ACS OMEGA

http://pubs.acs.org/journal/acsodf Article

Optimization of Profile Control and Oil Displacement Scheme Parameters Based on Deep Deterministic Policy Gradient

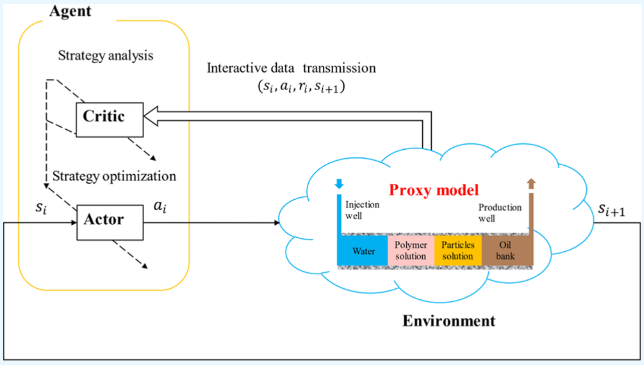
Chaodong Tan,* Chunqiu Wang, Jinjie Tian, HuiZhao Niu, Qi Wei, and Xiongying Zhang

Cite This: *ACS Omega* 2023, 8, 23739−23753

Read Online

ACCESS | Metrics & More | Article Recommendations

ABSTRACT: The parameter design of profile control and oil displacement (PCOD) scheme plays an important role in improving waterflooding efficiency and increasing the oil field production and recovery. In this paper, the parameter optimization model and solution method of the PCOD scheme based on deep deterministic policy gradient (DDPG) are constructed with the half-year increased oil production (Q_i) of injection well group as the objective function and the parameter range of PCOD system type, concentration, injection volume, and injection rate as constraints. Using the historical data of PCOD and extreme gradient boosting (XGBoost) method to construct a proxy model of PCOD process as the environment, the change rate of Q_i of well groups before and after optimization is taken as the reward function; the system type, concentration, injection volume, and injection rate are taken as the action; and the Gaussian strategy with noise is taken as the action exploration strategy. Taking XX block of offshore oil field as an example, the parameters of the compound slug PCOD process (pre-slug + main slug + protection slug) of the injection well group are analyzed, that is, parameters such as the system type, concentration, injection volume, and injection rate of each slug system are optimized. The research shows that the parameter optimization model of the PCOD scheme established based on DDPG can obtain higher oil production PCOD scheme for well groups with different PCOD, and has strong optimization and generalization ability compared with the particle swarm optimization (PSO) model.



■ INTRODUCTION

The parameter design of injection well profile control and oil displacement (PCOD) scheme plays an important role in improving the water drive effect and enhancing the oil field production and recovery.[1,2] The effect of PCOD is affected by many factors. Injection rate is one of the important factors to ensure the stable and balanced rise of block pressure. If the injection rate is too low, the time for the oil field development to reach the limit water cut will be increased, and the cost will further increase. If the injection rate is too high, part of the pores of the reservoir will be blocked, greatly reducing the permeability of the reservoir and eventually leading to difficulties in the injection of the PCOD system.[3] At the same time, the PCOD system can be used as a displacement phase to improve the unfavorable mobility ratio, improve the oil displacement efficiency, and finally achieve the goal of improving waterflood recovery. Therefore, the viscosity of the PCOD system directly affects the effect of the PCOD, and the concentration of the PCOD system is an important factor affecting its viscosity.[4,5] The injection volume of the PCOD system will also affect the PCOD effect. If the injection volume is too small, it is difficult to achieve the purpose of plugging the high permeability layer. If the injection volume is too large, the cost will increase.[6] Therefore, it is very important to optimize reasonable PCOD parameters before taking measures.

At present, the optimization design methods of PCOD schemes mainly include laboratory experiment and numerical simulation. Jia et al.[7] carried out microscopic tests of polymer microspheres with different particle sizes and indoor experiments of simulating core flow; optimized the particle sizes of microspheres at low, medium, and high water cut stages; and simulated and optimized the deep PCOD injection process parameters of polymer microspheres at different water cut stages. Although a large number of indoor experiments can obtain more comprehensive PCOD results, it is expensive and time-consuming. Therefore, numerical simulation is used to optimize the PCOD parameters. Xiao et al.[8] used numerical simulation to optimize the polymer gel injection scheme and

Received: March 25, 2023
Accepted: June 5, 2023
Published: June 19, 2023

analyzed the PCOD mechanism of polymer gel in the WAG process of fractured reservoirs. However, both the indoor experiment and numerical simulation methods are based on the orthogonal design to design the PCOD scheme. The optimal parameter combination found is only the local optimal solution in the designed PCOD scheme, not the global optimal solution under specific reservoir geological conditions, which cannot solve the parameter optimization problem quickly and comprehensively. In recent years, machine learning methods have received increasing attention due to their ability to extract information from a large amount of available data and have been widely applied in oil and gas development. Sun et al.[9] proposed a machine-learning-assisted multiobjective optimization protocol to design and optimize alkali−surfactant−polymer flooding processes in the presence of multiple technical and economic objective functions. Sun et al.[10] proposed a method for screening and optimization of polymer flooding projects using artificial neural network (ANN)-based proxies, which can obtain quick techno-economical assessments of polymer injection projects. Due to the reinforcement learning (RL) method that carries out trial and error learning through agent and environment interaction and explores the optimal behavior of the system independently, it has the advantage of not easily falling into the local optimal solution. In this paper, the RL method is applied to optimize the parameters of the PCOD scheme.

RL has strong exploration ability and self-learning ability, as well as has outstanding advantages in solving complex and high-dimensional problems, such as intelligent navigation problem[11] and commodity recommendation problem.[12] In recent years, RL has also been applied in the oil and natural gas industry. The RL method can not only be used for decision-making in the petroleum field but also for real-time adjustment of field operation parameters. He et al.[13] proposed using the proximal policy optimization algorithm (PPO) to optimize the number of wells. This method can learn the basic reservoir engineering principles without prior knowledge, such as setting wells at favorable positions with high porosity and permeability, selecting a reasonable number of wells, and maintaining a good well spacing. Shi et al.[14] established an optimization model based on deep reinforcement learning (DRL) and selected the optimal artificial lifting method by using the influence factors and effect evaluation function. The model has fast convergence speed and high prediction accuracy. It is a reliable, practical, and intelligent method. Talavera et al.[15] established a predictive control model based on RL to control oil production by optimizing injection volume. This method supports some characteristics of the reservoir system, such as strong nonlinearity, long delay of system response, and multivariable characteristics. Miftakhov et al.[16] established a DRL model based on pixel data to maximize the net present value (NPV) of water injection by changing the water injection rate. Bhowmik et al.[17] established the optimization design method of submarine pipeline based on RL, which can minimize the pipeline route length and costs related to pipeline stability, and is an efficient and economic method. Pollock et al.[18] established an automatic drilling method based on RL, which can optimize the rate of penetration, reduce the drilling bending, reduce the number of personnel on board, and improve the directional drilling efficiency. Saini et al.[19] proposed a prediction and optimization method for hole cleaning and stuck pipe prevention based on the combination of digital twin and reinforcement learning, which can automatically identify the status of the hole cleaning system and determine the best hole cleaning action. The optimization of PCOD is also a decision-making problem. Because there is a mutual relationship between the injection rate, concentration, and injection volume of the PCOD system, the level value of each parameter cannot be determined relatively independently. The optimization of PCOD parameters is a typical high-dimensional and complex problem. RL algorithm is more suitable to solve the optimization problem of PCOD parameters because of its characteristics of indefinite step search and multivariable synchronous optimization.

In this paper, the proxy model of PCOD scheme is established as the environment, and DDPG is used to optimize PCOD parameters. The optimization process of parameters is modeled and analyzed by taking the half-year increased oil production of well group (Q_i) as the optimization goal. The object of agent learning in DDPG is the environment action mapping function, which can process continuous action and state space without discretizing the control action. By comparing with the classical intelligent algorithm particle swarm optimization (PSO), the superiority of DDPG for parameter optimization of PCOD schemes is verified, and it has great application potential in solving complex problems.

■ PROXY MODEL OF PCOD SCHEME

Reservoir heterogeneity, petrophysical properties, and local physicochemical environment are important factors affecting the effect of PCOD. The pore structure of the reservoir plays a great role in controlling its reservoir permeability performance. When the PCOD system injects large pores with different pore diameters or fractures with different fracture widths and migrates for a long time, it is bound to be subject to different degrees of shear action, affecting its PCOD effect. The larger the permeability difference, the more uneven the remaining oil saturation distribution. Alvaro et al.[20] showed through experimental research that the PCOD system also has a certain impact on the nontarget layer, and the degree of impact is related to the permeability difference of the reservoir. Salinity is the sum of the amounts of various electrolytes in the system. Electrolytes will affect the distribution of polymer molecular groups. Too high salinity will make the molecular groups curl and affect the performance of the PCOD system. In a word, geological reservoir conditions are also important factors affecting the effect of PCOD. Only by designing and optimizing the optimal parameters of PCOD scheme for specific geological reservoir conditions can the oil production be increased and the reservoir potential be fully developed.

It is very difficult to build a mathematical model directly between the operation parameters and the Q_i because the PCOD is a dynamic process and involves complex dynamics involving nonlinear, high-order, time-varying, and potentially highly heterogeneous reservoirs. In this paper, the proxy model of PCOD scheme is established by using the PCOD history database and four machine learning algorithms, including linear regression (LR), random forest (RF), light gradient boosting machine (LightGBM), and extreme gradient boosting (XGBoost).

LR is a basic statistical learning method that predicts the relationship between a response variable and one or more explanatory variables by linearly fitting a dataset. RF is an ensemble learning method consisting of multiple decision trees. Each decision tree is trained using a randomly selected subset of the data and features, which gives each tree some independence and reduces the risk of overfitting. XGBoost and LightGBM are machine learning algorithms based on gradient boosting decision trees. XGBoost is characterized by the use of regularization methods and feature sub-sampling to avoid overfitting and improve model generalization. LightGBM algorithm speeds up the learning process of decision trees by proposing a gradient-based method, while also introducing feature parallelism to reduce training time. It also uses a depth-first approach to grow trees, which can more quickly learn the features and structural information of the data, while also reducing training time.

The sample data in this paper comes from the XX block of the offshore oil field, consisting of oil field data and numerical simulation data from that block, with a total of 319 data. The data distribution is shown in Table 1. Numerical simulation data refers to the virtual samples of three kinds of PCOD systems, which are established based on the geological model of the target area using the CMG software. The input parameters of proxy model include reserves of well group, porosity, permeability, horizontal section length or vertical depth of vertical well, permeability ratio, crude oil viscosity, salinity of formation water, temperature, formation pressure, water saturation, system type, system concentration, system injection volume, and injection rate of the PCOD. The output parameter is the half-year Q_i.

Table 1. Basic Data[a]

parameter	symbol	unit	minimum	mean	maximum
reserves of well group	P_0	m^3	138.8	140.37	462.02
porosity	P_1	%	0.35	0.36	0.39
permeability	P_2	mD	2870	3007.98	4816.1
horizontal section length or vertical depth of vertical well	P_3	m	11	21.11	22
permeability ratio	P_4		18.39	18.44	25.08
crude oil viscosity	P_5	mPa·s	74	120.22	263.3
salinity of formation water	P_6	g/L	3061	4494.11	5200
temperature	P_7	°C	60	62.99	63
formation pressure	P_8	MPa	10	14.96	15
water saturation	P_9	%	60	69.74	85
system type of pre-slug	t_1		1	1.99	3
system concentration of pre-slug	c_1	% mg/L	0.05	0.28	0.8
system injection volume of pre-slug	v_1	m^3	60	2207.45	4500
system type of main slug system a	$t_{2\text{-a}}$		1	2.32	3
system concentration of main slug system a	$c_{2\text{-a}}$	% mg/L	0	0.43	0.7
system injection volume of main slug system a	$v_{2\text{-a}}$	m^3	2800	5858.49	12661.4
system type of main slug system b	$t_{2\text{-b}}$		0	1	3
system concentration of main slug system b	$c_{2\text{-b}}$	% mg/L	0	0.19	1
system injection volume of main slug system b	$v_{2\text{-b}}$	m^3	0	1404.50	6000
system type of protective slug system	t_3		1	2.33	3
system concentration of protective slug system	c_3	% mg/L	0.2	0.49	1
system injection volume of protective slug system	v_3	m^3	400	712.08	2772.2
Injection rate	r	m^3/d	197.03	350.27	746.7
Q_i	Q_i	m^3	0	676.01	3176.95

[a] 0, 1, 2, and 3 in the system type, respectively, represent no system, polymer system, gel system, and particle system.

During the training period, 90% of the data in the PCOD database is used as the training set and 10% is used as the test set. The hyperparameters for the RF, XGBoost, and LightGBM models are shown in Tables 2−4, respectively. Each hyperparameter was determined through grid search, in which a range of different parameter values were set and the best parameters for the model were obtained using the GridSearchCV method in the Sklearn module in Python library. And uses two indicators to evaluate the performance of the four proxy models of PCOD scheme. The two evaluation indicators are the coefficient of determination (R^2) and the root-mean-square error (RMSE), as shown in Table 5.

Table 2. Hyperparameters of the RF Model

n_estimators	n_jobs	random_state	min_samples_leaf
6	−1	50	5

Table 3. Hyperparameters of the XGBoost Model

booster	learning_rate	max_depth	subsample	colsample bytree
gbtree	0.05	5	0.7	0.8

The prediction effect of each model is shown in Figure 1, and the comparison of the performances of four models is shown in Figure 2. The LightGBM model and XGBoost model both have good predictive performance, but the XGBoost model has the smallest RMSE and the largest R^2 on the test set, which is better than the LightGBM model. Therefore, this paper chooses the XGBoost algorithm as a supervised learning algorithm to build a proxy model. After training and debugging, the final RMSE value of the model is less than 110, and R^2 is 0.94. In the future, with the accumulation of production data of PCOD wells, the data volume of the database will gradually increase, and the model parameters will be adjusted to improve accuracy and adaptability.

OPTIMIZATION MODEL OF PCOD SCHEME BASED ON DDPG

Taking XX block of offshore oil field as an example, the injection method of PCOD in this block is compound slug (pre-slug + main slug + protection slug). There are three types of systems: polymer, gel, and particle. On this basis, the type, concentration, injection volume, and injection rate of each slug system are optimized.

Principle of DDPG Algorithm. The basic elements of RL include state, action, policy, and reward function. Through the communication and feedback between the agent and the environment, the optimal policy is learned from random and continuous attempts, and the machine learning method that maximizes the long-term cumulative return is obtained. The interaction process between the agent and the environment is as follows: first, the agent obtains a state "*s*" of the environment by sensing the environment; Second, the agent selects an action "*a*" according to a decision rule; Finally, after the action is executed, the environment state is changed. At the next moment, the agent modifies its decision rules after obtaining a reward "*r*" from the environment, as shown in Figure 3.

According to the action selection of agents, RL algorithms can be divided into three categories: value-based, policy-based,

Table 4. Hyperparameters of the LightGBM Model

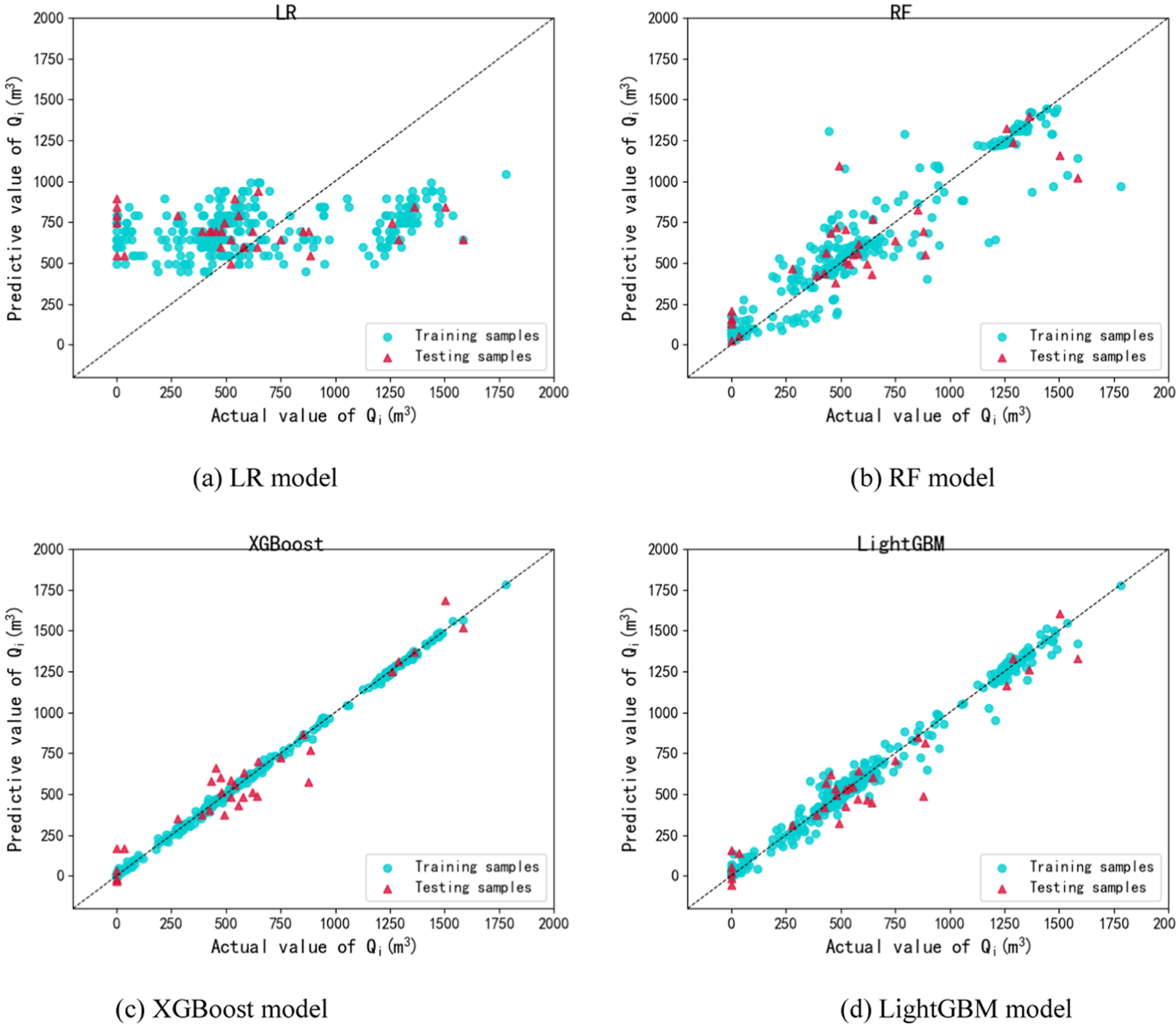
boosting_type	num_leaves	min_data_in_leaf	max_depth	learning_rate	feature_fraction	lambda_l1
gbdt	20	20	6	0.3	0.8	0.1

Table 5. Regression Model Evaluation Index[a]

evaluation index	calculation formula	criteria
RMSE	$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$	the smaller the RMSE, the smaller the error, the larger the RMSE, the larger the error
R^2	$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$	R^2 is between 0 and 1; the larger the value, the better the model fitting

[a] n represents the number of wells, y_i is true value,$\hat{y}_i$ is predicted value, $\bar{y}_i$ is the mean of true value.

and actor-critic. The value-based RL algorithm implicitly constructs the optimal policy by obtaining the optimal value function and selecting the action corresponding to the maximum value function.[21] The value function in RL is used to measure the expected cumulative reward of a state or a state-action pair. There are typically two types of value functions: state value function and action value function (also known as *Q* function). The state value function measures the expected cumulative reward that an agent can obtain in the current state, such as the SARSA algorithm,[22] while the action value function measures the expected cumulative reward that can be obtained by taking a certain action in the current state, such as the *Q*-learning algorithm.[23] The agent makes decisions based on the estimated results of the value function to maximize the expected cumulative reward. The value-based algorithm has high sample utilization rate and small value function estimation variance, so it is not easy to fall into local optimization. However, this kind of algorithm selects actions by optimizing



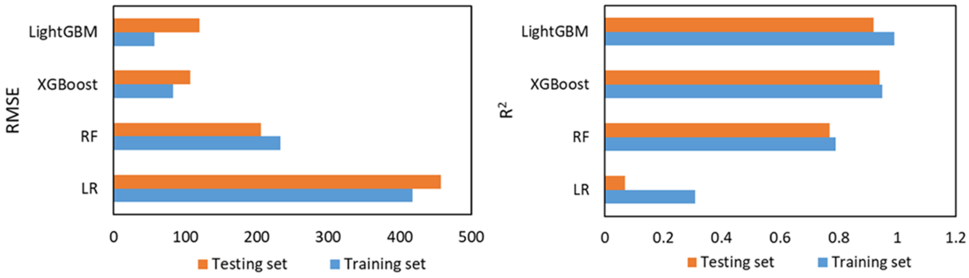
(a) LR model

(b) RF model

(c) XGBoost model

(d) LightGBM model

Figure 1. Four models' predictions for full sample.



(a) Evaluation index RMSE

(b) Evaluation index R^2

Figure 2. Comparison of the performances of four models.

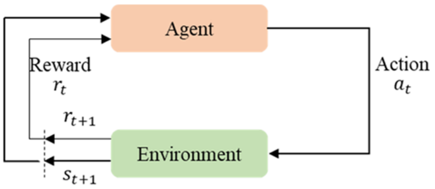


Figure 3. Interaction process between agent and environment.

the Q value, and the agent needs to calculate the corresponding Q value before selecting each action. It can only solve the discrete action space problem, which is prone to overfitting, and the complexity of the problem that can be handled is limited. For the complex and high-dimensional problem of parameter optimization in this paper, the value-based algorithm is difficult to apply.[24]

The policy-based RL algorithm directly searches for the best policy across the value function. Compared with value-based algorithm, policy-based algorithm can deal with discrete and continuous space problems and has better convergence. However, the policy-based algorithm has low sampling efficiency and requires a large amount of computing power as support, which greatly limits the application of the algorithm. At the same time, the trajectory variance of this algorithm is large, the sample utilization rate is low, and it is easy to fall into the dilemma of local optimization.

The actor-critic algorithm combines the value-based (corresponding reviewer, critic) method with the policy-based (corresponding performer, actor) method, and simultaneously learns the policy and value function. The actor trains the value function according to the feedback of critic, while the critic trains the value function and uses the temporal difference (TD) method for one-step update.[25] DDPG[26] is a typical actor-critic algorithm and an important milestone of the RL algorithm. Among them, the application of deep neural network enhances the feature extraction ability of the model and provides a possibility for the application of RL in high-dimensional continuous state space. At the same time, the DDPG algorithm utilizes an experience replay mechanism to store the agent's interactions with the environment in a memory buffer, enabling offline training from the memory to improve sample efficiency and stability. By employing neural networks as function approximators, DDPG can enhance its capability by increasing the complexity and parameter count of the networks, thereby adapting to more complex problems. DDPG learns both an actor network to capture policies and a critic network to estimate value functions, enabling the algorithm to perform policy optimization and value estimation, leading to enhanced stability and effectiveness. It is more suitable for the parameter optimization of PCOD in this paper.

The DDPG algorithm is based on policy gradient and DQN algorithm, which can solve the problem of continuous action space, as shown in Figure 4. The DDPG algorithm consists of four neural networks, namely, the actor network $\mu(s|\theta^{\mu})$, the target actor network $\mu(s|\theta^{\mu'})$, the critic network $Q(s, a|\theta^{Q})$, and the target critic network $Q(s, a|\theta^{Q'})$. Here, θ^{μ}, $\theta^{\mu'}$, θ^{Q}, and $\theta^{Q'}$ represent the internal parameters of these four neural networks. The actor network is used to determine the action a_t that the agent will take in the current state s_t. In order to explore the action space and avoid getting stuck in local optima, a noise function can be added to the actor network, as shown in eq 1.

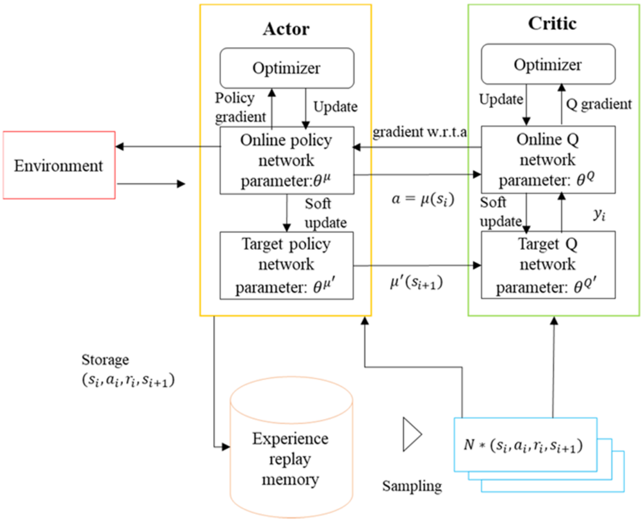

Actor
Optimizer
Policy gradient
Update
Online policy network parameter:θ^{μ}
Soft update
Target policy network parameter: $\theta^{\mu'}$
Critic
Optimizer
Update
Q gradient
Online Q network parameter: θ^{Q}
Soft update
y_i
Target Q network parameter: $\theta^{Q'}$
Environment
gradient w.r.t.a
$a = \mu(s_i)$
$\mu'(s_{i+1})$
Storage (s_i, a_i, r_i, s_{i+1})
Experience replay memory
Sampling
$N * (s_i, a_i, r_i, s_{i+1})$

Figure 4. Principle of the DDPG algorithm.

$$a_t = \mu(s_t|\theta^{\mu}) + N_t \tag{1}$$

where N_t refers to a noise function. The critic network is used to estimate the state-action value $Q(s_t, a_t)$. The target actor network is used to generate the action a_{t+1} that the agent should take in the next state s_{t+1}. The target critic network is used to calculate the state-action value $Q(s_{i+1}, a_{i+1})$. The target value for calculating $Q(s_t, a_t)$ based on the Bellman equation is as follows

$$y_i = r_i + \gamma Q'((s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'}) \tag{2}$$

where y_i is the target value of $Q(s_t, a_t)$, γ is the discount factor, and r_i is the reward feedback from the environment.

The DDPG algorithm uses an experience replay mechanism to break the correlation between interaction sequences τ, storing state transitions (s_i, a_i, r_i, s_{i+1}) in an experience pool. A certain number of state transitions are randomly selected from the pool for training and the parameters of the 4 neural networks are updated at the same time. The parameters of the critic network are updated in the direction of minimizing the loss function, which is as follows

$$L = \frac{1}{N}\sum i(y_i - Q(s_i, a_i|\theta^Q))^2 \tag{3}$$

where $Q(s_i, a_i|\theta^Q)$ represents the estimated value of $Q(s_i, a_i)$. The actor network parameters are updated in the direction of maximizing the cumulative reward.

$$\nabla_{\theta^\mu} J \approx \frac{1}{N}\sum_i \nabla_a Q(s, a|\theta^Q)|_{s=s_i,a=\mu(s_i)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)|_{s_i} \tag{4}$$

The target networks are used to improve the stability of the algorithm. The update rule for the target network parameters is shown in eqs 5 and 6.

$$\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'} \tag{5}$$

$$\theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'} \tag{6}$$

where τ is the update parameter, $\tau \ll 1$.

Optimization Model of PCOD Scheme Based on DDPG Algorithm. In this paper, the optimization model of PCOD scheme based on DDPG is established, and the maximum Q_i is achieved by automatically optimizing the type, concentration, injection volume and injection rate of PCOD system. The two key contents of RL are environment and agent. In order to express the optimization problem of PCOD scheme as an RL problem, the environment and agent need to be defined first. In the optimization process of PCOD scheme based on DDPG algorithm, the environment is a proxy model formed by data-driven modeling of PCOD process. The agent takes actions in this environment to try to maximize the best possible reward in the environment. In this paper, the agent is defined from the angle of injection well. Its possible action is to change the type, concentration, injection volume and injection rate of the PCOD system. According to the action selected by the oil well, the observed states are Q_i, type, concentration, injection volume and injection rate of the PCOD system. The elements in the DDPG method are shown in Table 6.

When the environment resets the first state, the agent selects the action through the Gaussian policy, and the selected action is transferred to the proxy model according to the parameters of the PCOD operation. According to the comparison between the output of the model and the optimization target, a reward value is allocated for the action. This reward serves as a feedback signal to let the agent know whether the action in a given state is beneficial to the target so that the agent can decide the next action. Table 7 shows the detailed optimization process of the DDPG model.

Table 6. Elements of DDPG Algorithm

elements	description
objective	find the best combination of PCOD parameters for each well to maximize Q_i.
environment	data-driven proxy model of PCOD scheme.
state	current Q_i, type, concentration, injection volume and injection rate of PCOD system.
action	change the values of the type, concentration, injection volume and injection rate of PCOD system.
reward	change of Q_i after current optimization compared with that before optimization.
behavior policy	Gaussian policy.

Table 7. Procedures of the DDPG Algorithm for the PCOD Process

DDPG algorithm for optimization process of PCOD scheme

```
environment: PCOD process
build a data-driven proxy model simulation environment
agent: DDPG
randomly initialize critic network $Q(s, a|\theta^Q)$ and actor $\mu(s|\theta^\mu)$ with weights $\theta^Q$
  and $\theta^\mu$.
initialize target network $Q'$ and $\mu'$ with weights $\theta^{Q'} \leftarrow \theta^Q$, $\theta^\mu \leftarrow \theta^\mu$.
initialize replay buffer R.
for episode = 1, $M$ do
    initialize a random process N for action exploration
    receive initial observation state $s_1$
    for $t$ = 1, T do
        select action $a_t = \mu(s_t|\theta^\mu) + N_t$ according to the current policy
        and exploration noise
        execute action $a_t$ and observe reward $r_t$ and observe new state
        $s_{t+1}$
        store transition $(s_t, a_t, r_t, s_{t+1})$ in R
        sample a random minibatch of N transitions $(s_i, a_i, r_i, s_{i+1})$ from
        R
        set $y_i = r_i + \gamma Q'\ (s_{i+1}),\ \mu'((s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
        update critic by minimizing the loss:
        $L = \frac{1}{N}\sum_i (y_i - Q(s_i, a_i|\theta^Q))^2$
        update the actor policy using the sampled policy gradient:
            $\nabla_{\theta^\mu} J \approx \frac{1}{N}\sum_i \nabla_a Q(s, a|\theta^Q)|_{s=s_i,a=\mu(s_i)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)|_{s_i}$
        Update the target networks:
            $\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}$
            $\theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}$
    end for
end for
```

DDPG-Environment. In this paper, the proxy model of PCOD scheme is used as the environment of DDPG model. During the interaction between agent and proxy model, agent will provide the current state as input to proxy model, and then the model will predict the next state and reward. Agents update their policy network and Q network according to the information provided by the proxy model. In this way, the intelligent agent can take into account the dynamic changes in environmental parameters during the trial and error process and make corresponding adjustments.

Through data-driven modeling, there is no need to study the mechanism reaction process of the object. It only needs to be driven by data and build the prediction model through the establishment of artificial intelligence algorithm. The real function of the proxy model is to simulate the real environment and provide corresponding feedback to the DDPG agent. The DRL model with numerical simulation as the environment needs to run numerical simulation during each optimization, which is very time-consuming. The advantages of establishing RL environment based on data-driven method are low cost,

Table 8. Value Range of Each Parameter Variable

symbol	t_1	c_1	v_1	$t_{2\text{-a}}$	$c_{2\text{-a}}$	$v_{2\text{-a}}$	$t_{2\text{-b}}$	$c_{2\text{-b}}$	$v_{2\text{-b}}$	t_3	c_3	v_3	s
min	1	0.05	60	1	0.25	2800	0	0	0	1	0.2	400	197.0
max	3	0.8	4500	3	0.7	12661.4	3	1	6000	3	1	2772.2	746.7

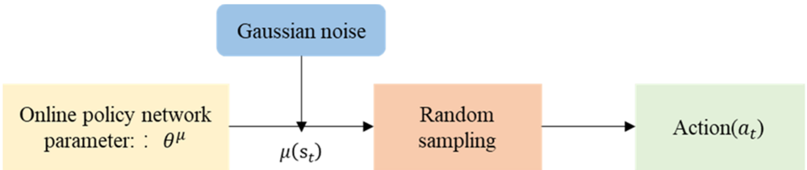


Figure 5. Schematic diagram of behavior policy.

Table 9. Basic Data of Well_1 and Well_2

symbol	P_0	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	t_1	c_1
well_1	138.8	0.36	3000	11	18.39	76	4500	60	10	60	2	0.18
well_2	283.9	0.35	2980	12	18.39	263.3	4500	63	15	80	1	0.1
symbol	v_1	$t_{2\text{-a}}$	$c_{2\text{-a}}$	$v_{2\text{-a}}$	$t_{2\text{-b}}$	$c_{2\text{-b}}$	$\mathrm{v}_{2\text{-b}}$	t_3	c_3	v_3	s	Q
well_1	1800	2	0.35	3600	0	0	0	2	0.42	2100	200	631.89
well_2	415	1	0.4	12661.4	0	0	0	1	0.2	747.1	746.7	1781.5

high efficiency, better adaptability to complex engineering scenarios and high flexibility.

DDPG-State. The state space is 14, that $S = [Q_i, \mathrm{t}_1, c_1, \mathrm{v}_1, t_{2\text{-a}}, c_{2\text{-a}}, v_{2\text{-a}}, t_{2\text{-b}}, c_{2\text{-b}}, v_{2\text{-b}}, t_3, c_3, v_3, r]$, Where t_1 represents the system type of pre-slug, c_1 is system concentration of pre-slug, v_1 is system injection volume of pre-slug, $t_{2\text{-a}}$ is system type of main slug system a, $c_{2\text{-a}}$ is system concentration of main slug system a, $\mathrm{v}_{2\text{-}a}$ is system injection volume of main slug system a, $t_{2\text{-b}}$ is system type of main slug system b, $c_{2\text{-b}}$ is system concentration of main slug system b, $v_{2\text{-b}}$ is system injection volume of main slug system b, t_3 is system type of protective slug system, c_3 is system concentration of protective slug system, v_3 is system injection volume of protective slug system, and r is injection rate. In the training process, the agent observes its current state, selects actions according to the current policy, changes correspondingly after execution in the environment, and then predicts the corresponding Q_i, t_1, c_1, v_1, $t_{2\text{-a}}$, $c_{2\text{-a}}$, $\mathrm{V}_{2\text{-a}}$, $t_{2\text{-b}}$, $c_{2\text{-b}}$, $\mathrm{V}_{2\text{-b}}$, t_3, c_3, v_3, r, thereby outputting a new set of states.

DDPG-Action. The action space is 13, that $A = [t_1, c_1, v_1, t_{2\text{-a}}, c_{2\text{-a}}, v_{2\text{-a}}, t_{2\text{-b}}, c_{2\text{-b}}, \mathrm{v}_{2\text{-}b}, t_3, c_3, \mathrm{v}_3, r]$. Due to operational constraints, the variation range of action variables shall not exceed its maximum and minimum values, The range of values for each parameter variable is shown in Table 8.

The activation function of the last layer of the actor network is the tanh function so that the action output of each layer is controlled between [−1,1]. Then, according to eq 7, the output of the neural network is scaled up in proportion to the action boundary to make the action output of the actor network meet the boundary value constraint.

$$a\_ = \frac{a_{\max} - a_{\min}}{b_{\max} - b_{\min}} \cdot (a - b_{\min}) + a_{\min} \tag{7}$$

where a represents the value output by the tanh function of the last layer in the actor network; $b_{\max}$ and $b_{\min}$, respectively, represent the maximum and minimum values of the output value of the tanh function; $a\_$ represents the value of the operation after being scaled up in proportion to the operation boundary; and $a_{\max}$ and $a_{\min}$, respectively, represent the maximum value and the minimum value of the actual action.

DDPG-Reward. In RL, the reward comes from the environment. Facing different tasks, the reward function needs to be carefully designed according to the characteristics of the task and the state of the environment. In this paper, the goal of the agent is to maximize the Q_i. Therefore, this paper gives rewards according to the change rate of the Q_i after the agent optimization compared with the initial Q_i. The reward function is shown in eq 8.

$$\mathrm{reward}_t = \frac{s_{Q_{i\_t}} - s_{Q_{i\_0}}}{s_{Q_{i\_0}}} \tag{8}$$

where $s_{Q_{i\_t}}$ represents the Q_i obtained by the agent at time t and $s_{Q_{i\_t\text{-}0}}$ represents the initial Q_i. If the agent takes an action at time t that reduces the Q_i relative to the initial Q_i, it will receive a negative reward, and the greater the reduction, the larger the absolute value of the negative reward. This shows the agent that this strategy has a negative effect, so it should change its strategy and try to obtain a positive reward. Conversely, the greater the increase in Q_i to the initial Q_i after taking an action at time t, the greater the positive reward, which will encourage the agent to take such actions to increase the reward value.

DDPG-Action Exploration Policy. DDPG algorithm is a deterministic policy algorithm. Although the algorithm efficiency of deterministic policy is high, it also has the defect of insufficient exploration ability. In order to explore potential better policy, random noise is introduced into the decision-making mechanism of action: the decision-making of action is changed from a deterministic process to a random process, and then the action is sampled from the random process and issued to the environment for execution. This policy is called the behavior policy. The behavior policy adopted in this paper is Gaussian policy with noise. The behavior policy is shown in Figure 5. Since the actor in the DDPG algorithm uses a certain value in the Gaussian distribution to output, controlling the variance of the Gaussian distribution can control the proportion of ″exploration″ and ″utilization″ of the actor. In this paper, the initial variance $\sigma = 0.1$. After the data in the memory pool reaches the upper limit, learning begins.

https://doi.org/10.1021/acsomega.3c02003

Experimental Study. In this paper, the two wells with different Q_i in this block are selected for optimization. The basic data is shown in Table 9. The purpose of well_1 PCOD is to expand the swept volume of injected water, improve the recovery degree of well group and improve the development effect of water drive through PCOD of well-group_1 reservoir. After PCOD, the Q_i of well-group_1 is 631.89 m^3, which has a large optimization space. The well_2 has good connectivity with other wells, and injection breakthrough is obvious. The remaining recoverable reserves of the well group are large, which has great potential for tapping. After the start of profile control and flooding, well cluster 2 has achieved a certain effect of precipitation and oil increase. The Q_i of well-group_2 is 1781.5 m^3, and the PCOD effect is very good.

The overall framework of parameter optimization of PCOD scheme based on the DDPG algorithm is shown in Figure 4. After repeated experimental debugging, the network parameters were optimized based on the model convergence and reward value curve, and the optimal network structure was ultimately selected. Both actor network (main network and target network) and critic network (main network and target network) contain one hidden layer network, and the number of neurons in each layer is 30 and 60, respectively. The activation function of the last layer of the actor network is the tanh function so that the action output of each layer is controlled between [−1,1]. The critic network uses relu activation function to evaluate the PCOD parameters obtained by actor network. After repeated experiments and debugging, the actor network learning rate is set to 0.001, the critic network learning rate is set to 0.002, the training round is set to 500, and the maximum time step number of each round is 200, which means that the agent will conduct 200 steps of exploratory learning in each round, and the maximum size of the memory buffer is 10,000.

■ RESULTS AND DISCUSSION

The change curve of reward value during the training of the DDPG optimization model of well_1 is shown in Figure 6, which shows the change curve of the rewards obtained by the agent in each episode. During the training process, the total reward gradually increases and fluctuates within the equilibrium range, which indicates that the DDPG agent can learn in the process of interacting with the environment to select actions with higher rewards. In the absence of any prior knowledge given to the agent, the policy adopted by the agent in the initial stages is difficult to improve the oil production, and hence the reward value is relatively low. As the agent learns, the oil production gradually increases, and the reward value also increases accordingly. The total reward for well_1 stabilizes around the 50th iteration, and the model has essentially learned the combination of variables to maximize the overall reward of the system, with the stimulation scheme parameters all within their parameter range. The fluctuation in the curve in the later stages of training is due to the introduction of the Gaussian policy in the action exploration process, which allows the model to continuously explore potential better policies. This also makes the model less likely to fall into local optimal solutions. The environment in this study is relatively complex and requires optimization of multiple parameters, with a total training time of about 40 min and an average time of 4.8 s per iteration. Compared to traditional numerical simulation methods, the computational cost is significantly reduced.

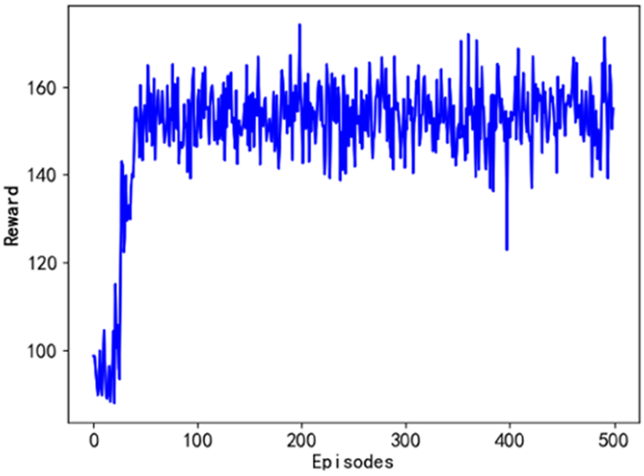


Figure 6. Change curve of reward value during the training of DDPG optimization model of well_1.

Each iteration of the model includes two processes: sampling and optimization. Sampling refers to sampling some trajectories from the environment under the current policy to update the value function and action policy of the policy. Optimization refers to updating the parameters of the policy based on the samples obtained from the sampling process. In the optimization process, the gradient of the policy needs to be computed, and the policy parameters are updated based on the gradient. To ensure the correctness and efficiency of policy updates, it is necessary to periodically test the performance of the policy to identify and solve problems in a timely manner. Through testing, the current policy reward performance indicator can be obtained in the current environment, which can be used to determine whether the current policy has been improved and adjust the algorithm hyperparameters or optimization methods in a timely manner to better improve the algorithm performance. During the testing phase, there is no need for the model to continue exploration because the optimal action has been determined, and it is desired to see the model execute the task with the optimal action. Therefore, no noise is added during testing. This article tests once per iteration and immediately after each sampling process.

The change curve of reward value during the testing of the DDPG optimization model of well_1 is shown in Figure 7. Since the model has been tested, there is no need to conduct action exploration. The total reward value of the parameter optimization model of PCOD scheme based on DDPG tends to be stable after 220 episodes. It shows that when the DDPG model is stable, the total reward value of the model is from punishment to reward at the beginning, iterative training is continued, and finally the optimal value is stable.

The change curve of various parameters during the testing of the DDPG optimization model of well_1 is shown in Figure 8. The initial Q_i of well_1 was 631.89 m^3. The intelligent agent stabilized at the maximum incremental oil volume of 1840.09 m^3 from the 200th round, which increased the oil volume by 1208.2 m^3, a 191.2% improvement. The optimized parameters are shown in Table 10. The type of segment plugging system changed after optimization, indicating that the particle + polymer composite injection system is suitable for this geological condition, rather than the gel system. At the same time, the injection volumes and rates of various segment

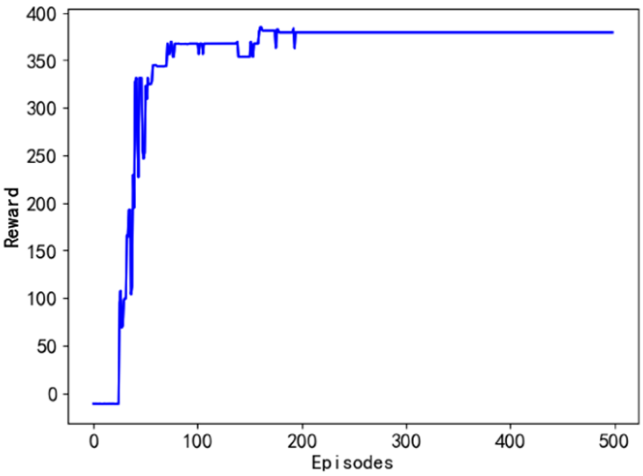


Figure 7. Change curve of reward value during the testing of DDPG optimization model of well_1.

plugging systems have also changed after optimization, indicating that the original PCOD parameters were not set appropriately, resulting in the underutilization of the reservoir potential.

The change curve of reward value during the training of the DDPG optimization model of well_2 is shown in Figure 9. The reward value of the model increases significantly around the 50th episode, which indicates that the model has begun to learn how to take action to obtain greater rewards. The model reward value is negative because the Q_i after the PCOD optimization for well_2 was 1781.5 m^3, which is relatively superior in the block. During the training process, the model continuously tried and erred, making it difficult to surpass the initial oil production increase.

The change curve of reward value during the testing of the DDPG optimization model of well_2 is shown in Figure 10. The total reward value of the DDPG optimization scheme model starts to be positive at around 200 rounds and tends to stabilize after 300 rounds.

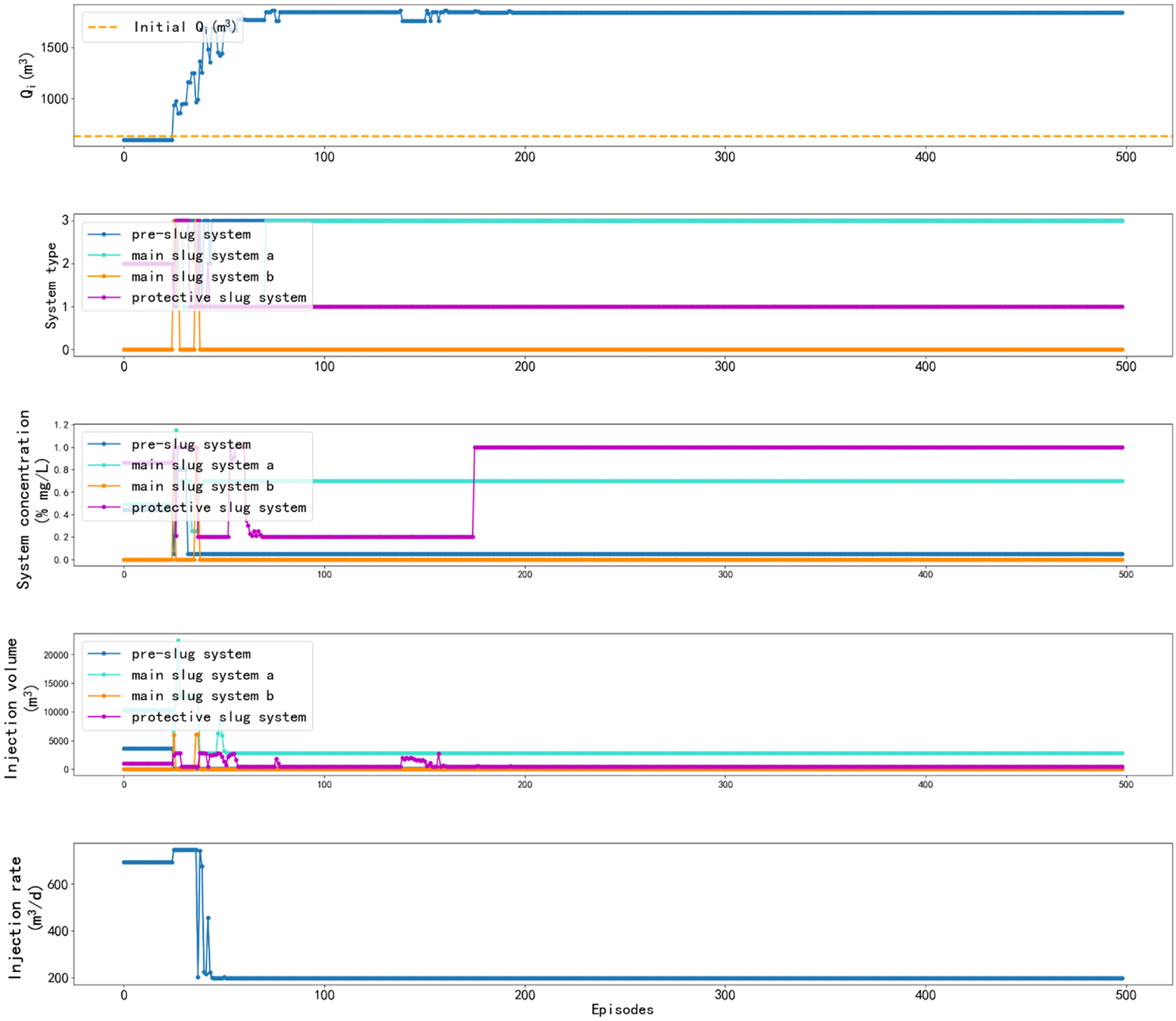


Figure 8. Change curve of various parameters during the testing of the DDPG optimization model of well_1.

Table 10. Initial and Optimized Parameters of Well_1

symbol	t_1	c_1	v_1	$t_{2\text{-a}}$	$c_{2\text{-a}}$	$v_{2\text{-a}}$	$t_{2\text{-b}}$	$c_{2\text{-b}}$	$v_{2\text{-b}}$	t_3	c_3	v_3	r	Q_i
initial	2	0.18	1800	2	0.35	3600	0	0	0	2	0.42	2100	200	631.89
optimized	3	0.05	60	3	0.7	2800	0	0	0	1	0.2	400	197.03	1840.09

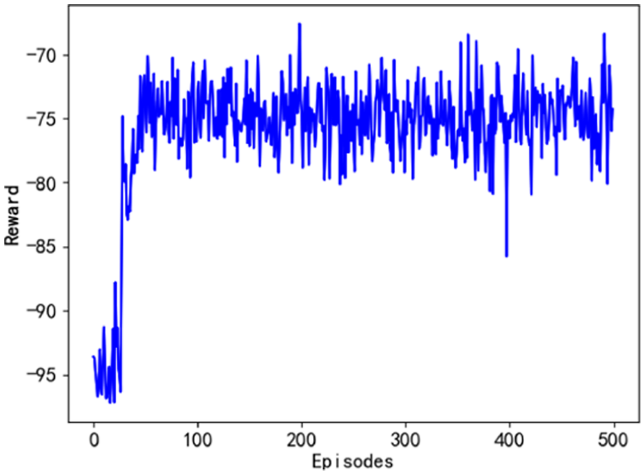


Figure 9. Change curve of reward value during the training of the DDPG optimization model of well_2.

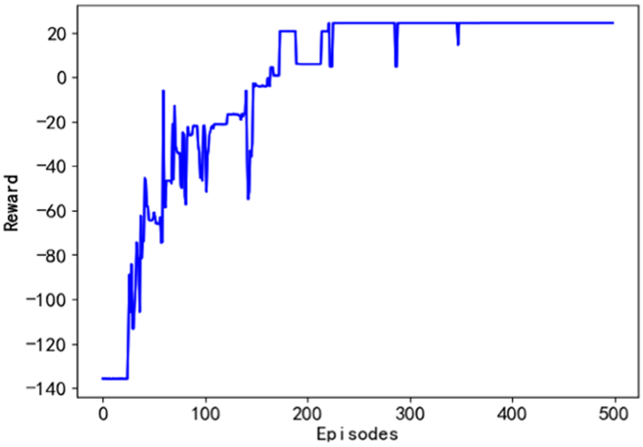


Figure 10. Change curve of reward value during the testing of the DDPG optimization model of well_2.

The change curve of various parameters during the testing of the DDPG optimization model of well_2 is shown in Figure 11, and the optimized parameters are shown in Table 11. The optimized values of each parameter are within a reasonable range, and the Q_i has increased by 218.76 m^3, an improvement of 12.3%. The type of the reservoir system has changed after optimization, indicating that the particle system is more suitable for injection under geological conditions rather than the polymer system. Although the water injection well_2 had achieved good results before optimization, the DDPG model optimization still significantly increased the Q_i, demonstrating that this model can optimize different PCOD effects of injection wells and has strong generalization ability.

In order to test the performance of the model, PSO algorithm, one of the most classical intelligent algorithms, is selected to establish PSO models for wells_1 and well_2, respectively, for case study. The essence of the PSO algorithm is that a particle exchanges experience with other particles in the population through memory, updates the existing memory, and adjusts its traveling direction to gradually approach the optimal position. It can provide a solution to the global optimization problem.[27]

The XGBoost proxy model of the PCOD scheme was used as the objective function of the PSO algorithm, which has both controllable and uncontrollable input parameters. The PSO algorithm was used to search for the maximum value of the objective function (Q_i) in order to infer the values of the controllable parameters, namely, the parameters of PCOD (system type, concentration, injection volume, and injection rate). There are a total of 23 variables in the objective function, including 10 uncontrollable variables and 13 controllable variables. In the optimization process with oil production as the objective function, the values of the uncontrollable variables are fixed for each sample, and the values of the controllable variables are inferred by optimizing the maximum value of the objective function. The constraints of the objective function are the construction intervals of the 13 controllable variables.

The objective function can be expressed as

$$\text{find } u_* \text{ as arg max } J(u) \tag{9}$$

$$u_{i,\min} \leq u \leq u_{i,\max} \tag{10}$$

where u represents the parameters of the optimization plan. The upper and lower limits of the variable u, $u_{i,\min}$ and $u_{i,\max}$, are obtained from the on-site construction constraints. The objective function J is the predicted Q_i by the XGBoost proxy model.

The internal parameters of the optimal algorithm are set as follows: the number of iterations is 500; the upper and lower limits of the velocity weight are −0.2 and 0.2, respectively; the population size is 5000; and the learning factors of the individual and social particles are 1.6 and 2, respectively. To dynamically change the movement speed of the particle swarm according to the system environment and ensure that the particle swarm can fully explore the global space in the early stage of operation and meet the needs of searching for local important areas in the later stage, a linearly decreasing inertia factor ω[28] is set in this study to modify the particle movement speed.

The fitness value change curve and parameter change curve of well_1 based on the PSO model are shown in Figures 12 and 13, respectively, and the fitness value change curve and parameter change curve of well_2 based on the PSO model are shown in Figures 14 and 15, respectively. The well_1 model began to converge at around 400 iterations, while the well_2 model started to converge at around 20 iterations, with a faster convergence rate. Compared to the DDPG optimization model, the PSO model took only 30 min to complete the optimization process, demonstrating a higher time efficiency. Tables 12 and 13 show the optimized parameters for well_1 and well_2, respectively, which fall within reasonable ranges. The well_1 was optimized using a particle system injection scheme, which differed slightly from the results of the DDPG model, but confirmed that the gel system is not suitable for this well. Although the Q_i of well_1 was improved, there is still

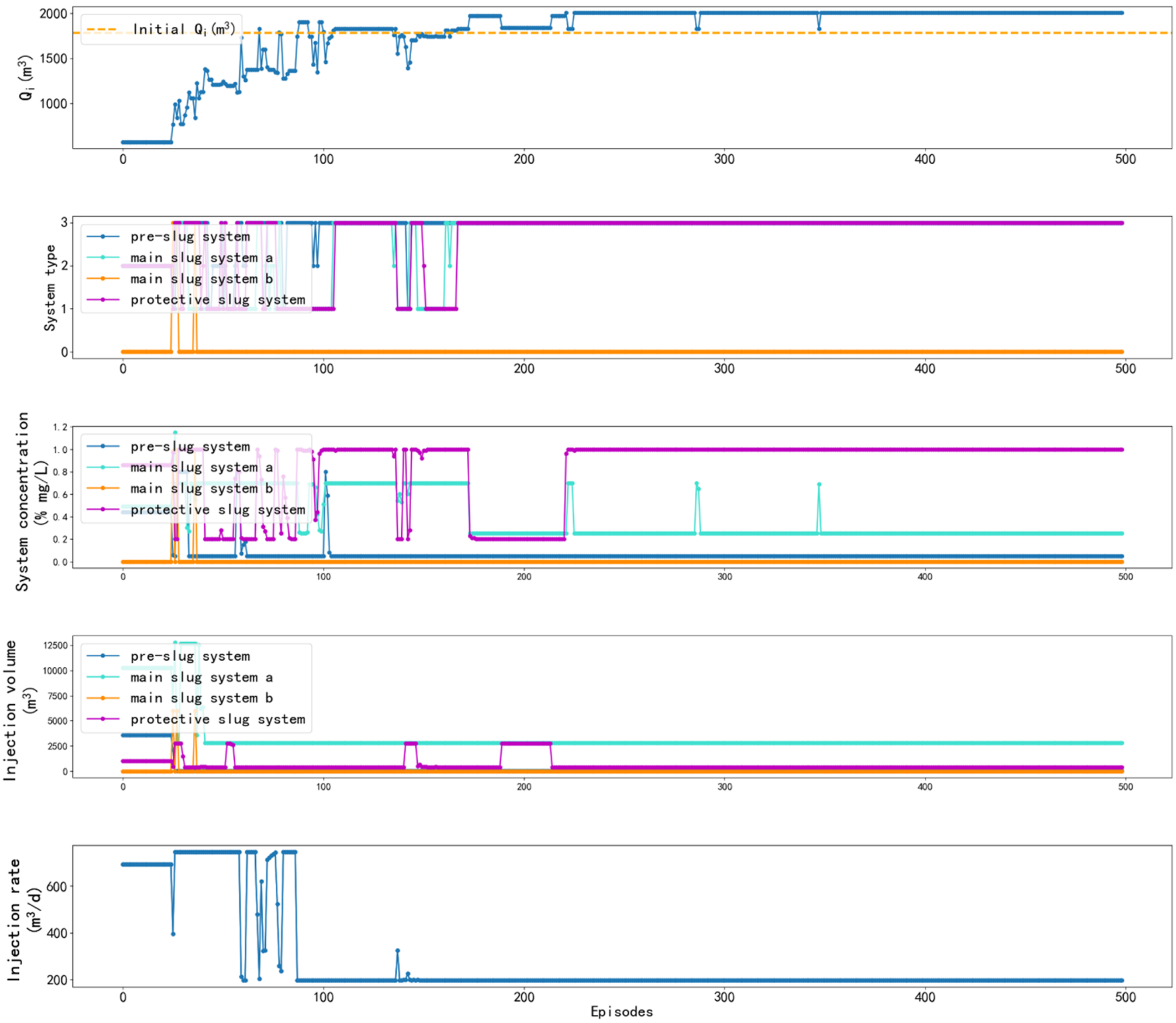


Figure 11. Change curve of various parameters during the testing of the DDPG optimization model of well_2.

Table 11. Initial and Optimized Parameters of Well_2

symbol	t_1	c_1	v_1	$t_{2\text{-a}}$	$c_{2\text{-a}}$	$v_{2\text{-a}}$	$t_{2\text{-b}}$	$c_{2\text{-b}}$	$v_{2\text{-b}}$	t_3	c_3	v_3	r	Q_i
initial	1	0.1	415	1	0.4	12661.4	0	0	0	1	0.2	17471.4	746.7	3563.0
optimized	3	0.05	60	3	0.61	2800	0	0	3	1	0.2	27709.18	197.03	6319.76

some gap compared to the DDPG model. Meanwhile, Q_i of well_2 did not improve after optimization but rather decreased compared to the pre-optimization result.

The aim of this study is to optimize the parameters for the design of pre-construction schemes for water injection wells to increase Q. While considering time efficiency, we focus more on the optimization effect of the model. PSO algorithm is a heuristic algorithm based on the concept of swarm intelligence, where each particle updates its own state by interacting with neighboring particles. However, particles can only consider their own and neighboring particle situations when updating their states, without considering the global optimal solution.[29] Therefore, if the initial position is not good or the search space is large, particles are prone to getting trapped in a local optimal solution. Although the PSO model has a faster convergence rate, the optimized effect is still inferior to the DDPG model in this paper, indicating that due to the complexity of the XGBoost proxy model, the search space is too large, leading to the PSO algorithm only being able to converge to a local optimal solution instead of a global optimal solution. At the same time, this study verified the DDPG model's ability to solve high-dimensional optimization problems, which is superior to the PSO model in optimizing different well stimulation effects in stimulation wells.

■ CONCLUSIONS

(1) The parameter optimization model of PCOD scheme established based on DDPG in this paper can greatly increase the half-year increased oil production (Q_i) of

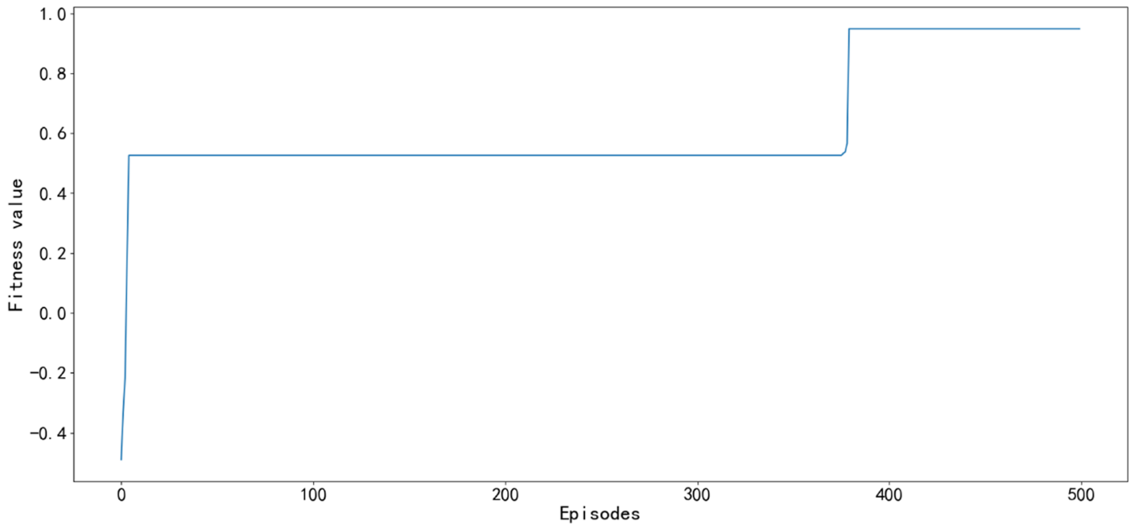


Figure 12. Change curve of fitness value of PSO optimization model of well_1.

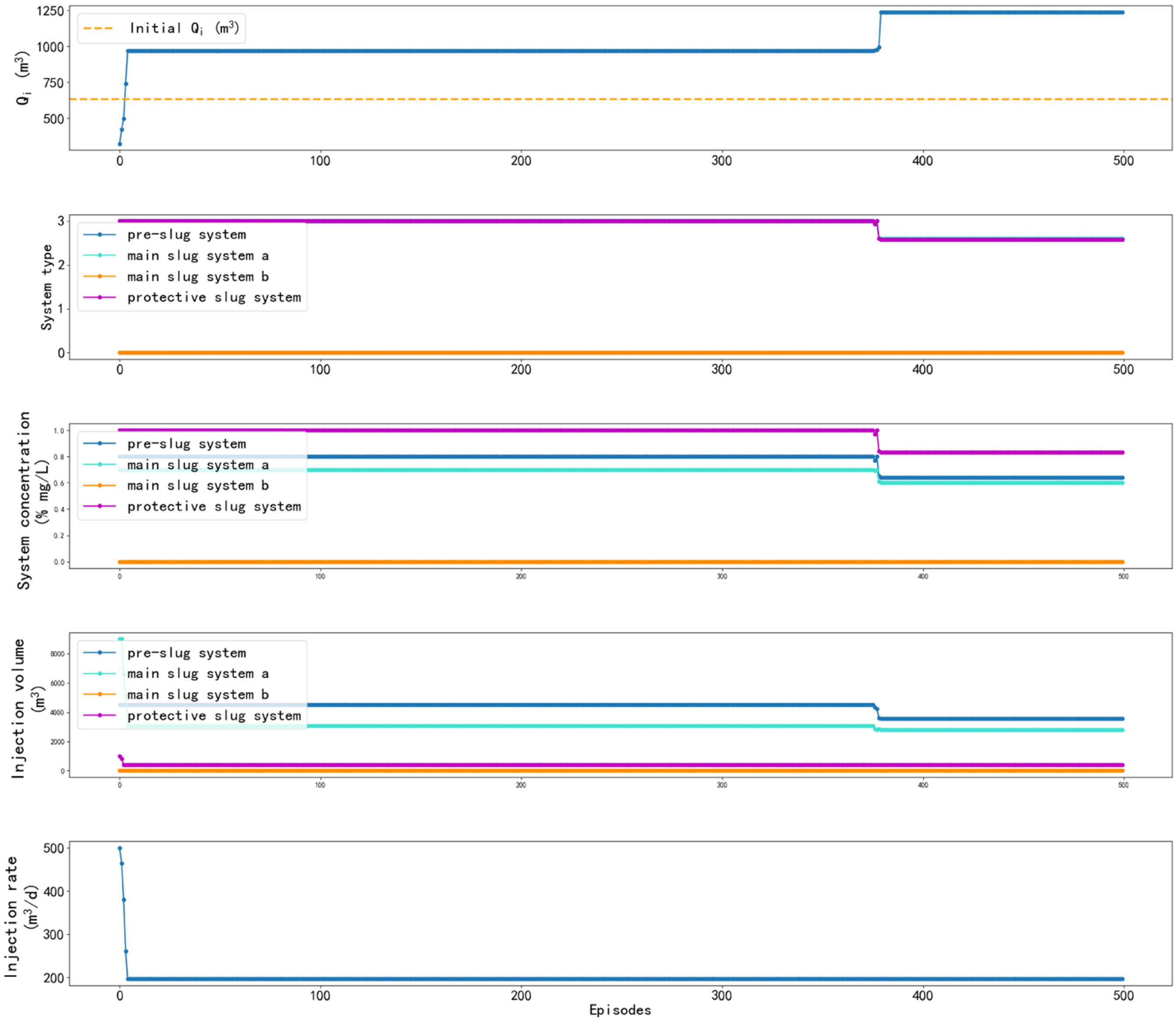


Figure 13. Change curve of parameters of PSO optimization model of well_1.

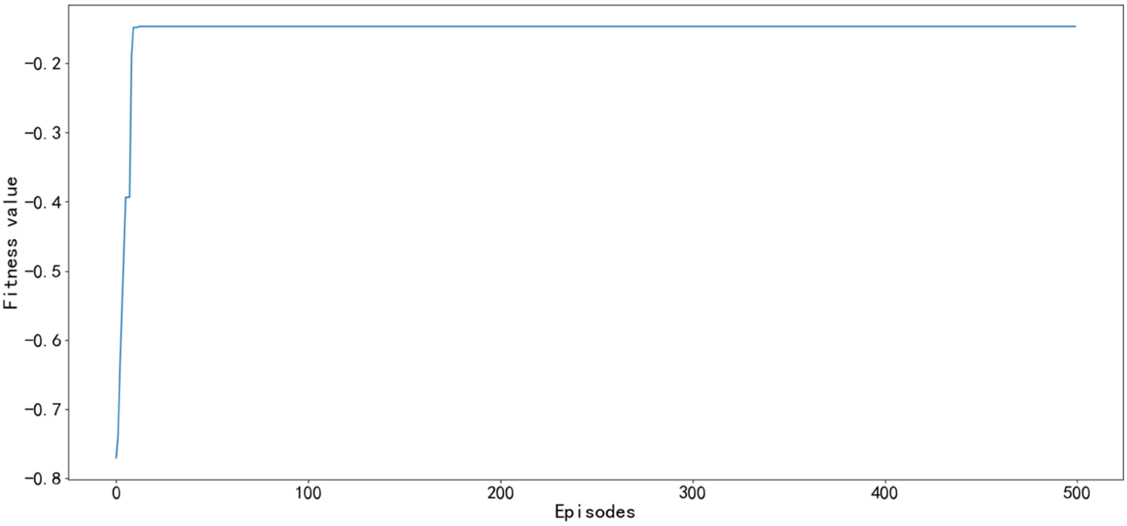


Figure 14. Change curve of fitness value of PSO optimization model of well_2.

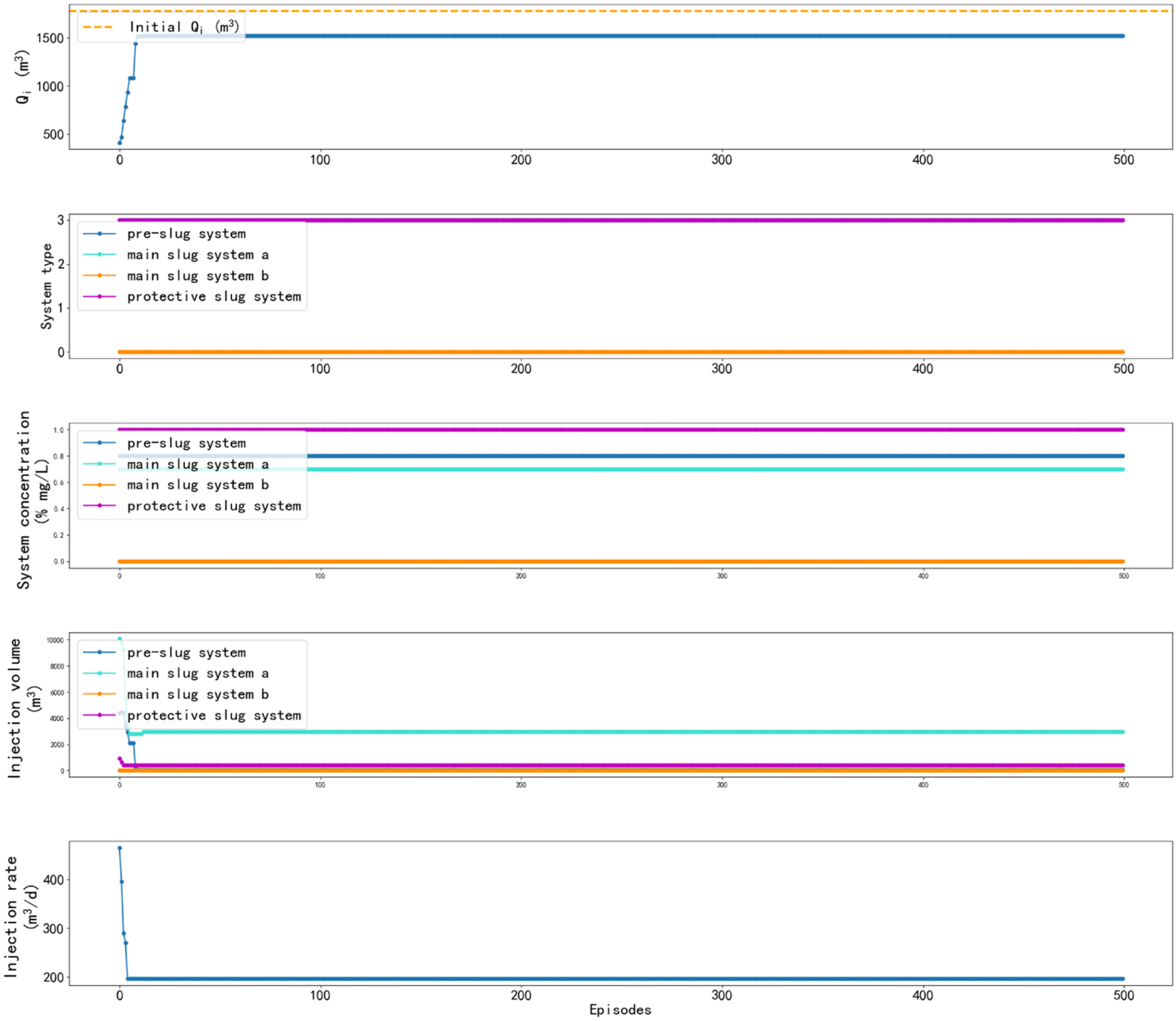


Figure 15. Change curve of parameters of PSO optimization model of well_2.

Table 12. Initial and Optimized Parameters of Well_1 of PSO

symbol	t_1	c_1	v_1	$t_{2\text{-a}}$	$c_{2\text{-a}}$	$v_{2\text{-a}}$	$t_{2\text{-b}}$	$c_{2\text{-b}}$	$v_{2\text{-b}}$	t_3	c_3	v_3	s	Q_i
initial	2	0.18	1800	2	0.35	3600	0	0	0	2	0.42	2100	200	631.89
optimized	3	0.62	3528.8	3	0.6	2873.28	0	0	0	3	0.83	400	197.03	1224.14

Table 13. Initial and Optimized Parameters of Well_2 of PSO

symbol	t_1	c_1	v_1	$t_{2\text{-a}}$	$c_{2\text{-a}}$	$v_{2\text{-a}}$	$t_{2\text{-b}}$	$c_{2\text{-b}}$	$v_{2\text{-b}}$	t_3	c_3	v_3	s	Q_i
initial	1	0.1	415	1	0.4	12661.4	0	0	0	1	0.2	747.1	746.7	1781.5
optimized	3	0.8	219.4	3	0.7	2867.14	0	0	0	3	1	468.4	197.03	1521.52

injection well group by optimizing the parameters of the PCOD scheme, which is of great significance for oil field development. At the same time, the proxy model in this paper is based on historical data and numerical simulation data, which can automatically select the optimal algorithm when applied in different blocks and has certain flexibility.

(2) Compared with the PSO model, the parameter optimization model of PCOD scheme established based on DDPG in this paper can obtain the PCOD schemes with higher Q_i for well groups with different PCOD effects and has strong optimization and generalization ability.

(3) The DDPG algorithm has more advantages for continuous action problem. In the future, it may be considered to further improve the model to realize the monitoring and optimization of the operation parameters such as the viscosity and displacement of the injection system.

AUTHOR INFORMATION

Corresponding Author

Chaodong Tan − *Department of Automation, China University of Petroleum, Changping, Beijing 102249, China; College of Petroleum Engineering, China University of Petroleum, Beijing 102249, China;* orcid.org/0000-0003-3787-1670; Phone: +86 13801331255; Email: tanchaodong@cup.edu.cn

Authors

Chunqiu Wang − *College of Petroleum Engineering, China University of Petroleum, Beijing 102249, China;* orcid.org/0000-0001-6238-3391

Jinjie Tian − *CNOOC Energy Development Co., Ltd. Engineering Technology Branch, Tianjin 300452, China*

HuiZhao Niu − *Beijing Yadan Petroleum Technology Development Co., Ltd., Beijing 102200, China*

Qi Wei − *College of Petroleum Engineering, China University of Petroleum, Beijing 102249, China*

Xiongying Zhang − *Beijing Yadan Petroleum Technology Development Co., Ltd., Beijing 102200, China*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.3c02003

Notes

The authors declare no competing financial interest.
The authors confirm that there are no known conflicts of interest associated with this publication.
All of the authors listed in the paper agree to sign their names. They have read the paper and agree to contribute to publication. The submission of this article follows the journal policies and regulations.

ACKNOWLEDGMENTS

This work was supported by the following two fund projects: National Natural Science Foundation of China “Integrated Optimization of Large-scale Non-pipelined Well Production and Tank Truck Scheduling Based on Data Analytics” (No. 51974327); Major special project of CNOOC Energy Development Co., Ltd. “Intelligent Technology Research for Injection and Production well” (NO. HFXMLZ-GJ2018-14).

REFERENCES

(1) Lane, R. H.; Sanders, G. S. In *Water Shutoff Through Fullbore Placement of Polymer Gel in Faulted and in Hydraulically Fractured Producers of the Prudhoe Bay Field*, SPE Production Operations Symposium; OnePetro: Oklahoma City, USA, 1995.

(2) Chunming, X.; Xiaofen, T. Technologies of water shut-off and profile control: An overview. *Pet. Explor. Dev.* **2007**, *34*, 83−88.

(3) Liu, D.; Hu, T. H.; Pan, G. M.; et al. Forecasting Model for Profile Control and EOR in Heavy Oil Reservoirs by Using Weak Gel. *Spec. Oil Gas Reservoirs* **2018**, *25*, 103−108.

(4) Tsau, J. -S.; Liang, J. -T.; Hill, A. D.; Sepehrnoori, K. Re-Formation of Xanthan/Chromium Gels After Shear Degradation. *SPE Reservoir Eng.* **1992**, *7*, 21−28.

(5) Lu, X.; Cao, B.; Xie, K.; et al. EOR mechanisms of polymer flooding in a heterogeneous oil reservoir. *Pet. Explor. Dev.* **2021**, *48*, 148−155.

(6) Aldhaheri, M.; Mingzhen, Wei.; Na, Zhang.; et al. A Review of Field Oil-Production Response of Injection-Well Gel Treatments. *SPE Reservoir Eval. Eng.* **2019**, *22*, 597−611.

(7) Jia, Y.; Zheng, M.; Yang, H.; et al. Optimization of Operational Parameters for Deep Displacement Involving Polymer Microspheres in Low Permeability Reservoirs of the Changqing Oilfield. *Pet. Drill. Tech.* **2018**, *46*, 75−82.

(8) Xiao, K.; Mu, L.; Wu, X.; Yang, L.; Xu, F.; Zhu, G.; Zhu, Y. In *Comprehensive Study of Polymer Gel Profile Control for Wag Process in Fractured Reservoir: Using Experimental and Numerical Simulation*, SPE EOR Conference at Oil and Gas West Asia; OnePetro: Muscat, Oman, 2016.

(9) Sun, Q.; Ertekin, T.; Zhang, M.; et al. A comprehensive techno-economic assessment of alkali−surfactant−polymer flooding processes using data-driven approaches. *Energy Rep.* **2021**, *7*, 2681−2702.

(10) Sun, Q.; Ertekin, T. Screening and optimization of polymer flooding projects using artificial-neural-network (ANN) based proxies. *J. Pet. Sci. Eng.* **2020**, *185*, No. 106617.

(11) Shi, H.; Shi, L.; Xu, M.; et al. End-to-end navigation strategy with deep reinforcement learning for mobile robots. *IEEE Trans. Ind. Inf.* **2020**, *16*, 2393−2402.

(12) Zhao, X. Y.; Xia, L.; Zhang, L.et al. In *Deep Reinforcement Learning for Page-wise Recommendations*, Proceedings of the 12th ACM Conference on Recommender Systems; ACM, 2018; pp 95−103.

(13) He, J.; Tang, M.; Hu, C.; et al. Deep Reinforcement Learning for Generalizable Field Development Optimization. *SPE J.* **2021**, *27*, 226−245.

(14) Shi, J.; Han, Q.; Ren, X.et al. In *The Application of Big Data Analysis in the Optimizing and Selecting Artificial Lift Methods*, SPE Middle East Artificial Lift Conference and Exhibition; OnePetro: Manama, Bahrain, 2018.

(15) Talavera, A. G.; Tupac, Y. J.; Vellasco, M. M. In *Controlling Oil Production in Smart Wells by MPC Strategy with Reinforcement Learning*, SPE Latin American and Caribbean Petroleum Engineering Conference; OnePetro: Lima, Peru, 2010.

(16) Miftakhov, R.; Al-Qasim, A.; Efremov, I. In *Deep Reinforcement Learning: Reservoir Optimization from Pixels*, International Petroleum Technology Conference; OnePetro: Dhahran, Kingdom of Saudi Arabia, 2020.

(17) Bhowmik, S. In *Machine Learning-Based Optimization for Subsea Pipeline Route Design*, Offshore Technology Conference; OnePetro: Texas, 2021.

(18) Pollock, J.; Stoecker-Sylvia, Z.; Veedu, V.et al. In *Machine Learning for Improved Directional Drilling*, Offshore Technology Conference; OnePetro: Texas, USA, 2018.

(19) Saini, G. S.; Ashok, P.; Oort, E. V. In *Predictive Action Planning for Hole Cleaning Optimization and Stuck Pipe Prevention Using Digital Twinning and Reinforcement Learning*, IADC/SPE International Drilling Conference and Exhibition; OnePetro: Texas, USA, 2020.

(20) Prada, A.; Civan, F.; Dalrymple, E. D. In *Evaluation of Gelation Systems for Conformance Control*, SPE/DOE Improved Oil Recovery Symposium; OnePetro: Tulsa, Oklahoma, 2000.

(21) Ma, C.; Xie, W.; Sun, W. Research on Reinforcement Learning Technology: A Review. *Command Control Simul.* **2018**, *B40*, 68−72.

(22) Rummery, G. A.; Niranjan, M. *On-line Q-learning Using Connectionist Systems*; University of Cambridge: Cambridge, 1994.

(23) Watkins, C. J. C. H.; Daya, P. Q-learning. *Mach. Learn.* **1992**, *8*, 279−292.

(24) Li, R.; Peng, H.; Li, R.; et al. Overview on Algorithms and Applications for Reinforcement Learning. *Comput. Syst. Appl.* **2020**, *29*, 13−25.

(25) Konda, V. R.; Tsitsiklis, J. N. On actor-critic algorithms. *SIAM J. Control Optim.* **2003**, *42*, 1143−1166.

(26) Lillicrap, T. P.; Hunt, J. J.; Pritzel, A.et al. In *Continuous Control with Deep Reinforcement Learning*, 4th International Conference on Learning Representations, 2016.

(27) Kennedy, J.; Eberhart, R. In *Particle Swarm Optimization*, Proceedings of ICNN'95 - International Conference on Neural Networks; IEEE, 1995.

(28) Yuhui, Shi.; Eberhart, R. C. *Parameter Selection in Particle Swarm Optimization*, Proceedings of the 7th International Conference on Computation Programming VII; Springer: London, U.K., 1998.

(29) Sengupta, S.; Basak, S.; Peters, R. A. Particle swarm optimization: a survey of historical and recent developments with hybridization perspectives. *Mach. Learn. Knowl. Extr.* **2018**, *1*, 157−191.