An AGC dynamic control method based on DQN algorithm

Su Jianjun¹, Ma Changhui¹, Li Shan¹, Li Xiaoyu², Zhang Bing³, Liu Wenxue¹

¹ State Grid Shandong Electric Power Research Institute, Jinan 250003, China
² Key Laboratory of Power System Intelligent Dispatch and Control of Ministry of Education (Shandong University), Jinan, China
³ State Grid Shandong Electric Power Company, Jinan 250001, China
*Corresponding author’s e-mail: 626211631@qq.com

Abstract. In order to cope with the incompatibility of traditional AGC control under CPS control performance standard, this paper proposes a hierarchical control framework based on DQN algorithm. It effectively solves the problem of dimensionality disaster of Q learning by using neural network instead of state action pair, and speed up the convergence. Through simulation verification, this method can effectively improve the system CPS control performance index and effectively reduce the system power generation cost.

1. Introduction
Automatic generation control (AGC) is one of the important aspects of energy management system (EMS). It is mainly divided into two processes: 1) tracking of total power command, 2) allocation of total power command. The CPS pass rate is usually used as an important criterion for evaluating AGC control performance[1].

The traditional load frequency control is the narrow AGC uses the negative feedback control, and the proportional integral link can always eliminate the error and reach the steady state. However, due to the nonlinear link of the thermal power system, the dynamic performance based on linear programming control is not satisfactory. At the same time, the proportional integral gain coefficient of the secondary control loop is not only related to the system structural parameters, but also related to the load change. It needs to be adjusted in real time with the change of the system load to meet the frequency change requirements of the load change[2-3].

The AGC system under the CPS standard can be regarded as “uncertain stochastic system”. The mathematical model is modeled by Gauss-Markov stochastic process, and the power dynamic allocation problem can be understood as a discrete time Markov decision process[4]. Traditional analytical methods rely on the accuracy of grid structure, parameters and operational measurement data. Complex iterative algorithms often have weak robustness. The reinforcement learning (RL) algorithm with Markov decision process (MDP) as the basis of strict mathematics only needs to respond to the evaluation information of current control effects, with higher control real-time and robustness, and conventional control methods. Gradual learning and optimization ability that is not available. In recent years, a large number of studies have emerged to widely apply RL algorithms to the decision-making of power system operation and planning[5-6]. However, these studies are generally based on the more mature Q learning algorithm, which discretizes the continuous state of the
grid state, and solves the maximum future reward expectation by iterative Berman optimal equation. It is difficult to avoid the phenomenon of "dimension disaster" when facing complex problems.

The Deep Reinforcement Learning (DRL) algorithm, which combines deep learning (DL) and reinforcement learning, replaces the “state-action” pair in traditional Q learning with deep neural network (DNN), which can directly form the mapping relationship between continuous state quantity and reward expectation. Through the organic combination with deep learning, not only the reinforcement learning algorithm has the ability to process complex continuous state quantities, but also the deep learning algorithm can pre-learn the deep neural network parameters to effectively improve the convergence speed of the reinforcement learning algorithm.

2. Introduction of deep reinforcement learning

2.1. Deep learning

The neural network is a special nonlinear function structure with high versatility. Its hierarchical structure is very suitable for solving the parameters by using the gradient descent method chain-guided rule back propagation. The basic principle of the deep learning algorithm is to solve the maximum likelihood estimation of the neural network parameters, that is, the optimization problem with the maximum likelihood function as the objective function. Each training of the neural network corresponds to an iterative process of the optimization problem, and the iterative process. The initial values are usually randomly generated by the program. The maximum likelihood estimation uses the probability model and is a kind of statistic-based phylogenetic tree reconstruction method. That is to say, if the training set is large enough, the noise generated by the training data due to measurement error or the like is regarded as a normal distribution \( N(0, \sigma^2) \) satisfying the mean value of 0. For a given input quantity and neural network parameters, the output is subject to the exponential family. Distribution, and the goal of deep learning is to predict the mean \( E(y) \) of the output.

The most essential feature of deep learning is that neural networks have the ability to transform features. The output of a layer in the middle of the network can be thought of as another expression of data, which is considered to be a feature learned through the network. The implementation of traditional algorithms usually requires artificially specifying some feature quantities, then inputting variables and then transforming them according to the specified feature quantities. The deep learning algorithm is difficult to determine each layer of nerves even after the network training is completed and put into practical use. What characteristics are corresponding to the feature transformation of the network. This is one of the characteristics of today's big data analysis. It does not require experts in various fields to carry out detailed principles and feature analysis. Instead, it uses automatic data summarization to form corresponding feature transformations. This is why neural networks are highly generalized. One of the reasons for sex. However, on the other hand, this is the shortcoming of the deep learning algorithm, and the comprehensibility is very poor. It is difficult for us to interpret the trained neural network in the traditional sense. The ability of this kind of feature transformation of neural networks has been widely used in methods such as migration learning and reinforcement learning.

In summary, deep learning is not a "smart" algorithm, but a type of regression analysis problem in the statistical sense of big data. It is an algorithm for predicting the mean value of the probability distribution of the output based on the training data fitting function. The training process is essentially a solution to the optimization problem.

2.2. Q-Learning

Reinforcement learning (RL) is one of the methods for studying multi-stage decision problems based on Markov decision process. In the field of control theory, some experts also call reinforcement learning "approximate dynamic programming", which is different from dynamic programming. Reinforcement learning does not require the state transition matrix of the corresponding Markov decision process, reward function and all state sets. As a known condition, only the action set and the
initial state of the system need to be known. The goal of reinforcement learning is to get the agent to get the most cumulative reward in the process of interacting with the environment.

The Q-learning algorithm has been proposed as early as 1989. The basic idea is to form a Q-table as shown in the figure after a large number of interactions with the environment after determining the discrete state set and action set of the multi-stage decision problem. The maximum future reward expectation of taking different actions in a given state is used as a criterion for evaluating the effect of the control action.

During the iterative process of the algorithm, the control unit selects a control action to act on the environment according to the current state s, and the environment feeds back to the control unit reward and punishment signal R and the new state s' to which the environment is transferred under the action of the control action a. The maximum future reward expectation is the reward value R of action a under the current state s, plus the maximum future reward expectation of the discount factor that can be obtained by the new state s' to which the environment is transferred under the action of this control action a.

\[
Q(s, a) = E \left[ R(s, s', a) + \gamma \max_{a' \in A} Q(s', a') \right]
\]

Where: \(R(s, s', a)\) is the reward value R obtained by taking action a under state s, \(\gamma\) is a discount factor, and \(\max_{a' \in A} Q(s', a')\) is the maximum value among the row of Q values corresponding to the new state s'.

The iterative solution formula for the Q value is:

\[
Q^{k+1}(s, a) = Q^k(s, a) + \alpha \left[ R(s, s', a) + \gamma \max_{a' \in A} Q(s', a') - Q(s, a) \right]
\]

When all the Q values in the Q table converge, we can select the optimal control action for each discrete state of the system by querying the Q table to form a "state-action" pair. Because the algorithm always chooses the control action that is most likely to get a bigger reward, Q-learning is generally considered a greedy algorithm that pursues only the greatest rewards without taking into account the risks.

\[
a^*(s) = \arg\max_{a \in A} Q(s, a)
\]

The Q-learning algorithm is essentially a statistical method. After ensuring that each discrete case in the Q table has experienced enough times and iterating to convergence, the algorithm for selecting the optimal control action according to the Q table is only It can cope with the state that has experienced enough times, and can’t do anything about the state of insufficient or even unknown experience. It is difficult to avoid the problem of “dimensional disaster” in the face of complex problems.

2.3. Deep Q Network

In order to solve the shortcoming of the Q-learning algorithm, the Q value is no longer stored in a tabular form, but a function approximation (FA) method is used to construct a mapping relationship between states, actions and Q values. However, at that time, the academic community generally believed that large-scale nonlinear approximation functions for value function theory could not guarantee the convergence of the algorithm, instead of using large nonlinear approximation functions, Q-learning algorithm can only be used to solve some simple problems, lacking practical value. The research has entered the bottleneck. Until 2013, Google's deep mind team proposed the DQN algorithm, using the large-scale nonlinear function structure of deep neural network (DNN) as an approximation function, and guaranteed the convergence of the algorithm through special processing, which broke the recognition of people. know.
The DQN algorithm, which introduces neural network as an approximation function, faces severe convergence problems of neural network parameters. Deep learning requires training samples to be independent of each other, and there is a correlation between the state before and after the multi-stage decision-making problem of reinforcement learning. At the same time, in deep learning, each training (iterative) parameter is advanced toward the same target value, and its training process target distribution is fixed. In reinforcement learning, especially in the initial stage of strengthening learning to explore the state space, the new data added by each training may trigger the change of the convergence target of the neural network parameters. The reinforcement learning training process belongs to the optimization problem of the variable objective function, which makes it difficult to converge neural network parameters. In order to solve the problem of convergence, the DQN algorithm uses experience replay and target Q network, which is a method of establishing a memory and a dual neural network using time difference update. Experience replay establishes a memory and uses storage-random sampling to break the correlation between data. The target Q network introduces an auxiliary neural network whose structure is identical to the main neural network, but the parameters do not participate in the iterative operation. Its role is to form a parameter time difference update with the main neural network, and solve the convergence problem by reducing the correlation between the two neural networks.

By using a neural network instead of a Q-table, the DQN algorithm no longer divides the system state into a number of independent points like the Q-learning algorithm, but treats each state experienced during the interaction with the environment as a continuous state. A point on the domain that makes the DQN algorithm somewhat capable of coping with new states that have not been experienced in the training process. At the same time, the combination with the neural network enables deep reinforcement learning to be pre-learned by means of deep learning algorithms to speed up the convergence.

3. Design of AGC dynamic control method based on DQN algorithm

3.1. Hierarchical control framework

The closed-loop control of AGC dynamic optimization process under CPS standard is divided into two processes: CPS control and CPS instruction allocation. The two-layer control framework of this paper corresponds to these two processes.

The first layer of the framework is CPS control, and the AGC total scheduling instruction is calculated according to the system frequency deviation. The second layer of the framework allocates CPS commands. The distribution factor is calculated according to the characteristics of the AGC total dispatching command and the remaining capacity of each unit of the system. The distribution power of each unit is the product of the total CPS command and the distribution factor. The control purpose is to realize the economic system run. In the initial stage of training, the two parts are separately trained. When training the CPS control part, the instruction allocation is implemented using the PROP algorithm based on the adjustable capacity ratio allocation; when training the CPS instruction allocation part, the set load variation is directly used as the AGC total scheduling instruction. After the two parts are separately trained to converge, the two parts are combined for random disturbance simulation to further train to convergence and use the control effect of the CPS standard test algorithm.

3.2. Design of reward function

The power system frequency should be kept within the range of 50 ± 0.2 Hz during normal operation, so the segmentation reward function for CPS control is:

$$ R = \begin{cases} -\mu_1^* |\Delta f| & |\Delta f| < 0.2 \\ -\mu_2^* |\Delta f| & |\Delta f| > 0.2 \end{cases} $$

(4)

$\mu_1^*$, $\mu_2^*$ are the segment weights corresponding to the bonus function. This paper is set to 1 and 5.
The purpose of CPS instruction allocation is to realize the minimum cost operation of the system considering constraints such as unit capacity limitation and climbing speed. The unit capacity constraint and the climbing speed are controlled by the hard constraint participation algorithm, and only the system power generation cost is used as a reward. Function, no need to consider the weight factor.

\[ R = \sum_{i=1}^{n} C_i \Delta P_{G_i} \]  

(5)

3.3. Flow of DQN algorithm

The unified control process of the above two-layer control framework is as follows:

1) Initialize algorithm parameters such as Q-value neural network, initial state \( s \), learning speed, and discount factor.

2) Select the control action \( a \) according to the current state, and perform additional processing considering hard constraints such as unit capacity and climbing speed.

3) Applying the processed control action \( a \) to the power system simulation environment, obtaining the feedback system new state \( s' \), and calculating the reward \( R \) corresponding to the control action according to the bonus function formula.

4) Update the Q value neural network parameters according to the reward \( R \) of the control action.

5) Update the system state to \( s' \) and increase the number of iterations by one. When the current iteration number is greater than the maximum number of iterations, the algorithm ends; otherwise, jump to step 2).

4. Examples

Refer to the AGC unit data of a provincial power grid, and divide 77 units into 10 groups according to the characteristics of the unit. The information of the AGC unit is detailed in the literature[7]. The AGC control period takes 2s. The system model and its controller are built on the Python language environment. The CPS control and CPS instruction allocation part of the algorithm parameters are shown in the table.

Table 1. parameters of DQN algorithm

|                | Learning rate | Reward decay | \( \varepsilon \) greedy | \( \tau \) replace | Memory size | Batch size |
|----------------|---------------|--------------|------------------------|-------------------|-------------|------------|
| CPS control    | 0.02          | 0.9          | 0.8                    | 0.02              | 500         | 32         |
| CPS distribution | 0.005        | 0.9          | 0.9                    | 0.01              | 1500        | 64         |

In view of the random exploration of the state space by the depth enhancement algorithm, this paper runs the CPS controller part and the CPS dynamic instruction allocation part of the training process 50 times each time, taking the best-performing neural network parameters as the initial parameters of the final AGC dynamic control training part.

In order to verify the effectiveness of the proposed algorithm, the traditional PI-PROP control and DQN algorithm are used to solve the AGC dynamic scheduling problem respectively. The CPS index and system power generation cost are shown in the table.

Table 2. CPS indicator comparison

|                | CPS1/% | CPS2/% | CPS/% | Costs/w¥ |
|----------------|--------|--------|-------|----------|
| PI-PROP        | 156.47 | 100    | 84.52 | 179.36   |
| DQN            | 183.42 | 100    | 100   | 146.38   |

From the table, the CPS index of the proposed algorithm is significantly better than the traditional PI-PROP algorithm, and the total cost of system power generation is significantly reduced.
5. Conclusion

Aiming at the problem that the dynamic performance of the existing AGC unit dynamic optimization scheduling is poor, and the CPS assessment cost is not considered, the DQN hierarchical control algorithm is proposed. The simulation results show that the algorithm can effectively improve the CPS index and reduce the total cost of system power generation.

The success of the deep reinforcement learning algorithm represented by Alpha-go shows its potential and possibility beyond the traditional classical algorithms in the face of complex scenes, surpassing the existing level of human capabilities. Although the deep reinforcement learning algorithm is not yet perfect, it is undoubtedly one of the algorithms closest to artificial intelligence.

References

[1] Ai X, Liao S, Wen J. An AGC control performance index for main tie-lines in interconnected power grids[J]. Automation of Electric Power Systems, 2013, 37(21):111-117.
[2] LI H, YAN Z. Application of Q-learning Approach with Prior Knowledge to Non-linear AGC System [J][J]. Automation of Electric Power Systems, 2008, 23.
[3] Li H, Yan Z. Application of reinforcement learning method in a hydro-thermal hybrid automatic generation control system[J]. Dianli Xitong Zidonghua(Automation of Electric Power Systems), 2010, 34(9): 39-43.
[4] Y , Liu R, Li W. Comparison of CPS and its assessment between North America and China[J]. Dianli Xitong Zidonghua/Automation of Electric Power Systems, 2012, 36(15):63-72.
[5] Yu T, Liang H, Zhou B. Smart power generation control for microgrids islanded operation based on R (λ) learning[J]. Power System Protection and Control, 2012, 40(13): 7-13.
[6] Zhang X, Yu T, Tang J. Optimal CPS command dispatch based on hierarchically correlated equilibrium reinforcement learning[J]. Autom Elect Power Syst, 2015, 39(8): 80-86.
[7] Yu T, Wang Y, Ye W, et al. Multi-objective dynamic optimal dispatch method for CPS order of interconnected power grids using improved hierarchical reinforcement learning[C]//Zhongguo Dianji Gongcheng Xuebao(Proceedings of the Chinese Society of Electrical Engineering). Chinese Society for Electrical Engineering, 2011, 31(19): 90-96.