Contents lists available at ScienceDirect

Nuclear Engineering and Technology

journal homepage: www.elsevier.com/locate/net

Original Article

Reinforcement learning-based control with application to the once-through steam generator system

Cheng Li, Ren Yu*, Wenmin Yu, Tianshu Wang

Naval University of Engineering, Wuhan, 430033, China

ARTICLE INFO

Article history:
Received 18 May 2022
Received in revised form
31 May 2023
Accepted 1 June 2023
Available online 21 June 2023

Keywords:
LSTM
PPO algorithm
Pretraining
Once-through steam generator

ABSTRACT

A reinforcement learning framework is proposed for the control problem of outlet steam pressure of the once-through steam generator(OTSG) in this paper. The double-layer controller using Proximal Policy Optimization(PPO) algorithm is applied in the control structure of the OTSG. The PPO algorithm can train the neural networks continuously according to the process of interaction with the environment and then the trained controller can realize better control for the OTSG. Meanwhile, reinforcement learning has the characteristic of difficult application in real-world objects, this paper proposes an innovative pretraining method to solve this problem. The difficulty in the application of reinforcement learning lies in training. The optimal strategy of each step is summed up through trial and error, and the training cost is very high. In this paper, the LSTM model is adopted as the training environment for pretraining, which saves training time and improves efficiency. The experimental results show that this method can realize the self-adjustment of control parameters under various working conditions, and the control effect has the advantages of small overshoot, fast stabilization speed, and strong adaptive ability.

1. Introduction

The once-through steam generator(OTSG) is the hub of the primary and secondary loop, which has the characteristics of small volume and strong heat transfer ability. The compact structure is suitable for an integrated layout [1]. For the casing OTSG, the length of the single-phase section in the heat transfer tube is shortened in the low load operation condition, and the secondary side has the risk of flow instability, which will seriously affect the operation safety of OTSG. Because of the characteristics of strong coupling, the control of the OTSG's outlet steam pressure is difficult [2]. To control the outlet steam pressure, Zhang Yue proposed a structure using the PID method to adjust the secondary feedwater flow rate [3]. A new kind of artificial immune algorithm is applied to the control of the OTSG, the result shows that the algorithm can improve the dynamic characteristics of OTSG [4]. Literature [5] proposed a control scheme based on T-S fuzzy neural to the OTSG's feedwater control system. The system uses a three-impulse control method [6] to control the main feedwater valve which can contribute to controlling the steam pressure of the OTSG.

Reinforcement learning(RL) is to learn experiences through the interaction between the agent and the environment to obtain reward feedback, and finally explore the optimal strategy. It is core guiding principle is to maximize return [7,8]. After Google's AlphaGo defeated the world's top Go player Lee Sedol [9], deep reinforcement learning (DRL) has become a hot research topic again, and more and more research has been conducted on it.

So far, DRL has been widely used in robotics, energy, heating, ventilation, air conditioning(HVAC), unmanned aerial vehicle(UAV), and so on. To solve the job shop scheduling problems in the environment of resource preemption, Wang Xiaohan [10] used a multi-agent RL method to learn the scheduling strategies. Deng Xiangtian [11] proposes a novel non-stationary DQN method for the control of the HVAC. The proposed DQN method reduces unnecessary energy consumption and is superior to existing DQN methods in multi-zone control tasks. The moving UAV's attitude control has strong nonlinear and coupling characteristics. To solve this problem, Qiu Xiaoqi [12] proposed a DRL attitude controller to realize end-to-end control. Grando Ricardo Bedin used the Twin Delayed Deep Deterministic Policy Gradient (TD3) and Soft Actor-Critic (SAC) models to solve the 3D mapless navigation for UAVs [13]. Zhang Rong proposed an RL algorithm to optimize the task sequence for the human-robot collaborative [14]. JaeKwan Park adopts a DRL method to state diagnose the

* Corresponding author.
E-mail address: 296379210@qq.com (R. Yu).

https://doi.org/10.1016/j.net.2023.06.001

safety function status when monitoring the safety functions of nuclear facilities [15].

Based on the above discussion, the RL is a promising technology that can solve complex problems, and especially plays an important role in the control system. Since there is little research on RL in the field of nuclear energy, this paper focuses on the feasibility of applying reinforcement learning to practical OTSG control problems. In the process of reinforcement learning development, PPO(Proximal Policy Optimization) can be regarded as one of the most classic algorithms, which mainly benefits from its three obvious advantages: the algorithm is simple and easy to understand, has strong adaptability, and has excellent effect. It is a random policy search method. Its learning process is to obtain the reward values of the environment feedback and find the optimal strategy in the continuous interaction process. In order to apply reinforcement learning algorithm to practical control problems, this paper puts forward an effective approach that the algorithm should be pre-trained on the environment model and then deploy to the actual target, so as to realize the application of the method to the actual environment and achieve better results quickly.

2. Problem statement and preliminaries

2.1. System introduction

This paper regards the integrated pressurized water reactor(IPWR) as the research object. The reactor core and the OTSG are installed on the pressure vessel as shown in Fig. 1. The primary pump is placed on the pipe connected to the pressure vessel.

OTSG is the key heat exchange component between the primary and secondary loop in the pressurized water reactor. As can be seen from Fig. 1 below, the reactor coolant is driven by the primary pump and flows down into the lower chamber of the reactor from the connection pipe of the primary pump to the pressure vessel. And then it bends upward and flows into the core, carrying away the heat generated by the core's self-sustaining chain fission reaction. The coolant flowing from the core continues to flow upward into the annular compartment of the OTSG, and then enters the connection pipe of the primary pump after heat exchange with the feedwater of the secondary side. The main function of the feed water is to exchange heat with the primary coolant in the OTSG. After absorbing heat, the feed water becomes a steam-water mixture. The saturated steam from the OTSG eventually flows into the turbine and drives it to generate electricity.

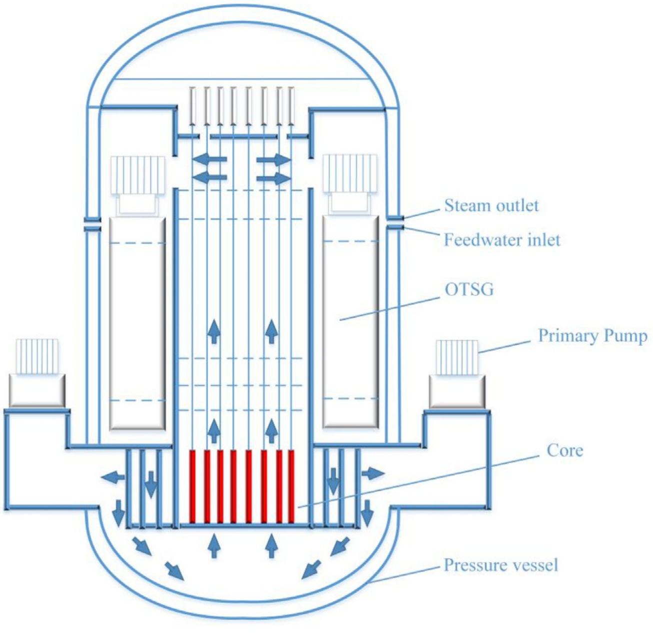

Fig. 1. Schematic diagram of an IPWR.

2.2. Problem statement

This paper focuses on the control of OTSG. In the process of tracking control, the main principle is to keep the steam pressure constant. The feedwater control system monitors the steam pressure and adjusts the feedwater flow according to the steam pressure control strategy. The problem is transformed into the control research of the feedwater pressure and flow of the OTSG, which aims to ensure the smooth change of operation parameters in conjunction with the primary system in the star-stop process, reducing the thermal impact on the OTSG.

So the overall goal of control is to keep the outlet steam pressure stable when there is a bounded disturbance, that is,

$$\lim_{t\to\infty}(p(t)-p_d)=0$$

Where $p(t)$ is the steam pressure, p_d is the target pressure.

3. System modeling

In order to reduce the training time of the agent and the training cost of the reinforcement learning algorithm, first of all, we need to build the model of the IPWR as the environment for interacting with the agent. It is difficult to use the traditional mechanism modeling method because of the nonlinear relationship between the state parameters and the control parameters of the IPWR. Therefore, researchers have proposed a variety of data-driven modeling methods, such as data mining algorithms (artificial neural network(ANN), statistical models (regression), geometric models, and random models (probability density function approximation). In the above modeling methods, the neural network algorithm does not need the complicated modeling process with high accuracy, and it has more advantages in the modeling of nonlinear systems. Therefore, this paper uses a neural network to fit the IPWR. The establishment of the neural network model requires reasonable parameters to improve the comprehensibility, extensibility, and accuracy of the model.

3.1. Selection of parameters for the system model

The IPWR has a large number of strongly coupled process variables, and the variables in its distributed control system(DCS) even reach more than 100,000. It is unnecessary to extract all the variables for modeling. The control goal is to keep the pressure of the OSTG stable, and also facilitate a better evaluation of the model, we only need to extract some variables that affect the pressure of the OSTG and represent the real-time state of the IPWR.

In the process of power operation, the second loop adopts the control scheme of keeping the steam outlet pressure constant. Under normal operating conditions, the feedwater flow control system puts into automatic operation by the regulating valve and the feedwater pump to ensure the supply of feedwater flow. The feedwater control system monitors the steam pressure and adjusts the feedwater flow according to the steam pressure control strategy. The regulating valve is controlled by a PI controller. The main feedwater regulating valve adopts three impulse control scheme by steam pressure, feedwater flow, and steam flow, through the comparison of the measured steam pressure and set value, and the comparison of feedwater flow and steam flow, the control signal is generated to adjust the valve opening, the purpose is to ensure that the feedwater flow is matched with the demand load. The speed of the feedwater pump is controlled by the PI controller which produces a control signal to adjust the speed of the pump and changes the head of the water pump so that the differential pressure between the feedwater valve front and back keeps stable. Therefore, the position of the feedwater valve, the speed of the feedwater pump, the control parameters K_p and K_i of the PI controller for the feedwater pump, and the K_p and K_i of the PI controller for the feedwater valve are selected as the input parameters. Since nuclear power can represent the entire system state, it is also chosen as the model input. The output parameters are the outlet pressure of the OSTG, the differential pressure between the feed water valve front and back, and the average temperature of the primary loop.

3.2. Data acquisition and preprocessing

In this paper, the data from an IPWR simulator are selected as the training sample sets of the model. The main parameters are shown in Table 1. It includes 25000 sample data which contains a training data set (20000 samples) and test data set (5000 samples). Sample data is collected once every 0.25s, and the collection conditions are mainly normal power operation and transient accident operation.

To establish the model, the training data should be standardized first to eliminate the difference between data features and facilitate the calculation and convergence of the model. Due to the different dimensions of input variables and the large difference in the size and range of data values, the training speed of the network will be slow and even can not converge. Therefore, the training data are normalized and de-normalized. The linear function conversion method [16] is adopted to convert data into values within the range of 0–1.

3.3. Neural network structure

The long-term and short-term memory network(LSTM) is a special form of recurrent neural network (RNN), which is improved on the basis of RNN [17]. Compared with the ANN, the RNN can apply the relevant information of historical data to the prediction. The error backpropagation algorithm of RNN is just as simple as that of ANN, but it also has the problem of gradient explosion and gradient disappearance. To solve these two problems, the researchers have developed a neural network model with short and long-term memory [18], as shown in Fig. 2. Different from the RNN, the most obvious improvement of LSTM is that its structure consists of a cell state C and three gates, namely the forget gate f, the output gate o, and the input gate i.

At the time of updating weights through the error backpropagation, some errors can directly enter the input gate and then pass to the neurons of the next layer, some errors will be forgotten by the forget gate, thus solving the problem of the gradient explosion and disappearance, which effectively deal with the redundancy issues of relevant information in historical data [19,20]. The problem studied in this paper is a typical time series problem, so the LSTM algorithm is chosen for the IPWR prediction.Where, X_t represents the input at t, h_{t-1} represents the output of LSTM at t-1, and C_{t-1} represents the memory at t-1. σ represents sigmoid activation function.

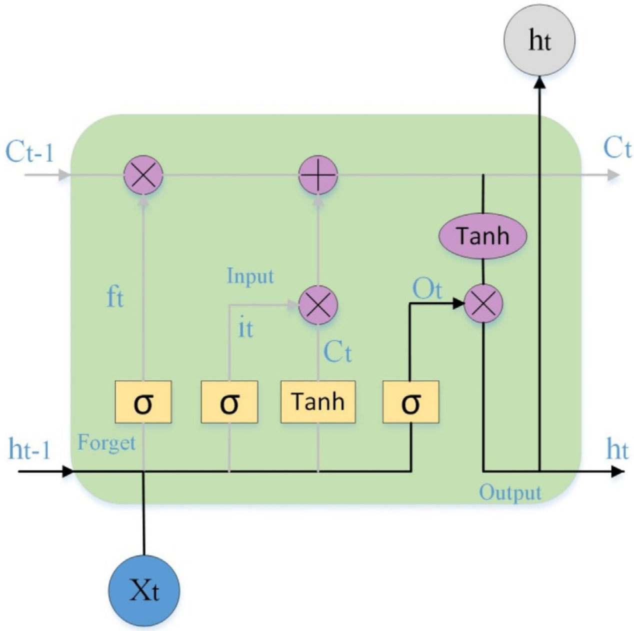

Fig. 2. Schematic diagram of the LSTM.

Forget gate f determines how much cell state Ct-1 is retained from cell state ct at time t.

The first step of LSTM network is to determine what old information should be forgotten from the previous cell state. Forget gate f_t determines how much cell state Ct-1 at time t-1 is retained to the cell state Ct at time t. b_f and W_f are the offset and input weight of forget gate f_t, respectively. The specific expression of f_t is as follows:

$$f_t = \sigma\left(W_f \bullet (X_t, h_{t-1}) + b_f\right) \tag{1}$$

The second step is to determine what new information should be entered for the current cell state. First, X_t and h_{t-1} are used in the input gate i_t to determine the cell information. Then X_t and h_{t-1} obtain new candidate cell information $\tilde{C}_t$ through tanh. b_i and W_i are the bias items and input weights of the input gate i_t, and b_c and W_c are the bias items and input weights of candidate cell states $\tilde{C}_t$, respectively. The specific calculation method is:

$$i_t = \sigma(W_i \bullet (X_t, h_{t-1}) + b_i) \tag{2}$$

$$\tilde{C}_t = \tanh(W_c \bullet (X_t, h_{t-1}) + b_c) \tag{3}$$

The third step is the update of the current cell state. The $f_t \bullet C_{t-1}$ indicates forgotten information. According to the above calculation, C_t cell status update value at time t can be calculated. The specific calculation method is as follows:

$$C_t = f_t \bullet C_{t-1} + i_t \bullet \tilde{C}_t \tag{4}$$

Table 1
Main design parameters of an IPWR.

Parameter	Nuclear power(%FP)	Turbine power(%FP)	Average temperature of the primary loop(°C)	Steam generator pressure(Mpa)	Steam flow(kg/s)	Feedwater flow(kg/s)	Folding step rods Position(step)	Pressurizer Pressure(Mpa)	Pressurizer level(m)
Value	100.363	100.119	300.76	4.538	600.656	599.802	602	15.503	1.965

After calculating the cell status update value, the final step is to calculate the output gate O_t value. b_o and W_o are the offset items and input weights of output gate O_t respectively, and the final output result is determined by O_t and C_t. The calculation formula is:

$$O_t = \sigma(W_o \bullet (X_t, h_{t-1}) + b_o) \tag{5}$$

$$h_t = O_t \bullet \tanh(C_t) \tag{6}$$

3.4. Analysis of the model performance

Different model parameter of LSTM has obvious effects on the model's predictive power, such as the number of neurons layer and the number of neurons in each layer can make the model's computation grows exponentially and affect the accuracy. The change in the learning rate will significantly affect the efficiency and accuracy of the model during training [21].

Therefore, different hyperparameters are set in this experiment. Through experimental measurement, when the number of neurons in the hidden layer is 8 and the number of iterations is 100, the model training effect is the best and the error of mean square root(RMSE) is the smallest. The comparison results are shown in Table 2.

In the current research paper about the LSTM neural network model, the model has the highest efficiency when the number of neuron layers is 2 or 3. The number of neurons in the hidden layer is 8, and the learning rate is usually selected as 0.001. In this paper, the learning rate of 0.005 is set as a comparison, and the number of iterations is set as 150 and 300. The activation function is the tanh function. Through the results of the experiment, we can determine the most effective hyperparameters.

The mean square error function (MSE), root mean square error (RMSE), and mean absolute error (MAE) are used to evaluate the model. Table 3 shows the experimental results under different parameters.

According to the results in Table 3, the MSE result is the smallest 0.0005 when the iteration number is 300, the activation function is the tanh, the learning rate is 0.001, and the neural layers are 2 with 8 hidden neurons. When the iteration number is 150, the activation function is the tanh, the learning rate is 0.001, the neural layers are 3 with 8 hidden neurons, and the RMSE and MAE are both the smallest. It is worth mentioning that the MSE value is also very respectable, only slightly higher than the minimum 0.0005 by 0.0002. Therefore, the hyperparameters of the LSTM model are selected as the iteration number is 150, the activation function is the tanh, the learning rate is 0.001, and the neural layers are 3 with 8 hidden neurons in this paper.

4. DRL controller design for OTSG

4.1. Background

RL algorithms are mainly divided into value function-based methods and policy gradient-based methods. Traditional RL is usually based on the value function method. Such as the q-learning algorithm, but when the state and action space is high-dimensional continuous, the Q value table cannot be stored. Google Deepmind team proposed the Deep Q-Network(DQN) algorithm [22], which can change the updating problem of the Q value table into a function fitting problem by the neural network. Then, the Deterministic Policy Gradient(DPG) is proved, and the Deep Deterministic Policy Gradient (DDPG) algorithm is proposed to realize the learning of continuous actions [23].

Table 2
The RMSE of the LSTM model under different conditions.

The number of neurons in the hidden layer	RMSE
6	0.0131
7	0.0137
8	0.0127
9	0.0150
10	0.0133
11	0.0146

Different from the value function-based methods, the policy gradient-based methods directly output the specific actions and directly modify policy parameters through gradient rise using the environmental feedback. Williams et al. proposed the RL algorithm which updates the policy gradient through the Monte Carlo sampling, but the state-value function guiding the update direction is obtained through backtracking, which has low sampling efficiency [24]. Schulman et al. proposed the Trust Region Policy Optimization(TRPO) method, which limited the updated range of the output, but affected the implementation efficiency and update speed [25]. Therefore, The OPENAI team proposed the PPO algorithm to achieve a more concise update method. PPO algorithm has a good performance for continuous control problems. Based on Markov Decision Process(MDP), it continues the step selection mechanism of the policy optimization algorithm, draws on the idea of the policy-based estimation, and inherits the experience of the dual network of policy and value in actor-critic methods [26].

4.2. Design of the controller based on PPO for OTSG

The function of the feedwater flow control system is to supply and adjust feedwater to the OTSG so that the feedwater of the OTSG and the load of the secondary loop are compatible. The speed control system of the feedwater pump compares the measured value of the differential pressure between the feedwater valve front and back with the setpoint, and then the control signal is generated to adjust the speed of the feedwater pump which changes the head of the pump so that the differential pressure between the feedwater valve front and back maintains around the setpoint. The feedwater regulating valve adopts the three-element control system of the steam pressure, feedwater flow, and steam flow, which compares the steam pressure and setpoint at first, and then comparing with the differential between the feedwater flow and steam flow, the control signal is generated to adjust the valve, so as to ensure the water supply flow rate and the demand load match.

According to the control strategy of IPWR, this paper innovatively adopts the PPO algorithm to control the OTSG of IPWR, as shown in Fig. 3. The IPWR is the environment of the PPO algorithm. The agent learns through repeated interaction with the environment. In each interaction, the agent takes some action to influence the environment. The purpose of RL is to obtain the optimal cumulative reward through constant interaction with the environment.

The specific structure of the controller is a double layer(Fig. 4), the bottom layer is consist of the PI controllers, which control the feedwater valves and pumps to keep the steam pressure stable. The layer of the PPO agent of the controller can realize the online self-adjustment of the control parameters of the PI controller after training.

The structure of the PPO algorithm is consist of three neural networks, namely the actor-old network, actor-new network, and critic network. As can be seen from Fig. 4, after receiving the environment state (s), the actor-new network outputs an action(a) through neural network calculation, which acts on the parameter adjustment of PI controller. After the environment (IPWR) status is

Table 3
Prediction results of the LSTM model under different parameters.

The number of iterations	The activation function	The learning rate	The number of neuron layer	The Number of neurons	MSE	RMSE	MAE
150	tanh	0.001	3	8	0.0007	0.0283	0.2101
300	tanh	0.001	3	8	0.0012	0.0629	0.2121
150	tanh	0.005	3	8	0.0071	0.0511	0.3365
300	tanh	0.005	3	8	0.0019	0.0423	0.2308
150	tanh	0.001	2	8	0.0017	0.0523	0.2582
300	tanh	0.001	2	8	0.0005	0.0629	0.2130
150	tanh	0.005	2	8	0.0006	0.0584	0.2345
300	tanh	0.005	2	8	0.0016	0.0323	0.2714

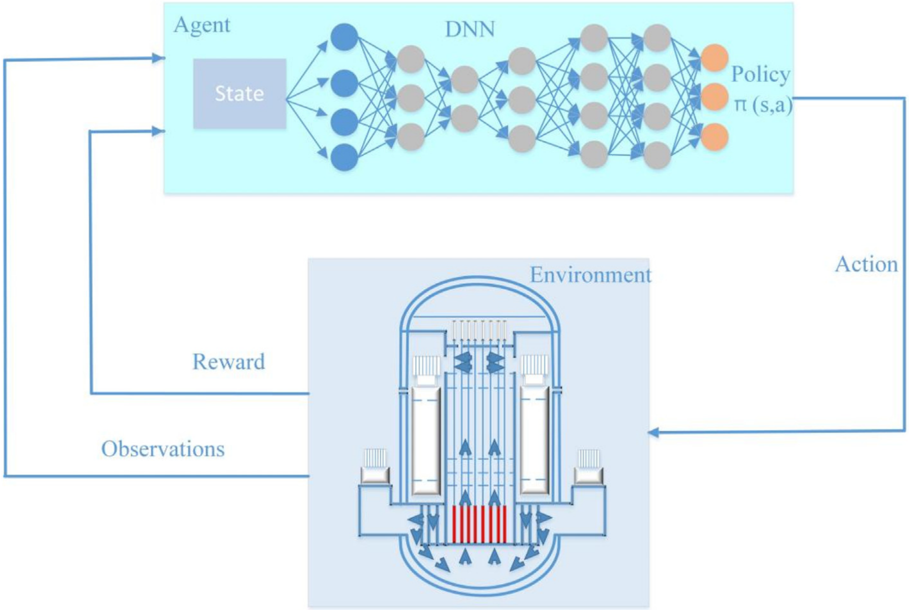

Fig. 3. Controller based on PPO of an IPWR.

adjusted, it becomes the next environment status (s_). The algorithm also calculates the reward(r) for the interaction. And what the actor-old network does is store the parameters of the neural network of the actor-new. To prevent the update steps from being too large, the PPO algorithm copies parameters from the actor-old network to the actor-new network before each batch size step. The critic network is is responsible for the state value function(v) calculation, which means the cumulative discount rewards when performing the current policy. The replay buffer stores historical experiences and then feeds the random samples to train and update actor and critic networks.

4.2.1. *Design of state-space*

In the process of power operation, the primary loop of IPWR adopts the control scheme of constant average temperature, and the secondary loop adopts the scheme of constant main steam pressure.

Therefore, according to the dynamic characteristic of the OTSG, this article selects state-spaces including the outlet pressure, the differential pressure between the feedwater valve front and back, and the average temperature of the primary loop.

4.2.2. *Design of action space*

The action space is the type and range of the output of the algorithm. In this paper, the PI control parameter is selected as the output action of the PPO algorithm, and the dimension is 4 which includes the PI parameters of the feedwater pump and the regulating valve, that is action = [K_{p1},K_{i1}, K_{p2},K_{i2}] and the range of the set points are K_{p1} = [0,10], K_{i1} = [0,10].

4.2.3. *Design of reward function*

The reward function is the core of the RL algorithms, which determines whether the training of the agent can be successful. A good reward function can guide the agent to quickly learn from the interaction process, while a bad reward function often leads to training failure.

In order to make the agent learn the control strategy effectively, a piecewise reward function is constructed. First, the relative error functions $r_{v,}$ r_p, and r_t are defined:

$$r_v = \frac{sv - y_v(t)}{sv}, r_p = \frac{sp - y_p(t)}{sp}, r_t = \frac{st - y_t(t)}{st} \tag{7}$$

Where, sv, sp, and st are the target set values, y_v is the measured value of the main steam pressure, y_p is the differential pressure between the feed water valve front and back, and y_t is the measured value of the average temperature of the primary loop.

According to the different ranges of absolute relative error, $r > 200\%$ is designated as the abnormal area, $r > 15\%$ as the large error area, and $r \leq 15\%$ as the low error area. In the abnormal area, if the system deviates too far from the target value, a large penalty value will be given. In the large error zone, the reward value is uniformly set as −1. In the low error zone, and the system state-space value is close to the target value, the reward value is positive.

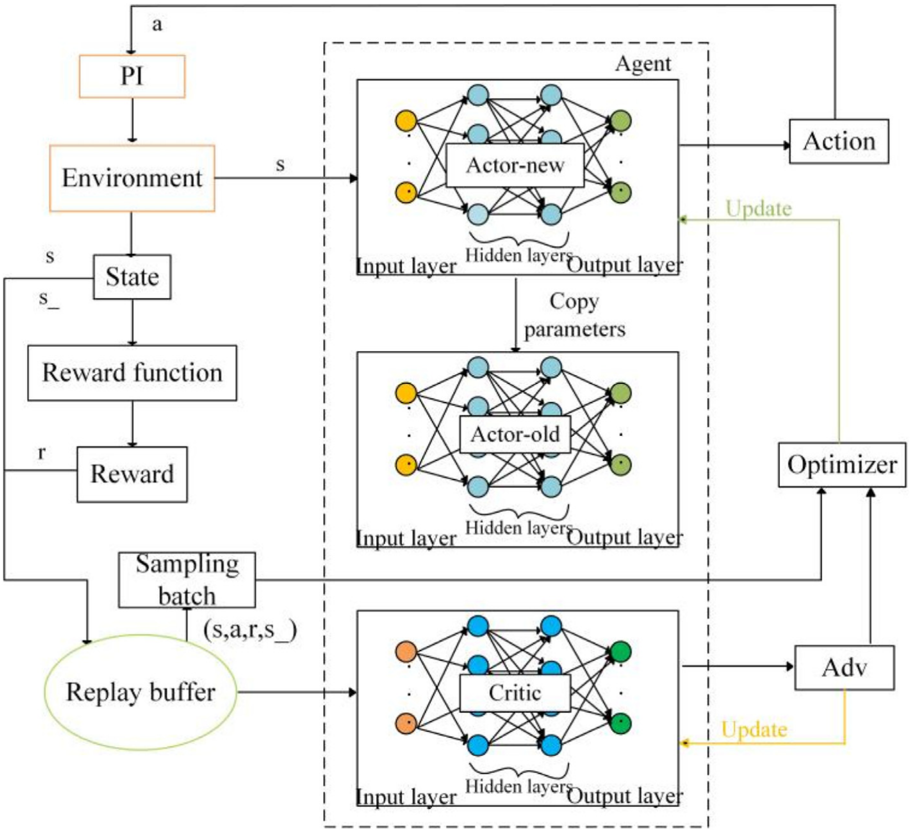

Fig. 4. The specific framework of the PPO controller.

$$r_1 = \begin{cases} -10, |r_v| > 200\% \\ -1, 15\% < |r_v| \le 200\% \\ 1, |r_v|, |r_v| \le 15\% \end{cases} \tag{8}$$

$$r_2 = \begin{cases} -10, |r_p| > 200\% \\ -1, 15\% < |r_p| \le 200\% \\ 1, |r_p|, |r_p| \le 15\% \end{cases} \tag{9}$$

$$r_3 = \begin{cases} -10, |r_t| > 200\% \\ -1, 15\% < |r_t| \le 200\% \\ 1, |r_t|, |r_t| \le 15\% \end{cases} \tag{10}$$

To sum up, the final reward function is:

$$R = a{*}r_1 + b{*}r_2 + c{*}r_3 \tag{11}$$

Where, a, b, and c are the adjustment coefficients.

4.3. PPO algorithm

The PPO algorithm belongs to the strategy gradient algorithm in essence. It solves the problem of step size determination in the policy gradient algorithm with the goal is to maximize the reward [27–29]. The objective function's parameter θ is updated as below, π is the random strategy:

$$L(\theta) = E[\log \pi(a_t|s_t;\theta)A_t(s_t,a_t)] \tag{12}$$

In the formula, $A_t(s_t, a_t)$ is the superiority function under the current policy:

$$A_t(s_t,a_t) = Q_t(s_t,a_t) - V_t(s_t) \tag{13}$$

The updating principle of parameter θ adopts the policy gradient algorithm, a is the adjustment coefficient, as follows:

$$\theta_{t+1} = \theta_t + a\nabla_\theta L(\theta_t) \tag{14}$$

The objective function is as below:

$$L(\theta) = E\left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}A_t\right] \tag{15}$$

θ_{old} represents the parameter before the policy is updated.

PPO adds the Kullback-Leibler (KL) divergence on the basis of the original objective function to measure the difference between two distributions. The bigger the difference is, the greater the difference is. δ is the precision limitation.

$$E\left[KL\left[\pi_{\theta_{old}}(a_t|s_t), \pi_\theta(a_t|s_t)\right]\right] \le \delta \tag{16}$$

At the same time, the PPO algorithm introduces the penalty term into the objective function, that is, the objective function is:

$$L(\theta) = E\left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}A_t - \beta KL\left[\pi_{\theta_{old}}(a_t|s_t), \pi_\theta(a_t|s_t)\right]\right] \tag{17}$$

Where, β is the punishing weight. The truncation function clip is used to replace the KL divergence to constrain $r_t(\theta)$ and prevent the big difference between the old and new policies. The ratio $r_t(\theta)$ of the old and new policies is:

Table 4
Settings for the neural network.

Parameter	Actor	Critic
Activation function	Tanh&softpus	Relu
Number of neurons in the input layer	3	3
Number of neurons in the hidden layer	15	15
Number of neurons in output layer	3	1

Table 5
Settings of hyper-parameter.

Parameter	Hyper-parameter
Learning rate of actor network	0.0001
Learning rate of critic network	0.0002
Discount factor	0.95
Max step	250
Max episode	300
Batch size	25
Truncation constant	0.2

$$r_t(\theta)\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)} \tag{18}$$

The objective function is:

$$L(\theta)=E[\min(r_t(\theta)A_t, clip(r_t(\theta), 1-\varepsilon, 1+\varepsilon)A_t)] \tag{19}$$

Where, ε is a hyperparameter.

5. Experiments and results

5.1. Pretraining

When the target is particularly complex, too large and difficult to represent, the RL algorithm is difficult to be directly trained and applied to nuclear power plant tasks due to the difficulty of training, long training time, and difficult representation of the environment. Since it is impossible to use the actual nuclear power plant for training, and in order to shorten the training time and improve the learning efficiency, this paper uses the trained LSTM model to replace the actual IPWR, and as the PPO algorithm training environment. After training the RL control method on the LSTM model, the control model will be saved.

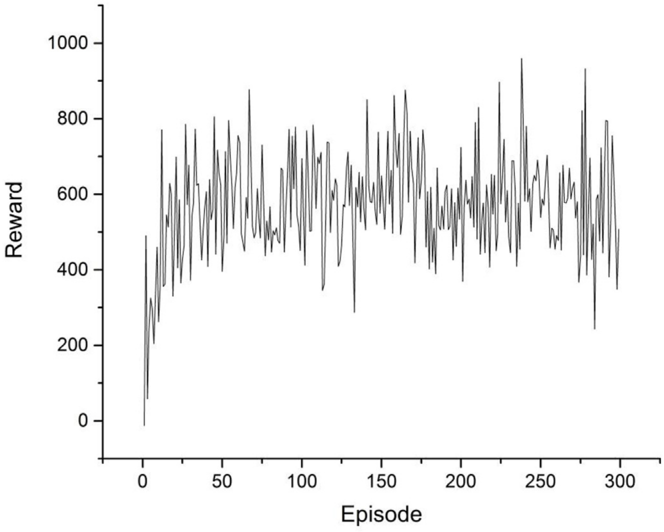

Fig. 5. Training effects of the PPO algorithm.

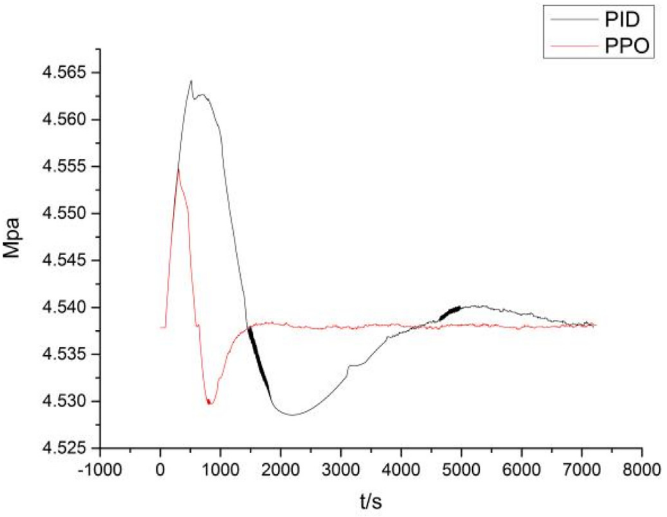

Fig. 6. Comparison curve of the steam pressure when reducing load.

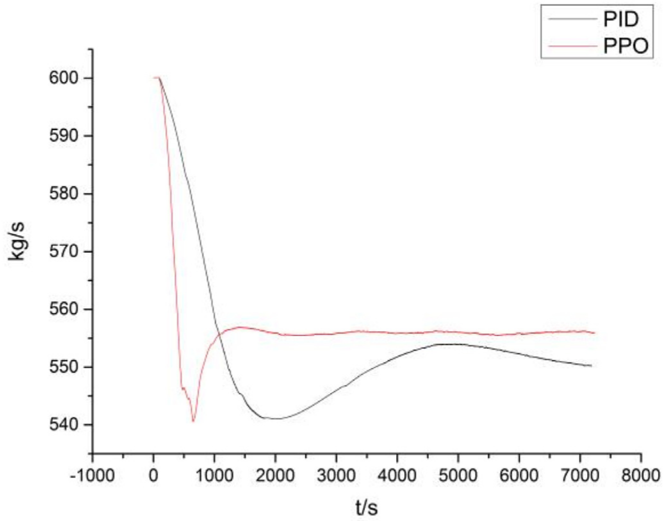

Fig. 7. Comparison curve of the feedwater flow when reducing load.

In order to verify the characteristics of the trained control model, the IPWR simulator is used instead of the actual power station. After saving the trained model, it is connected to the IPWR simulator for testing through the communication module. The RL control algorithm and LSTM environment model are developed with Python. At the same time, the communication module is developed to realize the data exchange between the control algorithm and the IPWR simulator.

5.2. Simulation setup

The setting of the PPO algorithm hyperparameter is divided into two parts: neural network parameters and algorithm hyper-parameter. The settings of neural network parameter are shown in Table 4. The hyper-parameters for the PPO algorithm are given in Table 5. The parameters used in the training and the experiment are the same.

The pretraining results are shown in Fig. 5. When the number of

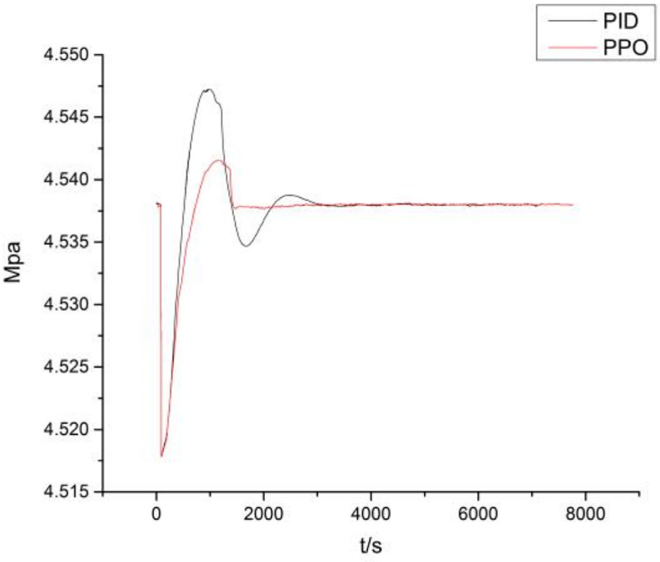

Fig. 8. Comparison curve of the steam pressure when increasing load.

algorithm iterations reaches the episode number, the training is stopped. Early in the training process, the reward value is low because the agent is a bit inexperienced and the number of training is not enough. The early stage is a process of exploration. With the increase of iteration number, the agent gradually learns more experience and get a good control effect, the cumulative reward value gradually begins to converge, that is to find the optimal control scheme.

5.3. Analysis of simulation results

In this chapter, transient tests and tracking tests are carried out on the IPWR simulator to test the performance of the proposed controller and the control effect is compared with that of the fixed parameter PI controller.

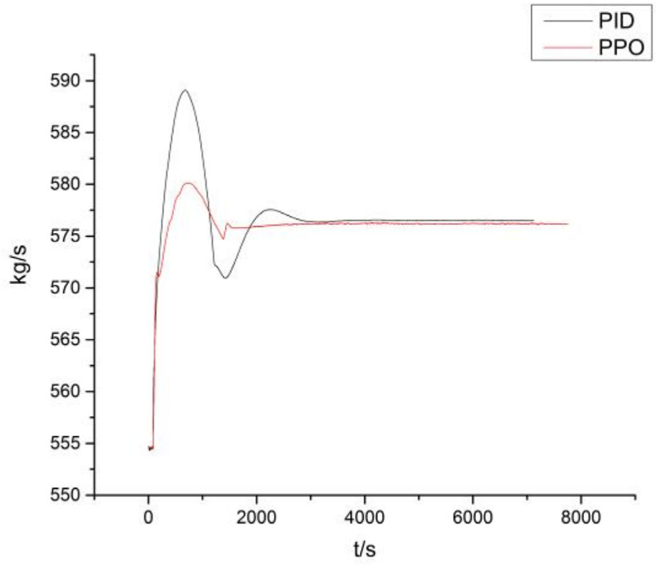

Fig. 9. Comparison curve of the feedwater flow when increasing load.

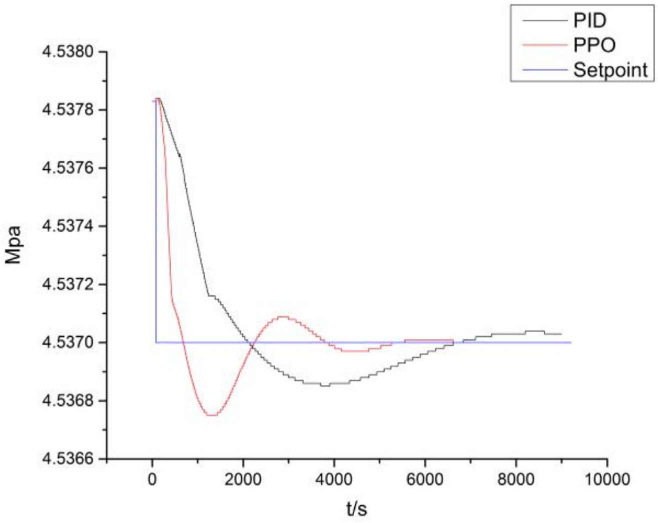

Fig. 10. Comparison curve of the steam pressure steps up to 4.537Mpa.

5.3.1. Transient test

In this section, the reducing and increasing load tests are carried out respectively, and the steam pressure setpoint for both tests is 4.538Mpa.

When the steam turbine load is reduced from 100%FP to 70%FP, the opening of the steam regulating valve decreases and the steam pressure increases gradually. As a result of the regulating effect of the control, the feedwater flow decreases and then the steam pressure decreases and becomes stable. Figs. 6–7 show the variation curves of steam pressure and feedwater flow when the steam turbine load decreases from 100%FP to 70%FP. The pre-trained RL controller not only has a smaller overshoot than the PID controller but also has a faster stabilization speed. When the steam turbine load increases from 70%FP to 100%FP, the comparison curve between steam pressure and feedwater flow is shown in Figs. 8–9. Compared with PID control, the RL controller has less overshoot and improves the response speed of the control system.

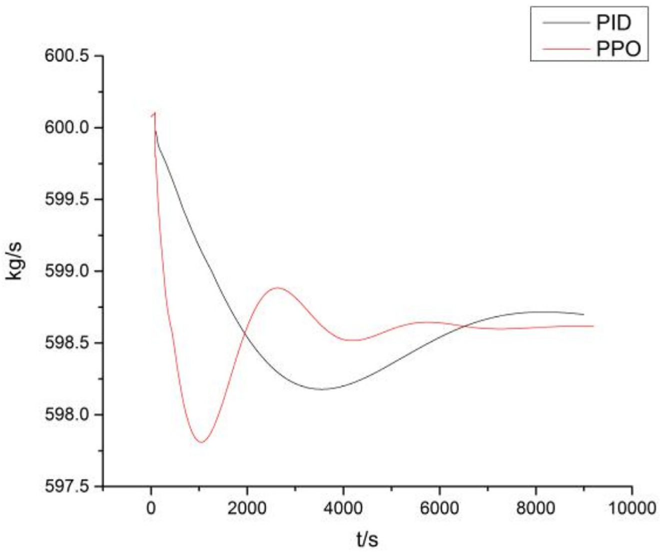

Fig. 11. Comparison curves of pressure and feedwater flow.

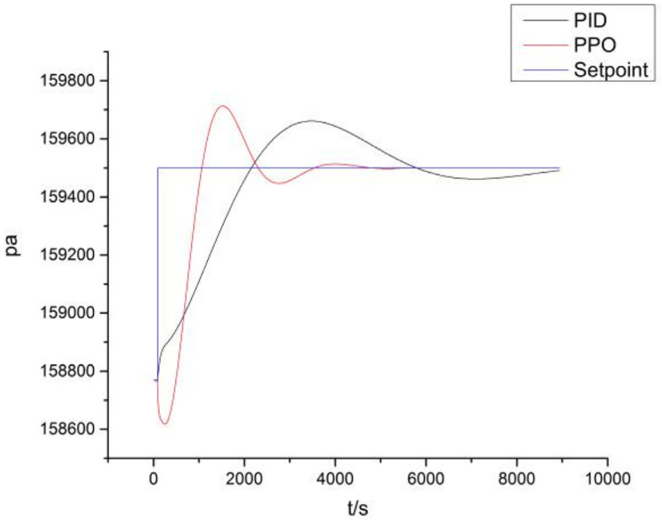

Fig. 12. Comparison curve of the differential pressure steps up to 0.1595Mpa.

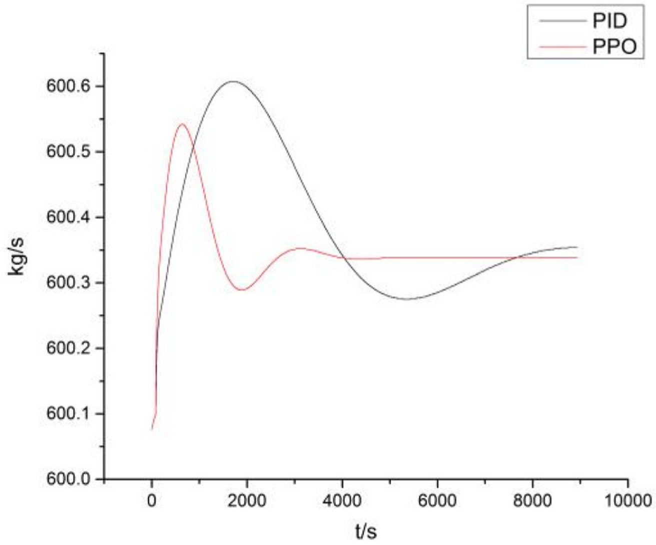

Fig. 13. Comparison curves of feedwater flow and feedwater temperature.

5.3.2. Tracking test

In order to better illustrate the performance of the method, the disturbance under the step function is added for comparison experiment, the step-change experiments of the setpoint of main steam pressure and the differential pressure between the feedwater valve front and back are carried out respectively.

Figs. 10–11 shows the comparison curves of the two schemes at the 100%FP when the pressure set point steps down from 4.538Mpa to 4.537Mpa at the 20s. Figs. 12–13 shows the comparison of the two methods when the differential pressure setpoint steps up from 0.1588Mpa to 0.1595Mpa when running at 100%FP for 20s during the test. Both schemes can effectively stabilize the pressure at the set point. Although the overshoot of the controller with PPO algorithms is slightly larger than the PI controller in these two experiments, it does not exceed 0.6% of the setpoint, and the stability time is significantly shorter than the PI controller.

6. Conclusions

In this paper, a novel controller of IPWR's OTSG with double-layer using the PPO algorithm is designed in this paper. This control structure can not only realize the learning of the parameter adjustment strategy of the upper agent, but also realize the adaptive adjustment of the parameters of the bottom PI controller, so as to realize the end-to-end control. In order to solve the problem that RL is difficult to apply, we creatively put forward the LSTM neural network as the environment of the PPO algorithm, which can reduce the training time of the agent and the training cost of the RL algorithm. The simulation results show that the controller adopts RL algorithm can realize the adaptive adjustment of PI parameters under various working conditions. Compared with the traditional PI control, it has the advantages of fast response speed and strong adaptive ability. The pre-training method can solve the problem of difficult training in the real environment, improve the training efficiency of the controller, and avoid dangerous states during the exploration.

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.

References

[1] H. Yao, G. Chen, K. Lu, Y. Wu, W. Tian, G. Su, S. Qiu, Study on the systematic thermal-hydraulic characteristics of helical coil once-through steam generator, Ann. Nucl. Energy 154 (2021), 108096.
[2] G. Zhao, Y. Zhao, J. Liu, Integral control strategy between the casing once-through steam generator and the turbine, Energy Conserv. Technol. 220 (2020) 162–166.
[3] Y. Zhang, M. Zheng, Z. Ma, J. Wu, Dynamic modeling ,simulation and control of helical coiled once-through steam generator, Appl. Sci. Technol. 313 (2020) 71–77.
[4] S. Cheng, C. Li, M. Peng, X. Liu, Research of pressure control based on artificial immune control of once -through steam generator, Nucl. Power Eng. 36 (2015) 62–65.
[5] Z. Chen, L. Liao, L. Liu, W. Li, Study on application of T-S fuzzy neural method in once-through steam generator feedwater control, Nucl. Power Eng. 33 (2012) 20–23.
[6] X. Hu, T. Yang, H. Qian, Research on control strategy of once-through steam generator for integrated reactor, J. Shanghai Univ. Electr. Power 37 (2021) 115–120.
[7] R.S. Sutton, A.G. Barto, R.J. Williams, Reinforcement learning is direct adaptive optimal control, IEEE Control Syst. Mag. 12 (1992) 19–22.
[8] C.J.C.H. Watkins, P. Dayan, Q-learn. Mach. Learn. 8 (1992) 279–292.
[9] T.P. Lillicrap, J.J. Hunt, A. Pritzel, et al., Continuous Control with Deep Reinforcement Learning, 2018, pp. 1–16. IN201847005934A.
[10] X. Wang, L. Zhang, T. Lin, C. Zhao, K. Wang, Z. Chen, Solving job scheduling problems in a resource preemption environment with multi-agent reinforcement learning, Robot. Comput. Integrated Manuf. 77 (2022) 102324.
[11] X. Deng, Y. Zhang, H. Qi, Towards optimal HVAC control in non-stationary building environments combining active change detection and deep reinforcement learning, Build. Environ. 211 (2022), 108680, 1-108680.16.
[12] X. Qiu, C. Gao, K. Wang, W. Jing, Attitude control of a moving MassA-ctuated UAV based on deep reinforcement learning, J. Aero. Eng. 35 (2022), 4021133.1-4021133.12.
[13] R.B. Grando, J.D. Jesus, V.A. Kich, et al., Double critic deep reinforcement learning for mapless 3D navigation of unmanned aerial vehicles, J. Intell. Rob. Syst. 104 (2022) 29–43.
[14] R. Zhang, Q. Lv, J. Li, J. Bao, T. Liu, S. Liu, A reinforcement learning method for human-robot collaboration in assembly tasks, Robot. Comput. Integrated Manuf. 73 (2022) 1–10.
[15] J.K. Park, T.K. Kim, S.H. Seong, Providing support to operators for monitoring safety functions using reinforcement learning, Prog. Nucl. Energy 118 (2022), 103123.
[16] T. Nishida, Data transformation and normalization, Rinsho Byori the Japanese Journal of Clinical Pathology 58 (2010) 990–997.
[17] M.S. David, S. Renjith, Comparison of word embeddings in text classification based on RNN and CNN, IOP Conf. Ser. Mater. Sci. Eng. 1187 (2021) 247–255.

[18] Q. Ye, Y. Wang, X. Li, J. Guo, Y. Huang, B. Yang, A power load prediction method of associated industry chain production resumption based on multi-task LSTM, Energy Rep. 8 (2022) 239–249.

[19] A. Zeng, W. Nie, Stock recommendation system based on deep bidirectional LSTM, Comput. Sci. 46 (2019) 84–89.

[20] J. Ren, J. Wang, C. Wang, Stock forecasting system based on elstm-l model, Stat. Decis. 35 (2019) 160–164.

[21] I. Papatsouma, N. Farmakis, Approximating symmetric distributions via sampling and coefficient of variation, Commun. Stat. 49 (2020) 61–77.

[22] V. Mnih, K. Kavukcuoglu, D. Silver, et al., Playing atari with deep reinforcement learning, CoRR abs/1312 5602 (2013) 1–9.

[23] T.P. Lillicrap, J.J. Hunt, A. Pritzel, et al., Continuous Control with Deep Reinforcement Learning, Computer ence, 2015, pp. 1–16.

[24] R.J. Williams, Simple statistical gradient-following algorithms for connectionist reinforcement learning, Mach. Learn. 8 (1992) 229–256.

[25] J. Schulman, et al., Trust region policy optimization, Int. Conf. Mach. Learn. 3 (2016) 244–259.

[26] P. Hämäläinen, et al., PPO-CMA: proximal policy optimization with covariance matrix adaptation, IEEE 30th Int. Workshop on Mach. Learn. Signal Proc. (2020) 1–6.

[27] J. Baxter, P.L. Bartlett, Infinite-horizon policy-gradient estimation, J. Artif. Intell. Res. 15 (2019) 319–350.

[28] D. Yan, C. Xi, Rein Houthooft, Bench marking deep reinforcement learning for continuous control, Int. Conf. Mach. Learn. 3 (2016) 2001–2014.

[29] Y. Wu, Z. Yu, C. Li, M. He, B. Hua, Z. Chen, Reinforcement learning in dual-arm trajectory planning for a free-floating space robot, Aero. Sci. Technol. 98 (2020), 105657.