables related to the layer configuration is 60.

In this model, the injection wells operate under a constant liquid injection mode, with injection rates ranging from 0 to 600 m^3/day, while ensuring that the bottom-hole pressure does not exceed 420 bar. The production wells are controlled by a bottom-hole pressure control mode, with the bottom-hole pressure set between 350 and 450 bar. The oil price is set at 503.2 USD/m^3, and both water injection and water production costs are 18.87 USD/m^3, with an annual discount rate of 0. The optimization period for the development scheme is 10 years, divided into 5 timesteps, with one injection-production adjustment at each step. As a result, the number of decision variables related to the production scheme is 60, and the total number of decision variables for the integrated optimization is 120, making this a high-dimensional optimization problem for reservoir development.

In this study, each of the four algorithms was run independently 10 times, and the average results were analyzed, as shown in Fig. 9. The results demonstrate that the RRCODE method achieved significantly higher net present values (NPV) for the optimal integrated production and layer configuration schemes across the 10 independent tests compared to the two other machine

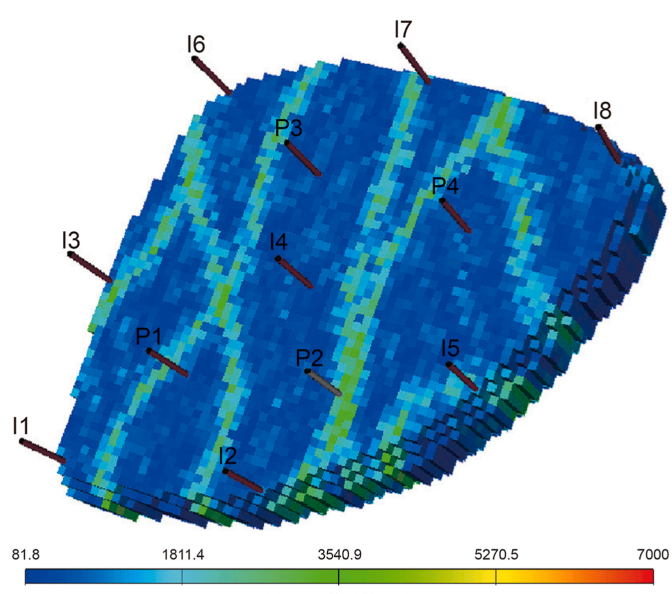

Fig. 7. Egg model and well placement distribution.

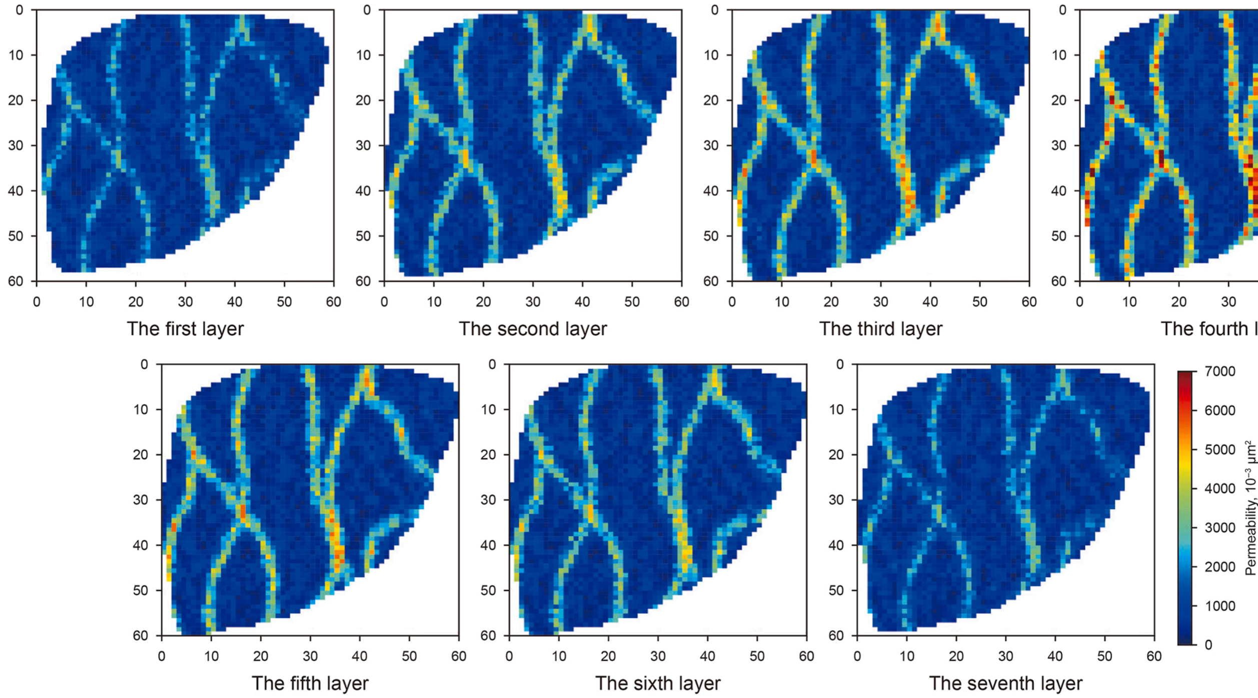

Fig. 8. Permeability field of each layer in the Egg model.

learning-based surrogate models (GLSADE and SADE), as well as the DE method based on classical optimization theory. Moreover, an analysis of the convergence trends shows that the RRCODE method consistently outperformed the other three methods throughout the optimization process. This indicates that the RRCODE method has strong potential for practical application in oilfield operations, as it can still obtain relatively superior integrated development schemes even when the number of real evaluations is reduced, while maintaining comparable computational efficiency.

The optimal integrated production and layer configuration schemes obtained by the various optimization methods are shown in Fig. 10. In the DE and SADE optimization schemes, many of the injection wells have most of their layers in the open state, such as injection wells I3, I4, I5, and I8 in Fig. 10(a), and I2, I3, and I7 in Fig. 10(b). Similarly, in the SADE and GLSADE optimization schemes, many of the production wells have most of their layers open, such as production wells P1 in Fig. 10(b) and P2 in Fig. 10(c). In contrast, the RRCODE optimization scheme exhibits a more balanced distribution of open and closed layers for both injection and production wells, with no single well having most of its layers open for injection or production. This indicates that the RRCODE method can more effectively identify the interrelationships between layers and injection-production strategies, allowing for injection and production to be conducted at the necessary layers. Under the same injection-production volumes, this method maximizes the development potential of the more favorable layers, avoiding the waste of displacement energy in areas that have already been fully developed.

The changes in cumulative oil production, cumulative water injection, cumulative water production, and water cut during the production process are shown in Fig. 11. As seen in Fig. 11(a), the difference in cumulative oil production among the four methods is relatively small. However, Fig. 11(b) and (c) shows that the RRCODE method results in significantly lower cumulative water injection and cumulative water production compared to the other three methods. This not only reduces the overall development costs, thereby increasing economic benefits, but also maintains almost the same cumulative oil production as the other methods with less total water injection. This indicates that the RRCODE method achieves a better displacement coordination between the layer configuration scheme and the production strategy, enabling the similar oil recovery effect with less water injection. Furthermore, Fig. 11(d) shows that the water cut in the RRCODE method increases more slowly during the early and middle stages of development compared to the other three methods, and the final water cut is also lower than in the SADE and GLSADE methods. Therefore, the RRCODE method provides an integrated production and layer

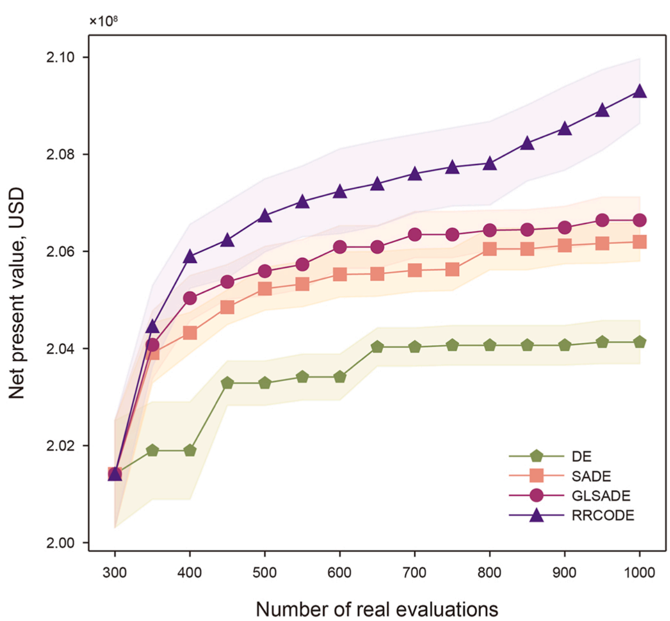

Fig. 9. Optimization curves of NPV for different algorithms in Egg reservoir model.

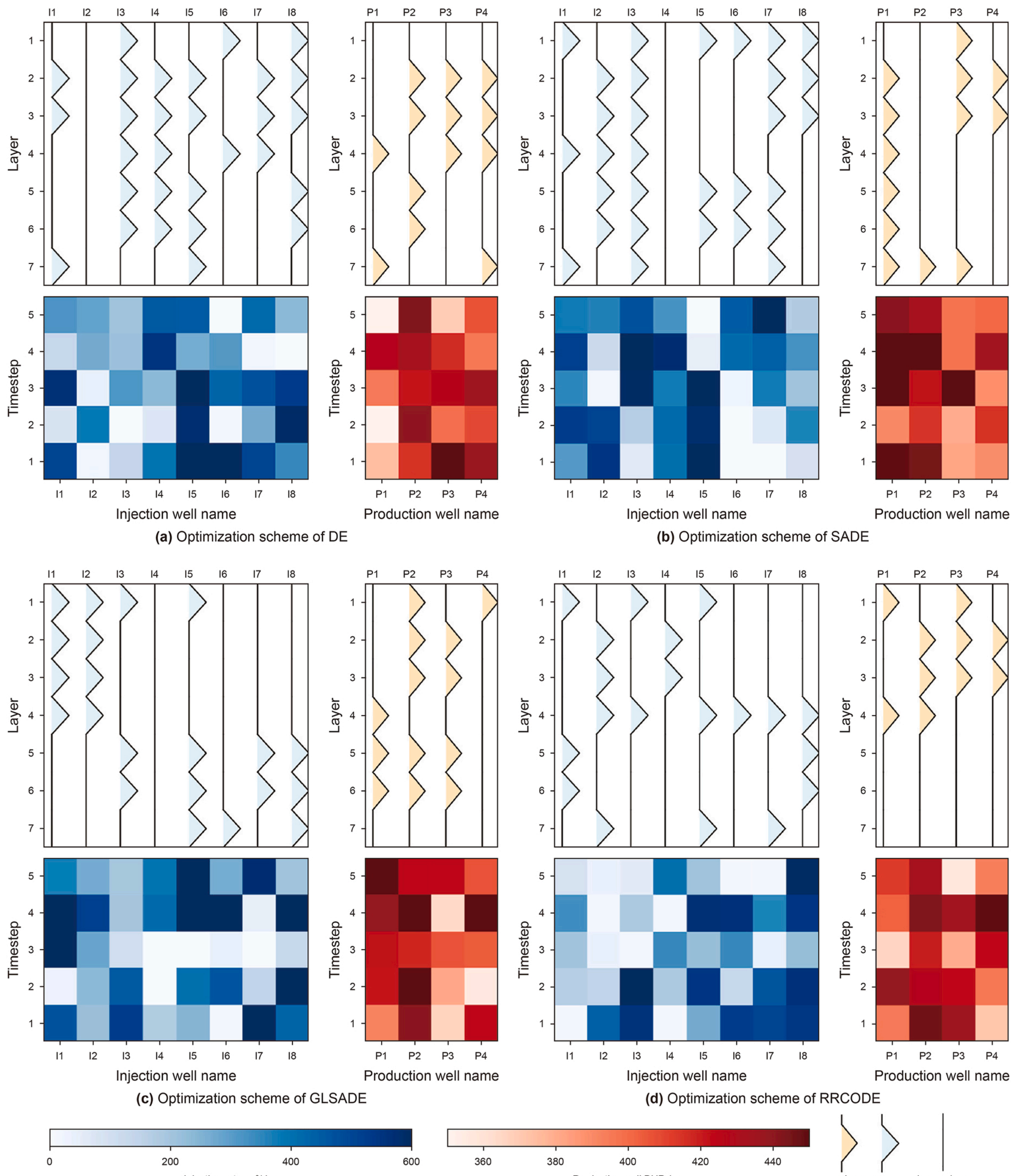

Fig. 10. Production and layer development schemes optimized by different algorithms in Egg reservoir model.

configuration scheme that is better suited to the actual needs of oilfield operations.

4.3. Real reservoir case study

To further assess the robustness and generalization capability of the proposed RRCODE framework under realistic conditions, a case study was carried out on a real reservoir model, hereinafter referred to as the S reservoir. The model features a complex, multi-segmented geological architecture, partitioned into 10 distinct zones labeled A through J, as depicted in Fig. 12. The corresponding permeability distribution, shown in Fig. 13, reveals substantial heterogeneity and highlights the challenges posed by irregular flow patterns across the reservoir.

The S reservoir model comprises a total of 19 production wells and 11 injection wells, among which 4 production wells (P4, P8, P9, and P19) have been shut-in, and 1 injection well (I1) is designated as a gas injection well. Consequently, the remaining 15 active production wells and 10 water injection wells were included in the integrated optimization of production and layer configuration schemes. This reservoir has already been in production for a certain period under a historical production strategy. Thus, the

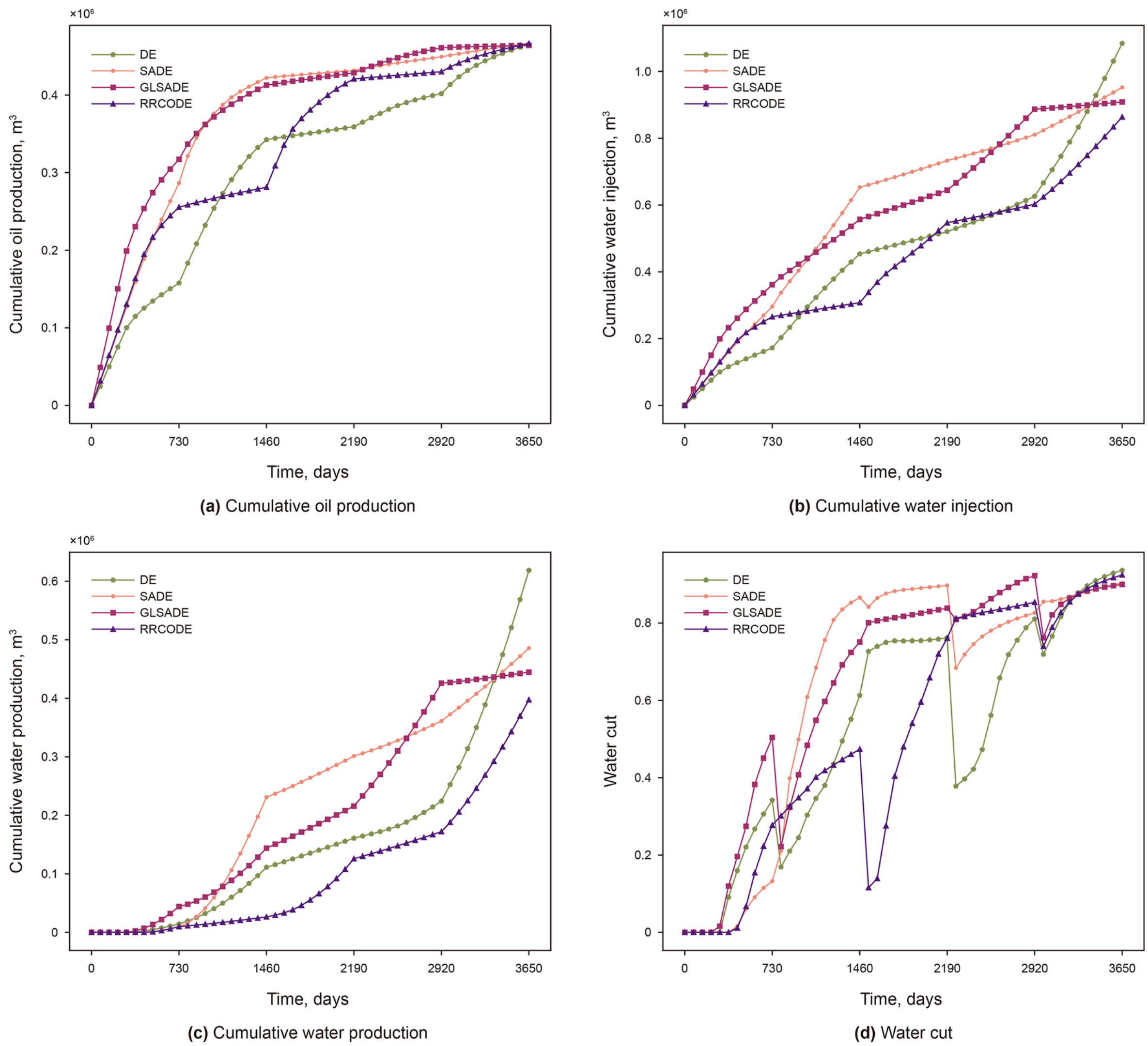

Fig. 11. Cumulative oil production, cumulative water injection, cumulative water production, and water cut curves for optimized schemes by different algorithms in Egg reservoir model.

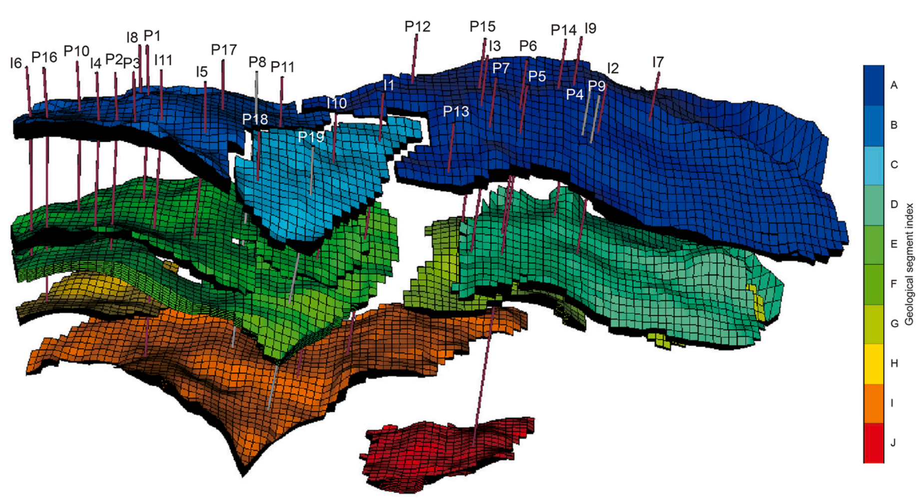

Fig. 12. Geological segment division and well distribution in the S reservoir.

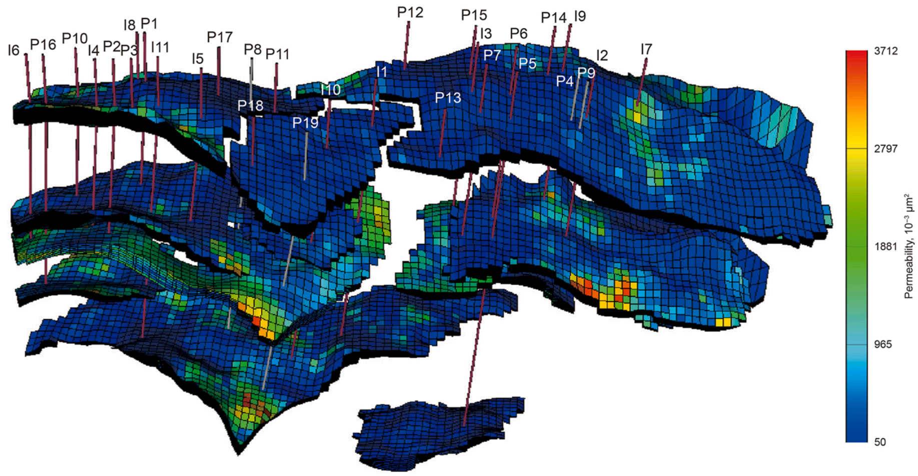

Fig. 13. Permeability field distribution of the S reservoir.

Table 2
Optimizable geological segments for each well in S reservoir.

Well name	Available geological segment	Well name	Available geological segment	Well name	Available geological segment
I2	A, D, G	I3	A, G, I, J	I4	B, E, F, H
I5	B, E, I	I6	B, E, F, H	I7	A, D, G
I8	B, E, F, H, I	I9	A, D, G, J	I10	C, E, F, I
I11	B, E, H, I	P1	B, E, F, H, I	P2	B, E, F, H
P3	B, E, F, H, I	P5	A, D, G, J	P6	A, D, G, J
P7	A, D, G, I, J	P10	B, E, F, H, I	P11	B, E, F, I
P12	A, I, J	P13	A, G, I, J	P14	A, D, G, I, J
P15	A, G, I, J	P16	B, E, F, H, I		
P17	B, E, I	P18	C, E, F, I		

Table 3
Comparison of cumulative recovery in S reservoir.

Method	Cumulative recovery, %	Improvement over historical baseline, %
Historical	27.61	–
RRCODE	30.32	2.71
GLSADE	30.16	2.55
SADE	29.99	2.38
DE	29.14	1.53

optimization conducted here aimed at enhancing reservoir performance over the subsequent 6-month period. Specifically, the optimization focused on adjusting the injection rates of wells I2–I11 and the liquid production rates of the 15 active production wells. The permissible injection rates ranged between 30 and 650 m^3/day, while the liquid production rates were constrained between 20 and 500 m^3/day. Additionally, due to differing completion intervals among wells across the ten distinct geological segments (segments A–J), individualized layer configuration schemes were optimized simultaneously over two consecutive timesteps, each spanning 3 months. The specific geological segments available for optimization in each injection and production well are detailed in Table 2.

The optimization performance of the proposed RRCODE framework was benchmarked against GLSADE, SADE, and DE, using cumulative oil recovery as the evaluation metric. The historical cumulative recovery prior to optimization served as a baseline reference. Table 3 summarizes the cumulative recovery values obtained by each optimization method and their respective improvements compared to the historical performance.

As indicated in Table 3, the RRCODE framework attained the highest cumulative recovery of 30.32%, representing a notable improvement of 2.71% over the historical baseline of 27.61%. GLSADE and SADE also showed reasonable enhancements, achieving cumulative recoveries of 30.16% and 29.99%, respectively, corresponding to increases of 2.55% and 2.38% compared to the historical scenario. DE provided the smallest improvement, resulting in a cumulative recovery of 29.14% (1.53% above the baseline). These outcomes clearly demonstrate that the integration of relational and regression surrogate models within the proposed optimization framework effectively enhances oil recovery, even when faced with the complexities of heterogeneous and multi-segmented real reservoir conditions.

Overall, the results from this realistic case study of the S reservoir reinforce the robustness and practical value of the proposed RRCODE framework. The optimization performance demonstrates that the proposed framework is not only effective for economic indicators like NPV but also exhibits strong adaptability and extensibility to other critical performance indicators, such as cumulative oil recovery. Thus, this case study highlights the broader applicability and versatility of the proposed optimization framework in addressing diverse reservoir management objectives.

4.4. *Accuracy and computational efficiency of surrogate models*

To illustrate the effectiveness and computational efficiency of the surrogate models employed in the RRCODE framework, the multi-layer multi-channel reservoir case described in Section 4.1 is taken as a representative example. Based on the data generated from this case, a detailed evaluation of both the relational

Table 4
Performance evaluation of surrogate models used in RRCODE.

Relational model			Regression model		
GNB accuracy	GNB training time, s	GNB prediction time, s	RBFN R^2	RBFN training time, s	RBFN prediction time, s
0.8864	4.4925	6.9702×10^{-6}	0.8642	0.1456	3.9873×10^{-5}

surrogate model (GNB) and the regression surrogate model (RBFN) was performed to demonstrate their lightweight characteristics and suitability for high-dimensional and nonlinear optimization problems. The overall results are summarized in Table 4.

The relational surrogate model, constructed using a Gaussian naive Bayes (GNB) classifier, was trained on 300 high-quality samples selected from a training set of 800 samples, which was randomly drawn from the full simulation-evaluated dataset. To assess the classification capability of the GNB model, 19,900 non-redundant pairwise comparisons were generated using an independently held-out test set of 200 samples. The model achieved a classification accuracy of 0.8864, which is significantly higher than the 0.5 baseline for random selection, indicating its ability to reliably identify superior solutions and effectively guide the evolutionary search direction. Notably, while the GNB model assumes feature independence, experimental results demonstrate that its classification accuracy remains high even under complex inter-feature dependencies, validating its practical utility in surrogate-assisted evolutionary optimization. The training process required 4.4925 s, and the average prediction time per pair was 6.9702×10^{-6} s, demonstrating its suitability for large-scale comparison tasks in population-based optimization.

The regression surrogate model, based on a radial basis function network (RBFN), was trained using 300 samples selected near the best-performing solutions identified during the optimization process. Testing was conducted on an independent set of 50 samples not used during training. The model achieved a coefficient of determination of $R^2 = 0.8642$, reflecting a strong approximation capability considering the problem dimensionality (169 decision variables) and limited training data. The training process required 0.1456 s, and the average prediction time per sample was 3.9873×10^{-5} s, which is sufficiently fast for practical use in simulation-based reservoir optimization tasks. Fig. 14 shows a strong overall alignment between predicted and true NPV values, with most samples lying near the diagonal line, confirming the model's generalization ability.

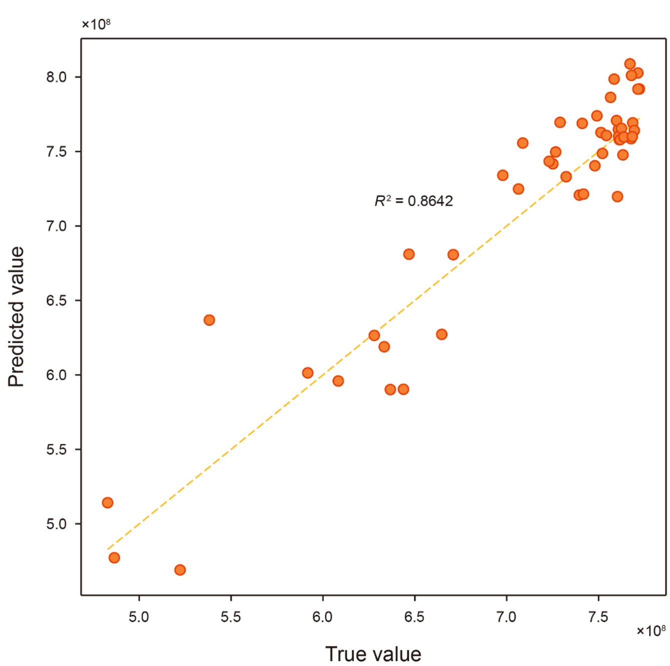

Fig. 14. Comparison between RBFN-predicted and true NPV values on the test dataset.

While the R^2 value may appear moderate compared to those reported in low-dimensional benchmark cases, it remains competitive and satisfactory within the context of complex reservoir optimization scenarios. It is worth noting that many of the higher R^2 values reported in existing studies are obtained on idealized mathematical test functions, which lack the complex heterogeneity and operational constraints inherent in real-world reservoir problems. In contrast, the present case involves 169 decision variables and a limited number of high-fidelity simulation samples, making the achieved performance particularly meaningful. Moreover, the results also highlight the limitations of relying solely on regression models in such high-dimensional, nonlinear settings, thereby justifying the incorporation of a relational surrogate model. By leveraging accurate pairwise comparisons, the relational model complements the regression component and more effectively guides the evolutionary search process.

5. Conclusions

This study systematically introduced relational machine learning surrogate models into the optimization of reservoir development schemes, marking the first application of this approach in the field. Unlike traditional regression model based optimization methods, relational models directly learn the relative superiority between samples, reducing reliance on specific target values. This enhances the generalization ability of the model, particularly in handling high-dimensional and nonlinear optimization problems. The results of this study highlight the potential of relational models in reservoir development optimization.

In addition, a novel relational and regression based composite differential evolutionary framework (RRCODE) was proposed, which integrates a Gaussian naive Bayes-based relational model with a radial basis function network regression model. This approach leverages the comparative and selection properties of evolutionary algorithms, expanding the number of training samples and learning the relationships between them. RRCODE significantly improves both the efficiency and accuracy of the optimization process, making it more adaptable to the complexities of reservoir development.

The optimization tests on the multi-layer multi-channel model, the Egg reservoir model, and a real-field reservoir model (the S reservoir) further validated the superior performance of the RRCODE method. Compared to leading machine learning-assisted optimization methods for reservoir development, RRCODE provided better integrated development schemes, demonstrating its potential for addressing complex and multifaceted reservoir optimization challenges. Furthermore, the additional real-field study illustrates the robustness and practical applicability of RRCODE in realistic, heterogeneous reservoir conditions.

Although this study primarily addresses static or pre-determined reservoir development plans, the lightweight nature and fast training capability of the employed surrogate models indicate that RRCODE can feasibly support periodic optimization tasks through repeated retraining when reservoir conditions change. Future research could explore incorporating transfer learning techniques or integrating deep learning approaches, to potentially enhance the method's adaptability and effectiveness in dynamic, data-rich scenarios. This lays the foundation for future research into more scalable and intelligent reservoir optimization systems.

CRediT authorship contribution statement

Qin-Yang Dai: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Li-Ming Zhang:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Data curation, Conceptualization. **Kai Zhang:** Writing – review & editing, Supervision, Resources, Methodology, Funding acquisition, Data curation, Conceptualization. **Hao Hao:** Software, Resources, Methodology, Conceptualization. **Guo-Dong Chen:** Validation, Resources, Methodology, Data curation. **Xia Yan:** Visualization, Investigation. **Pi-Yang Liu:** Validation, Data curation. **Bao-Bin Zhang:** Writing – review & editing, Investigation. **Chen-Yang Wang:** Writing – review & editing, Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under Grant 52325402, 52274057, and 52074340, the National Key R&D Program of China under Grant 2023YFB4104200, the Major Scientific and Technological Projects of CNOOC under Grant CCL2022RCPS0397RSN, 111 Project under Grant B08028, and China Scholarship Council under Grant 202306450108.

References

An, Z.B., Zhou, K., Hou, J., Wu, D.J., Pan, Y.P., 2022. Accelerating reservoir production optimization by combining reservoir engineering method with particle swarm optimization algorithm. J. Petrol. Sci. Eng. 208, 109692. https://doi.org/10.1016/j.petrol.2021.109692.

Broomhead, D.S., Lowe, D., 1988. Multivariable functional interpolation and adaptive networks. Complex Syst. 2 (3), 321–355.

Chang, H.J., Liu, Y.X., Lei, Y., Zhang, Q., 2020. A comprehensive workflow for real time injection-production optimization based on equilibrium displacement. Advan. Geo-Energy Res. 4 (3), 260–270. https://doi.org/10.46690/ager.2020.03.04.

Chen, G.D., Zhang, K., Xue, X.M., et al., 2020a. Surrogate-assisted evolutionary algorithm with dimensionality reduction method for water flooding production optimization. J. Petrol. Sci. Eng. 185, 106633. https://doi.org/10.1016/j.petrol.2019.106633.

Chen, G.D., Zhang, K., Zhang, L.M., et al., 2020b. Global and local surrogate-model-assisted differential evolution for waterflooding production optimization. SPE J. 25 (1), 105–118. https://doi.org/10.2118/199357-PA.

Chen, G.D., Zhang, K., Xue, X.M., et al., 2022. A radial basis function surrogate model assisted evolutionary algorithm for high-dimensional expensive optimization problems. Appl. Soft Comput. 116, 108353. https://doi.org/10.1016/j.asoc.2021.108353.

Dai, Q.Y., Zhang, L.M., Zhang, K., et al., 2023. An efficient infill well placement optimization approach for extra-low permeability reservoir. J. Energy Resour. Technol. 145 (3), 033001. https://doi.org/10.1115/1.4055198.

Das, S., Suganthan, P.N., 2011. Differential evolution: A survey of the state-of-the-art. IEEE Trans. Evol. Comput. 15 (1), 4–31. https://doi.org/10.1109/TEVC.2010.2059031.

Desbordes, J.K., Zhang, K., Xue, X.M., et al., 2022. Dynamic production optimization based on transfer learning algorithms. J. Petrol. Sci. Eng. 208, 109278. https://doi.org/10.1016/j.petrol.2021.109278.

Dong, C., Zhang, B.B., Yang, E.L., Lu, J.H., Zhang, L.M., 2023. Identification of water flooding advantage seepage channels based on meta-learning. Energies 16 (2), 687. https://doi.org/10.3390/en16020687.

Du, S.Y., Zhao, X.G., Xie, C.Y., et al., 2023. Data-driven production optimization using particle swarm algorithm based on the ensemble-learning proxy model. Pet. Sci. 20 (5), 2951–2966. https://doi.org/10.1016/j.petsci.2023.04.001.

Emerick, A.A., Silva, E., Messer, B., et al., 2009. Well placement optimization using a genetic algorithm with nonlinear constraints. In: SPE Reservoir Simulation Symposium. https://doi.org/10.2118/118808-MS.

Feng, Q.H., Li, S.S., Zhang, X.M., Gao, X.F., Ni, J.H., 2022. Well production optimization using streamline features-based objective function and Bayesian adaptive direct search algorithm. Pet. Sci. 19 (6), 2879–2894. https://doi.org/10.1016/j.petsci.2022.06.016.

Golzari, A., Haghighat Sefat, M., Jamshidi, S., 2015. Development of an adaptive surrogate model for production optimization. J. Petrol. Sci. Eng. 133, 677–688. https://doi.org/10.1016/j.petrol.2015.07.012.

Gu, J.W., Liu, W., Zhang, K., et al., 2021. Reservoir production optimization based on surrograte model and differential evolution algorithm. J. Petrol. Sci. Eng. 205, 108879. https://doi.org/10.1016/j.petrol.2021.108879.

Guo, Z.Y., Reynolds, A.C., 2018. Robust life-cycle production optimization with a support-vector-regression proxy. SPE J. 23 (6), 2409–2427. https://doi.org/10.2118/191378-PA.

Hao, H., Zhou, A.M., 2023. A relation surrogate model for expensive multiobjective continuous and combinatorial optimization. In: International Conference on Evolutionary Multi-Criterion Optimization. Springer Nature Switzerland, Cham, pp. 205–217. https://doi.org/10.1007/978-3-031-27250-9_15.

Hao, H., Zhang, J.Y., Lu, X.F., Zhou, A.M., 2020. Binary relation learning and classifying for preselection in evolutionary algorithms. IEEE Trans. Evol. Comput. 24 (6), 1125–1139. https://doi.org/10.1109/TEVC.2020.2986348.

Hao, H., Zhou, A.M., Qian, H., Zhang, H., 2022. Expensive multiobjective optimization by relation learning and prediction. IEEE Trans. Evol. Comput. 26 (5), 1157–1170. https://doi.org/10.1109/TEVC.2022.3152582.

Jin, Y.C., Wang, H.D., Chugh, T., Guo, D., Miettinen, K., 2019. Data-driven evolutionary optimization: An overview and case studies. IEEE Trans. Evol. Comput. 23 (3), 442–458. https://doi.org/10.1109/TEVC.2018.2869001.

Kim, Y.D., Durlofsky, L.J., 2021. A recurrent neural network-based proxy model for well-control optimization with nonlinear output constraints. SPE J. 26 (4), 1837–1857. https://doi.org/10.2118/203980-PA.

Kim, Y.D., Durlofsky, L.J., 2023. Convolutional-recurrent neural network proxy for robust optimization and closed-loop reservoir management. Comput. Geosci. 27 (2), 179–202. https://doi.org/10.1007/s10596-022-10189-9.

Liu, Z., Reynolds, A.C., 2021. Gradient-enhanced support vector regression for robust life-cycle production optimization with nonlinear-state constraints. SPE J. 26 (4), 1590–1613. https://doi.org/10.2118/204236-PA.

Ma, X.P., Zhang, K., Zhang, L.M., et al., 2021. Data-driven niching differential evolution with adaptive parameters control for history matching and uncertainty quantification. SPE J. 26 (2), 993–1010. https://doi.org/10.2118/205014-PA.

Ma, X.P., Zhang, K., Zhang, L.M., et al., 2022. A distributed surrogate system assisted differential evolutionary algorithm for computationally expensive history matching problems. J. Petrol. Sci. Eng. 210, 110029. https://doi.org/10.1016/j.petrol.2021.110029.

Nwankwor, E., Nagar, A.K., Reid, D.C., 2013. Hybrid differential evolution and particle swarm optimization for optimal well placement. Comput. Geosci. 17 (2), 249–268. https://doi.org/10.1016/j.aei.2012.03.007.

Onwunalu, J.E., Durlofsky, L.J., 2010. Application of a particle swarm optimization algorithm for determining optimum well location and type. Comput. Geosci. 14 (1), 183–198. https://doi.org/10.1007/s10596-009-9142-1.

Park, J., Sandberg, I.W., 1993. Approximation and radial-basis-function networks. Neural Comput. 5 (2), 305–316. https://doi.org/10.1162/neco.1993.5.2.305.

Volkov, O., Bellout, M.C., 2017. Gradient-based production optimization with simulation-based economic constraints. Comput. Geosci. 21 (5), 1385–1402. https://doi.org/10.1007/s10596-017-9634-3.

Wang, J.L., Zhang, K., Zhang, L.M., et al., 2023. Adaptive basis function selection enhanced multisurrogate-assisted evolutionary algorithm for production optimization. SPE J. 28 (6), 3192–3209. https://doi.org/10.2118/217432-PA.

Wang, J.L., Zhang, L.M., Zhang, K., et al., 2024. Multi-surrogate framework with an adaptive selection mechanism for production optimization. Pet. Sci. 21 (1), 366–383. https://doi.org/10.1016/j.petsci.2023.08.028.

Wang, L., Yao, Y.D., Luo, X.D., et al., 2023. A critical review on intelligent optimization algorithms and surrogate models for conventional and unconventional reservoir production optimization. Fuel 350, 128826. https://doi.org/10.1016/j.fuel.2023.128826.

Wang, Y., Cai, Z.X., Zhang, Q.F., 2011. Differential evolution with composite trial vector generation strategies and control parameters. IEEE Trans. Evol. Comput. 15 (1), 55–66. https://doi.org/10.1109/TEVC.2010.2087271.

Wang, Z.Z., Zhang, K., Zhang, J.D., et al., 2022. Deep reinforcement learning and adaptive policy transfer for generalizable well control optimization. J. Petrol. Sci. Eng. 217, 110868. https://doi.org/10.1016/j.petrol.2022.110868.

Wang, Z.Z., Zhang, K., Chen, G.D., et al., 2023. Evolutionary-assisted reinforcement learning for reservoir real-time production optimization under uncertainty. Pet. Sci. 20 (1), 261–276. https://doi.org/10.1016/j.petsci.2022.08.016.

Xu, J.C., Zhou, W.X., Li, H.Y., 2023. Efficient production optimization for naturally fractured reservoir using EDFM. Pet. Sci. 20 (4), 2268–2281. https://doi.org/10.1016/j.petsci.2023.02.007.

Xue, X.M., Zhang, K., Li, R.P., et al., 2020. A topology-based single-pool decomposition framework for large-scale global optimization. Appl. Soft Comput. 92, 106295. https://doi.org/10.1016/j.asoc.2020.106295.

Xue, X.M., Chen, G.D., Zhang, K., et al., 2022. A divide-and-conquer optimization paradigm for waterflooding production optimization. J. Petrol. Sci. Eng. 211, 110050. https://doi.org/10.1016/j.petrol.2021.110050.

Yan, X., Qin, G.Y., Zhang, L.M., et al., 2024. A dual-porosity flow-net model for simulating water-flooding in low-permeability fractured reservoirs. Geoenergy Sci. Eng. 240, 213069. https://doi.org/10.1016/j.geoen.2024.213069.

Zhang, H., 2004. The optimality of naive Bayes. In: Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference (FLAIRS 2004). AAAI Press, Menlo Park, California.

Zhang, K., Zhao, X.G., Chen, G.D., et al., 2021. A double-model differential evolution for constrained waterflooding production optimization. J. Petrol. Sci. Eng. 207, 109059. https://doi.org/10.1016/j.petrol.2021.109059.

Zhang, K., Sun, Z.F., Zhang, L.M., et al., 2024. A maximum entropy deep reinforcement learning method for sequential well placement optimization using multi-discrete action spaces. Geoenergy Sci. Eng. 240, 213004. https://doi.org/10.1016/j.geoen.2024.213004.

Zhao, M.J., Zhang, K., Chen, G.D., et al., 2020a. A surrogate-assisted multi-objective evolutionary algorithm with dimension-reduction for production optimization. J. Petrol. Sci. Eng. 192, 107192. https://doi.org/10.1016/j.petrol.2020.107192.

Zhao, M.J., Zhang, K., Chen, G.D., et al., 2020b. A classification-based surrogate-assisted multiobjective evolutionary algorithm for production optimization under geological uncertainty. SPE J. 25 (5), 2450–2469. https://doi.org/10.2118/201229-PA.

Zhao, X.G., Zhang, K., Chen, G.D., et al., 2020c. Surrogate-assisted differential evolution for production optimization with nonlinear state constraints. J. Petrol. Sci. Eng. 194, 107441. https://doi.org/10.1016/j.petrol.2020.107441.

Zhong, C., Zhang, K., Xue, X.M., et al., 2022. Surrogate-reformulation-assisted multitasking knowledge transfer for production optimization. J. Petrol. Sci. Eng. 208, 109486. https://doi.org/10.1016/j.petrol.2021.109486.