Energy Proceedings
Vol 60, 2025
ISSN 2004-2965

A Data-driven Multi-Step Temporal Difference Pretraining Framework of Deep Q-Network for Boiler Combustion Control

Jiaxuan Pu [1], Jingyu Wu [1], Yan Jiang [1*], Yaran Wang [1*], Huan Zhang [1]

1 School of Environmental Science and Engineering, Tianjin University, Tianjin,300350, PR China

(Corresponding Author: Jyan@tju.edu.cn, yaran_wang@tju.edu.cn)

ABSTRACT

Deep Q-network (DQN) has shown significant potential in industrial control, yet its high exploration costs and slow convergence during early deployment hinder practical adoption. To address this, this study proposes a data-driven pretraining framework for DQN, based on a multi-step temporal difference (TD) algorithm. Historical operation data and well-designed reward functions are employed to pretrain the agent, significantly reducing post-deployment exploration. The proposed framework is demonstrated through a case study on boiler combustion control system. The agent built on an enhanced recurrent neural network (RNN) architecture regulates coal feeding and air supply to meet dynamic steam load demands. The effects of different reward formulations and TD steps on pretraining efficacy are analyzed. Results show that agents pretrained with linear and nonlinear rewards both outperform historical control strategies. The 3-step TD pretraining strategy outperforms single-step TD, achieving 83.4% consistency with expert actions and realizing a 5.8% improvement in energy efficiency compared to manual operation.

Keywords: deep reinforcement learning, multi-step temporal difference algorithm, intelligent control, boiler combustion optimization.

NONMENCLATURE

Abbreviations	
DQN	Deep Q-network
MDP	Markov decision process
RL	Reinforcement learning
RNN	Recurrent neural network
TD	Temporal difference
Symbols	
a	Action
a_r	Reward function coefficient
b_r	Reward function coefficient
c_r	Reward function coefficient
Q	Q-value
R	Return
r	Reward
s	State
x	Evaluated value
x_{target}	Reference value

1. INTRODUCTION

Deep Q-network (DQN) is a classical reinforcement learning algorithm [1], well-suited for problems with continuous state spaces and discrete action spaces [2]. DQN is a potential tool for control tasks in industry, offering advantages such as simple network structure and strong stability [3]. Despite its potential, the application of DQN in industrial scenarios is limited by high exploration costs and safety risks during online learning [4]. Therefore, some research in RL for industrial applications focus on offline training and pretraining skills to mitigate these limitations [5-9]. However, few studies integrate multi-step temporal difference (TD) learning methods into pretraining frameworks.

In this work, a data-driven pretraining framework of DQN for boiler combustion control system is proposed to reduce exploration costs of initial period. The framework incorporates multi-step TD algorithm [10], tailored reward structures and historical field data to improve pretraining performance. The agent is built on an enhanced recurrent neural network (RNN) architecture to accommodate the latency of the boiler combustion system while guaranteeing an accurate estimation of the value function. The pretraining performances of different reward function formulations and TD steps are analyzed. The pretraining effectiveness is tested by a real-world boiler combustion control task and compared with manual operation and expert rule-based strategy.

This is a paper for the 5th Applied Energy Symposium and Forum: Renewable Energy Matrix (REM2025), Oct. 29-31, 2025, Yancheng, China.

2. METHOD

2.1 Pretraining framework

The proposed framework consists of offline data processing, reward shaping, multi-step TD learning, and Q-network training. It aims to initialize a DQN agent with a policy close to optimal, minimizing the reliance on online exploration.

2.2 Network architecture

The Q-network architecture is illustrated in Fig. 1. The Q-network uses an enhanced RNN-based structure to handle sequential input. The output layer adopts a dueling network structure [11], in which both the dominance stream and the state-value stream are modeled using fully connected neural networks. The outputs of the two networks are combined to estimate the Q-values [12]. Normalization and dropout layers improve training stability and generalization. The RNN-based part, dominance network and state value network all comprise one hidden layer with 32 neurons.

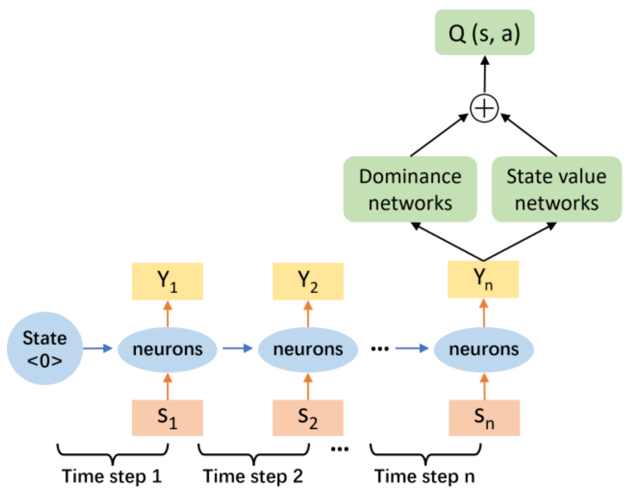


Fig. 1 The Q-network architecture

2.3 Reward design

The reward function should be designed according to the specific objectives of the control system. For the boiler combustion case, the main control goals include: (1) the output steam flow should meet the steam load demand; (2) the oxygen concentration in the flue gas should remain at a reference value; and (3) the drum pressure should stay within a safe range. The reward function consists of the sum of these three components.

Linear and nonlinear reward structures are investigated to evaluate their effect on policy quality during pretraining. The linear reward penalizes absolute deviations from target values:

$$y = a_r \, | x - x_{target} | + b_r$$

The nonlinear reward functions are constructed using Gaussian forms to enhance sensitivity near optimal operating points and suppress extreme actions:

$$y = a_r \exp\left(b_r \left(\frac{| x - x_{target} |}{c_r} \right)^2 \right)$$

In this case, both the steam flow and oxygen concentration parts are continuous functions. The vapor pressure part sets an allowable interval, with a penalty value only outside the interval and without penalty inside the interval.

2.4 Multi-step TD learning

The Q-value update in the single-step TD approach is given by:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t))$$

The Q-value update in the n-step TD approach is given by [13]:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(R_t^{(n)} + \gamma^n Q(s_{t+n}, a_{t+n}) - Q(s_t, a_t))$$

$$R_t^{(n)} = \sum_{k=0}^{n-1} \gamma^k r_{t+k}$$

Compared to single-step TD approach, multi-step TD approach incorporates a richer set of future rewards, which helps alleviate the overestimation bias and improve the stability of learning [14], particularly in environments with delayed rewards.

2.5 Case formulation

In the case study, the boiler combustion control system includes two actuators: coal feeder and air blower. The goal is to maintain desired steam output by adjusting these actuators based on real-time sensor readings.

The system is formulated as a Markov decision process (MDP) with four state features (oxygen concentration, steam flow, drum pressure, target steam load) and five action outputs (adding, subtracting, or holding adjustments of the two actuators).

The dataset is collected from a real-world boiler system. The raw operation data is sampled at a frequency of 1 second, with control instructions issued every 20 seconds. The agent is pretrained offline and then evaluated in a simulated environment to verify the effectiveness of the pretraining strategy.

2.6 Evaluation metrics

(1) Action rationality: the conformity of selected actions to physical laws and expert control experience. (2) Action similarity: Match between agent and expert actions; (3) efficiency gain: reduction in energy consumption compared to historical data.

3. RESULTS AND DISCUSSION

Using real historical data as state input, the actions generated by the pretrained agents were examined. The oxygen concentration and steam flow data from the selected historical data segment are shown in Fig. 2.

However, comparison between Fig. 3 (a) and Fig. 3 (b) reveals that the agent trained with the linear reward function is more sensitive to small deviations between actual and reference values, leading to more frequent actuator adjustments. Such frequent adjustments may increase the risk of actuator wear or failure. This is because the linear reward function maintains a constant gradient regardless of proximity to the target value, while the nonlinear reward functions, having smaller gradients near the reference point, allow the agent to be less reactive to minor deviations. Therefore, agents pretrained with nonlinear reward formulations exhibit lower operational risk in control decisions.

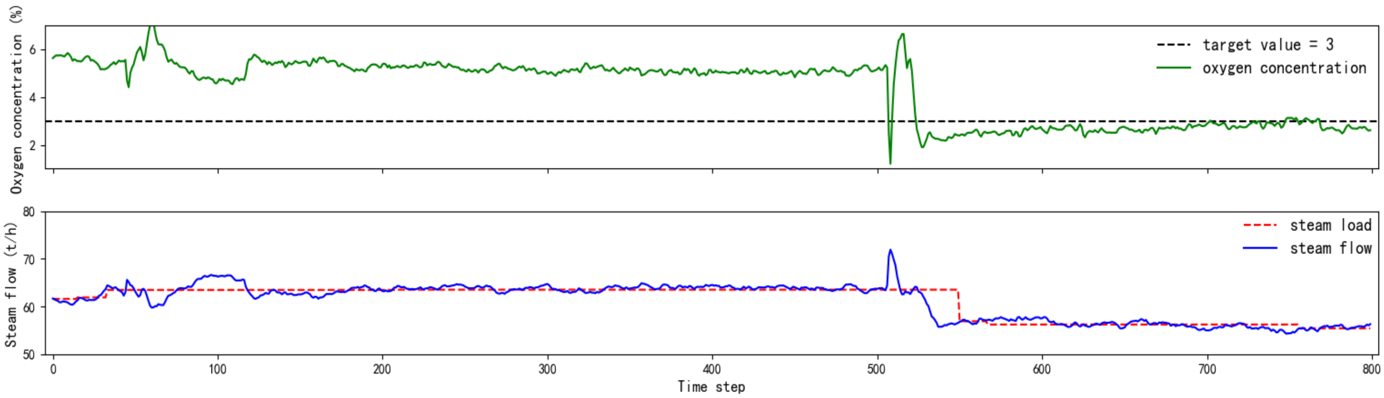
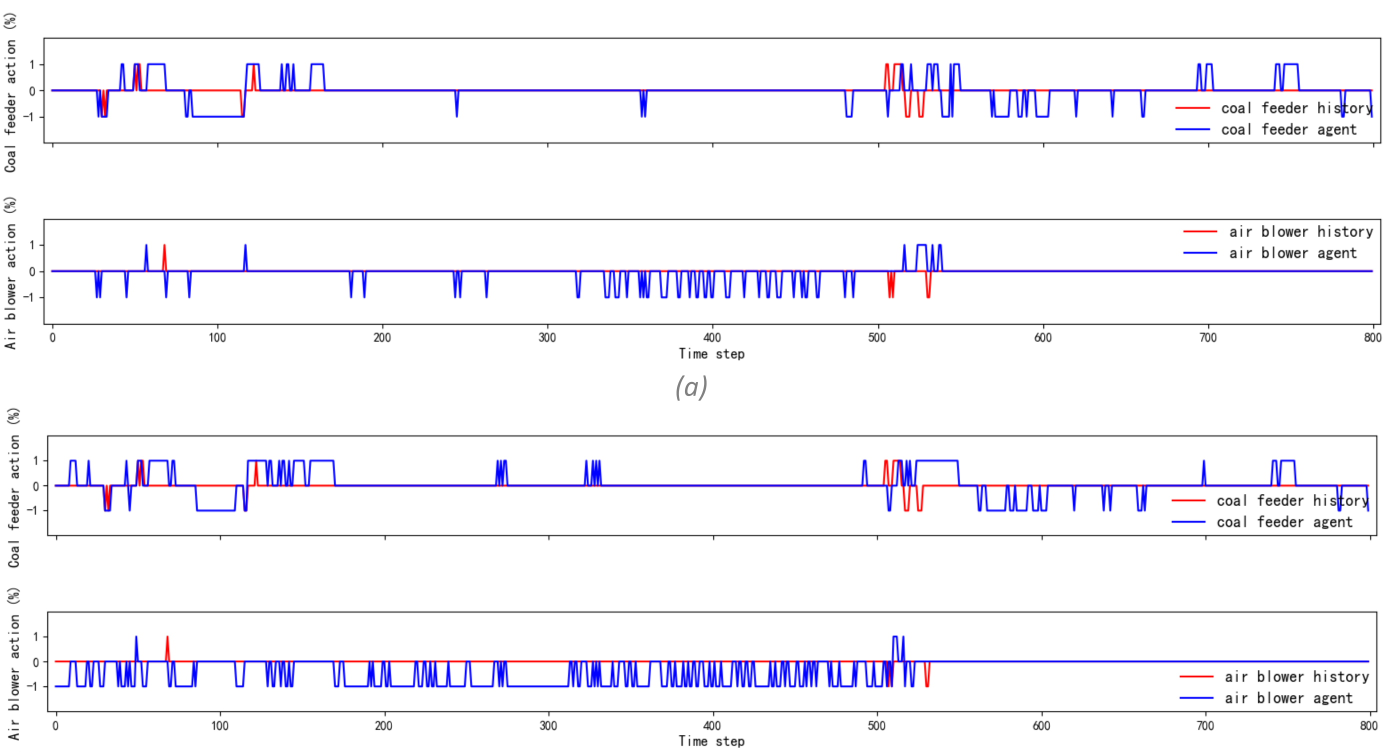


Fig. 2 Historical data segments of steam flow and oxygen concentration.

3.1 Impact of reward function formulations

The control actions selected by the agents pretrained with linear and nonlinear reward function formulations are illustrated in Fig. 3. Results show that both linear and nonlinear reward formulations enable the training of agents that generate reasonable control actions. Compared to real-world boiler operations, these agents exhibit faster response behaviors.



(a)

(b)

Fig. 3 The control actions chosen by the 3-step-nonlinear agent (a) and the 3-step-linear agent (b).

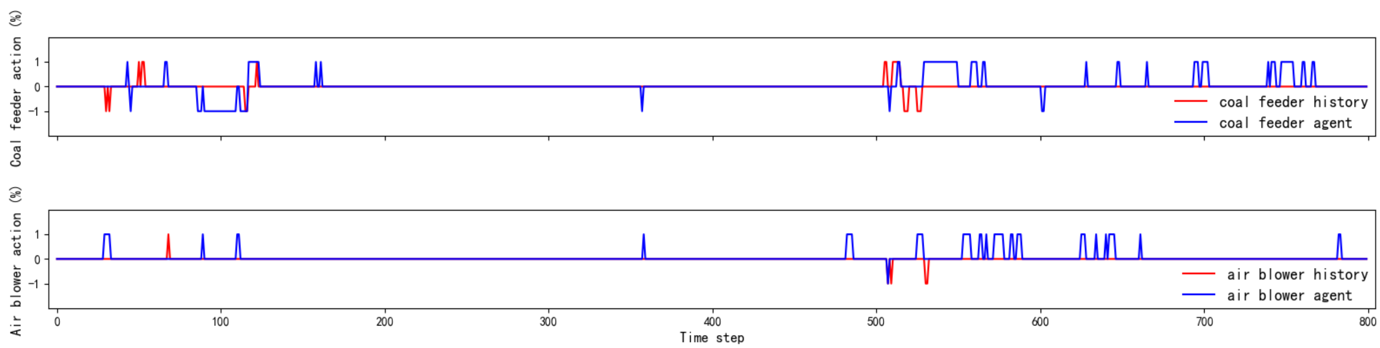
3.2 *Impact of TD step size*

The control actions selected by the agents pretrained with different TD step learning strategies are illustrated in Fig. 4. Significant performance differences are observed between the single-step and multi-step TD learning pretraining strategies by comparing Fig. 4 (a) and Fig. 4 (b). The 1-step TD agent often selects inappropriate actions, such as reducing coal feed despite insufficient steam output or increasing air flow when oxygen concentration is already high. This behavior improves as the TD step increases. The agent trained with 3-step TD learning achieves an action rationality rate of 83.4%.

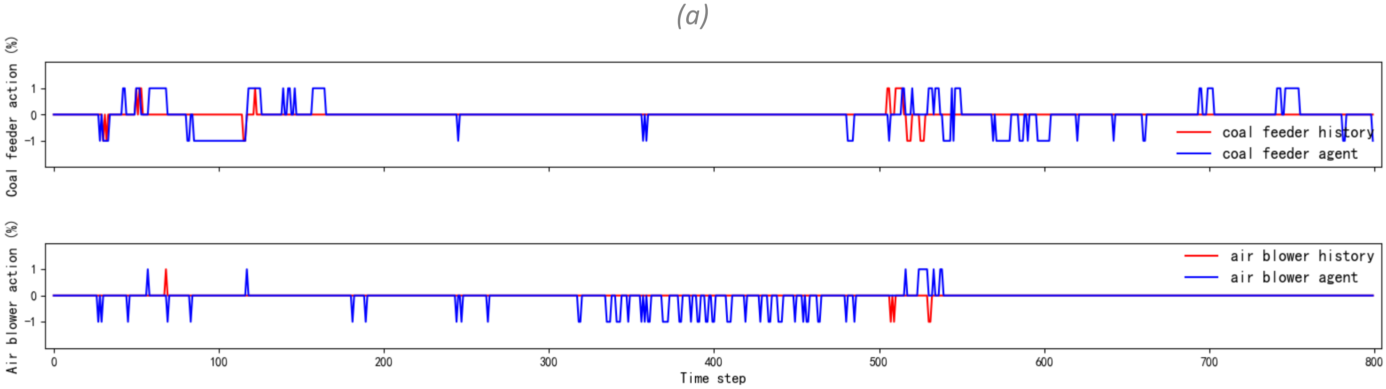
As shown in Fig. 4 (c), the performance of the agent trained with 5-step TD tends to decline. This is attributed to approximation errors introduced when the return is calculated using a finite sequence of actual observations to estimate expected returns. As the step size increases, the approximation error may also grow. Therefore, increasing the TD step does not always lead to better performance. It is necessary to choose an appropriate step size based on the system characteristics. Repeated testing shows that the agent pretrained with 3-step TD achieves the best overall results in this case.

3.3 *Environmental considerations*

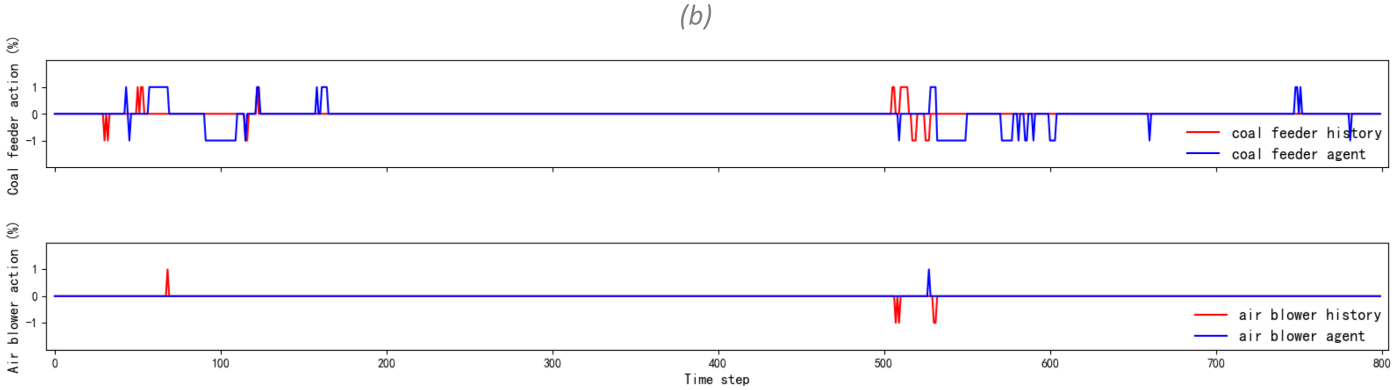
Based on the above, the 3-TD-nonlinear agent demonstrates the best overall performance and is used to evaluate the environmental friendliness of the proposed DQN-based control strategy. Fig. 5 compares the actuator speeds controlled by the DQN agent and



(a)



(b)



(c)

Fig. 4 The control actions chosen by the 1-step agent (a), the 3-step agent (b) and the 5-step agent (c).

manual operation. It can be observed that, while meeting the control objectives, the DQN agent tends to minimize the operation speeds of the coal feeder and the primary air blower. According to the calculation, this control strategy achieves a 5.8% reduction in energy consumption compared to manual control. Moreover, the reduced coal feeding contributes to lower emissions of pollutants such as nitrogen oxides (NO_x) and carbon monoxide (CO).

The offline pretraining approach also enhances system safety by avoiding trial-and-error during online learning, which is critical in industrial thermal systems. In addition, the modularity and generalizability of the proposed framework support potential transfer to other combustion equipment or broader industrial applications, promoting scalable adoption in low-carbon transition efforts.

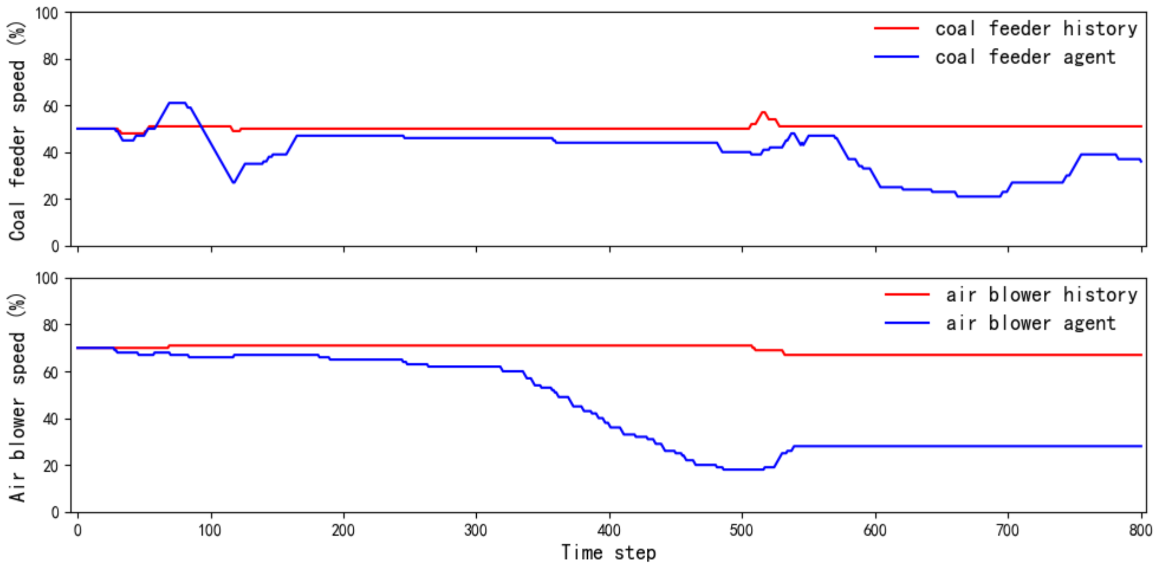


Fig. 5 The actuator speeds controlled by the 3-TD-nonlinear agent and manual operation.

4. CONCLUSIONS

This paper presents a data-driven pretraining framework for Deep Q-Network using multi-step temporal difference learning. The following conclusions can be drawn based on the experimental results and analysis:

(1) Both linear and nonlinear reward functions are effective in training agents capable of generating reasonable control actions, but nonlinear formulations reduce actuator wear.

(2) Multi-step TD learning significantly outperforms single-step TD in policy quality, with 3-step TD yielding the best balance between performance and stability.

(3) The proposed DQN control strategy with 3-step TD learning and nonlinear reward achieves 5.8% energy savings.

Future work will explore real-time deployment and generalization to other industrial control systems.

ACKNOWLEDGEMENT

In addition to the listed authors, the authors would like to express their special thanks to Mingguang Zhu for his valuable assistance in this research.

REFERENCE

[1] Dogru O, Xie J, Prakash O, Chiplunkar R, et al. Reinforcement learning in process industries: Review and perspective. IEEE/CAA J. Autom. Sinica, 2024; 11(2):283–300.

[2] Fu Q, Li Z, Ding Z, et al. ED-DQN: An event-driven deep reinforcement learning control method for multi-zone residential buildings. *Building and Environment*, 2023; 242:110546.

[3] Wang Z, Yin Y, Yao G, et al. BoilerNet: Deep reinforcement learning-based combustion optimization network for pulverized coal boiler. *Energy*, 2025; 318:134804.

[4] Park J, Jung H, Kim JW, et al. Reinforcement Learning for Process Control: Review and Benchmark Problems. *International Journal of Control, Automation and Systems*, 2025; 23:1-40.

[5] Park J, Park M, Lee D. Pretraining a shared Q-network for data-efficient offline reinforcement learning. *arXiv preprint* arXiv:2505.05701; 2025.

[6] Nakamoto M, Zhai Y, Singh A, et al. Cal-QL: Calibrated Offline RL Pre-Training for Efficient Online Fine-Tuning. *NeurIPS* 2023.
[7] Zhang L, Qi Q, Wang J, Sun H, Liao J. Multi-task Deep Reinforcement Learning for Scalable Parallel Task Scheduling. *Institute of Electrical and Electronics Engineering (IEEE)* 2021.
[8] Kumar A, Zhou A, Tucker G, Levine S. Conservative Q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems* 2020;33:1179–91.
[9] Agarwal R, Schuurmans D, Norouzi M. An optimistic perspective on offline reinforcement learning. *arXiv preprint* arXiv:1907.04543; 2019.
[10] De Asis KD, Sutton RS. Per-decision Multi-step Temporal Difference Learning with Control Variates. *arXiv preprint* arXiv:1807.01830; 2018.
[11] Wang Z, Schaul T, Hessel M, Van Hasselt H, Lanctot M, De Freitas N. Dueling Network Architectures for Deep Reinforcement Learning. *International Conference on Machine Learning (ICML)*; 2016.
[12] Hu W, Zhou Y, Ho HW. Mobile Robot Navigation Based on Noisy N-Step Dueling Double Deep Q-Network and Prioritized Experience Replay. *Electronics* 2024; 13(12):2423.
[13] De Asis KD, Sutton RS. Per-decision Multi-step Temporal Difference Learning with Control Variates. *arXiv preprint* arXiv:1807.01830; 2018.
[14] Ly A, Dazeley R, Vamplew P, Cruz F, Aryal S. Elastic step DQN: A novel multi-step algorithm to alleviate overestimation in Deep Q-Networks. *Neurocomputing* 2024; 576:127170.