

Article

Development of Chiller Plant Models in OpenAI Gym Environment for Evaluating Reinforcement Learning Algorithms

Xiangrui Wang ¹, Qilin Zhang ¹ , Zhihua Chen ², Jingjing Yang ¹ and Yixing Chen ^{1,3,*} ¹ College of Civil Engineering, Hunan University, Changsha 410082, China;

wxr18684748039@hnu.edu.cn (X.W.); qilinzhang@hnu.edu.cn (Q.Z.); yang226@hnu.edu.cn (J.Y.)

² Department of Building Science, School of Architecture, Tsinghua University, Beijing 100084, China; chenzhihua@tsinghua.edu.cn³ Key Laboratory of Building Safety and Energy Efficiency of Ministry of Education, Hunan University, Changsha 410082, China

* Correspondence: yixingchen@hnu.edu.cn

Abstract: To face the global energy crisis, the requirement of energy transition and sustainable development has emphasized the importance of controlling building energy management systems. Reinforcement learning (RL) has shown notable energy-saving potential in the optimal control of heating, ventilation, and air-conditioning (HVAC) systems. However, the coupling of the algorithms and environments limits the cross-scenario application. This paper develops chiller plant models in OpenAI Gym environments to evaluate different RL algorithms for optimizing condenser water loop control. A shopping mall in Changsha, China, was selected as the case study building. First, an energy simulation model in EnergyPlus was generated using AutoBPS. Then, the OpenAI Gym chiller plant system model was developed and validated by comparing it with the EnergyPlus simulation results. Moreover, two RL algorithms, Deep-Q-Network (DQN) and Double Deep-Q-Network (DDQN), were deployed to control the condenser water flow rate and approach temperature of cooling towers in the RL environment. Finally, the optimization performance of DQN across three climate zones was evaluated using the AutoBPS-Gym toolkit. The findings indicated that during the cooling season in a shopping mall in Changsha, the DQN control method resulted in energy savings of 14.16% for the cooling water system, whereas the DDQN method achieved savings of 14.01%. Using the average control values from DQN, the EnergyPlus simulation recorded an energy-saving rate of 10.42% compared to the baseline. Furthermore, implementing the DQN algorithm across three different climatic zones led to an average energy savings of 4.0%, highlighting the toolkit's ability to effectively utilize RL for optimal control in various environmental contexts.

Keywords: reinforcement learning; chiller plant; OpenAI Gym; AutoBPS; optimal control

Academic Editor: Chun-Cheng Lin

Received: 20 March 2025

Revised: 23 April 2025

Accepted: 25 April 2025

Published: 27 April 2025

Citation: Wang, X.; Zhang, Q.; Chen, Z.; Yang, J.; Chen, Y. Development of Chiller Plant Models in OpenAI Gym Environment for Evaluating Reinforcement Learning Algorithms. *Energies* **2025**, *18*, 2225. <https://doi.org/10.3390/en18092225>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The buildings and construction sector accounts for 32% of global energy consumption, making improvements in energy efficiency crucial for mitigating the global energy crisis [1]. However, improving energy efficiency through conventional approaches, such as code adoption and minimum performance standards, is difficult. The challenge underscores the importance of controlling building energy management systems (BEMS) [2]. The optimal control of HVAC plays a critical role in advanced BEMS, as the end-use consumption for HVAC systems accounts for 50% of building operational energy [3]. For large-scale commercial buildings, more than 50% of the energy consumed by HVAC systems is concentrated in the chiller plant systems. As the facilities are designed according to peak building

load but always operate under partial load, the chiller plant system usually operates in a sub-optimal state [4]. Therefore, an amount of energy can be saved during the operation of the chiller plant system. A chiller plant system consists of two main loops: a chilled water loop and a condenser water loop, including core components such as chillers, water pumps, and cooling towers. The condenser water loop significantly impacts the overall operation energy cost of the chiller plant system [5]. Maintaining optimal operating conditions for each device in the condenser water loop is worth studying.

Different control methods were applied to optimize the control strategies of cooling water systems, and it was found that energy and cost can be saved by controlling the operation parameters. Adjusting the speed of cooling tower fans, varying the condenser water flow rate, and setting the water outlet temperature of cooling towers are measures that can be used to optimize the performance of the condenser water loop. Kim et al. [6] explored the performance of predictive control on optimizing condenser water setpoint temperature, and total cooling energy consumption was saved by 5.6%. Huang et al. [7] also studied the optimization of the condenser water setpoint temperature of the cooling water system. Model predictive control (MPC) was utilized for a real legacy chiller plant, and annual energy consumption savings achieved by chillers and cooling towers were up to around 9.67%. The performance of MPC relies on the model's accuracy, but constructing a complex model is full of challenges. The generalization of a built model is weak, and the model cannot be easily used in another situation. Control methods based on data-driven methods reduce the dependence on physical models. Wang et al. [8] applied random forest to achieve the ground optimization of a chiller plant system and proposed a stepwise optimization strategy. The strategy obtained an energy-saving rate of 6.41% and 13.56% on two research days. The parameters optimized on the cooling side consisted of the cooling water outlet temperature and the cooling water flow. Ma et al. [9] proposed a hybrid programming particle swarm optimization (HP-PSO) algorithm to reduce the energy consumption of cooling water systems. By adjusting the number of chillers and pumps, the water mass flow rate of a single pump, and the air mass flow rate of the cooling tower, the energy consumption of the cooling water system was reduced by 15.3% compared with the rule-based constant temperature difference optimization. However, these data-driven control approaches require quality and quantity of historical data. Reinforcement learning (RL) is a model-free control method that does not depend on an accurate model and has low requirements for historical data and prior knowledge. It is suitable for optimizing the control of building energy systems. Qiu et al. [10] studied the performance of RL in a chilled water system for a Guangzhou subway station. By adjusting the frequency of the pumps and the cooling tower fans in the cooling water loop, the control of the chilled water outlet temperature setpoint was realized, and the energy saving rate was stabilized to 12% after the second applied cooling season. Fu et al. [11] proposed a Multi-Agent deep RL method for the building cooling water system; the performance of pumps, chillers, and cooling towers was optimized by adjusting the load distribution, cooling tower fan frequency, and cooling water pump frequency. Compared with the rule-based control method, the RL method showed an 11.1% improvement in energy-saving performance. Studies demonstrated that RL performed well in the optimal control of the cooling water system.

More and more advanced RL algorithms were utilized to optimize the operation of HVAC systems. The Q-learning algorithm is the most classic RL algorithm. Chen et al. [12] optimized the on/off strategy of the air-conditioner and window by utilizing the Q-learning algorithm in two residential zones. In Miami and Los Angeles, the energy consumption saving rate compared with heuristic control reached 13% and 23%, respectively. The deep-Q-network (DQN) algorithm is an improvement of Q-learning. In the DQN algorithm, the traditional Q-table used for storing and updating values based on actions and states

was replaced by a neural network. Ahn et al. [13] used a DQN method to minimize the energy consumption of a building while maintaining the indoor CO₂ concentration; a 14.9% energy saving potential was found between DQN and baseline operation. To overcome problems existing in DQN, such as the fact that it can only solve problems on a discrete action space, the deep deterministic policy gradient (DDPG) algorithm was developed. Du et al. [14] utilized the DDPG algorithm to optimally control the HVAC system of a multi-zone residential building and verified its advantages over the DQN. Peng et al. [15] proposed an enhanced DDPG to optimize the energy consumption predicted by a convolutional neural network and long short-term memory (CNN-LSTM) model; they compared the energy efficiency ratio with a proximal policy optimization (PPO) algorithm and reached a 49% performance improvement. More RL algorithms were proposed and used to optimize the control of building energy management systems by simulation. Fu et al. [16] integrated MPC and twin delayed deep deterministic policy gradient (TD3) to optimize the energy consumption of an HVAC system and demonstrated a 16% cost-saving performance better than DDPG. These studies focused on developing more advanced algorithms to improve the convergence speed of RL and control performance. However, constructing advanced RL algorithms for different HVAC systems requires the researcher to have proficient programming skills and considerable knowledge of specialized knowledge. Moreover, the algorithms are related to the HVAC systems the researchers built, resulting in a lack of universality for other applications.

Many open-source RL algorithms developed in other fields are accessible through GitHub (<https://github.com>, accessed on 19 March 2025). Although these algorithms are not developed for optimal control in the HVAC field, they can still simplify developing RL algorithms in HVAC systems. Biemann et al. [17] demonstrated a way to use a library of open-source RL algorithms for optimal control of HVAC systems. They compared the performance of four different actor–critic algorithms (soft actor–critic (SAC), TD3, PPO, and trust-region policy optimization (TRPO)) in the HVAC system of a data center. The algorithms utilized by Biemann came from the Stable-Baselines3 library [18].

OpenAI Gym provides a framework for constructing an RL environment and contains a collection of environments [19]. With the OpenAI Gym Application Programming Interfaces (APIs), agents with different open-source RL algorithms can be deployed to interact with the environment. Moriyama et al. [20] wrapped an EnergyPlus program into an OpenAI Gym environment and used the TRPO algorithm as the agent. The data center's cooling system was optimized, and the controller performance obtained by RL was 22% higher than a built-in controller. Zhang et al. [21] used the same method to construct an OpenAI Gym environment as Moriyama. The RL controller was deployed for a radiant heating system in a one-floor office building and reduced heating demand by 16.7% compared with a rule-based controller. Arroyo et al. [22] reported a framework coupling the OpenAI Gym environment and building optimization performance tests (BOPTEST). A floor heating system for a single-zone residential building was chosen as the testing case for the framework. Wang et al. [23] and Chen et al. [24] developed two platforms based on the OpenAI Gym interactive interface to analyze and evaluate the performance of regional H₂-electricity network systems and building electric vehicle systems. These virtual testbeds, constructed based on OpenAI Gym, provided a more convenient way to analyze and optimize the control of BEM systems. However, most environments were coupled with other software, such as EnergyPlus. The co-simulation during the RL training process costs a huge amount of computing power and requires a long time for simulation. Moreover, for the optimization of the sub-system of the HVAC system, it is not necessary to calculate all of the operation states of a whole building.

In summary, RL is an appropriate optimization control method for complex systems such as HVAC. However, developing specialized RL algorithms for HVAC systems is a challenge. Utilizing open-source RL algorithms to control HVAC systems is a new way, but it usually requires co-simulation with other software, resulting in a long training cycle.

Motivated by the interactive environment framework of OpenAI Gym and the function of the automated building performance simulation (AutoBPS) platform, this paper aims to develop a tool that can quickly generate interactive environments for HVAC systems and realize a simplified application of open-source RL algorithms to HVAC systems. A cooling water system in the chiller plant system was selected as the study objective, and a toolkit called AutoBPS-Gym was developed. Based on the building energy models generated by AutoBPS between different building types and climate zones, interactive environments were generated and combined with RL algorithms to test and explore the energy-saving potential of cooling water. In addition, the developed toolkit reduces repetitive modeling work in traditional studies on RL and effectively alleviates the computational power demand in the traditional co-simulation process. Furthermore, it reduces the time consumed in the training process of RL.

The rest of the paper is organized as follows: Section 2 describes the details of the development of AutoBPS-Gym, including parameters of case building, physical modeling of equipment, and related information for the environment and RL. The results of energy-saving performance and control strategies obtained by the developed tool when optimizing cooling water systems are demonstrated in Section 3. Section 4 discusses some shortcomings in developing the tool, and Section 5 concludes the results achieved in this study.

2. Materials and Methods

Figure 1 illustrates the workflow of the developed AutoBPS-Gym, divided into three parts. First, AutoBPS was used to generate building EnergyPlus models. The necessary parameters, including temperature, hourly water flow rate, and sizing parameters of HVAC facilities, were obtained by running the generated models. Second, an OpenAI Gym (Gymnasium v1.1.1) environment of the cooling water system was constructed based on these parameters. The environment comprised three main equipment: chillers, pumps, and cooling towers. These models were used to simulate the operation of the cooling water system, calculate energy consumption, and obtain the states. Third, agents based on different RL algorithms were developed to interact with the constructed environment. Control strategies were obtained by exploration in the action space according to the states and rewards.

2.1. Building Models Generation

In this section, the tool for generating building energy models is first introduced. The thermal performance parameters of the case building model generated by AutoBPS are shown, and the parameters of the cooling water system in the case building are presented.

2.1.1. Modeling Tool: AutoBPS

AutoBPS is an urban building energy modeling tool developed by Hunan University, China [25]. With four function modules—AutoBPS-GEO, AutoBPS-OSS, AutoBPS-RM, and AutoBPS-PV—AutoBPS can be used to calculate building energy demand, realize building energy retrofit, and explore the potential of rooftop photovoltaic. Different toolkits were developed to expand the function of the tool. Chen et al. [3] developed AutoBPS-BIM to transfer the building information model (BIM) to the building energy model (BEM). AutoBPS-Param was developed by Chen et al. [26] and Xi et al. [27] to create BEMs using building parameterized geometry data. Peng et al. [28] proposed AutoBPS-DR to integrate

demand response strategies. Yang et al. [29] considered the user-friendliness of AutoBPS and developed the toolkit called AutoBPS-Prototype to automatically establish prototype-building energy models in China. With the climate zone, building type, and standard vintage defined, AutoBPS-Prototype can generate a default prototype BEM based on a pre-built library. This study used the AutoBPS-Param toolkit to generate a building energy model of the typical shopping mall in Changsha.

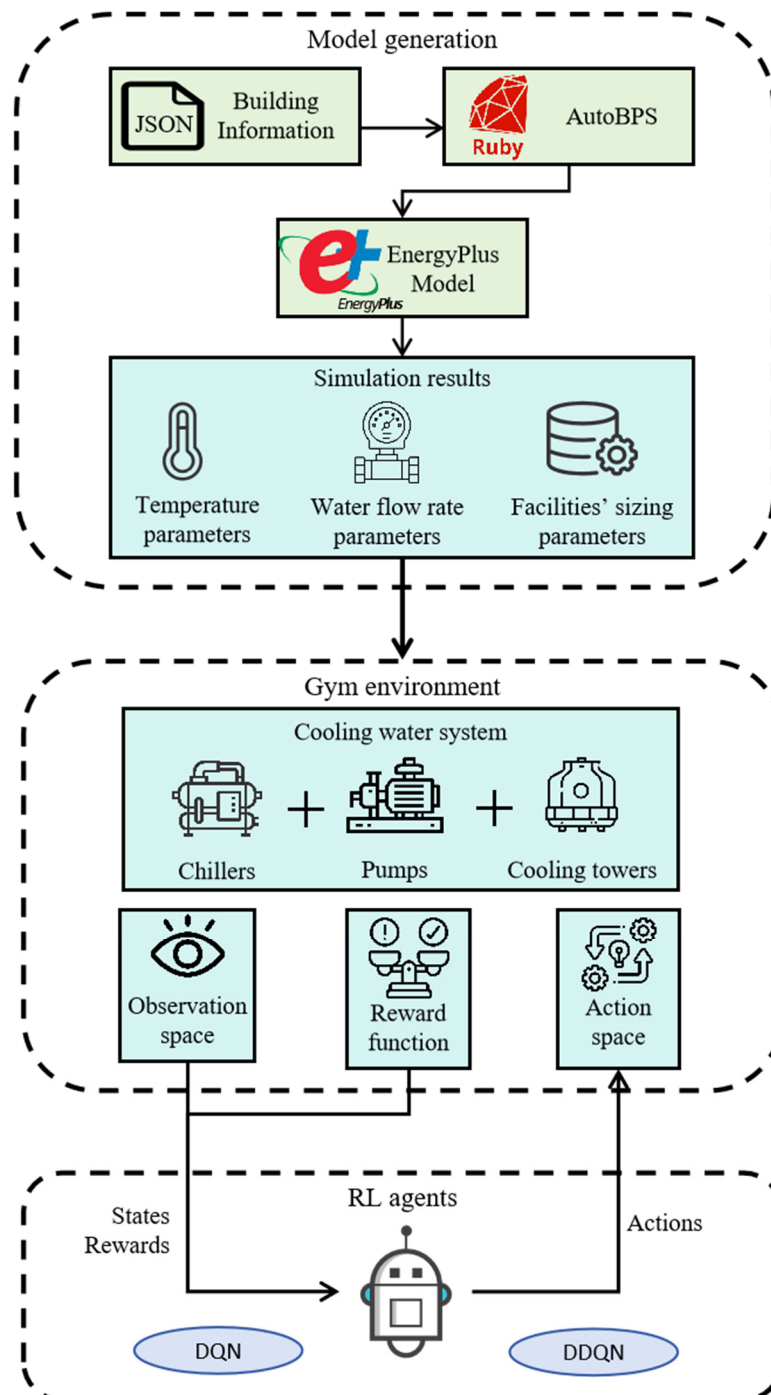


Figure 1. Flow chart of AutoBPS-Gym.

2.1.2. Case Study Building Model

A shopping mall in Changsha was selected as the case building to test the environment generated by AutoBPS-Gym; the climate in Changsha is described as a hot summer and cold winter with high humidity. This shopping mall has a footprint of 29,888 m² and a

net conditioned building area of 149,002.50 m². It has two stories of parking basement and five stories of above-ground space, with each floor having a height of 4.7 m. The windows in the model were distributed in the bottom story and were simulated by setting the windows-to-wall ratios (WWR) at 0.35 on the east, 0.56 on the south, 0.35 on the west, and 0.3 on the north. Table 1 presents detailed parameters of the building envelope and room configurations.

Table 1. Details of building parameters.

Parameters		Shopping Mall	
U-factor (W/(m ² ·K))		Exterior wall	0.59
		Roof	0.38
		Glass	2.50
Solar heat gain coefficient of glass		0.40	
Internal load value			
Room type	Lights (W/m ²)	Equipment (W/m ²)	People (person/m ²)
Corridor	4.95	1.00	0.07
Parking	5.00	1.00	0.05
Office	11.00	20.00	0.07
Clothing	12.00	13.00	0.13
Cinema	12.00	13.00	0.20
Food	13.00	13.00	0.10
Supermarket	13.00	13.00	0.10
Entertainment	18.00	5.00	0.20

2.1.3. Cooling Water System

A central cooling water system was used to handle the cooling load from the shopping mall building. The system consists of a water-cooled centrifugal chiller, a fan-cycling open cooling tower, and a variable-speed pump. The period for the simulated operation of the cooling water system lasted from 1 June to 30 September. The hourly operational status of the cooling water system was simulated, resulting in a total of 2928 time steps. AutoBPS automatically calculated the cooling load that needed to be handled, and the parameters of the facilities were sized. The total energy consumption of this cooling water system during the cooling season is 4.65 GWh, with an average building area energy consumption of 31.19 kWh/m². The equipment models and parameters are provided in Table 2.

Table 2. Parameters of cooling water system facilities.

Parameters		Value	Unit
Chiller	Cooling capacity	21.80	MW
	Chilled water flow rate	0.82	m ³ /s
	Condenser water flow rate	1.10	m ³ /s
	Coefficient of Performance (COP)	5.50	W/W
Condenser pump	Water flow rate	1.10	m ³ /s
	Power	281.21	kW
Cooling tower	Water flow rate	1.10	m ³ /s
	Air flow rate	847.25	m ³ /s
	Rated power	222.18	kW
	Nominal capacity	29.78	MW

2.2. Physical Model Construction

This section introduces the modeling process of the cooling water system. This study focused on the behavior of the main plant (chiller, pump, and cooling tower), and the auxiliary equipment and pipes in the cooling water system were simplified.

2.2.1. Chiller Model

The Department of Energy (DOE) provided the compression chiller model (COMREF) in their building energy simulation program; the detailed chiller model can be found in the file Engineering Reference provided by EnergyPlus. DOE models were referred to and integrated with the framework of the OpenAI Gym to develop the chiller model.

Since the user-supplied chiller parameters are typically the nameplate parameters at rated operating conditions, and it is not easy to operate the chiller under such conditions, the core of the chiller model provided by DOE is the calculation of non-rated performance using three curves: Cooling Capacity Function of Temperature Curve, Energy Input to Cooling Output Ratio Function of Temperature Curve, and Energy Input to Cooling Output Ratio Function of Part Load Ratio Curve. These three curves were modeled in the chiller model shown in Figure 2, and the needed parameters were listed.

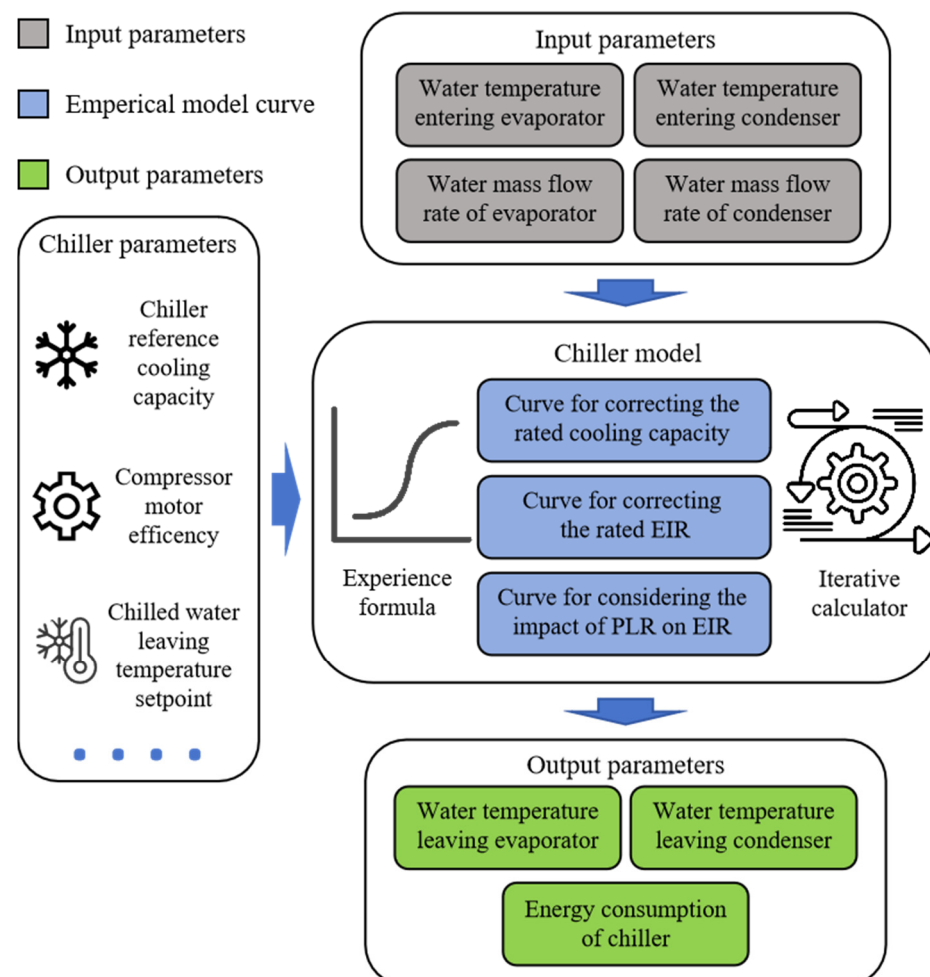


Figure 2. Correlation of input and output parameters in the chiller model.

When simulating the performance of a chiller, a factor was obtained to calculate the available cooling capacity of the chiller under operating conditions, firstly. After determining whether the water outlet temperature of the chiller could reach the setpoint temperature, the energy input to cooling output ratio (EIR) under the actual operating

condition of the chiller was calculated according to the actual water outlet temperature of the chiller. Then, the part-load ratio (PLR) was calculated according to the actual cooling load afforded by the chiller, and the chiller power was calculated under the simulation condition. The compressor motor efficiency and false loading were also considered to calculate the actual water temperature leaving the condenser. False loading is assumed to occur whenever the PLR is below the minimum unloading ratio specified.

The chiller model was then described as a function of four input variables to calculate the output parameters, such as the chilled water temperature leaving the evaporator and condenser, and the energy consumed by the chiller compressor. The three curves are fitted by empirical formulas, which can be used to calculate the power of the chiller and the outlet temperature of the condenser.

$$T_{chw,l}, T_{cond,l}, E_{chiller} = f(T_{chw,e}, T_{cond,e}, m_{evap}, m_{cond}) \quad (1)$$

$$\left. \begin{aligned} CapFTemp &= a_1 + b_1(T_{chw,ls}) + c_1(T_{chw,ls})^2 + d_1(T_{cond,e}) + e_1(T_{cond,e})^2 + f_1(T_{chw,ls})(T_{cond,e}) \\ EIRFTemp &= a_2 + b_2(T_{chw,l}) + c_2(T_{chw,l})^2 + d_2(T_{cond,e}) + e_2(T_{cond,e})^2 + f_1(T_{chw,l})(T_{cond,e}) \\ EIRFPLR &= a_3 + b_3(PLR) + c_3(PLR)^2 \\ PLR &= Q_{evap} / Q_{ref} \cdot CapFTemp \\ P_{chiller} &= \frac{Q_{ref} \cdot CapFTemp}{COP_{ref}} \cdot EIRFTemp \cdot EIRFPLR \\ T_{cond,l} &= T_{cond,e} + \frac{P_{chiller} \cdot eff_{motor} + Q_{evap}}{m_{cond} \cdot C_{p,cond}} \\ E_{chiller} &= P_{chiller} \cdot TimeStep \cdot 3600 \end{aligned} \right\} \quad (2)$$

where: a_i to f_i are the coefficients in the curve-fitting equation, Q_{evap} is the cooling load that needs to be disposed of by the chiller, W; Q_{ref} is the cooling capacity of the chiller under the reference condition, W; $T_{chw,ls}$ is the chilled water setpoint temperature leaving the evaporator (between 4.44 to 8.89 °C, with the default value being 6.67 °C); $T_{chw,l}$ is the chilled water temperature leaving the evaporator (between 4.44 to 8.89 °C); $P_{chiller}$ is the power of the chiller in the simulation operating condition, W; COP_{ref} is the rated coefficient of performance of the chiller; $T_{cond,l}$ is the condenser water temperature leaving the condenser, °C; $E_{chiller}$ is the energy consumed by the chiller compressor, J; $T_{chw,e}$ is the chilled water temperature entering the evaporator, °C; $T_{cond,e}$ is the condenser water temperature entering the condenser (between 26.23 to 39.12 °C); m_{evap} is the evaporator mass flow rate, kg/s; m_{cond} is the condenser mass flow rate, kg/s; eff_{motor} is the motor efficiency of the chiller compressor; and $TimeStep$ is equal to the simulation step of EnergyPlus.

2.2.2. Pump Model

In this study, the pumps were modeled as variable-speed pumps according to the pump model provided by DOE. The PLR primarily determines the performance of a pump. The model constructed a function to represent the performance curve when the pump operates under a partial load condition, and the fraction of full-load power could be calculated with the given PLR. Figure 3 lists the pump model's input and output parameters, with the model's key steps also demonstrated.

The inputs of a pump model are the water temperature entering the pump and the mass flow rate. With the PLR calculated, the curve determines the pump's power and shaft power consumed. Heat generation due to pump friction was also considered, resulting in a temperature rise between the outlet and inlet temperatures of the pump.

The pump model was described as a function of two input variables, which calculated the water temperature leaving the pump and the energy consumed by the pump.

$$T_{water,l}, E_{pump} = f(T_{water,e}, m_{water}) \quad (3)$$

$$\left. \begin{aligned}
 PLR &= \frac{m_{water}/\rho_{water}}{m_{v,design}} \\
 FracFullLoadPower &= C_1 + C_2 \cdot PLR + C_3 \cdot PLR^2 + C_4 \cdot PLR^3 \\
 P_{pump} &= FracFullLoadPower \cdot P_{pump,design} \\
 T_{water,l} &= T_{water,e} + \frac{P_{pump} \cdot eff_{motor} \cdot (1 - eff_{pump})}{m_{water} \cdot C_{p,water}} \\
 E_{pump} &= P_{pump} \cdot TimeStep \cdot 3600
 \end{aligned} \right\} \quad (4)$$

where C_1 , C_2 , C_3 , and C_4 are the coefficients of the empirical formula of pump performance; $T_{water,l}$ is the water temperature leaving the pump, °C; $P_{pump,design}$ is the rated pump power for the design of the cooling water system, W; P_{pump} is the actual pump power for the operating condition, W; E_{pump} is the energy consumed by the pump, W; $T_{water,e}$ is the condenser water temperature entering the pump, °C; m_{water} is the cooling water mass flow rate, kg/s; $m_{v,design}$ is the rated volume flow rate of the pump, m³/s; $C_{p,water}$ is the specific heat capacity of water at the inlet temperature, kJ/(kg·°C); eff_{motor} is the efficiency of the motor in the pump; and eff_{pump} is the efficiency of the pump.

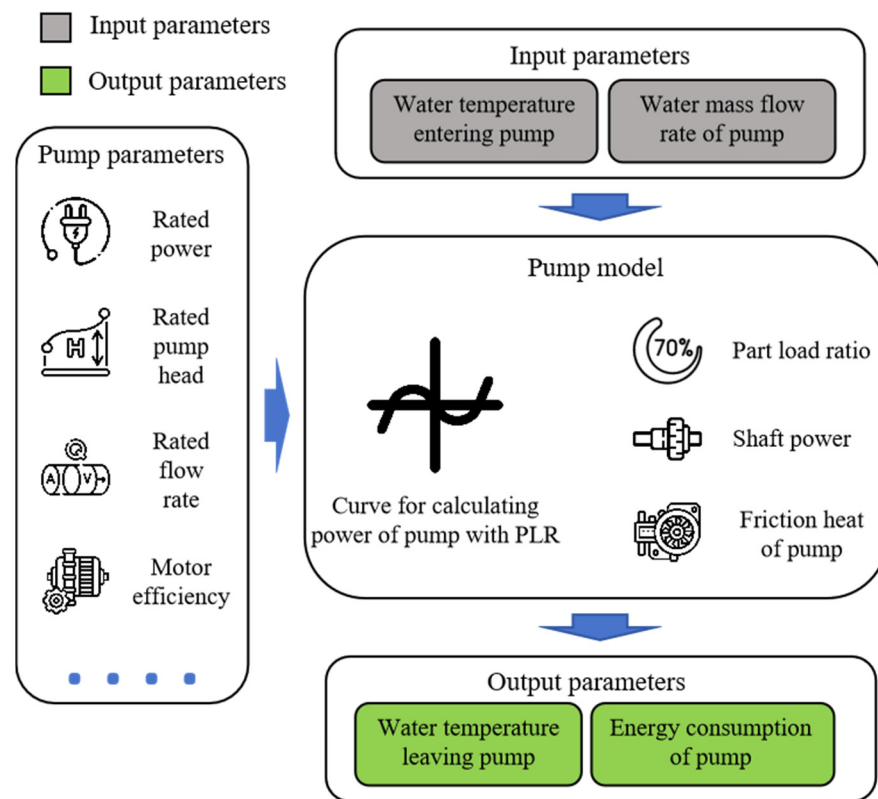


Figure 3. Correlation of input and output parameters in the pump model.

2.2.3. Cooling Tower Model

DOE provided three types of cooling tower models: single-speed, two-speed, and variable-speed cooling towers. This paper constructed the cooling tower model based on the empirical models of variable-speed cooling towers, as demonstrated in Figure 4. The core of this model is an empirical equation that calculates the approach temperature between the outlet water temperature and the inlet air wet-bulb temperature; it is a function of the inlet air wet-bulb temperature, range temperature, and the flow rate ratio of liquid and air. The initial range temperature, defined as the difference between the inlet and outlet water temperature, was specified by users. However, whether the actual range temperature can be achieved must be confirmed by the model iteratively. The iterative calculators contained two toolkits; one was used to obtain the actual range temperature, and the other was used to calculate the cooling tower fan's corresponding air flow rate

ratio. Once the range temperature is fixed, the cooling tower model calculates the actual water temperature leaving the cooling tower. It can be described as four steps to simulate the outlet water temperature:

First, the cooling tower model calculated the outlet water temperature at the maximum fan power. If the outlet water temperature exceeded the setpoint, the fan ran continuously at the maximum speed in the simulation time step.

If the outlet water temperature at the maximum fan power was lower than the setpoint, the fan was turned off, and the free convection condition was considered. When free convection could handle the cooling load transferred from the chiller, the fan remained off during the simulation step.

If the outlet water temperature under free convection was above the setpoint, the tower fan was turned on at minimum fan speed. If the outlet temperature was still below the setpoint, the fan was cycled on and off to maintain the outlet water temperature.

Finally, if the outlet water temperature at minimum fan speed remained above the setpoint temperature, the fan was turned on, and the model determined the required airflow rate and corresponding fan speed.

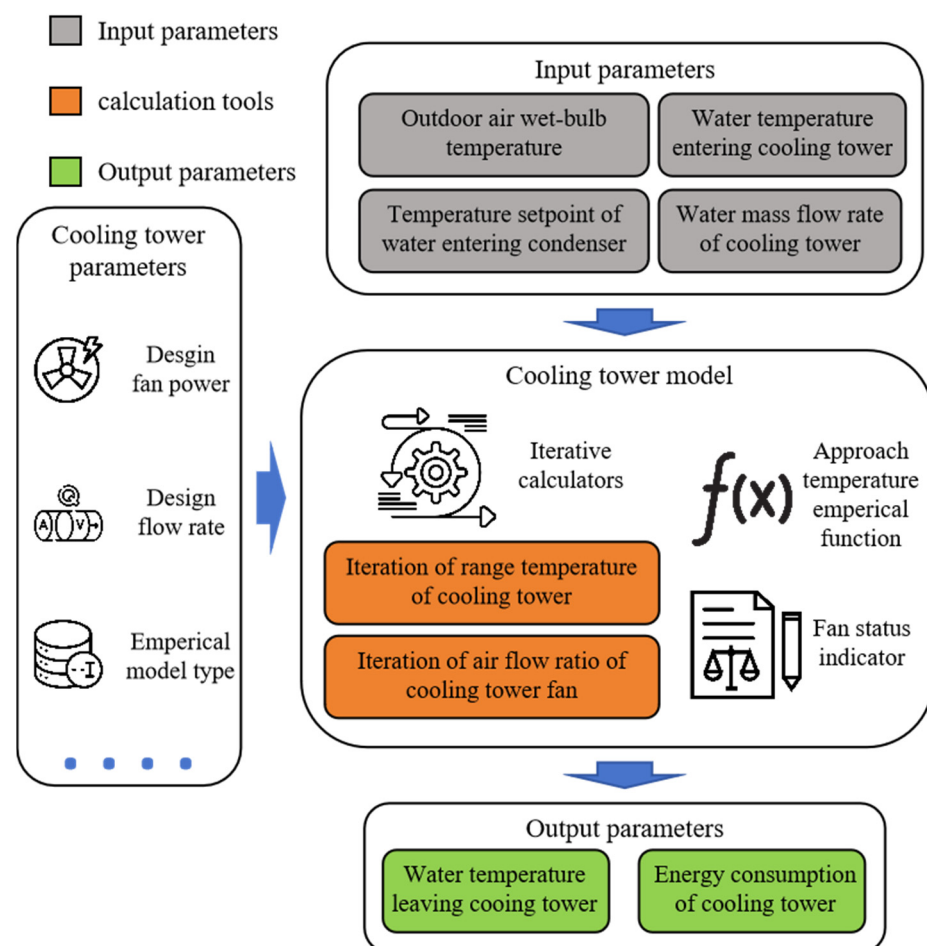


Figure 4. Correlation of input and output parameters in the cooling tower model.

The cooling tower model was described as a function of four input variables to calculate the output parameters, such as the water temperature leaving the cooling tower (i.e., the condenser water temperature entering the condenser) and the energy consumed by the cooling tower fan.

$$T_{cond,e}, E_{coolingtower} = f(T_{env,wb}, T_{pump,l}, T_{cond,es}, m_{cond}) \quad (5)$$

$$\left. \begin{aligned} T_{approach} &= f(FR_{air}, FR_{water}, T_{range}, T_{env,wb}) \\ \left\{ \begin{aligned} T_{cond,e} &= T_{env,wb} + T_{approach} \\ T_{cond,e} &= T_{pump,l} - T_{range} \end{aligned} \right\} \\ \rightarrow T_{pump,l} - T_{range} - T_{env,wb} &= f(FR_{air}, FR_{water}, T_{range}, T_{env,wb}) \\ P_{fan} &= P_{fan,design} \cdot PLR_{fan} \cdot (FR_{air})^3 \\ E_{coolingtower} &= P_{fan} \cdot TimeStep \cdot 3600 \end{aligned} \right\} \quad (6)$$

where: $T_{cond,e}$ is the condenser water temperature entering the condenser, °C; $E_{coolingtower}$ is the energy consumed by the cooling tower fan, W; $T_{approach}$ is the approach temperature of the cooling tower (between 1.1 to 11.1) °C; T_{range} is the temperature difference between the inlet and outlet of the cooling tower (between 1.1 to 11.1 °C); FR_{air} is the flow rate ratio of air through the cooling tower (between 0.15 to 1.0); FR_{water} is the flow rate ratio of water through the cooling tower, (between 0.3 to 1.0); $T_{env,wb}$ is the ambient air wet-bulb temperature (i.e., inlet air wet-bulb temperature) (between −1 to 26.67 °C); $P_{fan,design}$ is the rated fan power of the cooling tower, W; P_{fan} is the actual fan power of the cooling tower, W; $T_{pump,l}$ is the water temperature leaving the pump, °C; $T_{cond,es}$ is the condenser water setpoint temperature entering the condenser, °C; and m_{cond} is the condenser mass flow rate, kg/s.

2.2.4. Model Coupling for Cooling Water System

After the three models were developed, the facilities were coupled to each other based on the actual connections. When coupling the models, the system obeys the energy conservation law. The pipes among the facilities were considered ideal, and there was no energy loss. There are several constraints in each node of the models:

- For the chiller, the cooling load supplied by the chiller's condenser equals the demand cooling load from the evaporator plus the load caused by the chiller's compressor.
- The inlet water temperature of the pump is equal to the outlet temperature of the chiller condenser.
- For the pump, the fluid enthalpy at the outlet equals the fluid enthalpy at the inlet, plus the electrical energy delivered through the shaft, minus the energy loss due to friction.
- The inlet water temperature of the cooling tower is equal to the outlet temperature of the pump.
- For the cooling tower, the heat carried away by the air equals the enthalpy difference between the water entering and leaving the cooling tower, plus the electrical energy input to the fan.

The coupling model workflow can be seen in Figure 5. Since this paper used the chilled-side parameters output from EnergyPlus as input for the cooling water system, the coupling model used the chiller model as a starting point. The chiller model calculated the evaporator outlet temperature, condenser outlet temperature, and energy consumption based on the chilled water inlet temperature, chilled water flow rate, and the initial condenser water inlet temperature read from the Comma Separated Values (CSV) input file; the condenser outlet temperature was used as the water pump inlet temperature, and the pump outlet temperature and pump energy consumption was calculated using the condenser water pump model; the pump outlet temperature was passed to the cooling tower as inlet temperature and input into the cooling tower model. The actual outlet temperature and energy consumption were calculated based on the temperature setpoint of water leaving the cooling tower. The cooling tower outlet temperature output from the cooling tower model would be used as the condenser inlet temperature in the next time step, which would be input into the chiller model for cycle calculation.

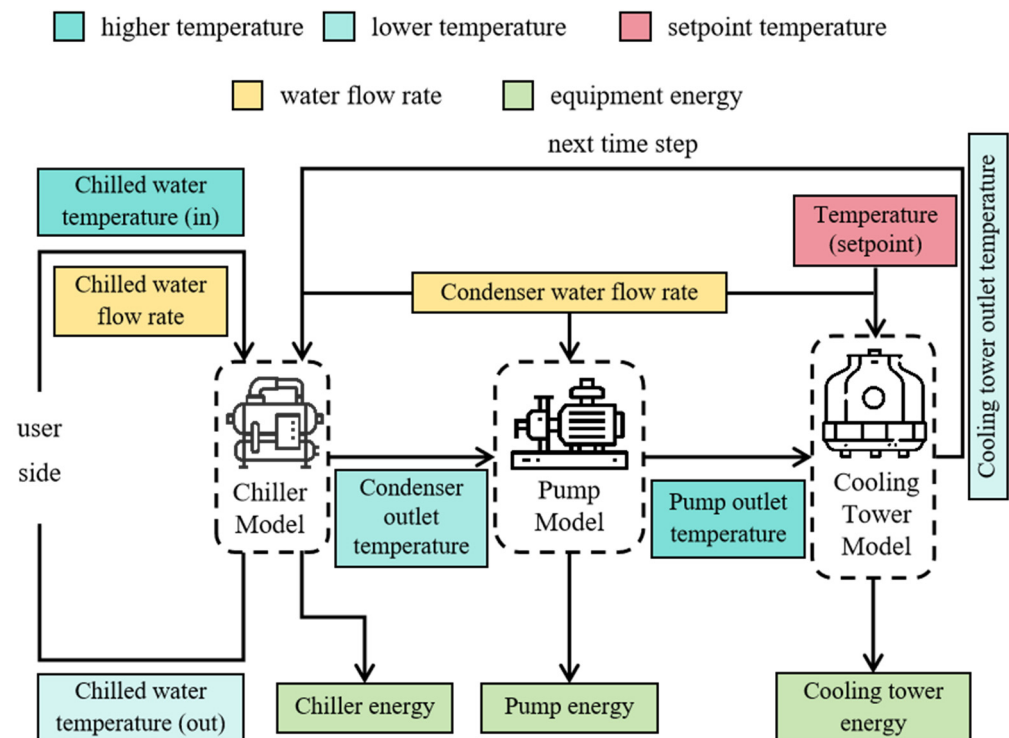


Figure 5. Models coupling workflow.

2.3. Reinforcement Learning

RL learns through interaction with the environment via a trial-and-error process. This interaction involves observing the environment's state and applying control actions to influence it. Each trial yields a reward, which provides feedback on whether the chosen action suits the given environment. The goal of an RL agent is to maximize its expected cumulative reward over time [30]. It is crucial to define key components of the algorithm, such as the state, action, reward function, and hyperparameters. This section introduces these fundamental elements in detail.

2.3.1. State Space

The state space is used to store the variables that represent the state of the environment. Agents observe states of the environment according to the state space. It contains the parameters from the external environment and parameters calculated by the OpenAI Gym environment.

The choice of state determines the reliability of the final strategy obtained by the algorithm. If all the parameters affecting energy consumption are included as state parameters, the RL algorithm can find the optimal strategy through sufficient trial and error. However, the state cannot be described completely, and typically, a few parameters that have the greatest influence on energy consumption are selected as states to simplify the study. Therefore, this paper selects seven parameters closely related to energy consumption as state parameters: water temperature entering and leaving the evaporator, evaporator water mass flow rate, outdoor air wet-bulb temperature, water temperature entering and leaving the condenser, and condenser water mass flow rate. The range of these parameters is presented in Table 3.

Table 3. State space.

State	Value Range	Unit
Outdoor air wet-bulb temperature	(15~30)	°C
Water temperature entering the evaporator	(10~20)	°C
Water temperature leaving the evaporator	(6~8)	°C
Water temperature entering the condenser	(15~35)	°C
Water temperature leaving the condenser	(15~40)	°C
Condenser water mass flow rate	(0~1300)	kg/s
Chilled water mass flow rate	(0~1300)	kg/s

2.3.2. Action Space

The actions are controllable variables that influence the performance of the controlled system. The following actions for the cooling water system were defined: the approach temperature setpoint and cooling water mass flow rate. The approach temperature setpoint is the difference between the condenser water inlet temperature setpoint and the outdoor air wet-bulb temperature. It controls the outlet temperature of the cooling tower, which can be adjusted by indirectly controlling the frequency of the cooling tower fan. The RL algorithms search for an optimal setpoint temperature between 1.5 and 5.0 °C at each time step, with a modification step of 0.5 °C. The controlled cooling water mass flow rate is the total mass flow rate of the cooling water system, which affects the energy consumption of the pump and the operation of the chiller and the cooling tower. The minimum mass flow rate is limited to 30% of the rated mass flow rate of the cooling water system, and the agent makes decisions with a 10% modification step. The range of these actions is presented in Table 4.

Table 4. Action space.

Action	Value	Step	Unit
Approach temperature setpoint	(1.5, 5.0)	0.5	°C
Cooling water mass flow rate ratio	(30%, 100%)	10%	-

2.3.3. Reward Function

The reward is the criterion for evaluating a control strategy in RL. The learning process is to get the maximum reward. Therefore, how to define a reward function is crucial. In this paper, the purpose was to reduce the energy consumption of the controlled cooling water system. As a result, the reward function contained the energy consumption of cooling water facilities. However, using only energy consumption as a reward function can cause the RL agent to over-pursue low energy consumption and ignore the demand of the user side, leading to an irrational control strategy. The difference between the cooling load on the supply and the demand side was introduced as a penalty to weigh the effect of the control action on energy reduction against the effect of not meeting the demand.

$$\left. \begin{aligned} \text{reward} &= -(E_{\text{chiller}} + E_{\text{pump}} + E_{\text{coolingtower}}) - \text{punish} \\ \text{punish} &= Q_{\text{target}} - Q_{\text{met}} \\ Q_{\text{target}} &= cm_{\text{evap}} \Delta(T_{\text{chw},e} - T_{\text{chw},l,\text{setpoint}}) \\ Q_{\text{met}} &= cm_{\text{evap}} \Delta(T_{\text{chw},e} - T_{\text{chw},l}) \end{aligned} \right\} \quad (7)$$

where Q_{target} is the cooling load that needs to be met in the evaporator according to the user demand side, W; Q_{met} is the cooling load met by the cooling water system after being controlled by the RL algorithm, W; $T_{\text{chw},l,\text{setpoint}}$ is the setpoint temperature of the chilled

water leaving the evaporator, °C; and c is the specific heat capacity of the chilled water inlet temperature, kJ/(kg·°C).

2.3.4. Hyperparameters

Since the control actions in the developed environment were discrete, algorithms designed for continuous action control, such as DDPG, were unsuitable for this study. Unlike DDPG, DQN and Double Deep-Q-Network (DDQN) leverage a value-based approach, which is better suited for discrete control problems. Therefore, DQN and DDQN were employed as agents for optimizing the control of the cooling water system.

Both algorithms were trained under identical hyperparameter settings to compare their feasibility in calling the developed environment, as shown in Table 5. The selected hyperparameters were tuned based on empirical experiments and prior research. The training process spanned 5000 episodes, with an epsilon-greedy exploration strategy that decayed from 1.0 to 0.001 through an exponential decay method to balance exploration and exploitation. A discount factor (Gamma) of 0.99 was used to prioritize long-term rewards. A batch size of 48 was chosen to balance computational efficiency and training stability. At the same time, the maximum step count of 2928 was set based on the total number of training steps from 1 June to 30 September in each hour. Experience replays and a target network enhanced learning stability and prevented Q-value divergence. All episodes in which the cooling water system's energy consumption decreased during training were recorded for further analysis. This facilitated a detailed comparison of the two algorithms regarding optimization speed and overall effectiveness.

Table 5. Default hyperparameters in both algorithms.

Hyper-Parameters	Value	Hyper-Parameters	Value
Gamma	0.99	Learning rate	0.001
Train episodes	5000	Maximum step	2928
Epsilon start	1.0	Epsilon end	0.001
Memory capacity	2928	Batch size	48
Hidden layer	2	Hidden dimension	64

2.3.5. Deep-Q-Network and Double Deep-Q-Network

Among the various algorithms for RL, Q-learning is a simple value iteration algorithm that learns the optimal strategy by estimating the state-action value (Q value). The Q value is calculated through a Q function and stored in the Q-table in Q-learning.

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha(r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)) \quad (8)$$

where s_t , a_t , and r_t represent the state, action, and reward at the t step; γ is the discount factor representing the importance of the reward in the future; and α is the learning rate that decides the weight for updating the network.

The Q-table is limited when applying higher dimensions or continuous state spaces. Deep learning and Q-learning were combined to expand the application scenarios of Q-learning, leading to the development of DQN. The DQN algorithm replaces the Q function with a deep neural network and performs value iteration by updating the parameters of the neural network. These parameters are represented as θ in the following equation:

$$\theta \leftarrow \theta + \alpha(r_t + \gamma \max_{a'} Q(s_{t+1}, a'; \theta^-) - Q(s_t, a_t; \theta)) \quad (9)$$

However, the maximum Q value is estimated by the Q-network in DQN, which may be overly optimistic, leading to an overestimation of the Q value in the learning process. DDQN is proposed to reduce this overestimation. DDQN uses two neural networks (Q-network and Target Q-network) to select the action and estimate the Q value.

$$\theta \leftarrow \theta + \alpha(r_t + \gamma Q(s_{t+1}, \arg\max_{a'} Q(s_{t+1}, a'; \theta); \theta^-) - Q(s_t, a_t; \theta)) \quad (10)$$

where $\arg\max_{a'} Q(s_{t+1}, a'; \theta)$ is the selected action by the Q-network according to the Q-value.

3. Results

This section introduces the research results of this paper. Firstly, the accuracy of the OpenAI Gym cooling water system environment is explained. Then, the performance of the optimized control of the cooling water system environment by the DQN algorithm is demonstrated. In addition, the difference in the control performance of the two algorithms, DQN and DDQN, is compared. Finally, the control performance of the DQN algorithm is compared in different climatic zones.

3.1. Validation of the OpenAI Gym Environment Model Accuracy

According to [31], the model accuracy of the cooling water system environment developed based on the OpenAI Gym can be verified through the “Comparison to other models” method. This paper used the EnergyPlus model generated by AutoBPS to validate the simulation results.

The energy consumed by the cooling water system model was verified when the control variables were the same as the EnergyPlus model. The approach temperature was set to 3 °C, and the condenser water flow rate ratio remained at 100% of the rated water flow rate (1085 kg/s). Figure 6 shows the results for 1 July. The solid line represents the results of the EnergyPlus simulation, while the dotted line represents the results from the environmental model calculation. As seen in the figure, there is almost no difference in the energy consumption results of the chiller and water pump, although there is a slight error in the results of the cooling tower model.

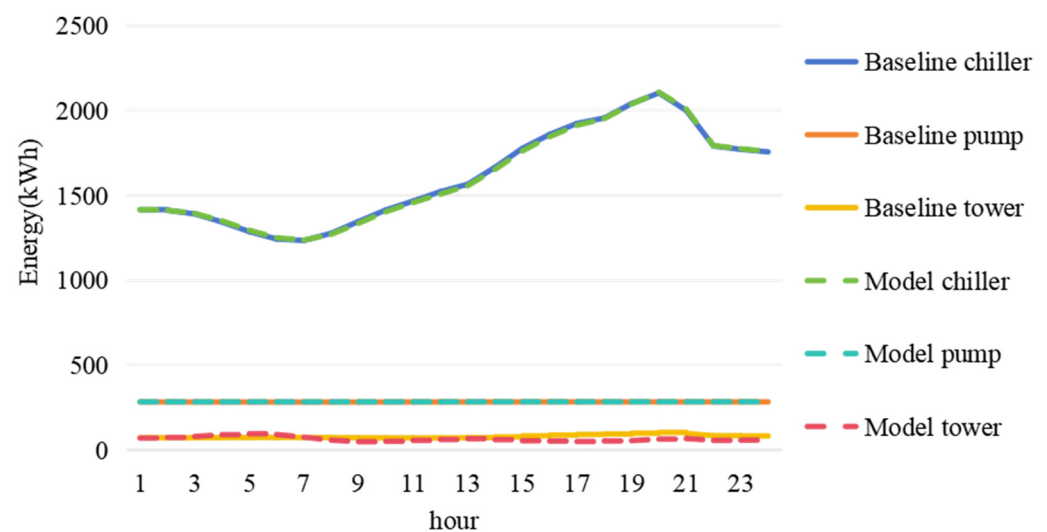


Figure 6. Comparison of hourly energy consumption between EnergyPlus and Gym models on 1 July. The solid line represents the result from the EnergyPlus simulation, and the dotted line represents the result from the environmental model calculation.

To calibrate the model error rate, the hourly energy consumption of each model of the cooling water system was evaluated using two metrics: the normalized mean bias error (NMBE) and the coefficient of variation of the root mean square error (CVRMSE). The equations are shown in Equations [11] and [12], where y is the energy consumption result calculated by the Gym models, $\hat{y}$ is the EnergyPlus simulation results, $\bar{y}$ is the average hourly energy consumption for the cooling season, and n is the number of simulated cooling season hours.

$$NMBE(y, \hat{y}) = \frac{\sum (y_i - \hat{y}_i)}{(n - 1) \times \bar{y}} \times 100 \quad (11)$$

$$CVRMSE(y, \hat{y}) = \frac{\sqrt{\frac{\sum (y_i - \hat{y}_i)^2}{(n - 1)}}}{\bar{y}} \times 100 \quad (12)$$

ASHRAE Guideline 14 provides limit values for these metrics for different scenarios, generally specifying that the NMBE should not be higher than 5% and the CVRMSE should not be higher than 15%, or the perceived accuracy of the model may be unconvincing.

The results of the accuracy validation of the equipment energy consumption data, calculated by the Gym model compared to the energy consumption results from the EnergyPlus simulation, are shown in Table 6.

The NMBE of the three types of cooling water system equipment is less than the limit value of 5%, and the CVRMSE of the chiller and pump is less than the limit value of 15%. However, the CVRMSE value of the cooling tower is higher than 15%, which indicates a certain degree of error in the simulation process between the calculation model of the cooling tower and the calculation model of EnergyPlus. From the overall energy consumption of the cooling water system, the energy consumption of the chiller and the cooling water pump is the main part, accounting for about 95% of the energy consumption of the system. In contrast, the energy consumption of the cooling tower accounts for a smaller part of the system, accounting for only about 5%. The existence of errors in the simulation of the entire cooling water system has a small impact on the simulation error. After calculating the energy consumption of the cooling system, the model shows an NMBE of 0.81% < 5% and a CVRMSE of 1.65% (< 15%), which shows that the accurate performance of the constructed Gym environment model is convincing.

Table 6. Cooling water system model accuracy validation results.

Equipment Model	NMBE	CVRMSE
Chiller	0.01%	0.79%
Pump	−0.43%	0.46%
Cooling tower	4.55%	25.28%
Cooling water system	0.81%	1.65%

The energy consumption during the cooling season, as simulated by the environment model and EnergyPlus, is shown in Figure 7. The energy consumption of the cooling tower in the cooling season simulated by EnergyPlus was 0.22 GWh, while the total energy consumption calculated by the model was 0.21 GWh, and the error of the chiller energy consumption was 4.76%. The energy consumed by the chiller and pump was kept at the same level between the EnergyPlus and the environment model, at 3.60 GWh and 0.83 GWh, respectively. Considering the total energy consumption of the cooling water system in the cooling season, the EnergyPlus simulation result was 4.65 GWh, while the model's calculation was 4.64 GWh, with an error of 0.15%.

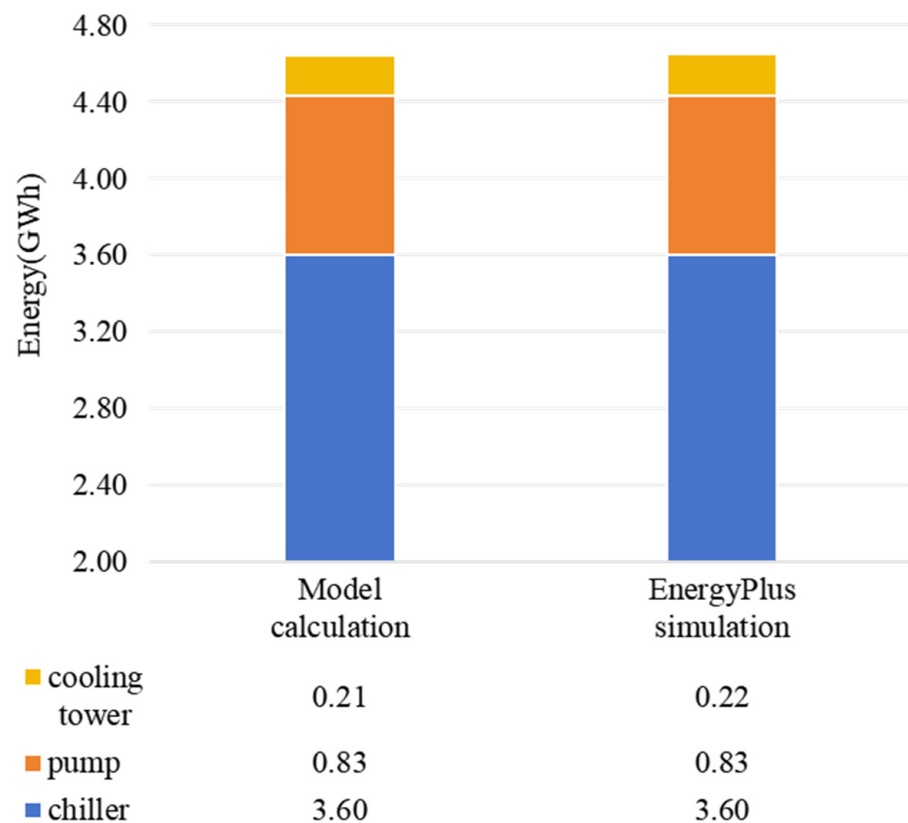


Figure 7. Comparison of energy consumption between EnergyPlus and Gym models during the cooling season.

With the verification completed, the error generated by the energy consumption calculation of the developed cooling water system environment model was within an acceptable range. Therefore, the results of the environmental calculation were used as the benchmark to compare and analyze the optimization effects of the two RL algorithms.

3.2. Optimization Results Using the DQN Algorithm

DQN was first used as an RL agent to interact with the developed environment model. By analyzing the energy consumption and distribution of control actions between different training episodes, the optimization performance of DQN was demonstrated in this section.

Figure 8 illustrates the optimization process of DQN during the training episodes. The greedy factor was set as a decay function of the exponential function, starting at 1.0 and decreasing gradually to 0.001 with optimization. At the beginning of optimization, the energy reward concentrated near the maximum value (4.26 GWh). With the high epsilon value at the beginning of training, the agent explored the control action space more to learn about the energy-saving effect of different actions. As epsilon gradually decreased, the energy reward exhibited a clear downward trend, though the rate of decrease slowed over time. The agent gradually completed exploring and began optimizing the control actions using the learned experience in each step. The minimum energy consumption was reached when the epsilon was close to 0.001. After 4000 episodes, the energy reward had reduced from 4.26 GWh to 3.99 GWh. However, in the final 1000 episodes, the agent could not discover a more effective control strategy, suggesting that the optimization process had reached a plateau.

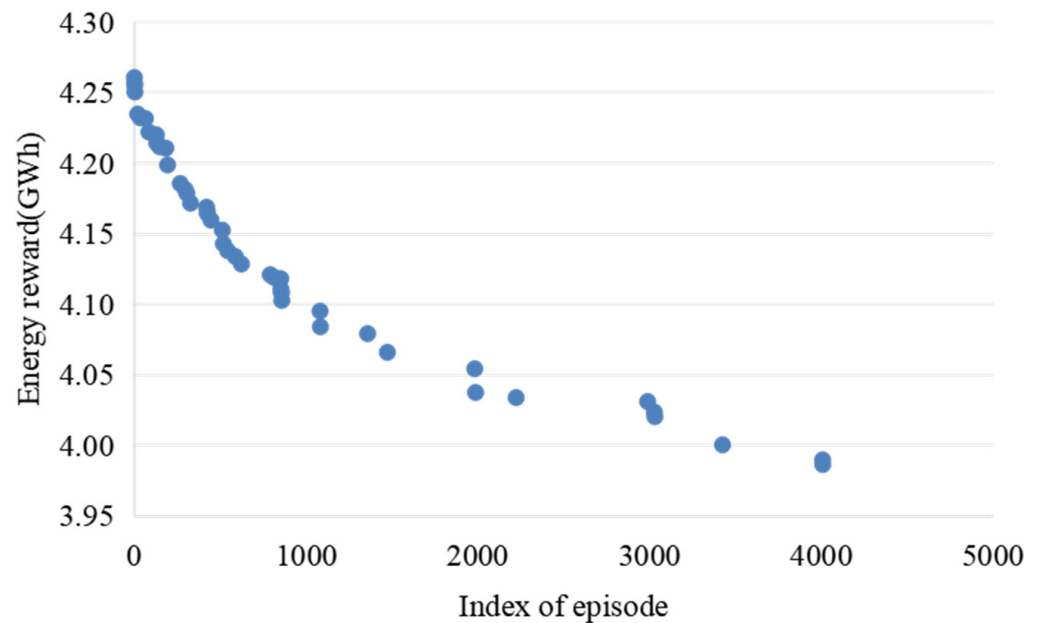


Figure 8. Convergence of energy reward during training for DQN.

To investigate the energy-saving potential of the DQN algorithm during the cooling season, the energy consumption of the cooling water system was analyzed during the optimization process. The cooling season spanned from 1 June to 30 September. A total of 47 episodes with reduced energy consumption were recorded in the training process. For a more intuitive presentation, 10 episodes were selected, among which the first 5 episodes were in the first 2000 episodes, representing the initial stage of the training, while the last five represented relatively convergent episodes. The selected indexes of episodes were 0, 150, 423, 858, 1982, 2223, 2989, 3031, 3425, and 4008. The energy consumption of the chiller, cooling tower, and pump in these episodes was counted and shown in Figure 9.

At the start of optimization, the agent fully explored the control actions in the action space. The energy consumption of the cooling water system was 4.26 GWh, which is 0.38 GWh lower than the baseline calculation. As the optimization process continues, the energy consumption of the cooling water system progressively decreases. After 4008 episodes, the energy consumption has been reduced to 3.99 GWh. Compared with the baseline model, the energy-saving rate increased from 8.19% to 14.16%. Further analysis of the energy consumption of each facility in the cooling water system revealed the following: The energy consumption of the cooling tower showed a continuous decreasing trend, from 0.2 GWh to 0.12 GWh, which means that in each selected episode, the control strategy had a positive effect on the operation of the cooling tower. The energy consumption of the condenser pump decreased sharply from 0.35 GWh to 0.16 GWh in the first 2000 episodes, while the energy consumption of the chiller increased from 3.71 GWh to 3.76 GWh. The decrease in energy consumption of the water pump and cooling tower came at the expense of an increase in the cooling machine's energy consumption. After 2000 episodes, the agent used its experience to gradually balance the energy consumption between the chiller and the pump, eventually returning the chiller's energy consumption to the initial level of 3.71 GWh. Although the energy consumption increased compared with the baseline model of 3.60 GWh, the energy consumption of the pumps and cooling towers had improved significantly.

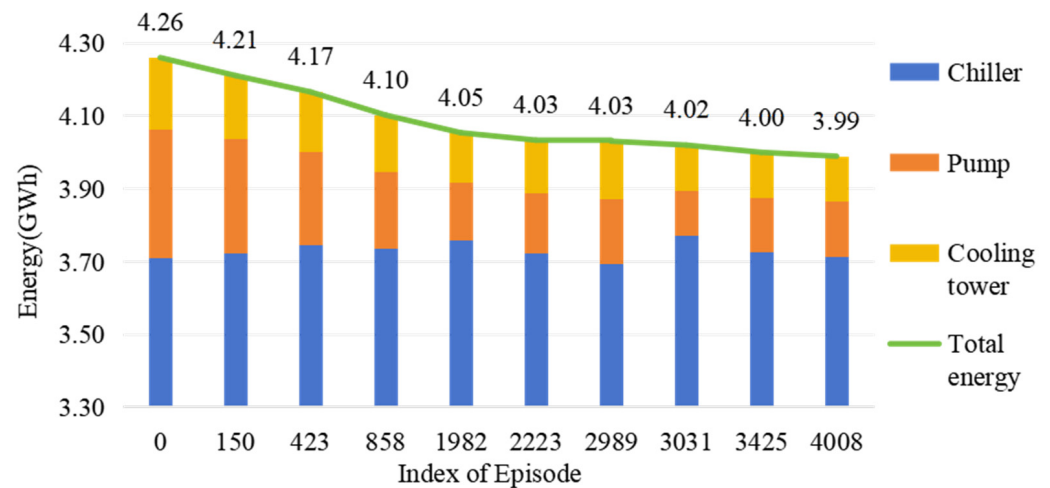


Figure 9. Energy consumption of facilities between different episodes.

To demonstrate the tendency of the DQN algorithm to select actions during the training process, the distribution of control actions in each episode is analyzed. The percentage distribution of the control action situation is shown in Figure 10. The color depth in the table represents the selection frequency of different control actions in different training episodes, and the deeper the color is, the more times the action is selected in the corresponding episode. Based on the previous analysis of energy consumption change, the reasons for the change in control action and its influence on energy consumption optimization were discussed. In the early stage of training (episodes 0–858), the approach temperature and water flow rate ratio selection frequency were scattered uniformly in the action space. DQN was in the exploration stage, and the influence of different control strategies on energy consumption had not been effectively judged. In the middle of the training period (episodes 1982–2989), the selection frequency for approach temperatures of 2 °C and 3 °C increased significantly (up to 26–29%). The selection frequency for low flow ratios (0.3 to 0.5) increased significantly, especially in episodes 1982 and 2223. This indicates that DQN begins to learn that a lower approach temperature helps to reduce the condensing pressure and improve the operating efficiency of the chiller. In contrast, a lower cooling water flow can reduce the pump’s energy consumption without significantly affecting the cooling effect, thus reducing the system’s total energy consumption. At the later stage of training (episode 3031–4008), the approach temperature setting was stabilized at about 2.5 °C, while the flow ratio of 0.3 to 0.5 was still the main choice. It is worth noting that the selection of some high approach temperatures and high flow ratio control actions still exist, which may be due to the following reasons: (1) Under some low-load conditions, appropriately increasing the approach temperature can reduce the energy consumption of the cooling tower, thereby improving overall energy efficiency. (2) During some periods of high load, DQN still selected a flow ratio of 0.6–0.7 to optimize the overall system’s energy efficiency. (3) DQN tried other options at some exploratory steps to prevent falling into local optimality.

The details of control actions selection and energy consumption during the 24 h on July 1 were analyzed to study the impact of the approach temperature and condenser water flow rate. The control actions of four episodes and the baseline model were demonstrated. Figure 11 shows the control actions of each episode. The approach temperature ranges from 1.5 to 5 °C, and the condenser water flow ratio ranges from 0.3 to 1.0. The approach temperature in the baseline model was set to 3 °C, and the condenser water flow was set to 100% of the rated flow. At the beginning of the training, the agent randomly selected the control actions in the action space with a high probability. Therefore, the control actions

in episode 0 and episode 423 were random, and the agent chose the control actions with significant differences per hour. With the increased training episodes, the agent gradually reduced the probability of randomly selecting control actions and used the optimal actions in each simulation step as the control strategy. After 2223 episodes, the greedy factor is less than 0.05, resulting in the agent's tendency to choose the best control action in each simulation step. The approach temperature mainly varies between 2 and 3 °C, and the flow rate ratio ranges between 0.4 and 0.6.

Low frequency High frequency

Index Action	0	150	423	858	1982	2223	2989	3031	3425	4008
1.5°C	12%	10%	14%	15%	5%	13%	20%	9%	5%	6%
2.0°C	13%	11%	10%	12%	25%	17%	16%	14%	2%	2%
2.5°C	14%	12%	12%	17%	26%	11%	10%	21%	64%	33%
3.0°C	12%	16%	9%	7%	11%	29%	22%	34%	15%	28%
3.5°C	12%	12%	12%	12%	7%	7%	14%	5%	4%	20%
4.0°C	12%	14%	17%	14%	9%	8%	8%	8%	4%	8%
4.5°C	13%	13%	13%	11%	13%	9%	4%	5%	4%	2%
5.0°C	13%	12%	12%	11%	4%	6%	6%	4%	3%	1%

(a) Approach temperature distribution

Index Action	0	150	423	858	1982	2223	2989	3031	3425	4008
0.3	13%	13%	25%	20%	45%	17%	14%	53%	16%	24%
0.4	12%	16%	15%	28%	10%	40%	23%	23%	47%	21%
0.5	11%	15%	17%	13%	24%	21%	30%	7%	17%	31%
0.6	12%	13%	10%	17%	8%	9%	22%	8%	11%	16%
0.7	12%	12%	9%	8%	3%	7%	3%	3%	2%	7%
0.8	14%	11%	10%	6%	6%	2%	5%	5%	5%	0%
0.9	12%	10%	8%	5%	3%	1%	3%	1%	1%	2%
1.0	13%	10%	7%	3%	2%	3%	0%	1%	2%	1%

(b) Water flow rate ratio distribution

Figure 10. Action distribution of DQN in different episodes.

The agents selected the optimal control actions under the corresponding observation states through the neural network at every simulation step. Therefore, applying the control strategy optimized by RL to actual engineering leads to frequent adjustments of equipment control parameters. To avoid hourly control of the equipment, the control strategies generated by the algorithm need to be simplified. The average value of the control parameters was used as the actual control action to explore the energy savings of the cooling water system under this control action.

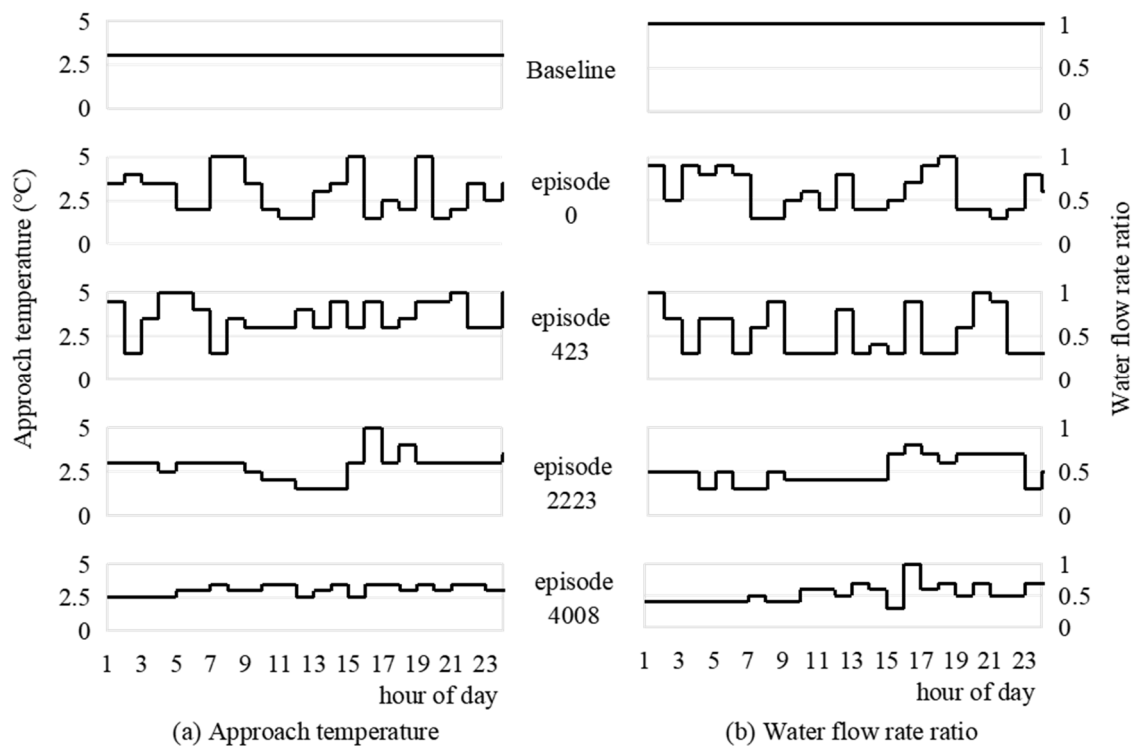


Figure 11. Control actions in different episodes on 1 July.

First, the energy management system (EMS) programs were added to the baseline input data file (IDF) of EnergyPlus to control the flow of the cooling water system. Then, the approach temperature was set to the average value by modifying the SetpointManager:FollowOutdoorAirTemperature field in the IDF. In this way, the baseline model was modified, and the energy consumption of the cooling water system under this control strategy was obtained through simulation.

The statistical analysis of control strategies generated by DQN yielded the average values of the control actions. The approach temperature and cooling water flow rate ratio were calculated as 2.96 °C and 47.12%. The energy consumption comparison is shown in Figure 12.

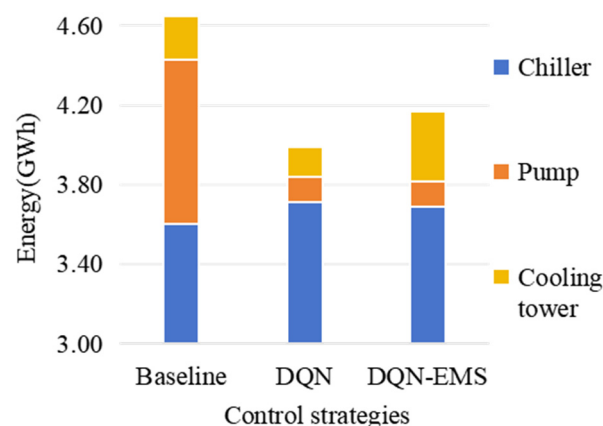


Figure 12. Energy consumption for different facilities by different control strategies.

By setting a rule-based control strategy (DQN-EMS) for the cooling water system through the EMS, the energy-saving performance of the feasible control scheme generated by DQN was demonstrated. There was a 10.29% reduction in energy consumption compared to the baseline model, with a significant reduction in the pump's energy con-

sumption and an increase in the energy consumption of both the chiller and the cooling tower. The reduction of condenser water flow is directly related to the reduction of cooling water pump energy consumption. However, the smaller water flow increased the cost of heat release by the chiller, so the energy consumption of the chiller compressor increased slightly. The reduction of the temperature setpoint of the approach temperature caused the cooling tower fan to expend more energy to release heat into the ambient air. The overall effect was a reduction in the energy consumption of the cooling water system. The energy consumption of the cooling tower mainly caused the difference between DQN and DQN-EMS. Since the control strategy of DQN can adjust the action according to the state of the cooling water system, and the simple EMS control strategy currently implemented sets the control action to a fixed value, the adaptive control effect of the dynamic change of the air conditioning system is not as good as that of DQN.

3.3. Comparison of Optimization Performance Across Different Algorithms

This paper aimed to develop an AutoBPS-Gym toolkit to realize the rapid generation of an interactive environment for RL. However, there are many RL algorithms, and whether the built environment can adapt to different RL algorithms remains to be studied. Based on the previous study of the DQN algorithm, this section introduces the DDQN algorithm to verify the applicability of the generated interactive environment to different RL algorithms. In the same interactive environment generated, the DDQN algorithm was also trained for 5000 episodes to compare with the DQN algorithm. The convergence of the two algorithms is shown in Figure 13.

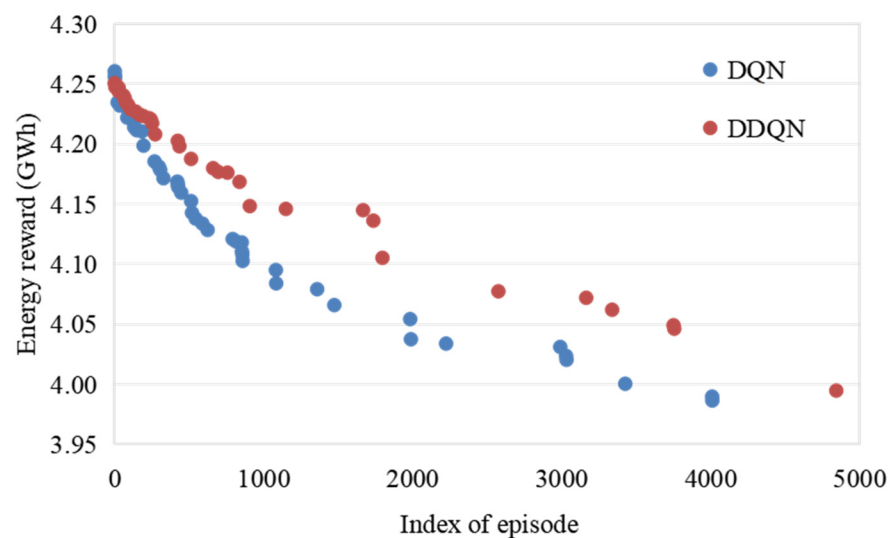


Figure 13. Convergence of energy reward for the two algorithms.

From the figure, it can be seen that the convergence process of DDQN is slower than that of DQN, and the final state is difficult to evaluate whether the results have converged. This may be because DDQN is more conservative in evaluating actions and takes longer to explore high-quality actions, which can save more energy in the cooling water system. In addition, DDQN is more sensitive to parameters such as learning rate and target network update frequency. Using the same hyperparameters of DQN directly may cause unstable updating of the Q value. The exact cause was not confirmed in this study, and further exploration needs to be carried out in future work. In this paper, the algorithm gradually converges, indicating that the generated RL environment can effectively interact with DDQN and achieve certain energy-saving effects.

The final energy consumption of the cooling water system obtained by DDQN is close to the optimization result of the DQN algorithm. The statistical results are shown in Figure 14. The energy consumption of the chiller is 3.75 GWh, which is 0.4 GWh higher than DQN, while the energy consumption of the cooling tower is 0.3 GWh lower than DQN. The energy-saving rate of the DDQN algorithm reaches 14.01%.

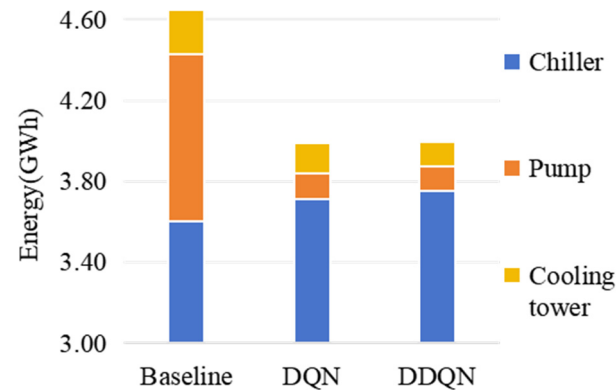


Figure 14. Energy consumption between different RL algorithms.

The comparison of DDQN and DQN control strategies is shown in Figure 15. The approach temperature selected by DDQN is slightly higher than that of DQN, while the cooling water flow is lower than that of DQN. The higher approach temperature reduces the energy consumption of the cooling tower, and the lower cooling water flow increases the energy consumption of the chiller. The overall comparison shows that the energy consumption of the cooling water system is close.

Low frequency High frequency

Approach temperature (°C)			Water flow rate ratio		
Action	Distribution		Action	Distribution	
	DQN	DDQN		DQN	DDQN
1.5	6%	8%	0.3	24%	37%
2	2%	22%	0.4	21%	8%
2.5	33%	18%	0.5	31%	47%
3	28%	43%	0.6	16%	4%
3.5	20%	3%	0.7	7%	2%
4	8%	2%	0.8	0%	1%
4.5	2%	1%	0.9	2%	0%
5	1%	4%	1.0	1%	0%

Figure 15. Control strategy comparison between the two algorithms.

3.4. Optimization Performance Across Different Climate Zones

The developed AutoBPS-Gym can generate environments in different climate zones and building types to interact with RL algorithms for optimal control of HVAC systems. This section takes multiple climate zone shopping mall buildings generated by AutoBPS-Gym as a case to test the optimization of the DQN algorithm in different climate zones. The cooling water system is usually used in the climate zone that needs cooling, according to

the Standard of Climatic Regionalization for Architecture (GB50178-93). It mainly includes three major climate zones: Hot Summer and Cold Winter Zone (HSCWZ), Hot Summer and Warm Winter Zone (HSWWZ), and Temperate Zone (TZ). A total of 20 sub-climate zones are divided into AutoBPS, seven of which can be used to analyze the optimal control of cooling water systems. More detailed information for climate zones and representative cities can be found in [29].

Three climate zones were selected in this study to test the performance of DQN, HSCWZ-3A (Shanghai, China), HSCWZ-3B (Changsha, China), and HSWWZ-4A (Shenzhen, China). Shanghai and Changsha are located in the same climate zone, but one is a coastal city, and the other is an inland city. Shenzhen and Changsha are cities with similar longitudes but in different climatic zones. At the same time, to test the environment with different numbers of cooling water equipment, we modified the generated models so that the number of chillers and cooling towers in the cooling water system was increased to two. The condenser water pump was modified to a variable-speed pump.

Figure 16 illustrates the comparative analysis of cooling water system energy consumption between baseline models generated by AutoBPS-Gym and DQN-optimized control strategies across three climatic zones. The baseline energy consumption in Shenzhen exhibits the highest values among the three cities, with chiller, pump, and cooling tower consumptions reaching 3.15 GWh, 0.14 GWh, and 0.41 GWh, respectively. These values surpass those of Shanghai (2.09, 0.10, 0.31 GWh) and Changsha (2.18, 0.10, 0.31 GWh), a phenomenon attributed to Shenzhen's elevated cooling demand, driven by its year-round high-temperature and high-humidity climate conditions.

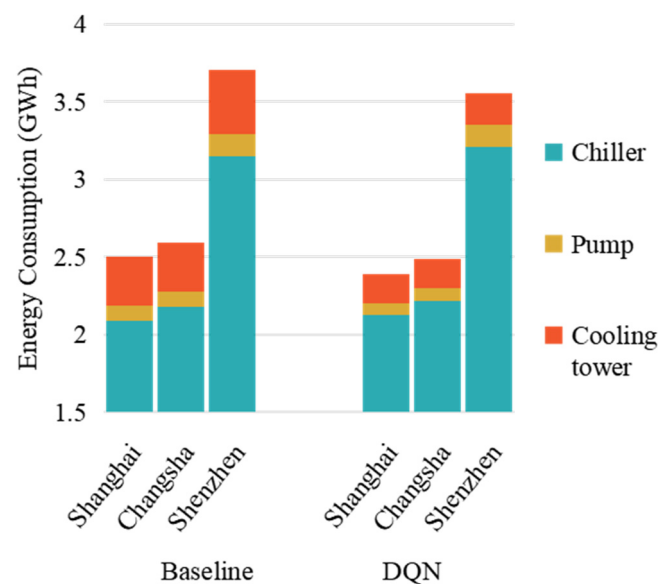


Figure 16. Energy consumption in different climate zones.

DQN-based control optimization reduced the total energy consumption of the cooling water system in all regions. Specifically, Shenzhen achieved a 4.05% energy saving (from 3.70 GWh to 3.55 GWh), while Shanghai and Changsha exhibited reductions of 0.11 GWh (4.40%) and 0.10 GWh (3.86%), respectively. At the component level, the pump energy consumption shows limited optimization potential (e.g., Shenzhen: 0.14 GWh to 0.14 GWh), primarily due to the inherent efficiency improvement from replacing fixed-speed pumps with variable-speed pumps in the baseline model. Furthermore, the DQN algorithm balances the trade-off between chiller and cooling tower energy consumption through dynamic optimization of the approach temperature. This strategy results in a notable reduction in cooling tower energy consumption (e.g., Shenzhen: 0.41 GWh to 0.20 GWh) at

the expense of a marginal increase in chiller energy consumption (e.g., Shenzhen: 3.15 GWh to 3.21 GWh), highlighting the algorithm's capability to prioritize system-level efficiency over individual component performance.

3.5. Time Consumption Between Different Ways of Utilizing the Deep-Q-Network

Before the research in this paper was carried out, an attempt at co-simulation between Python 3.7.16 and EnergyPlus V9-3-0 was carried out. The EnergyPlus model was packaged through the Functional Mock-Up Unit (FMU) and interacted with the DQN agent built in Python. All simulations were performed on a computer with an 11th Gen Intel(R) Core (TM) i7-1165G7 @2.80GH CPU and an NVIDIA GeForce MX450 GPU.

During the co-simulation process, the training time for each round of building energy information simulation using the FMU module is about 44.3 s on average, and the total time required is about 61 h if 5000 training rounds are to be performed. In the research process of this paper, the training time for conducting 5000 rounds was about 11 h, with the average training time per round being about 7.92 s. The time comparison graph of the interaction of RL algorithms through different methods is shown in Figure 17, and the results show that the interaction environment for generating locally optimized objects through AutoBPS-Gym can greatly reduce the time requirement for RL training.

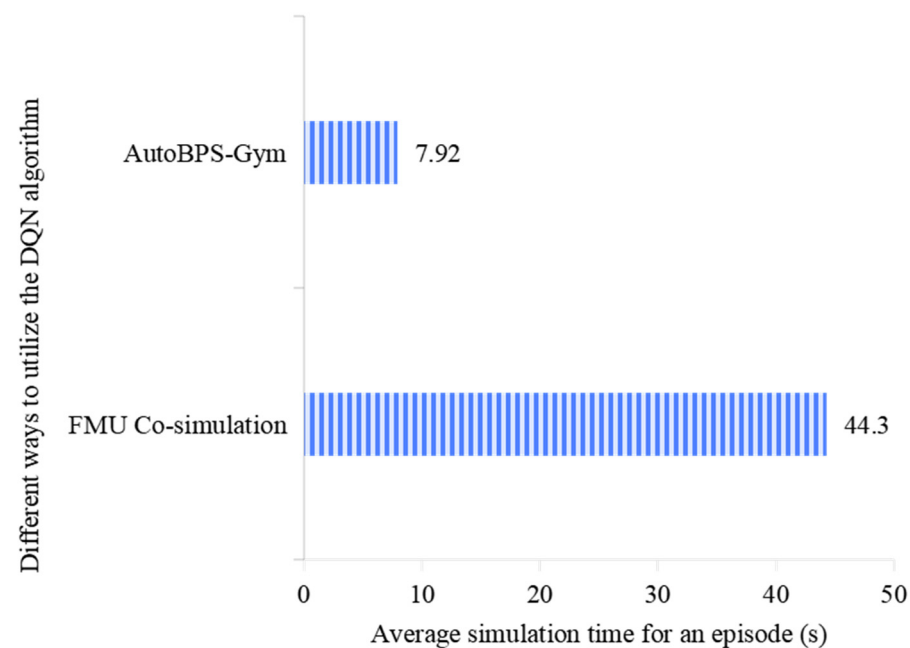


Figure 17. Average simulation time for an episode by different ways to utilize the DQN algorithm.

4. Discussion

The current research still has shortcomings that need to be improved and further explored in the following aspects of the follow-up research:

(a) The parameters contained in the state space may have an impact on the results of the research. Currently, the parameters contained in the state space mainly include the flow rate, the temperature in the cooling water system, and the outdoor wet-bulb temperature, among others. The timestamp and electricity price of the corresponding state can be added to the state space in the future.

(b) The current control action space contains only two control parameters, and the segmentation of control actions is rough, so there are certain differences in the control strategies that optimize the control distance. In addition, the current algorithm can only solve the discrete control action space. If the control action is segmented more finely or

increased, the action space will be too large, and the time for optimization will increase sharply. Algorithms such as DDPG can optimize the continuous action space, and multi-agent RL algorithms can simplify the problem of controlling multiple actions. More RL algorithms can be implemented in future work.

(c) Currently, the feasibility study of the control strategy optimized by the RL algorithm is tested only using the average value of control actions. A finer control strategy can lead to higher energy savings in the actual control process. In the future, we will further optimize our feasibility strategy study and propose a more operational control strategy for the operation control of the cooling water system by using the AutoBPS-Gym tool.

(d) In addition, when testing the building models of different climate zones generated by AutoBPS-Gym, due to the various modeling methods of the devices in EnergyPlus, the device models in the current environment cannot cover all the device models. As a result, there are still some errors in the results. In the future, we will add more device models to the environment to ensure that errors are minimized when generating the environment.

(e) The current studies focus on optimization and testing in the virtual environment. Although the cooling water system can produce an energy-saving effect of 14.01% in theory, whether the control strategy can produce an effect if applied to a practical cooling water system still needs further evaluation. In the future, we will build a scaled experiment platform to verify the control strategy of RL and explore the possibility of applying RL to practice.

(f) During the development of AutoBPS-Gym, the coefficients of the empirical equations of the plant models were not validated by regression, and the coefficients in EnergyPlus were taken directly, which can lead to problems in the practical application of the model. In the future, we will optimize AutoBPS-Gym's model to achieve more accurate cooling water system optimization.

5. Conclusions

The main research content of this paper was the development of a toolkit for AutoBPS, which can easily build RL interactive environments for cooling water systems to interact with open-source RL algorithms—even those not developed for heating, ventilation, and air-conditioning (HVAC) systems. In this way, the energy-saving potential of cooling water systems in different scenarios can be explored by RL, while reducing computational power demands compared to co-simulation with other software.

The conclusions of this paper are as follows:

- The constructed cooling water system model was improved to have the same simulation capability as the baseline model simulated by EnergyPlus, with energy consumption gaps of 0.01%, 0.43%, and 4.54% for the chiller, pump, and cooling tower, respectively. When considering the energy consumption of the cooling water system during the cooling season (June to September), the error decreased to 0.15%.
- The Deep-Q-Network and Double Deep-Q-Network algorithms demonstrated effective optimization results, with the energy-saving rates of the two algorithms reaching 14.16% and 14.01% in a Changsha shopping mall, where the pump is a constant speed pump.
- The AutoBPS-Gym toolkit was employed to generate three building models in distinct climate zones, aiming to evaluate the energy-saving potential of a cooling water system comprising multi-chiller systems, a variable-speed pump, and multiple cooling towers. Applying a Deep-Q-Network algorithm achieved an average energy saving rate of approximately 4% across all three climatic regions, demonstrating the algorithm's adaptability to heterogeneous thermal load conditions.

- In the co-simulation process between EnergyPlus and Python through the Functional Mock-Up Unit, a single episode took about 44.3 s to invoke the EnergyPlus simulation. In comparison, the cooling water system environment built in this paper only took about 7.92 s to simulate in the optimization process, greatly speeding up the optimization process.

Author Contributions: Methodology, X.W., Q.Z., Z.C. and J.Y.; Supervision, Y.C.; Writing—original draft, X.W.; Writing—review and editing, Q.Z., Z.C., J.Y. and Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This paper was supported by the National Natural Science Foundation of China (NSFC) through Grant No. 52478088. This work was also funded by “A Project Supported by the Scientific Research Fund of Hunan Provincial Education Department, China (No. 2023JGZD027)”.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: During the preparation of this work, the authors used Grammarly (v1.2.153.1650) and ChatGPT (GPT-4-turbo) to improve readability and detect spelling/grammar mistakes. After using these tools, the authors reviewed and edited the content as needed and took full responsibility for the publication’s content.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. IEA. *World Energy Outlook 2024*; IEA: Paris, France, 2024. Available online: <https://www.iea.org/reports/world-energy-outlook-2024> (accessed on 19 March 2025).
2. United Nations Environment Programme; Global Alliance for Buildings and Construction. Not Just Another Brick in the Wall: The Solutions Exist-Scaling Them Will Build on Progress and Cut Emissions Fast. Global Status Report for Buildings and Construction 2024/2025. 2025. Available online: <https://wedocs.unep.org/20.500.11822/47214> (accessed on 19 March 2025).
3. Chen, Z.; Deng, Z.; Chong, A.; Chen, Y. AutoBPS-BIM: A Toolkit to Transfer BIM to BEM for Load Calculation and Chiller Design Optimization. *Build. Simul.* **2023**, *16*, 1287–1298. [CrossRef]
4. Yang, S.; Yu, J.; Gao, Z.; Zhao, A. Energy-Saving Optimization of Air-Conditioning Water System Based on Data-Driven and Improved Parallel Artificial Immune System Algorithm. *Energy Convers. Manag.* **2023**, *283*, 116902. [CrossRef]
5. Lu, L.; Cai, W.; Soh, Y.C.; Xie, L.; Li, S. HVAC System Optimization—Condenser Water Loop. *Energy Convers. Manag.* **2004**, *45*, 613–630. [CrossRef]
6. Kim, T.Y.; Lee, J.M.; Yoon, Y.; Lee, K.H. Application of Artificial Neural Network Model for Optimized Control of Condenser Water Temperature Set-Point in a Chilled Water System. *Int. J. Thermophys.* **2021**, *42*, 172. [CrossRef]
7. Huang, S.; Zuo, W.; Sohn, M.D. Improved Cooling Tower Control of Legacy Chiller Plants by Optimizing the Condenser Water Set Point. *Build. Environ.* **2017**, *111*, 33–46. [CrossRef]
8. Wang, X.; Liu, K.; You, W.; Zhang, X.; Ma, H. Stepwise Optimization Method of Group Control Strategy Applied to Chiller Room in Cooling Season. *Buildings* **2023**, *13*, 487. [CrossRef]
9. Ma, K.; Liu, M.; Zhang, J. Online Optimization Method of Cooling Water System Based on the Heat Transfer Model for Cooling Tower. *Energy* **2021**, *231*, 120896. [CrossRef]
10. Qiu, S.; Li, Z.; Li, Z.; Li, J.; Long, S.; Li, X. Model-Free Control Method Based on Reinforcement Learning for Building Cooling Water Systems: Validation by Measured Data-Based Simulation. *Energy Build.* **2020**, *218*, 110055. [CrossRef]
11. Fu, Q.; Chen, X.; Ma, S.; Fang, N.; Xing, B.; Chen, J. Optimal Control Method of HVAC Based on Multi-Agent Deep Reinforcement Learning. *Energy Build.* **2022**, *270*, 112284. [CrossRef]
12. Chen, Y.; Norford, L.K.; Samuelson, H.W.; Malkawi, A. Optimal Control of HVAC and Window Systems for Natural Ventilation through Reinforcement Learning. *Energy Build.* **2018**, *169*, 195–205. [CrossRef]
13. Ahn, K.U.; Park, C.S. Application of Deep Q-Networks for Model-Free Optimal Control Balancing between Different HVAC Systems. *Sci. Technol. Built Environ.* **2020**, *26*, 61–74. [CrossRef]
14. Du, Y.; Zandi, H.; Kotevska, O.; Kurte, K.; Munk, J.; Amasyali, K.; McKee, E.; Li, F. Intelligent Multi-Zone Residential HVAC Control Strategy Based on Deep Reinforcement Learning. *Appl. Energy* **2021**, *281*, 116117. [CrossRef]
15. Peng, Y.; Shen, H.; Tang, X.; Zhang, S.; Zhao, J.; Liu, Y.; Nie, Y. Energy Consumption Optimization for Heating, Ventilation and Air Conditioning Systems Based on Deep Reinforcement Learning. *IEEE Access* **2023**, *11*, 88265–88277. [CrossRef]

16. Fu, C.; Zhang, Y. Research and Application of Predictive Control Method Based on Deep Reinforcement Learning for HVAC Systems. *IEEE Access* **2021**, *9*, 130845–130852. [\[CrossRef\]](#)
17. Biemann, M.; Scheller, F.; Liu, X.; Huang, L. Experimental Evaluation of Model-Free Reinforcement Learning Algorithms for Continuous HVAC Control. *Appl. Energy* **2021**, *298*, 117164. [\[CrossRef\]](#)
18. Raffin, A.; Hill, A.; Gleave, A.; Kanervisto, A.; Ernestus, M.; Dormann, N. Stable-Baselines3: Reliable Reinforcement Learning Implementations. *J. Mach. Learn. Res.* **2021**, *22*, 1–8.
19. Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; Zaremba, W. OpenAI Gym. *arXiv* **2016**, arXiv:1606.01540.
20. Moriyama, T.; De Magistris, G.; Tsubori, M.; Pham, T.H.; Munawar, A.; Tachibana, R. *Reinforcement Learning Testbed for Power-Consumption Optimization*; Springer: Singapore, 2018; Volume 946, ISBN 9789811328527.
21. Zhang, Z.; Chong, A.; Pan, Y.; Zhang, C.; Lam, K.P. Whole Building Energy Model for HVAC Optimal Control: A Practical Framework Based on Deep Reinforcement Learning. *Energy Build.* **2019**, *199*, 472–490. [\[CrossRef\]](#)
22. Arroyo, J.; Manna, C.; Spiessens, F.; Helsen, L.; Leuven, K.U. *An OpenAI-Gym Environment for the Building Optimization Testing (BOPTTEST) Framework*; Abstract Key Innovations; Flemish Institute for Technological Research (VITO): Mol, Belgium, 2021; pp. 175–182.
23. Wang, Z.; He, Y. AlphaHydrogen: A Virtual Platform for Simulating and Evaluating Station-Based Regional Hydrogen-Electricity Networks with Distributed Renewables, Buildings, and Fuel-Cell Vehicles. *Energy Convers. Manag.* **2023**, *280*, 116802. [\[CrossRef\]](#)
24. Chen, W.; He, Y.; Li, N.; Wang, Z.; Peng, J.; Xiang, X. A Smart Platform (BEVPro) for Modeling, Evaluating, and Optimizing Community Microgrid Integrated with Buildings, Distributed Renewable Energy, Electricity Storage, and Electric Vehicles. *J. Build. Eng.* **2024**, *87*, 109077. [\[CrossRef\]](#)
25. Deng, Z.; Chen, Y.; Yang, J.; Causone, F. AutoBPS: A Tool for Urban Building Energy Modeling to Support Energy Efficiency Improvement at City-Scale. *Energy Build.* **2023**, *282*, 112794. [\[CrossRef\]](#)
26. Chen, Y.; Wei, W.; Song, C.; Ren, Z.; Deng, Z. Rapid Building Energy Modeling Using Prototype Model and Automatic Model Calibration for Retrofit Analysis with Uncertainty. *Buildings* **2023**, *13*, 1427. [\[CrossRef\]](#)
27. Xi, H.; Zhang, Q.; Ren, Z.; Li, G.; Chen, Y. Urban Building Energy Modeling with Parameterized Geometry and Detailed Thermal Zones for Complex Building Types. *Buildings* **2023**, *13*, 2675. [\[CrossRef\]](#)
28. Peng, C.; Chen, Z.; Yang, J.; Liu, Z.; Yan, D.; Chen, Y. Assessment of Electricity Consumption Reduction Potential for City-Scale Buildings under Different Demand Response Strategies. *Energy Build.* **2023**, *297*, 113473. [\[CrossRef\]](#)
29. Yang, J.; Zhang, Q.; Peng, C.; Chen, Y. AutoBPS-Prototype: A Web-Based Toolkit to Automatically Generate Prototype Building Energy Models with Customizable Efficiency Values in China. *Energy Build.* **2024**, *305*, 113880. [\[CrossRef\]](#)
30. Nagy, Z.; Henze, G.; Dey, S.; Arroyo, J.; Helsen, L.; Zhang, X.; Chen, B.; Amasyali, K.; Kurte, K.; Zamzam, A.; et al. Ten Questions Concerning Reinforcement Learning for Building Energy Management. *Build. Environ.* **2023**, *241*, 110435. [\[CrossRef\]](#)
31. Sargent, R.G. Verification and Validation of Simulation Models. In Proceedings of the Winter Simulation Conference, Baltimore, MD, USA, 5–8 December 2010; pp. 166–183.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.