Optimized Automatic Temperature and Humidity Control for Tobacco Storage Using TwinCAT and Deep Reinforcement Learning

Zhen Liu, Jili Wang*, Shihao Song, Qiang Hua
Qingdao ETSONG Technology Co.Ltd, Qingdao 266001, China

***Abstract*—With the rapid development of the tobacco industry, precise temperature and humidity control in storage environments has become essential for maintaining tobacco leaf quality. Traditional manual control methods suffer from low efficiency and limited accuracy, failing to meet modern storage demands. This study proposes an optimized automatic control system integrating TwinCAT and deep reinforcement learning (DRL) to enhance climate regulation in tobacco warehouses. Leveraging TwinCAT's real-time control capabilities and DRL's adaptive decision-making, the system achieves precise environmental regulation. Experimental results demonstrate that temperature and humidity control errors are reduced to ±0.5 °C and ±3%, respectively. Compared to conventional methods, the proposed system lowers energy consumption by 20% and reduces the mildew rate of stored tobacco by 15%, significantly improving storage quality. This work offers a novel technical framework for intelligent environmental control in tobacco storage and provides valuable insights for broader applications in similar domains.**

***Keywords*—TwinCAT; deep reinforcement learning; tobacco storage; temperature and humidity control; system optimization**

I. Introduction

In today's rapidly developing tobacco industry, tobacco storage is an important link between production and sales, and the importance of environmental control is increasingly prominent [1, 2]. As an agricultural product extremely sensitive to environmental conditions, the quality of tobacco leaves is closely related to the temperature and humidity in the storage environment. Improper temperature and humidity conditions will not only lead to mildew and moth-eating tobacco leaves but also affect their colour, aroma and taste, thus causing irreversible effects on the quality of tobacco products [3, 4]. Therefore, accurately and efficiently controlling the temperature and humidity in the tobacco storage environment has become an urgent technical problem that needs to be solved in the tobacco industry.

The rapid progress of automation technology, especially the wide application of industrial automation software TwinCAT (The Windows Control and Automation Technology), provides a new solution for tobacco storage environment control [5, 6]. With its powerful real-time flexibility and openness, TwinCAT can realize real-time monitoring and precise control of storage environment parameters [7]. However, traditional control strategies are often based on fixed rules or models, which are difficult to adapt to the complex and changeable storage environment, resulting in unsatisfactory control effects.

In this context, the rise of Deep Reinforcement Learning (DRL) technology has brought new ideas for controlling the tobacco storage environment [8]. By combining the perception ability of deep learning with the decision-making ability of reinforcement learning, DRL can enable the control system to learn and optimize itself in complex environments, thus achieving more intelligent and efficient control [9, 10]. Combining TwinCAT with DRL to build a new automatic temperature and humidity control system for a tobacco storage environment can not only make full use of the real-time control capabilities of TwinCAT but also further improve the performance and adaptability of the control system with the help of the intelligent optimization characteristics of DRL.

This study aims to explore the feasibility and effectiveness of this fusion technique. An intelligent optimisation algorithm based on DRL is designed and implemented through an in-depth analysis of the characteristics of the tobacco storage environment and its demand for temperature and humidity, combined with the real-time monitoring and control system of TwinCAT. The algorithm can autonomously adjust the control strategy according to the real-time environmental data to best control temperature and humidity. At the same time, this study will verify the actual effect of the proposed system through experiments and compare it with the traditional control methods in order to provide a new and more efficient technical means for tobacco storage environment control.

Existing tobacco storage temperature and humidity control methods have limitations such as insufficient integration of industrial control systems and intelligent algorithms, difficulty in adapting to complex and variable storage environments, and less ideal control accuracy and energy efficiency. The research fills the gap by combining TwinCAT industrial control technology with deep reinforcement learning, realizing a more intelligent, adaptive and high-performance automatic temperature and humidity control system for tobacco storage, which effectively addresses the aforementioned shortcomings.

This research will involve cutting-edge knowledge in many fields, such as automation control theory, deep learning, and reinforcement learning. By organically combining these theories, it aims to build an automatic control system with powerful and intelligent optimisation abilities. At the practical level, this study will focus on the practical application effect of the system, including control accuracy, stability, energy consumption and other aspects, to ensure that the proposed technology can truly meet the actual needs of tobacco storage.

*Corresponding Author

The optimization research on the automatic control system of temperature and humidity in tobacco storage environment by integrating TwinCAT and DRL has important theoretical value and broad application prospects. This research can provide new technical support for the warehousing environment control of the tobacco industry and promote the development of the industry in a more intelligent and efficient direction.

The basic theory section on TwinCAT and DRL elaborates on the TwinCAT platform, including its features, components, and real-time control capabilities, as well as the principles of DRL, such as Markov decision processes, state-action value functions, and the integration of deep neural networks with reinforcement learning. Subsequently, in the optimization design and implementation of the automatic temperature and humidity control system for tobacco storage environment, the design of the temperature and humidity automatic control system based on TwinCAT was introduced, covering the software and hardware aspects and system architecture, and the temperature and humidity control algorithm based on DRL was also developed to solve the coordination and optimization problems. The experimental and result analysis section provides experimental data through various graphs, comparative analysis with other methods, and discusses the performance of the proposed system, demonstrating the improvement in control accuracy, energy efficiency and stability. Finally, the conclusion summarizes the main findings of this study, highlights the effectiveness of the integrated system, and proposes future research directions.

In the field of temperature and humidity control of tobacco storage, other researchers have proposed a variety of solutions. Some studies use the traditional PID control algorithm to adjust the temperature and humidity by setting fixed parameters, which realizes basic control, but it is difficult to cope with the complex and changeable interference in the storage environment, and the control accuracy and adaptability are limited. There are also studies on the use of fuzzy control methods to deal with uncertainty factors by using fuzzy rules, but there are shortcomings in dynamic response speed and optimization ability. In addition, some scholars try to combine simple machine learning algorithms with control strategies, although they have certain effects in specific scenarios, but lack of deep integration of industrial control systems. At present, the existing schemes generally have problems such as low integration with industrial control platforms, difficulty in adapting to changes in complex storage environments, and unsatisfactory control accuracy and energy efficiency, which are the directions to be further studied. In this study, TwinCAT industrial control technology and deep reinforcement learning are deeply integrated, which not only gives full play to the advantages of TwinCAT in real-time control and system integration, but also makes up for the shortcomings of existing schemes in complex environment adaptability and control performance with the help of deep reinforcement learning's adaptive learning and optimization decision-making ability, and forms a more intelligent, efficient and suitable for the actual industrial scene of tobacco storage temperature and humidity control scheme, which has made significant breakthroughs in technology integration and control effect.

II. Basic Theory of TwinCAT and DRL

A. Introduction to the Twincat Platform

TwinCAT is a pure software controller, which is the core component of the Beckhoff controller, with excellent openness and expansion potential [11, 12]. It has an interface that can be connected to a common field. The development environment integrates Microsoft Visual Studio software, supports special programming languages such as IEC61131 and PLC, and is compatible with high-level languages such as C, C + + and MATLAB/Simulink. Users can choose programming tools according to task characteristics. In addition, TwinCAT has multi-core processing capabilities, which can use all cores to improve performance according to the controller's condition. The PC can be regarded as a calculator, PLC and motion controller when its running core is installed.

TwinCAT's real-time system includes multiple industrial control software packages, covering various motion control software modules, such as TwinCATPLC, TwinCAT NC, TwinCAT CNC Scope View, etc. They can operate independently, exchange information, and work collaboratively through the TwinCAT ADS interface. The TwinCAT task manager controls the real-time process [13, 14].

TwinCAT PLC part is the core of Beckhoff equipment to implement robot motion control, including various logic instructions to control motor motion and some key sub-modules commonly used in programming [15]. Scope View is a system variable monitoring and analysis tool of TwincCAT software, which can display program variables in graphical form, which is convenient for users to monitor, analyze and control system variables in real-time. It is also equipped with Cursor Cursor, Trigger Trigger and other tools for easy operation [16]. Scope Array Bar Project monitors array variables with histograms; Scope project monitors a single variable over a time program; The Scope YTNC project monitors axis variables by axis number; Scope YT project with reporting analyzes the YT graph with reporting function; Scope XY project observes the corresponding relationship between any two variables; Scope XY project with reporting analyzes the XY graph with reporting function. TwinCAT NC is responsible for axis motion control in the TwinCAT system, which can improve axis control performance and play a key role in logic and function control. As a core module, it accelerates platform development and broadens application scope. Its compatibility enables connection with other manufacturers' equipment, and it has the advantages of superior performance, fast speed, high efficiency, and convenience [17, 18].

In the TwinCAT system, the motor control process is divided into three layers: ① PLC axis, axis variables defined under the TwinCAT PLC module; ② NC axis, a virtual axis added to the NC axis configuration interface under the Motion module; ③ Physical axis, the motion execution and position feedback hardware obtained by I/O module scanning and added to the TwinCAT system, each layer has different functions and is interrelated.

B. Deep Reinforcement Learning (DRL) Principles

Reinforcement learning (RL) uses the interaction between agents and the environment to continuously learn by trial and error to find the best strategy [19]. Specifically, RL does not rely on the real-time supervision signal to guide the learning path but relies on the reward signal evaluation strategy to indirectly guide the agent to learn towards the reward maximum and reduce the dependence on the accurate system model [20]. Under the RL framework, the agent makes decisions according to the real-time state of the environment. After execution, the environment enters a new state and feeds back the reward. The infinite loop of this decision and reward feedback constitutes its training process [21].

The core of training is agent decision-making to maximize long-term benefits [22]. The interaction between agent and environment in RL is often modeled as Markov decision process (MDP) [23]. MDP is characterized by a quadruple *(S, A, P, r)*, *S* is the state set, *A* is the action set, *P* describes the probability that the state $s \in S$ transitions to $s' \in S$ after executing the action $a \in A$, and *r* is the reward obtained by the agent when it transitions from state S to the next state S' after executing the action. Under the MDP framework, the agent selects the action $a_t \in A$ through the policy function $\pi(a_t|s_t)$ based on the current state $s_t \in S$ at time *t*, and gets a reward $r(s_t, a_t)$ after execution. The next state is randomly determined by the transition probability $P(s_{t+1}|s_t, a_t)$. The sequence of states, actions and rewards experienced by the agent before reaching the termination state constitutes a round, and its goal is to act to maximize the sum of all rewards at the end of the round, that is, the reward, which is defined in Eq. (1).

$$R_t = \sum_{k=0}^{N-1} r_{t+k} \tag{1}$$

In the formula, *N* represents the total number of steps of the discrete step size. In order to distinguish the relative importance of immediate reward from future reward, a reward discount factor γ (ranging from [0, 1]) is introduced. When γ = 0, immediate reward dominates; When γ = 1, future rewards are more important. Therefore, the return R_t is redefined as Eq. (2).

$$R_t = \sum_{k=0}^{N-1} \gamma^k r_{t+k} \tag{2}$$

The fundamental purpose of an agent is to find the optimal strategy π^*, from which it can determine the best actions it should perform in each state and ensure the maximum discount reward. In order to explore the optimal strategy π^*, a state value function $V(s)$ can be constructed to evaluate the advantageous degree of reaching the current state *s*, as shown in Eq. (3).

$$V^{\pi}(s_t) = E\left[R_t \mid s_t = s\right] = E\left[\sum_{k=0}^{N-1} \gamma^k r_{t+k} \mid s_t = s\right] \tag{3}$$

The state-valued function $V(s)$ is a means to evaluate the pros and cons of a strategy. For any $s \in S$, if the expected return according to the strategy π is higher than π', then $V_{\pi}(s) \geq V_{\pi'}(s)$, indicating that the strategy π is better, and the agent will tend to π when choosing the strategy function. Therefore, the optimal strategy π^* is defined in Eq. (4).

$$V^*(s) = \max_{\pi} V_{\pi}(s), \forall s \in S \tag{4}$$

Although the state value function can compare different strategies, it cannot find the optimal strategy [24]. Bellman created the state-action value function $Q_{\pi}(s, a)$, which is used to measure the degree of return of taking action a according to strategy π in state *s*, as shown in Eq. (5).

$$Q^{\pi}(s_t, a_t) = E\left[R_t \mid s_t = s, a_t = a\right] = E\left[\sum_{k=0}^{N-1} \gamma^k r_{t+k} \mid s_t = s, a_t = a\right] \tag{5}$$

The optimal Q*(s, a) can be defined as Eq. (6).

$$Q^*(s,a) = \max_{\pi} Q_{\pi}(s,a), \forall s \in S, a \in A \tag{6}$$

In order to determine the strategy π, it is necessary to pick out the action matching the maximum value from the action state value function $Q_{\pi}(s, a)$ for a specific state *s*. This is called greedy strategy, which is defined in Eq. (7).

$$\pi(a \mid s) = \begin{cases} 0, if \ \ a = \arg\max_{a'} Q(s,a') \\ 0, if \ \ a \neq \arg\max_{a'} Q(s,a') \end{cases} \tag{7}$$

When the agent operates according to the optimal strategy π^*, the expression of $V^*(s)$ is shown in Eq. (8).

$$V^*(s) = \max_{\pi} Q_{\pi^*}(s,a) \tag{8}$$

The corresponding Bellman equation can be obtained by recursive extension of Eq. (8), as shown in Eq. (9).

$$V^*(s) = \max_{a} \sum_{s',r} p(s', r \mid s, a)[r + \gamma V^*(s')] \tag{9}$$

By recursive derivation of Eq. (6), the optimal Bellman equation of $Q^*(s, a)$ can be obtained, Eq. (10) can be obtained.

$$Q^*(s,a) = \sum_{s'} p(s', r \mid s, a)[r + \gamma \max_{a'} Q^*(s', a')] \tag{10}$$

The premise of the research is to clarify the state transition probability and reward value of the agent, and the Bellman optimal equation can be solved iteratively, which is dynamic programming [25]. Algorithms with known state transition probabilities and rewards are collectively called model-based algorithms. Most RL problems assume that the state transition probability is unknown, and the algorithm set designed for this is called a model-free algorithm.

Traditional reinforcement learning is limited, with small action and sample spaces and mostly discrete environments, which makes it difficult to cope with high-dimensional input data such as images and sounds [26, 27]. DRL combines high-dimensional input of deep neural networks (DNNs) with reinforcement learning. DNNs have the latest technological advances in speech recognition, image classification, machine translation, robot control, etc. The commonly used architecture of DNNs is a feedforward neural network (FNN), a multi-layer perceptron model consisting of the input layer, one or more hidden layers, and an output layer [28].

After the input signal is introduced into the input layer, it will be transmitted layer by layer in the network. The hidden layer and output layer are composed of multiple nodes (perceptrons) [29]. When each perceptron receives the input vector of the previous layer, it will assign a weight ω to each vector element x and sum it. The operation result can be expressed by Eq. (11).

$$z = \sum_i x_i \omega_i + b \tag{11}$$

Where b is the deviation coefficient. After the summation operation is completed and the result is obtained, the activation function f is used to generate the neuron output y, and the calculation process of the neuron model is shown in Eq. (12).

$$y = f(\sum_i x_i \omega_i + b) \tag{12}$$

The multiplication operation of adjusting the input weight of the perceptron can change its behaviour and that of the network. It is necessary to train the neural network to adjust the weight in a specific way so that the network behaviour meets the expected results [30]. Fig. 1 shows the DRL core architecture. The deep learning module captures target observation data from the environment and provides environmental state information. Then, the reinforcement learning component corresponds the current state to the corresponding action, evaluates the value according to the estimated return, optimizes the decision-making action through the interactive process, and iteratively updates the network parameters to obtain the best strategy.

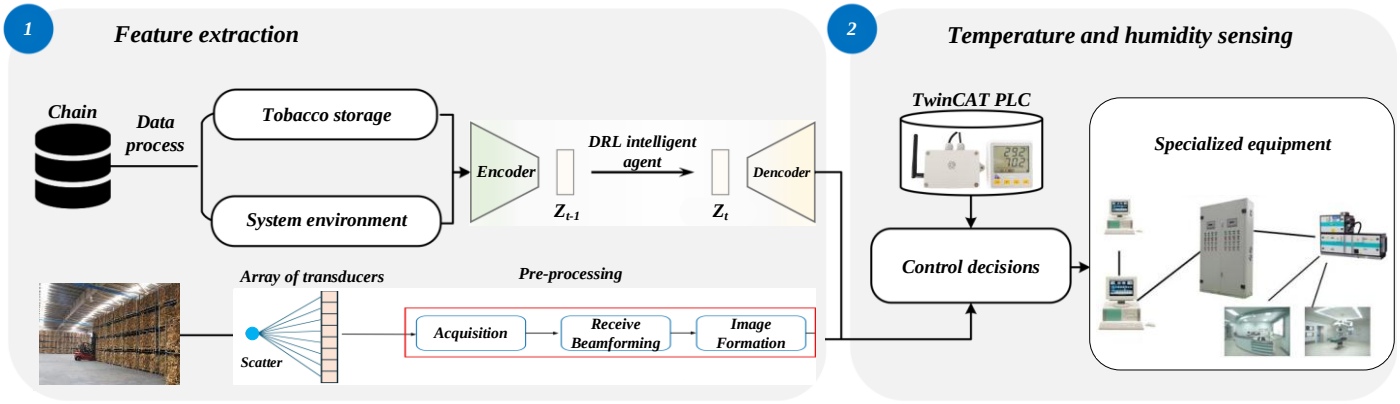

Fig. 1. Deep reinforcement learning architecture.

III. Optimization Design and Implementation of Automatic Control System for Temperature and Humidity in Tobacco Storage Environment

A. Automatic Control System of Temperature and Humidity in Tobacco Storage Environment

The inspection machine control system software software includes seven functional modules: start-stop speed control, servo motor control, etc. Accordingly, the software and hardware design process is carried out.

Regarding software design, the inspection machine involves the interface design of the host computer based on the Qt framework, visual inspection algorithm program and Beckhoff PLC programming based on TwinCAT. The Qt host computer interface uses the Qt Creator development tool on the Windows platform (Qt 5.12. 2 versions). The host computer and Beckhoff PLC interact with each other through ADS communication protocol, and a communication bridge is built in Qt Creator environment with C + + language combined with ADSDLL library of Beckhoff TwinCAT to realize the reading and writing of PLC memory data by the host computer. Beckhoff PLC is the ADS server in this architecture, and the host computer is the client. It uses an asynchronous notification mechanism to send ADS requests, and the server sends responses through the Call-back mechanism until the request is cancelled. Its advantage is that TwinCAT only transmits data when updated, which can avoid the loss of program running efficiency.

The TwinCAT integrated development platform is used for the lower computer Beckhoff PLC program design. TwinCAT is a PC-based real-time control system that can accurately control and respond to real-time systems, supports multi-task parallel processing, and assigns different priorities to tasks as needed. Each task contains one or more program blocks to execute specific control logic to cope with complex control challenges. In addition, TwinCAT is compatible with various communication protocols, which facilitates data interconnection with other devices and systems. The control system architecture adopted in this paper is shown in Fig. 2.

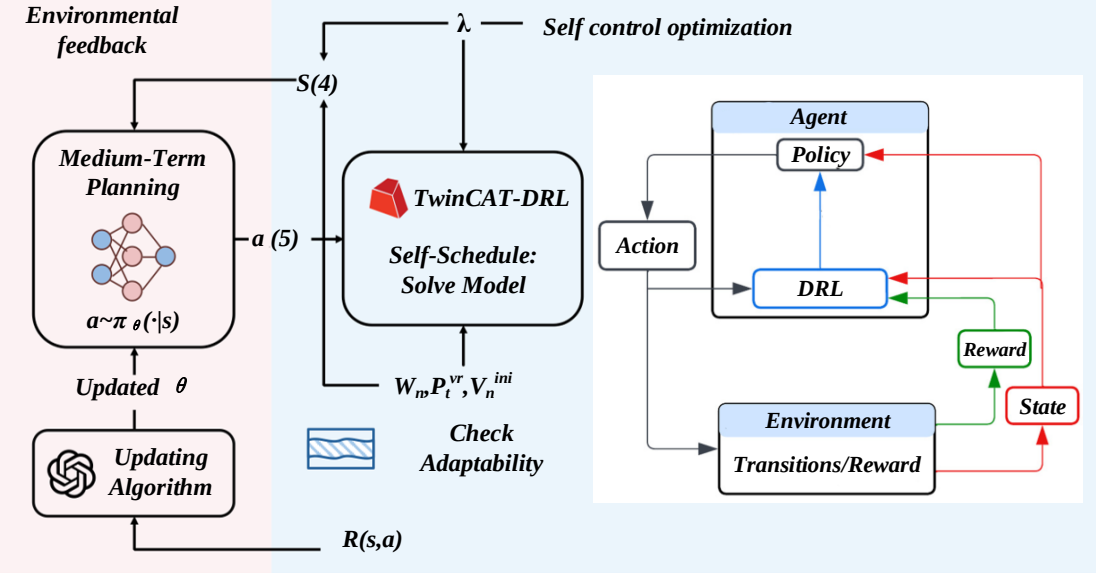

Fig. 2. Control system architecture.

B. Temperature and Humidity Control Algorithm Based on DRL

Because DRL aims to control frequency deviation minimization and tie-line power stability, GCD pursues control cost minimization. It is difficult to comprehensively optimize and control frequency deviation and control cost by combining the two algorithms, especially in the special situation of the current control system, and the actual operation time may exceed the AGC control interval limit, resulting in performance degradation. Therefore, to solve the mismatch problem between the two, this study proposes an integrated frequency regulation architecture based on an intelligent controller, which gives regulation instructions on the premise of balancing the two objectives according to the real-time state of the system to achieve multi-objective optimization.

In the automatic temperature and humidity control system, due to the wide variety of equipment involved and their unique characteristics, their frequency response speeds are also different, which requires the intelligent controller to accurately generate the matching power regulation instructions according to the unique performance characteristics of each frequency regulation unit. However, using traditional algorithms to complete this complex process often requires a lot of time and energy to carry out tedious parameter adjustment experiments to ensure the system's normal operation. Even so, when the system faces large-scale random disturbance, the dynamic performance of traditional algorithms is often unsatisfactory, and it is difficult to cope effectively with complex and changeable actual working conditions, which limits the stability and reliability of the temperature and humidity automatic control system to a certain extent.

Therefore, this study proposes an integrated frequency control strategy for a multi-regional interconnected temperature and humidity automatic control system, using the algorithm as the intelligent controller, sensing relevant data and outputting control instructions simultaneously, and designing the reward function to consider multiple dimensions to solve coordination problems comprehensively. Through pre-training, the algorithm masters the dynamic performance of each frequency modulation unit and the characteristics of system frequency change and generates reasonable instructions in the execution stage to ensure that the deviation is within the specified range. Each region is regarded as an independent agent, and the control instructions are obtained by a multi-agent cooperative game. The advantages of this strategy are: it can avoid the problem of AGC performance degradation, the algorithm is a multi-output algorithm, the calculation time is less than the command cycle limit, the strategy can be updated online without relying on the model, and it can generate instructions based on the real-time status of each unit to exert its frequency modulation potential.

IV. Experiment and Result Analysis

Experimental data Table I shows that the detection performance of the improved scheme is significantly improved. Under the mAP @ 50: 5: 95 evaluation standards, the improved version improves by about 1.5 percentage points on average compared with the previous one. Except for SSD, the mAP50 value of other methods exceeded 99.0%, and the mAP50 value of this method was the highest, reaching 99.5%. It can also be seen from Table 1 that although the improved method has a time delay of about 0.8 milliseconds compared with the original model, the processing speed is still about 130 fps, far exceeding the 20 fps of the two-stage target detector Faster-RCNN.

TABLE I Comparative Experimental Results

Detection model	mAP @ 50: 5: 95	mAP50	Average detection time per picture/ms
Base	95.6%	98.5%	8
This article improves	97.1%	98.5%	9
Faster R-CNN	97.7%	98.3%	60
SSD	48.8%	88.1%	42

Fig. 3 shows the average reward of each round of the algorithm agent. The training consists of 3000 rounds, each containing 60 time slots. At the beginning of the time slot, the agent interacts with the environment, makes decisions and updates the network model. During this period, the average reward of the DDPG module and DQN module climbed and converged with the increase in training rounds. The average reward of the DDPG module increases rapidly after 300 rounds and stabilizes after 1000 rounds. The DQN module gradually converges from the beginning of learning, and its average reward tends to be stable after 300 rounds. As the number of rounds increases, the DQN module will reduce the exploration frequency and generate greater actions, making its output decisions and rewards tend to be stable.

Fig. 4 explores the impact of warehousing computing resources on average weighted user energy consumption. The energy consumption of each scheme decreases with the increase of warehousing computing resources. When warehousing computing resources are scarce, the computing delay of users who choose to offload computing is too large, which does not meet the maximum delay permission conditions. Users prefer to execute tasks locally, resulting in higher user energy consumption.

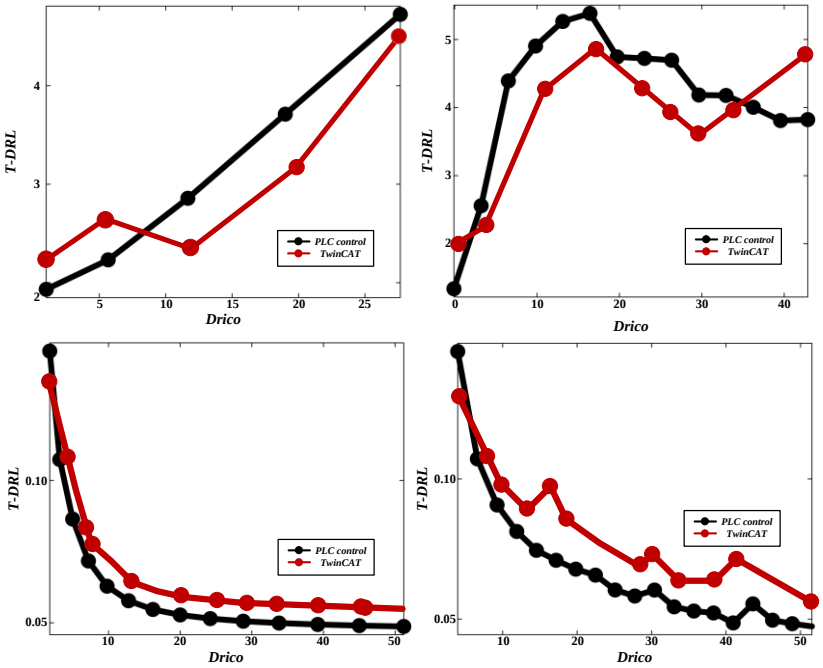

Fig. 3. DDPG unit average reward of the algorithm.

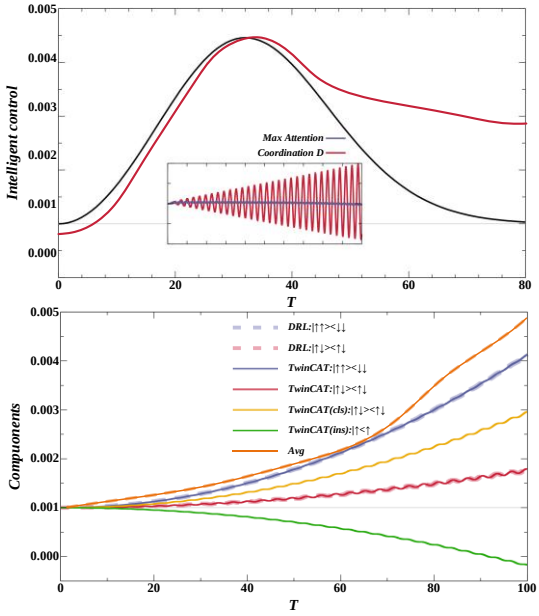

Fig. 4. Average weighted user energy consumption under different warehousing computing resources.

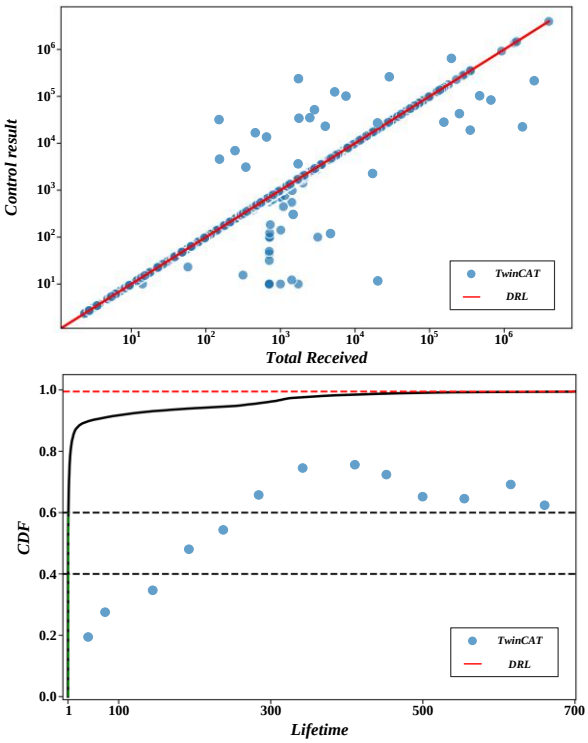

Fig. 5. Training results of different algorithms.

As can be seen from Fig. 5, the distributed training architecture uses many agents in multiple parallel systems to conduct distributed exploration, which can accelerate the process, improve agent training efficiency, and reduce computing overhead. Compared with centralized automatic control control (AGC) based on PPO and DDPG, distributed AGC based on deep reinforcement learning (DRL) has a better training effect and greatly increases training time. At the same time, PPO and DDPG algorithms have great volatility in the learning process, and the reward convergence value fluctuates, obviously. Hence, it is difficult to cope with random interference.

It can be seen from Fig. 6 that compared with DRL, PPO, DDPG and PID + PRPO, the frequency deviation at the same time point is smaller, and the frequency does not exceed 0.2 Hz during the control period. Because PID + PRPO cannot flexibly enable the quick response adjustment component to adjust the frequency, instability or over-adjustment problems will occur due to unreasonable parameter configuration, which may cause the frequency deviation to exceed 0.2 Hz and waste frequency adjustment resources.

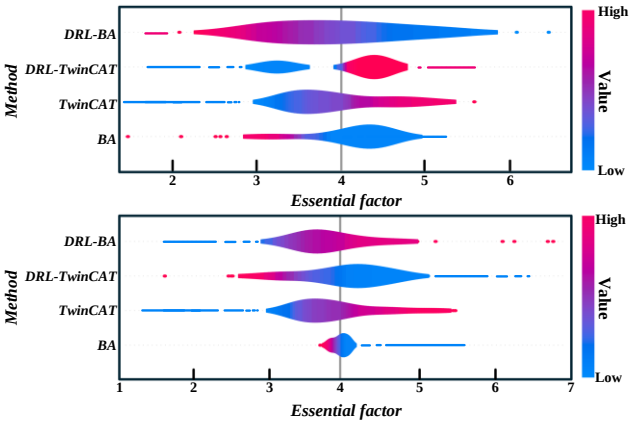

Fig. 6. Frequency deviation variation diagram of different algorithms.

It can be seen from Fig. 7 that the frequency control strategy can better fit the disturbance curve. Compared with the traditional PID + PRPO frequency modulation method, DRL technology's frequency modulation scheme performs better. DRL combines the essence of fuzzy control and neural network control, which can provide real-time feedback, adjust environmental changes during operation, and promote the control system to better adapt to the nonlinear dynamic environment. Compared with other DRL algorithms, it has smaller power tracking errors, can complete frequency adjustment as quickly as possible, and is more suitable for fluctuations in load and distributed resources in temperature and humidity automatic control systems.

It can be seen from Table II that each system index of the proposed strategy exceeds other strategies. Compared with other methods, this algorithm reduces the mean of Af by 19.3% to 58.1%, the mean of ACE by 12.6% to 77.4%, the mean of CPS1 by 0.15% to 28.6%, and the mean of CPS2 by 2.06% to 20.1%. By comparison, the controller has the smallest change in performance and the best adaptability and robustness.

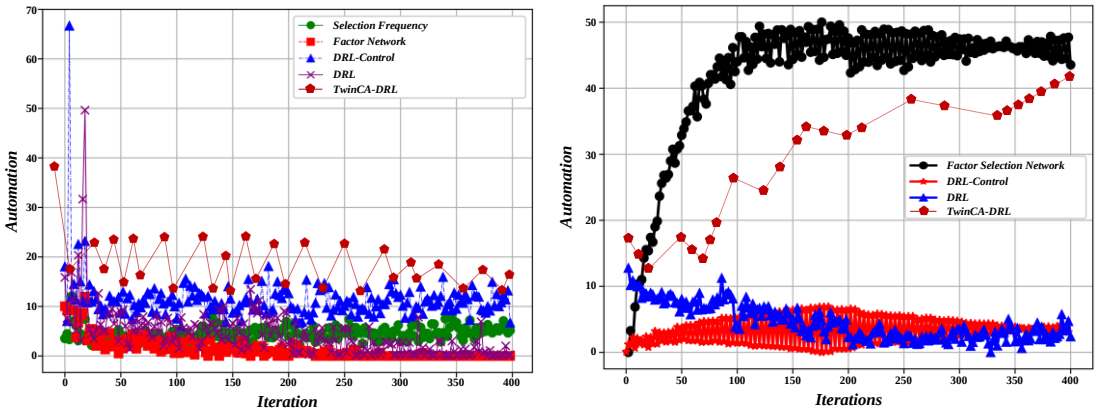

Fig. 7. Control effects of different strategies under white noise disturbance.

TABLE II MODEL STATISTICAL EXPERIMENTAL RESULTS UNDER RANDOM WHITE NOISE DISTURBANCE

Region	Index		DRL	PPO	DDPG	PI + PRPO
Region 1	Af/Hz	0.0694	0.0860	0.0912	0.1097	0.1689
	ACE/MW	13.3749	17.1171	21.1794	24.4013	44.2013
	CPS1/%	219.7602	219.4291	206.6361	203.8982	188.4960
	CPS2/%	106.1830	104.0160	98.2960	95.1940	88.3960
Region 2	Af/Hz	0.0563	0.0875	0.1159	0.1191	0.1838
	ACE/MW	11.1672	12.7006	19.9661	27.7134	44.9834
	CPS1/%	219.1354	219.0133	205.0983	201.8940	186.6480
	CPS2/%	104.8850	103.3780	97.2620	93.7310	87.5490

As shown in Fig. 8, the average algorithm score is 2.799. As the number of iterations increases and the exploration rate decreases, the average score of ERDQN algorithm grows slowly, while the average score of DRL-ERD3QN algorithm shows an upward trend, and its score rises rapidly in nearly 3000 rounds, which indicates that the network of this algorithm achieves rapid update in about 3000 rounds, while the neural network update speed of the other two algorithms is relatively slow.

As shown in Fig. 9, the DQ-DRL algorithm does not overestimate the agent policy learning. This shows that the algorithm effectively avoids the overestimation problem by taking two minimum values of Critic current network to update. Although this approach may lead to underestimation, underestimation is more acceptable than overestimation.

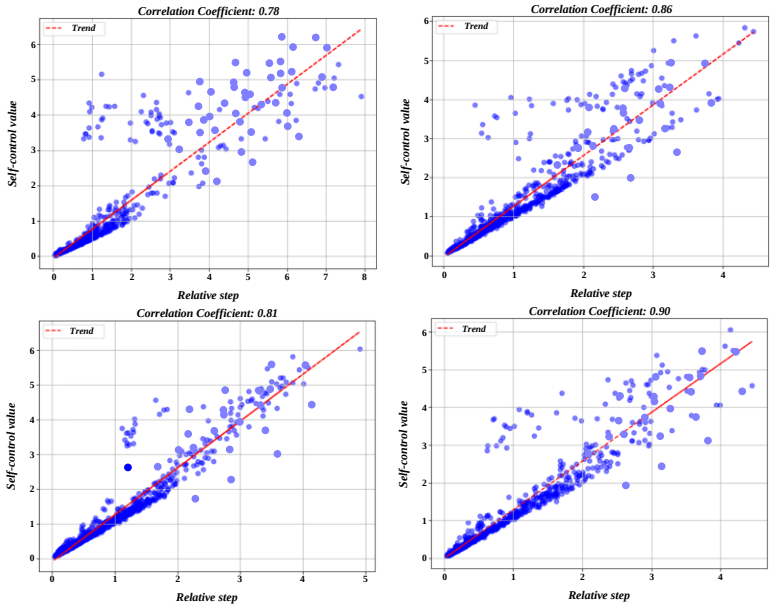

Fig. 8. Average score.

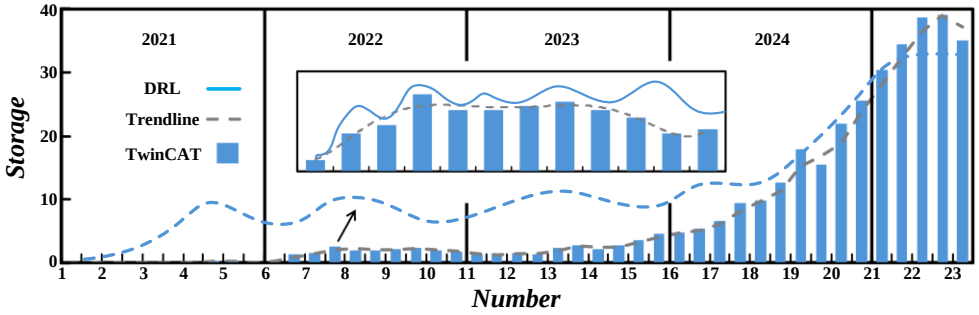

Fig. 9. Comparison of estimation bias.

Fig. 10 shows the changing trend of the cumulative total reward of the DQ-DRL algorithm and the follower agent trained by the DRL algorithm in each round when the group contains 4 units. It can be seen from the Fig. that in the early stage of training, the reward curves of both algorithms rise steadily. After about 35,000 rounds, the reward value of the DQ-DRL algorithm tends to stabilize and converge, while the reward curve of the DRL algorithm lags until about 55,000 rounds before converging. Moreover, the total reward value of the final convergence of the DQ-DRL algorithm is higher. This shows that although both algorithms allow the agent group to learn the stable formation control strategy, the DQ-DRL algorithm not only speeds up the convergence speed of the model but also improves the overall total reward of the agent group and enhances the superiority and robustness of the formation control model.

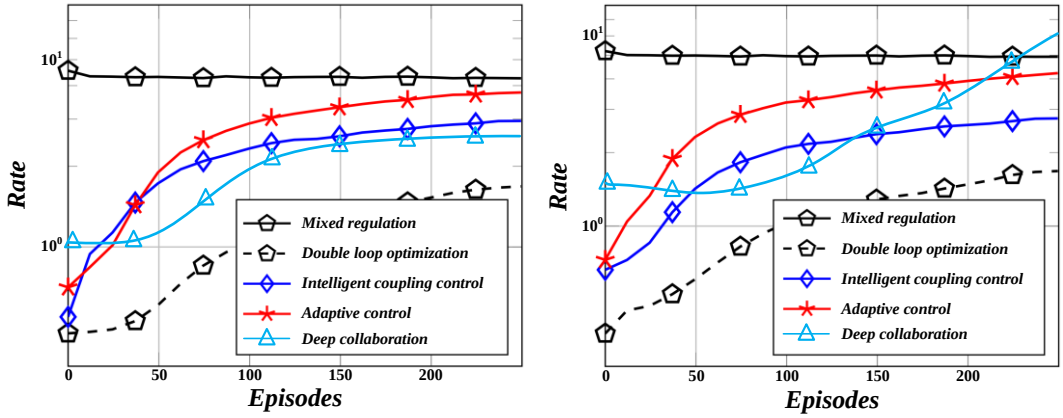

Fig. 10. Total reward curve obtained in each round during the training period.

V. Discussion

The integration of TwinCAT with Deep Reinforcement Learning (DRL) for temperature and humidity control in tobacco storage has led to a promising paradigm shift. Our results clearly show that this fusion not only solves the limitations of traditional control methods, such as poor adaptability to complex environments and poor accuracy, but also takes full advantage of the real-time processing power of TwinCAT and the adaptive decision-making capabilities of DRL to achieve better performance. The experimental results show that the temperature and humidity control error is reduced to $\pm 0.5^{\circ}$ C and $\pm 3\%$, the energy consumption is reduced by 20%, and the tobacco mildew rate is reduced by 15%, which highlights the practical value of the method. Our approach's ability to learn and adjust strategies in real-time is a key advantage compared to existing solutions such as PID control, which relies on fixed parameters and struggles to cope with dynamic changes, and fuzzy control that lacks self-optimizing capabilities. It is important to note that the success of the system depends on the seamless interaction between TwinCAT's industrial control infrastructure and the DRL algorithms, a synergy that enables the rapid acquisition, processing and execution of data, which is crucial in tobacco storage, where environmental fluctuations require a timely response. However, we also realize that the complexity of DRL model tuning and the need for sufficient training data may pose challenges to the widespread adoption of this method. Another aspect to consider is the versatility of our approach, and while it has been proven to be effective in tobacco storage, it is still worth exploring for its application to other agricultural or industrial storage scenarios, as different environments may have unique limiting factors. In addition, future improvements can focus on enhancing the robustness of the model to extreme weather conditions, as well as reducing the computational overhead of the DRL component, making it easier to apply in small storage facilities. Overall, this study highlights the potential of combining industrial control systems with advanced machine learning techniques to revolutionize storage environment management, providing a balance of precision, efficiency, and adaptability that is difficult to achieve with traditional methods.

VI. Conclusion

This study explores the optimal application of fusing TwinCAT and deep reinforcement learning (DRL) technology in tobacco storage environments' automatic temperature and humidity control systems. By constructing an intelligent control system, the high-precision and high-efficiency regulation of the tobacco storage environment can be realized, thus ensuring the quality of tobacco and reducing storage losses.

(1) Firstly, traditional temperature and humidity control system performance is benchmarked. The results show that under the unoptimized conditions, the control accuracy of the system for temperature and humidity is ± 2 ℃ and ± 5% RH, respectively, and there is obvious hysteresis. This result reveals the limitations of conventional control systems in coping with complex environmental changes.

(2) The system is preliminarily optimised by introducing the TwinCAT platform and using its powerful real-time control and data acquisition capabilities. The experimental results show that through the integrated management of TwinCAT, the control accuracy of the system has been significantly improved, and the control accuracy of temperature and humidity has been improved to ± 1 ℃ and ± 3% RH, respectively. At the same time, the system's response speed has been accelerated, and the lag phenomenon has been effectively alleviated.

(3) To further improve the system performance, this study introduces DRL technology and constructs an intelligent control

model based on deep reinforcement learning. The model achieves accurate prediction and adaptive regulation of temperature and humidity changes by continuously learning environmental data and control strategies. The final experimental results show that the system combining TwinCAT and DRL can stabilize the control accuracy of temperature and humidity within ± 0. 5 ℃ and ± 2% RH, respectively, and almost eliminate the hysteresis phenomenon. In addition, the system's energy consumption has also been effectively reduced, saving about 15% of energy consumption compared with traditional systems.

The tobacco storage environment's automatic temperature and humidity control system integrating TwinCAT and DRL shows excellent performance and great application potential. This study provides a new technical path for the intelligent management of tobacco storage environment and a useful reference for optimising automatic control systems in related fields. In the future, we will continue to deepen research, explore more innovative applications, and contribute more to the development of tobacco storage and other related industries.

The existing solutions include traditional PID control (fixed parameters, difficult to resist interference), fuzzy control (insufficient dynamic response and optimization), and simple machine learning combined control (lack of deep integration of industrial systems). These schemes generally have the problems of insufficient integration of industrial control and intelligent algorithms, weak ability to adapt to complex environments, and poor accuracy and energy efficiency. In this study, TwinCAT and deep reinforcement learning are integrated, which not only gives full play to the advantages of real-time control of the former, but also improves the adaptability and optimization capabilities of the latter, makes up for the shortcomings of the existing schemes, and forms a better control scheme.

ACKNOWLEDGMENT

This work was supported by ETSONG Tobacco (Group) Co., Ltd, Research and application of automatic control system based on nitrogen-filled pest control technology, contract number: 2025370200740017.

REFERENCES

[1] J. Gao, W. Xie, D. Shi, J. Wu, and R. Wang, “Synchronized optimization of the logistics system of a tobacco high-bay warehouse under production task fluctuations,” Engineering Research Express, vol. 6, no. 4, 2024.

[2] H. Zhou, J. Wu, X. Zheng, H. Zhu, G. Lu, Y. Zhang, and Z. Shen, “Fuzzy-PID controller based on improved LFPSO for temperature and humidity control in a CA ripening system,” Journal of Food Process Engineering, vol. 47, no. 6, 2024.

[3] Y. Sun, C. Liu, G. Du, H. Wang, and Z. Tian, “An almost disturbance decoupling temperature and humidity control strategy of air-handling units,” International Journal of Control, vol. 98, no. 1, pp. 1-17, 2025.

[4] S. Nakayama, T. Takada, R. Kimura, and M. Ohsumi, “Model Predictive Control of Humidity Deficit and Temperature in Winter Greenhouses: Subspace Weather-Based Modelling and Sampling Period Effects,” Machines, vol. 12, no. 1, 2024.

[5] A. Sterk-Hansen, B. H. H. Saghaug, D. Hagen, M. F. Aftab, and Ieee, "A ROS 2 and TwinCAT Based Digital Twin Framework for Mechatronics Systems," International Conference on Control Mechatronics and Automation. pp. 485-490, 2023.

[6] E. Okafor, D. Udekwe, Y. Ibrahim, M. B. Mu'azu, and E. G. Okafor, “Heuristic and deep reinforcement learning-based PID control of trajectory tracking in a ball-and-plate system,” Journal of Information and Telecommunication, vol. 5, no. 2, pp. 179-196, 2021.

[7] M. B. Mohiuddin, I. Boiko, R. Azzam, and Y. Zweiri, “Closed-loop stability analysis of deep reinforcement learning controlled systems with experimental validation,” Iet Control Theory and Applications, vol. 18, no. 13, pp. 1649-1668, 2024.

[8] Z. Liu, C. Wang, and W. Wang, “Online Cyber-Attack Detection in the Industrial Control System: A Deep Reinforcement Learning Approach,” Mathematical Problems in Engineering, vol. 2022, 2022.

[9] Y. Liu, Y. Ni, C. Dong, J. Chen, and F. Liu, “Task scheduling for control system based on deep reinforcement learning,” Neurocomputing, vol. 610, 2024.

[10] Q. Liu, T. Xia, L. Cheng, M. van Eijk, T. Ozcelebi, and Y. Mao, “Deep Reinforcement Learning for Load-Balancing Aware Network Control in IoT Edge Systems,” Ieee Transactions on Parallel and Distributed Systems, vol. 33, no. 6, pp. 1491-1502, 2022.

[11] D. Liu, Y. Wu, Y. Kang, L. Yin, X. Ji, X. Cao, and C. Li, “Multi-agent quantum-inspired deep reinforcement learning for real-time distributed generation control of 100% renewable energy systems,” Engineering Applications of Artificial Intelligence, vol. 119, 2023.

[12] D. Lin, and Y. Liu, “Discrete phase shifts control and beam selection in RIS-aided MISO system via deep reinforcement learning,” China Communications, vol. 20, no. 8, pp. 198-208, 2023.

[13] R. Liang, H. Lyu, and J. Fan, “A Deep Reinforcement Learning-Based Power Control Scheme for the 5G Wireless Systems,” China Communications, vol. 20, no. 10, pp. 109-119, 2023.

[14] X. Li, X. Zhang, J. Li, F. Luo, Y. Huang, and X. Zhang, “Blocklength Allocation and Power Control in UAV-Assisted URLLC System via Multi-agent Deep Reinforcement Learning,” International Journal of Computational Intelligence Systems, vol. 17, no. 1, 2024.

[15] X. Wang, X. Wang, J. Wu, K. Zheng, Y. Pang, and S. Gang, “Research on quality traceability of cigarette by combining PDCA quality cycle with information strategy based on fuzzy classification,” Journal of Intelligent & Fuzzy Systems, vol. 40, no. 4, pp. 8217-8226, 2021.

[16] H. Liu, Q. Chen, N. Pan, Y. Sun, Y. An, and D. Pan, “UAV Stocktaking Task-Planning for Industrial Warehouses Based on the Improved Hybrid Differential Evolution Algorithm,” Ieee Transactions on Industrial Informatics, vol. 18, no. 1, pp. 582-591, 2022.

[17] Y. Xia, K. Chang, Y. Lin, and C. Zhu, “Feedforward decoupling control of indoor temperature and humidity using a single direct expansion air conditioning unit,” Science and Technology for the Built Environment, vol. 30, no. 6, pp. 626-643, 2024.

[18] S.-C. Vanegas-Ayala, J. Baron-Velandia, and D.-D. Leal-Lara, “A Systematic Review of Greenhouse Humidity Prediction and Control Models Using Fuzzy Inference Systems,” Advances in Human-Computer Interaction, vol. 2022, 2022.

[19] Sunardi, A. Yudhana, and Furizal, “Tsukamoto Fuzzy Inference System on Internet of Things-Based for Room Temperature and Humidity Control,” Ieee Access, vol. 11, pp. 6209-6227, 2023.

[20] Y. Sun, Y. Zhang, D. Guo, X. Zhang, Y. Lai, and D. Luo, “Intelligent Distributed Temperature and Humidity Control Mechanism for Uniformity and Precision in the Indoor Environment,” Ieee Internet of Things Journal, vol. 9, no. 19, pp. 19101-19115, 2022.

[21] S. Liu, X. Liu, T. Zhang, C. Wang, and W. Liu, “Joint optimization for temperature and humidity independent control system based on multi-agent reinforcement learning with cooperative mechanisms,” Applied Energy, vol. 375, 2024.

[22] C. Li, Y. Cheng, and X. Hou, “Humidity Diffusion Process Analysis and Life Prediction of a VSC-HVDC Control Protection Device Based on a Finite Element Simulation Method,” Electronics, vol. 13, no. 15, 2024.

[23] Y. Jiang, S. Zhu, Q. Xu, B. Yang, and X. Guan, “Hybrid modeling-based temperature and humidity adaptive control for a multi-zone HVAC system,” Applied Energy, vol. 334, 2023.

[24] D. Guo, D. Luo, Y. Zhang, X. Zhang, Y. Lai, and Y. Sun, “Application of deep reinforcement learning to intelligent distributed humidity control system,” Applied Intelligence, vol. 53, no. 13, pp. 16724-16746, 2023.

[25] F. Garcia-Manas, T. Hagglund, J. L. Guzman, F. Rodriguez, and M. Berenguel, “A practical solution for multivariable control of temperature

and humidity in greenhouses," European Journal of Control, vol. 77, 2024.

[26] M. Dong, J. Zhang, L. Zhang, L. Liu, and X. Zhang, "Research on Relative Humidity and Energy Savings for Air-Conditioned Spaces without Humidity Control When Adopting Air-to-Air Total Heat Exchangers in Winter," Buildings, vol. 14, no. 4, 2024.

[27] M. H. Demir, S. Cetin, O. Haggag, H. G. Demir, W. Worek, J. Premer, and D. Pandelidis, "Independent temperature and humidity control of a precooled desiccant air cooling system with proportional and fuzzy logic plus proportional based controllers," International Communications in Heat and Mass Transfer, vol. 139, 2022.

[28] J. Zhou, L. Li, A. Vajdi, X. Zhou, and Z. Wu, "Temperature-Constrained Reliability Optimization of Industrial Cyber-Physical Systems Using Machine Learning and Feedback Control," Ieee Transactions on Automation Science and Engineering, vol. 20, no. 1, pp. 20-31, 2023.

[29] H. Zhou, J. Wu, X. Zheng, H. Zhu, G. Lu, Y. Zhang, and Z. Shen, "Fuzzy-PID controller based on improved LFPSO for temperature and humidity control in a CA ripening system," Journal of Food Process Engineering, vol. 47, no. 6, 2024.

[30] Q. Zhong, E. Xu, G. Xie, X. Wang, and Y. Li, "Dynamic performance and temperature rising characteristic of a high-speed on/off valve based on pre-excitation control algorithm," Chinese Journal of Aeronautics, vol. 36, no. 10, pp. 445-458, 2023.