energies

MDPI

Article

Evaluating Reinforcement Learning Algorithms in Residential Energy Saving and Comfort Management

Charalampos Rafail Lazaridis [1,2,†], Iakovos Michailidis [1,*,†], Georgios Karatzinis [1,2,†], Panagiotis Michailidis [1,2,†] and Elias Kosmatopoulos [1,2,†]

[1] Department of Electrical and Computer Engineering, Democritus University of Thrace, 67100 Xanthi, Greece; gkaratzi@iti.gr (G.K.); panosmih@iti.gr (P.M.); kosmatop@iti.gr (E.K.)
[2] Information Technologies Institute (I.T.I.), Centre for Research & Technology—Hellas (CE.R.T.H.), 57001 Thessaloniki, Greece
* Correspondence: michaild@iti.gr; Tel.: +30-2310-464160
† These authors contributed equally to this work.

Abstract: The challenge of maintaining optimal comfort in residents while minimizing energy consumption has long been a focal point for researchers and practitioners. As technology advances, reinforcement learning (RL)—a branch of machine learning where algorithms learn by interacting with the environment—has emerged as a prominent solution to this challenge. However, the modern literature exhibits a plethora of RL methodologies, rendering the selection of the most suitable one a significant challenge. This work focuses on evaluating various RL methodologies for saving energy while maintaining adequate comfort levels in a residential setting. Five prominent RL algorithms—Proximal Policy Optimization (PPO), Deep Deterministic Policy Gradient (DDPG), Deep Q-Network (DQN), Advantage Actor-Critic (A2C), and Soft Actor-Critic (SAC)—are being thoroughly compared towards a baseline conventional control approach, exhibiting their potential to improve energy use while ensuring a comfortable living environment. The integrated comparison between the different RL methodologies emphasizes the subtle strengths and weaknesses of each algorithm, indicating that the best selection relies heavily on particular energy and comfort objectives.

Keywords: reinforcement learning; energy efficiency; thermal comfort; buildings; residents; Energym

Citation: Lazaridis, C.R.; Michailidis, I.; Karatzinis, G.; Michailidis, P.; Kosmatopoulos, E. Evaluating Reinforcement Learning Algorithms in Residential Energy Saving and Comfort Management. *Energies* **2024**, *17*, 581. https://doi.org/10.3390/en17030581

Academic Editors: Ala Hasan and Hassam Ur Rehman

Received: 29 November 2023
Revised: 18 January 2024
Accepted: 21 January 2024
Published: 25 January 2024

1. Introduction

1.1. Motivation

Contrary to the outdated standpoint considering residents as static constructions, the modern perspective identifies them as multifunctional cyber-physical entities able to provide efficient thermal comfort and promote occupants' quality of life [1–5]. However, to provide adequate comfort to residents—using the potential integrated heating, ventilation, and air conditioning (HVAC) equipment—a specific amount of energy needs to be consumed. Such an amount portrays a significant portion of the overall energy consumption, rendering residential buildings as energy-intensive consumers and a significant contributor to the surge in greenhouse gas (GHG) emissions [6–8]. To this end, to maintain residential comfort in a viable way, the need to harmonize energy conservation while maintaining comfort levels has become an essential objective [9–12].

For many years, residents relied on manual approaches to control HVAC by adjusting thermostats, opening windows, or turning on fans based on their immediate comfort needs. While these actions provided quick adjustments to the indoor environment, the lack of real-time adaptability often resulted in energy wastage during unoccupied or temperate periods [12,13]. Recognizing this gap, scheduling devices emerged, facilitating pre-configured temperature preferences and timed functions, granting a level of autonomous management while ensuring a stable comfortable environment. However, their static configurations often

resulted in energy wastage during, e.g., vacant periods or unforeseen climatic changes, highlighting the necessity of a more sophisticated control mechanism [14].

Following the initial steps in automation, Rule-Based Control (RBC) emerged as the dominant strategy to strike a balance between energy saving and comfort. The utilized rules derived from hands-on observations and expert recommendations, setting actions for specific conditions, like reducing temperatures during off-hours or adjusting ventilation according to occupancy [15]. RBC, however, illustrated numerous limitations, struggling to adapt in real-time to varying factors such as changing weather, fluctuating occupancy, or equipment variations [16]. Additionally, RBC was primarily centered on upholding comfort standards, neglecting energy conservation or financial efficiency. The absence of a holistic understanding of the interplay between various HVAC elements also resulted in less ideal outcomes [16–18].

The complexity and unpredictability of managing HVAC led to the rise of algorithm-based control methods, such as RL [19–23]. At its core, RL algorithms learn from interactions with the environment, allowing decisions based on real-time data. Instead of relying on predefined rules such as RBC, RL methodologies continuously refine their strategy, ensuring optimal energy use without compromising comfort. Since residential environments are dynamic—with changing occupancies and external conditions—RL frameworks were adequate to anticipate and respond to different scenarios, whether it is a sudden weather change or varying resident preferences [23,24]. This continuous learning and adaptability mean that RL is sufficient to achieve long-term energy efficiency while always prioritizing the comfort of inhabitants [25–27].

However, choosing the optimal RL approach for enhancing energy efficiency in HVAC systems presents a significant challenge [28,29]. Given the dynamic nature of HVAC environments, influenced by fluctuating weather conditions, varying building occupancy, and equipment performance, no single RL algorithm stands out as a universal solution [29]. The plethora of RL algorithms, each tailored to specific scenarios and action spaces, further complicated the selection. The lack of a one-size-fits-all RL solution for HVAC systems suggests that evaluating numerous RL approaches is pivotal in identifying the best-suited strategy for specific applications, such as HVAC systems. Different RL algorithms operate uniquely when controlling residential HVAC due to their varied underlying principles, learning mechanisms, and optimization strategies [23]. Each algorithm is designed with certain assumptions and priorities, which means they respond differently to the dynamic nature of HVAC environments, influenced by fluctuating weather conditions, varying building occupancy, and equipment performance. Some might excel in rapidly changing conditions, while others might be more stable and consistent over longer periods [29]. By comparing different algorithms under consistent conditions, researchers may gain clarity on their respective performances, ensuring evaluations are both fair and insightful [30]. This process highlights the strengths and weaknesses of each algorithm, aiding in making informed decisions. Furthermore, it is not just about finding the best algorithm but understanding how each can be optimized or tailored for specific scenarios.

Motivated by the plethora of RL approaches, current work integrates a comparison between different RL control algorithms towards a conventional RBC approach for exhibiting their adequacy to support an efficient optimal control scheme for balancing energy saving and comfort. To this end, PPO, DDPG, DQN, A2C, and SAC methodologies are thoroughly assessed in a simulative environment for their ability to control the operation of HVAC and thus reduce energy consumption while ensuring indoor comfort. By testing their energy-saving capabilities in a standard residential apartment setting, the research reveals which algorithm stands out as the most suitable.

These algorithms have demonstrated superior performance in similar tasks in previous research, indicating their potential effectiveness for the specific application of HVAC control. Their ability to handle the complexities of an environment like a residential apartment requires balancing between energy saving and comfort. Defining which RL approach balances exploration and exploitation more efficiently is beneficial in finding the optimal

control strategies for HVAC and portrays a key factor for the current study. Each of these algorithms has unique strengths and characteristics which render them suitable for the problem of HVAC control for energy saving and comfort. For instance, PPO is known for its stability and reliability in different environments, making it a good choice for applications where safety and consistency are important. DDPG and SAC, being off-policy algorithms, are effective in environments with continuous action spaces, like HVAC control. Another key factor for the selection of the specific set of algorithms was influenced by their practicality in terms of implementation and the availability of support. Algorithms like PPO, DDPG, DQN, A2C, and SAC are often well-documented and supported by popular machine learning frameworks. This makes them more accessible for integration into existing systems, particularly in HVAC applications, fostering potential future real-life deployment.

It should be mentioned, that the concerned RL algorithms are meticulously evaluated against established criteria and user preferences, offering valuable insights into their performance trends. This approach adopts a user-centric perspective, aligning algorithmic control with individual comfort needs and energy efficiency goals, thereby providing customized solutions for various thermal comfort categories. Additionally, by emphasizing the distinct features of each algorithm, the study underscores the significance of choosing the appropriate algorithm based on the specific requirements of the application.

1.2. Related Literature Work

The literature exhibits numerous RL applications in residential buildings for the efficient control of different HVAC equipment. RL approaches such as PPO, DDPG, DQN, A2C, and SAC are commonly utilized to balance energy saving and comfort, reducing costs and the environmental impact on residents. More specifically, in [31], a comparison of the DQN methodology towards a thermostat controller was assessed. The findings indicate that the DQN-enabled intelligent controller surpassed the baseline controller. In the simulated setting, this advanced controller enhanced thermal comfort by approximately 15% to 30% while concurrently achieving a reduction in energy expenses ranging from 5% to 12%. The DQN algorithm in residents was also evaluated in [32], where a data-driven approach was aimed at managing split-type inverter HVACs, factoring in uncertainties. Data from similar AC units and homes were merged to balance out data disparities, and Bayesian convolutional neural networks (BCNNs) were employed to estimate both the ACs' performance and the associated uncertainties. Subsequently, the Q-learning RL mechanism was established to perform informed decisions about setpoints, using insights derived from the BCNN models. According to the outcome, the novel approach achieved slightly lower energy consumption (19.89 kWh) and discomfort (1.44 °C/h) compared to the rule-based controller [32].

In [33], a DDPG framework was tailored to regulate HVAC and energy storage systems without relying on a building's thermal dynamics model. This approach took into account a desired temperature bracket and various uncertain parameters. Comprehensive simulations grounded in real-world data attest to the potency and resilience of the suggested algorithm in comparison to the baseline ON/OFF control policy. According to the results, the proposed energy management algorithm was able to reduce the mean value of total energy cost by 15.21% compared to the baseline controller while also achieving a lower mean value of total temperature deviation, indicating improved thermal comfort. Similarly, in [34], researchers employed a dual-focused control strategy for HVAC systems that balanced energy costs with user comfort. By addressing such objectives simultaneously, an optimization model was structured toward energy cost predictions, past usage trends, and external temperatures. Utilizing the DDPG method, an optimal control strategy was achieved that was able to harmonize cost and comfort. According to the results, different weighting factor prices balancing energy cost saving and comfort provided different results. For instance, when the weighting factor reached 0.5, 38.5% energy cost saving was achieved. Increasing the weighting factor to 0.55, energy cost savings reached a 50% improvement in comparison to a predefined temperature schedule control approach. Thermal

Comfort Improvement Factor ranged from 42.75% to −28.7% across different weighting factor values.

A comparative analysis between DQN and DDPG methodologies was also carried out to control multi-zone residential HVAC systems in [35]. The concerned optimization objective was twofold: cut down energy expenses and ensure occupant comfort. By using the DDPG method, they effectively learned from continuous interactions in a simulated building setting, even without prior model insights. Their findings reveal that this DDPG-guided HVAC management surpassed the leading DQN, slashing energy costs by 15% and decreasing comfort breaches by 79%. Impressively, when pitted against a conventional RBC method, the DDPG approach reduced comfort infractions by a staggering 98%.

The PPO approach was evaluated in [36] to fine-tune the operation of a building's HVAC system, aiming to enhance energy efficiency, uphold thermal comfort, and meet set demand response targets. Simulated results showcased that leveraging RL for standard HVAC management was sufficient to achieve energy savings of nearly 22% in weekly energy consumption in comparison to a conventional baseline controller. Furthermore, during periods requiring demand response, employing a controller attuned to demand response with RL can lead to power fluctuations of about 50% over a week, in comparison to a conventional RL controller, all the while ensuring the thermal comfort of the inhabitants.

To assess the effectiveness of deep reinforcement learning (DRL)-based control systems, the researchers conducted evaluations of both PPO and A2C controllers. This evaluation focused on summer cooling performance over one month, followed by a test in the subsequent month using the pre-trained models [37]. Findings indicated that A2C generally outperformed the PPO methodology, particularly with medium-sized network estimation models, except in cool and humid climates, where a PPO control proved more effective. According to the outcome, the A2C control methodology delivered 4% and 22% lower energy consumption concerning the RBC methodology in cooling mode, all while ensuring thermal comfort.

The SAC methodology was evaluated in [38], where the combination of an RL with Long Short-Term Memory (LSTM) neural networks was aimed to steer heat pumps and storage systems across four buildings. Their simulation framework incorporates LSTM models, trained on an artificial dataset from EnergyPlus, to gauge indoor temperature dynamics. The engineered controller effectively sustained comfortable indoor conditions across various buildings, achieving a cost reduction of approximately 3% against the baseline RBC approach. Furthermore, this DRL controller facilitated a peak demand reduction by 23% and decreased the Peak-to-Average Ratio (PAR) by 20%. Additionally, the DRL controller successfully harnessed interactions among diverse sources of flexibility, enhancing the flexibility factor by 4%. Moreover, in [39], the SAC approach was also utilized to efficiently control the thermal storage of a four-building cluster with unique energy profiles. The goal was to optimize individual building energy usage while leveling the overall energy load. When compared to a traditionally set RBC, the novel methodology achieved 4% cost savings, reduced peak demand by up to 12%, and led to a 10% drop in daily average peak, showcasing the benefits of SAC in energy management.

1.3. Novelty

Current work stands out for its comprehensive analysis, exploration of energy and comfort trade-offs, performance evaluation against international standards, user-centric algorithm selection, and detailed characterization of each algorithm. These facets collectively contribute to the novelty and value of our research in advancing the field of residential energy management.

Grounded in a comprehensive analysis of several prominent RL algorithms, namely SAC, PPO, A2C, DDPG, and DQN, within the context of residential energy management, current research evaluates each algorithm's ability to balance energy efficiency and occupant comfort—a critical consideration in modern living spaces. A key aspect of the current effort lies in exploring the intricate trade-offs between energy reduction and thermal comfort.

Such attributes have been achieved by implementing specific weighting parameters—a novel approach that allows us to gauge each algorithm's performance meticulously. Such attributes not only assess the algorithms based on established criteria, but also factor in user preferences in line with international standards. This approach yields valuable insights into the general trends and performance nuances of the five algorithms under scrutiny, especially regarding how alterations in weight factors and user preferences impact their effectiveness. Moreover, additional emphasis has been given to the scalability and adaptability of these algorithms in different residential settings. Contrary to the majority of the literature, current efforts explore how these RL schemes may be effectively scaled up or down, catering to a wide range of residential environments, from small apartments to large houses, thus broadening the applicability and impact of findings.

Another interesting attribute highlighted in this study concerns the fact that some particular algorithmic schemes, like DQN, may not be directly comparable with others due to their intrinsic mechanisms. Conversely, it is observed that the PPO algorithm consistently maintains lower Predicted Percentage of Dissatisfied (PPD) values, illustrating its adaptability in shifting priorities between objectives. Such observations highlight the necessity of understanding the unique characteristics of each RL algorithm, underscoring the importance of selecting the most appropriate one for specific applications. To this end, contrary to the majority of existing literature approaches, current research does not merely provide a comparative analysis of these algorithms but also delves deep into their traits and suitability for varied applications in residential energy management.

Moreover, current effort enables a user-centric selection of algorithms to empower users to customize algorithmic control and align with their specific comfort needs and energy efficiency objectives. Such attributes pave the way for choosing the most suitable algorithm for each category of thermal comfort, thereby offering personalized solutions. The integration of dynamic user feedback into algorithm performance evaluation is a significant advancement. Unlike traditional static assessments, this approach allows for a more realistic and adaptable evaluation, considering how user preferences can evolve. To this end, current work reflects real-world scenarios accurately, where occupant preferences and comfort levels are not constant but change in response to various factors.

The environmental impact of the current research also concerns a critical aspect. By focusing on energy efficiency and sustainable living, the current study directly contributes to the broader goals of reducing carbon footprints and promoting eco-friendly practices in residential settings. This aligns with global environmental objectives and demonstrates the societal relevance of your work. Last but not least, the potential for real-world implementation and commercialization of these algorithms portrays a fruitful prospect. Current research efforts pave the way for developing new products or services that integrate these advanced algorithms into smart home systems, offering tangible benefits to consumers and industry stakeholders.

1.4. Paper Structure

The paper is structured as follows. In Section 1: Introduction, the motivation of the current work is assessed and the related previous work and the novelty of the current work are elaborated. Section 2: Joint Materials and Methods delivers the general mathematical overview of the RL methodology while providing the conceptual background of the algorithms concerned—PPO, DDPG, DQN, A2C, and SAC—regarding the implementation in HVAC. Section 3: Testbed Description elaborates on the aspects of the concerned simulative testbed description, while Section 4: Results and Discussion illustrates a thorough comparative analysis of RL algorithms' performance in energy saving and comfort measures. Last but not least, Section 5: Conclusions and Future Work concludes the outcomes of the current study and describes the future work generated by the current research effort.

2. Joint Materials and Methods

This section provides the mathematical description as well as the particularities of RL applications in HVAC control applications. Providing the generalized concept of the RL approach as well as the concept of the individual RL algorithms, the current work aims to establish the landscape of the concerned RL concepts, familiarizing the reader with the current state of the art and highlighting the relevance of the study at hand.

The General Reinforcement Learning Conceptual Background

In RL, an agent interacts with an environment over discrete time steps to achieve a certain objective. At each time step, the agent observes the current state of the environment, which provides a snapshot of all pertinent information about the environment at that moment. Based on this state, the agent selects an action according to its policy. The policy is essentially the strategy or behavior that the agent follows, and it can be deterministic (always giving the same action for a given state) or stochastic (providing a distribution over possible actions) [23,40].

Once the action is taken, the environment transitions to a new state and provides feedback to the agent as a reward (Figure 1). This reward offers a numerical value indicating how good or bad the action was in achieving the agent's objective. Over time, the agent aims to learn a policy that maximizes its expected cumulative reward, often referred to as the return. To aid in this learning, the agent often estimates a value function, which predicts the expected return from a given state when following a particular policy. This value function helps the agent to judge the long-term consequences of its actions, enabling it to favor actions that lead to higher cumulative rewards in the future [23,40].

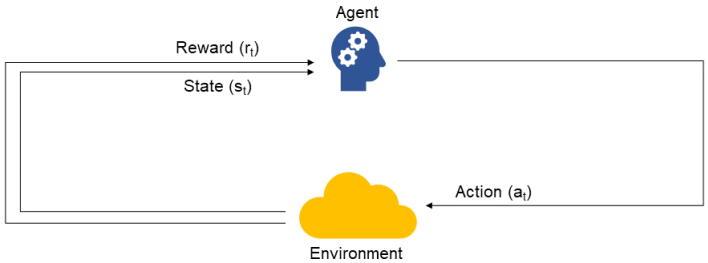


Figure 1. The general reinforcement learning framework, with an autonomous agent acting in an environment.

More specifically:

- **State:** The state, denoted as s, encapsulates the current environmental and system-specific conditions that are pertinent to the decision-making process of HVAC control. In the context of HVAC control, the state can be formally represented as [23]:

$$s = [T_{\text{room}}, T_{\text{setpoint}}, T_{\text{external}}, n_{\text{occupants}}, t, \ldots] \quad (1)$$

  where T_{room} is the current room temperature; T_{setpoint} is the desired temperature; T_{external} denotes the external temperature; $n_{\text{occupants}}$ is the number of occupants; t represents the current time or time of day.
- **Action:** The action space encompasses the set of all feasible actions that the HVAC system can take at any given state. Let a denote an action taken by the RL agent. In the HVAC context, this is represented as:

$$a \in \{\text{ON}, \text{OFF}, \Delta T_{\text{setpoint}}, \Delta\text{airflow}\} \quad (2)$$

  where ON and OFF denote the operational status of the HVAC; $\Delta T_{\text{setpoint}}$ represents the adjustment to the temperature setpoint; Δairflow signifies changes in the airflow rate.
- **Reward:** The reward function provides a quantitative measure of the quality of an action taken by the agent in a particular state. For HVAC systems, the reward function

aims to strike a balance between energy efficiency and occupant comfort. Formally, the reward $r(s, a)$ can be defined as:

$$r(s, a) = -\alpha E_{\text{consumed}} + \beta C_{\text{comfort}} - \delta |T_{\text{room}} - T_{\text{setpoint}}| \quad (3)$$

where α, β, and δ are weighting parameters; E_{consumed} is the energy consumed by the HVAC system; C_{comfort} quantifies the comfort level of occupants.

- **Policy:** The policy, denoted as π, provides a mapping from states to actions. It represents the strategy that the RL agent employs to act in the environment. Depending on the RL approach, this policy can be deterministic or stochastic. In deep RL contexts, this policy is often parameterized by a neural network, leading to a functional representation:

$$a = \pi_\theta(s) \quad (4)$$

where θ represents the parameters of the neural network.

- **Value Function:** The value function offers a prediction of the expected cumulative reward from a given state when following a particular policy. For a policy π, the value function $V^\pi(s)$ is defined as:

$$V^\pi(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) | s_0 = s \right] \quad (5)$$

Here, γ is a discount factor that ensures future rewards are discounted to reflect their temporal delay.

In the context of HVAC control, the goal is twofold: achieve energy saving while maintaining occupant comfort. The state typically captures variables that influence HVAC decisions, such as current room temperature, desired temperature setpoint, external temperature, and the number of occupants. This comprehensive state representation ensures that the RL agent has enough information to make informed decisions. The reward function is designed to balance energy efficiency and comfort. For instance, the agent could receive a positive reward for keeping room temperatures within a comfortable range and a negative reward proportional to the energy consumed. This way, the agent is encouraged to maintain comfort while using as little energy as possible [23,41].

The policy in this scenario maps from the rich state information to HVAC control actions, such as turning the system on/off or adjusting temperature setpoints. As the agent interacts with the environment (the residence and its HVAC system), it refines this policy to better achieve the dual objectives. The value function in the HVAC context provides insights into the long-term benefits of current actions. For instance, turning off the HVAC system might save energy now but could lead to discomfort later, resulting in a lower value. By considering such long-term consequences, the agent is adequate to provide decisions that balance immediate energy savings against future comfort levels [41].

3. Testbed Description

In this study, the simulative testbed concerned a four-floor residential building in Tarragona, Spain, with diverse equipment like thermostats and a central geothermal heat pump, which was integrated into the Energym open-source building simulation library. The setup employed the *Stable Baselines3* Python library, creating an environment where the RL agent interacts with the building model, and adjusts thermostat setpoints, while other settings are fixed. The agent's actions influence the building's thermal zones, with the environment providing feedback in terms of temperature, humidity, and energy consumption data. A multi-objective reward function guides the agent, balancing energy efficiency and thermal comfort, modifiable by adjusting weight parameters. The setup was benchmarked against a classic temperature control system, fostering a sufficient evaluation of the RL algorithms' effectiveness in optimizing residential energy management while maintaining occupant comfort.

3.1. Energym Framework

Energym [42] is a Python-based open-source library that is based on both EnergyPlus and Modelica, providing different benchmark building models that are interfaced using the Functional Mockup Interface (FMI) standard. This building framework consists of 11 simulation models providing diverse equipment installments (thermostat, heat pump, battery, air handling unit, electric vehicle, photovoltaic), distinct building usages (apartments, houses, offices, seminar center, and mixed-use), and different methods in the control settings (controlling thermostat setpoints and controlling the equipment directly). In this work, the *ApartmentsGrid-v0* case is adopted. This is a residential building located in Tarragona, Spain, consisting of four floors, each of them being an apartment, and there are eight thermal zones (two per floor). The thermal system of the building has a central geothermal heat pump (HP) directly connected to hot water tanks (one per apartment) used only for domestic hot water (DHW) consumption, and to a heating loop providing heat to the entire building. Therefore, regarding the equipment that is present in the building, there are four controllable thermostats (one per floor), a non-controllable heat pump, one battery, and one electric vehicle (EV). The simulation inputs (11 in total) involve thermostat setpoints for the four floors ($P1\_T\_Thermostat\_sp, \ldots, P4\_T\_Thermostat\_sp$), heat pump temperature setpoint ($Bd\_T\_HP\_sp$), temperature setpoints for each tank ($P1\_T\_Tank\_sp, \ldots, P4\_T\_Tank\_sp$), battery charging/discharging setpoint rate ($Bd\_Pw\_Bat\_sp$), and EV battery charging setpoint rate ($Bd\_Ch\_EVBat\_sp$). The output part consists of an extensive set (69 in total) of measurements with respect to the behavior of the building for a given input vector. These simulation outputs provide temperature ($Z01\_T \ldots Z08\_T$), humidity, and appliance energy measurements in different zones of the building, supply and return temperature for the heat pump, total energy consumption and HVAC energy consumption (Fa_E_HVAC), outdoor temperature (Ext_T), and other outputs related to the batteries. For more information about the building *ApartmentsGrid-v0*, including its thermal zones, components, inputs, and outputs, please refer to the *Energym* documentation https://bsl546.github.io/energym-pages/sources/apg.html (accessed on 28 November 2023).

3.2. Building Simulative Testbed

The overall workflow contains two gym-based environments which work in conjunction with the well-known Python library, *Stable Baselines3* [43]. The chosen model from the Energym framework, i.e., the *ApartmentsGrid-v0*, serves as a building model that responds with 69 output measurements for a given set of 11 input signals each time, whereas the second gym-based environment, named *IntermediateEnv*, establishes the interaction between the *ApartmentsGrid-v0* and the RL agent implementations of *Stable Baselines3*. Sections 3.3 and 3.4 describe the encapsulated operation within *IntermediateEnv*. Thus, the RL agent is encountered with *IntermediateEnv* with a Markov property interacting constantly. The overall workflow is depicted in Figure 2. More specifically, the *Energym* simulation model operates at a fine granularity, running for 480 time steps per day, with each step representing a 3 min interval. This detailed time scale allows for a nuanced simulation of the building environment's dynamic responses. In contrast, the *IntermediateEnv*, which facilitates the interaction between the RL agent and the *Energym* simulation, operates on a coarser time scale. It runs for 48 time steps per day, with each step corresponding to a 30 min interval. This difference in time step granularity is critical for the application of RL actions. Actions determined by the RL agent in the *IntermediateEnv* are applied to the *Energym* simulation model and held constant (clamped) for a duration of 10 *Energym* time steps, cumulatively amounting to 30 min. This approach ensures that each action has a sustained impact on the building environment, allowing the system enough time to reach a more stable state in response to the action. Also, it ensures that the system's response is not merely a transient reaction to the changes but rather a reflection of a more settled state. Such a setup is important in evaluating the effectiveness of the RL algorithms over a

realistically significant duration, accurately capturing the implications of each action on energy consumption and thermal comfort in the simulated building.

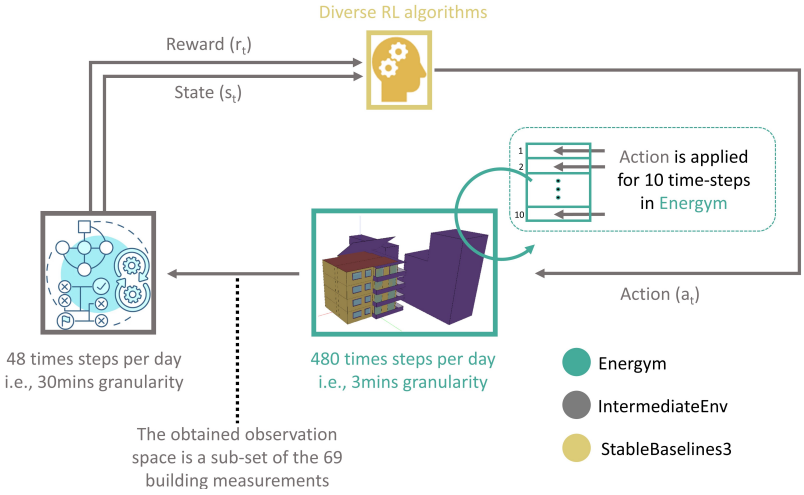


Figure 2. Operational scheme for the residential energy saving and comfort management application, coupling *Energym*, *IntermediateEnv*, and *Stable Baselines3* library.

As mentioned, *Energym* includes a wide set of input and output measurements on this specific building. The utilized input and output signals are depicted in Tables 1 and 2, respectively. Note that we change the notation of these measurements. To enhance the comprehensibility of the actions and states within our experimental setup, we have opted to use descriptive names that differ from the original *Energym* nomenclature. This decision was made to ensure clarity and ease of understanding for readers not familiar with the specific terminologies of the *Energym* platform.

Table 1. Action signals of the RL agent.

Parameter Description	Parameter Description	Symbol in Energym	Symbol in This Work
Thermostat setpoints for floors 1 to 4 (°C)	[16, 26]	$P\{\mathbf{X}\}\_T\_Thermostat\_sp$	$Thermostat_{\{\mathbf{X}\}}$
Heat pump temperature setpoint (°C)	[35, 55]	$Bd\_T\_HP\_sp$	$Heatpump$
Temperature setpoints for tanks 1 to 4 (°C)	[30, 70]	$P\{\mathbf{X}\}\_T\_Tank\_sp$,	$Tank_{\{\mathbf{X}\}}$
Battery charging/discharging setpoint rate	$[-1, 1]$	$Bd\_Pw\_Bat\_sp$	$Battery_{rate}$
EV battery charging setpoint rate	[0, 1]	$Bd\_Ch\_EVBat\_sp$	$EVBattery_{rate}$

Table 2. Subset of output measurements (state signals) from the building model that are inserted in the *IntermediateEnv*.

Parameter Description	Parameter Description	Symbol in Energym	Symbol in This Work
Zone temperatures for zones 1 to 8 (°C)	$Z0\{\mathbf{X}\}\_T$	[10, 40]	$Temp_{zone\{\mathbf{X}\}}$
Outdoor temperature	Ext_T	$[-10, 40]$	$Temp_{ext}$
HVAC energy consumption (Wh)	Fa_E_HVAC	[0, 2000]	E^{cons}

3.3. Actions and State

The ongoing interaction between the agent and the environment involves the agent selecting actions, to which the environment responds by providing rewards and introducing new states for the agent to consider. The action is of the following form:

$$action = \{(Thermostat_1 \quad \dots \quad Thermostat_4), \quad Heatpump, \quad \cdots \\ \cdots, \quad (Tank_1 \quad \dots \quad Tank_4), \quad Battery_{rate}, \quad EVBattery_{rate}\} \tag{6}$$

where $(Thermostat_1 \ \dots \ Thermostat_4)$ represent the thermostat setpoints for each floor, $Heatpump$ is the heat pump temperature setpoint that constantly takes a mean value in its operating bounds, $(Tank_1 \ \dots \ Tank_4)$ are the tank temperature setpoints for each floor constantly taking a mean value of their operating bounds, $Battery_{rate}$ stands for the battery charging/discharging setpoint rate that is constantly zero, and $EVBattery_{rate}$ is the EV battery charging setpoint rate that is also zero every time. The action variables that are left free to be adjusted/trained by the agent are the four thermostats, while the remaining setpoints for the tanks and the heat pump constantly take a mean operating value. In this work, we leave aside the electrical parts of the battery and EV battery, so these values will constantly be zero throughout the interaction with the building model. Thus, the action space is completely aligned with the input space of the *ApartmentsGrid-v0* building model. As mentioned, the building model returns 69 measurements for a given action vector. The observation space is a subset of those building responses. The state variables are defined as follows:

$$state = \{(Temp_{zone1} \dots Temp_{zone8}), \ Temp_{ext}, \ E^{cons}\} \tag{7}$$

where $(Temp_{zone1} \dots Temp_{zone8})$ and $Temp_{ext}$ represent the temperature measurements in degrees Celsius for the eight different building zones and outdoor conditions, respectively, and E^{cons} stands for the HVAC energy consumption, which is also measured continuously.

3.4. Reward Function

The objective here is to reduce energy consumption while sustaining thermal comfort for occupants controlling solely the thermostats of the building. Indeed, two contradictory factors together formulate the multi-objective reward function. Thus, the reward function is defined as:

$$reward = \{\alpha[E^{cons}(t)] - \beta[Th^{com}(t)]\} \tag{8}$$

where $E^{cons}(t)$ is the HVAC energy consumption that is straightforwardly measured from the building at each time instance, while $Th^{com}(t)$ is directly connected with the thermal comfort index. The emerging trade-off between HVAC energy consumption and thermal comfort is shifted into the tuning process of parameters α and β in reducing the first factor as much as possible while sustaining acceptable levels of comfort with minimum penalty. Different weights between the two tuning parameters present different results in favoring either the first or the second reward sub-term. In this work, we keep $\beta = (1 - \alpha)$, considering three weight sets $\{0.1, 0.5, 0.9\}$ towards testing three different operational modes. The intermediate scenario (weight = 0.5) induces a balance between the two reward factors, while the other two cases maintain slightly extreme cases that focus on either reducing electricity bills with a large thermal comfort penalty or increasing high levels of thermal comfort regardless of energy consumption.

3.5. Baseline Classic Controller Description

To evaluate the effectiveness of the adopted RL controllers, it is essential to compare their performance against a traditional, established control system. For this purpose, we utilize a classic controller, as implemented in the Energym framework, to serve as our baseline. This controller operates on a simple yet effective principle; it maintains a specified indoor temperature within a defined tolerance range. The operational mechanism of this classic controller is straightforward. It requires a set temperature and a tolerance limit. Whenever the indoor temperature deviates from the set value by more than the tolerance (in absolute terms), the controller activates to restore the temperature to the predetermined level. For our comparative analysis, we have set the average indoor temperature to 20.375 °C. This setpoint is coupled with a tolerance of 0.3 °C to allow for minor fluctuations without triggering the control mechanism unnecessarily.

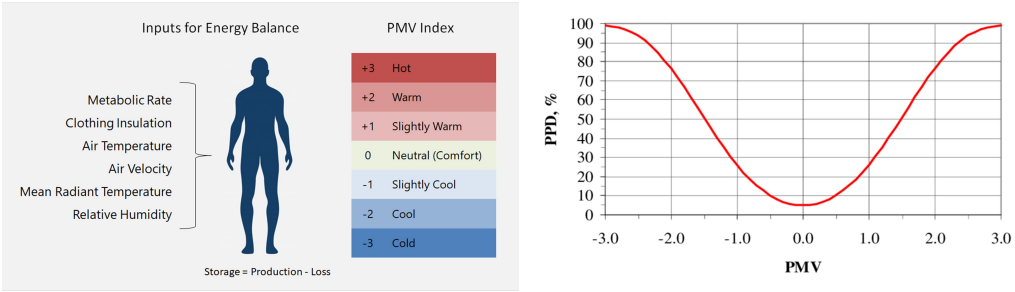
During the operation of the classic controller, we observed an average PPD index of 5.9%. This metric provides insight into the level of thermal comfort experienced by occupants and is crucial for assessing the practicality of the control strategy from a human-centric perspective. In terms of energy efficiency, the classic controller showed an average energy consumption of 430.183 Wh per time step. This consumption rate is a critical benchmark for evaluating the energy performance of our RL controllers under identical conditions. By comparing the performance of the RL controllers with this classic controller, we aim to ascertain not only their relative energy efficiency but also their ability to maintain occupant comfort, thereby determining their viability for practical applications in building energy management.

3.6. Thermal Comfort Metrics

The energy consumption of buildings is significantly influenced by various factors, such as indoor environmental conditions (including temperature, ventilation, and lighting) and the design and operation of the building and its systems. Simultaneously, these indoor conditions have a profound impact on the well-being, performance, and overall satisfaction of occupants within the built environment. It has been established that maintaining high-quality indoor environmental conditions can enhance work and learning performance, reduce absenteeism, and increase overall comfort. In addition, occupants who experience discomfort are more inclined to take actions to enhance their comfort, which might have implications for energy usage. Consequently, there is an increasing demand for well-defined criteria to guide building design and energy assessments [44]. To address these concerns, a series of indices have been developed, rigorously tested, and implemented to evaluate and optimize the indoor thermal environment. The wide set of international standards in this area include (i) ASHRAE 55 [45]—thermal environment conditions for human occupancy; (ii) ISO 7730 [46]—ergonomics of the thermal environment and analytical determination and interpretation of thermal comfort using calculation of the Predicted Mean Vote (PMV) and Predicted Percentage of Dissatisfied (PPD) and local thermal comfort effects; (iii) EN 16798 [47]—specification of criteria for measurements and methods for long-term evaluation of the indoor environment obtained as a result of calculations or measurements.

Thermal comfort assessment is a multifaceted process that takes into account several critical factors and aims to predict how a group of individuals perceives their thermal environment. This involves considering environmental parameters, including relative humidity (RH) and dry-bulb air temperature (tdb), and individual variables such as total clothing insulation (Icl) and metabolic rate (M). The PMV is the established reference for assessing thermal comfort in mechanically conditioned buildings, serving as a tool to anticipate individuals' perceptions of their thermal environment. For naturally conditioned buildings, the adaptive models of EN and ASHRAE are utilized. The PPD index provides insight into the percentage of people likely to feel too warm or too cool. Figure 3 illustrates the thermal sensation scale and the representation of PPD as a function of PMV [48,49]. The PMV values correlate with the PPD index, highlighting the balance between thermal comfort and dissatisfaction.

The application of these comfort models in practical scenarios is detailed in Table 3. The table categorizes levels of thermal comfort expectation, delineating the acceptable PMV ranges and corresponding PPD percentages, which formulate the assessment metrics for building thermal comfort. These categories range from Category I, which signifies a high level of thermal comfort expectation suitable for sensitive groups (expectation with less than 6% predicted dissatisfaction), to Category IV, indicating a lower expectation that is considered acceptable for only part of the year.



(**a**) (**b**)

Figure 3. Thermal sensation scale and representation of PPD as function of PMV. (**a**) Thermal comfort variables and PMV range: https://www.dexma.com/wp-content/uploads/2022/10/Predicted-Mean-Vote-index-Dexma.png (accessed on 28 November 2023). (**b**) Correlation between PMV and PPD indices.

Table 3. Different levels of criteria regarding thermal comfort based on EN 16798-2 standard [47].

Category	PPD %	PMV	Description
I	<6	$-0.2 < PMV < +0.2$	High level of expectation and also recommended for spaces occupied by very sensitive and fragile persons with special requirements like some disabilities, sick, very young children, and elderly persons, to increase accessibility.
II	<10	$-0.5 < PMV < +0.5$	Normal level of expectation.
III	<15	$-0.7 < PMV < +0.7$	An acceptable, moderate level of expectation.
IV	<25	$-1.0 < PMV < +1.0$	Low level of expectation. This category should only be accepted for a limited part of the year.

4. Results and Discussion

This section delves deeper into the comparative performance of various RL algorithms in the context of a multi-objective reward function focusing on energy consumption and thermal comfort. In the context of the classic controller, with an average energy consumption of 430.183 Wh per time step and an average PPD of 5.9%, these RL algorithms demonstrate a range of performances. This comparative analysis highlights the strengths and limitations of each RL algorithm in balancing energy efficiency and occupant comfort. Such insights are vital for selecting the most suitable algorithm for specific building environments and occupant needs, ultimately contributing to more intelligent and sustainable building management systems. The performance of these algorithms, as shown in Table 4, is evaluated under different weight scenarios (w = 0.1, w = 0.5, and w = 0.9) that prioritize either energy reduction or thermal comfort to varying degrees. The average values for each algorithm presented in Table 4 are derived from five distinct evaluation runs for generalization purposes. For metrics such as Predicted Percentage Dissatisfied (PPD), the average is computed across the building's eight zones, providing a comprehensive view of the occupant's comfort throughout the entire building. This methodical approach to averaging ensures that the reported values accurately reflect the overall performance of the algorithms in varying spatial contexts within the simulated environment. Note that the architecture and hyper-parameter configuration of the utilized RL algorithms are presented in Appendix B, i.e., Tables A1–A5.

In Appendix C, we provide a comprehensive collection of supplementary results (see Figures A1–A15) that encapsulate the extensive simulations conducted across various algorithmic cases. For each RL algorithm examined, we present detailed visual data under different weight scenarios, including (a) the Predicted Percentage Dissatisfied (PPD) for each thermal zone within the building, offering insights into the thermal comfort levels achieved; (b) the measured temperature for each building zone, which illustrates the algorithms' performance in maintaining the desired thermal conditions; (c) the HVAC energy consumption throughout the day, providing a quantitative measure of the algorithms'

energy efficiency. This ensemble of 45 images serves to augment the empirical findings discussed in the main text, allowing for a granular assessment of each RL algorithm's ability to navigate the trade-offs between thermal comfort and energy consumption. By presenting these data visually, we aim to facilitate a deeper understanding of the nuanced performance characteristics of each algorithm within a residential building energy management context.

Table 4. Performance comparison after 5 distinct runs for each algorithm.

Weight Case	RL Algorithm	Average HVAC Energy Consumption (Wh)	Average PPD (%)	Average Power Diff (%)
w = 0.1	SAC	119.0829	25.11	72.32
	PPO	307.7933	6.54	28.45
	A2C	82.4841	34.91	80.83
	DDPG	191.3231	24.40	55.53
	DQN	318.9417	7.78	25.86
w = 0.5	SAC	258.8230	9.63	39.83
	PPO	297.7426	6.8	30.79
	A2C	320.5797	7.04	25.48
	DDPG	214.5196	25.15	50.13
	DQN	507.4004	7.27	−17.95
w = 0.9	SAC	258.6270	8.68	39.88
	PPO	314.7583	6.50	26.83
	A2C	301.9630	6.92	29.81
	DDPG	199.7418	25.08	53.57
	DQN	467.2232	6.30	−8.61

4.1. Weight Implications on Performance

The weight factor in the reward function plays a crucial role in balancing between reducing energy consumption and maintaining thermal comfort, specifically:

Weight 0.1: This weight setting places a higher emphasis on energy reduction. Algorithms operating under this weight are expected to minimize energy usage, potentially at the expense of occupant comfort.

Weight 0.5: Represents a balanced approach, giving equal importance to both energy savings and maintaining a satisfactory PPD level.

Weight 0.9: Prioritizes thermal comfort, aiming to achieve a PPD level close to 6%, akin to the performance of the classic controller, and aligning with the standards of Category I, which represents a high level of thermal comfort expectation.

4.2. Analysis of RL Algorithms

Each RL algorithm demonstrates unique characteristics under the aforementioned weight settings:

Soft Actor-Critic (SAC): Under weight 0.1, SAC significantly reduces energy consumption but with a higher PPD, indicating a compromise in comfort. As the weight shifts towards thermal comfort (w = 0.9), SAC shows a balance, maintaining lower energy consumption while keeping the PPD close to the desired 6%. This performance makes SAC particularly suitable for environments that require normal levels of thermal comfort. Its ability to achieve a relatively low PPD while also providing substantial energy savings exemplifies its applicability in scenarios where both comfort and energy efficiency are important, but a perfect balance is not critical (like Category II).

Proximal Policy Optimization: PPO demonstrates moderate energy consumption across all weight settings, with consistently lower PPD values, indicating a steady performance in balancing energy efficiency and comfort. One of PPO's strengths is its ability to achieve lower PPD values, which is indicative of higher occupant thermal comfort. This is particularly significant in settings where maintaining a comfortable indoor environment is as important as energy efficiency. PPO shows a commendable adaptability to varying

weights in the reward function. Whether the focus is on energy efficiency or thermal comfort, PPO adjusts its strategy accordingly, showcasing its flexibility. Perhaps the most notable aspect of PPO is its balanced approach to energy efficiency and occupant comfort. Unlike some algorithms that may excel in one aspect but fall short in the other, PPO provides a harmonious balance, making it a versatile choice for a wide range of applications. Another aspect is related to the reliability it offers. In terms of operational predictability and reliability, PPO presents fewer fluctuations in performance metrics, which is beneficial for long-term planning and consistent building management operations. However, it is crucial to recognize that due to its intrinsic algorithmic design, PPO inherently lacks the granularity to precisely adjust the equilibrium between energy saving and thermal comfort objectives in this formulated problem. PPO is influenced by the reward function's design, neural network architecture, and entropy term. Adjustments to these factors can help fine-tune the algorithm's policy, potentially improving adherence to desired comfort levels. This limitation subtly guides its performance to align more closely with scenarios characteristic of Category I, irrespective of the weight variations in the reward function. PPO's operational framework, therefore, inherently favors occupant comfort optimization, a trait that becomes increasingly apparent under diverse reward function conditions. This inclination towards maintaining lower PPD values, despite shifts in prioritization, highlights PPO's aptness for environments where thermal comfort is paramount, yet also underscores a potential limitation in settings where a distinct emphasis on energy efficiency, with a more flexible balance, is essential.

Advantage Actor-Critic: At a weight of 0.1, A2C demonstrates the lowest energy usage among all tested algorithms, highlighting its strong inclination towards energy conservation, but with the highest PPD, suggesting a strong bias towards energy saving over comfort. As the weight shifts towards prioritizing thermal comfort (such as at weight 0.9), A2C shows a slight improvement in maintaining comfort levels. However, this improvement is marginal, suggesting that while A2C can adapt to different priorities, its strength lies predominantly in energy saving with a small fraction of penalty in thermal comfort. A2C's performance profile makes it a decent candidate for energy-critical applications, especially in scenarios where energy budgets are tight, and slight compromises in comfort can be tolerated. One of A2C's advantages is its predictability in energy-saving outcomes, making it a reliable option for long-term energy management strategies where consistent low energy usage is paramount.

Deep Deterministic Policy Gradient: DDPG's performance in terms of PPD and energy consumption remains relatively consistent across different weight settings, as indicated by its PPD values ranging from 24.4% to 25.15%. However, it is important to note that these PPD values, hovering around 25%, signify a lower level of occupant thermal comfort, aligning more with Category IV standards (Low Expectation, $PPD < 25\%$). While DDPG demonstrates a certain level of stability in its performance, it does so at a relatively lower standard of thermal comfort. This aspect is crucial for applications where higher thermal comfort is a priority.

Deep Q-Network: The performance of the DQN algorithm across different weights suggests a tendency towards higher energy consumption without proportionate gains in thermal comfort with an exception to w = 0.1, where it provides an adequate energy reduction with a small penalty on thermal comfort. Even at a weight of 0.9, where the focus is more on comfort, DQN consumes considerably more energy compared to the classic controller (-8.61%), while achieving only marginally better PPD values (6.3%). This trend is more pronounced at weights 0.5 and 0.9, where DQN's energy consumption far exceeds the baseline set by the classic controller, indicating inefficiency. This suggests that DQN, despite its potential to achieve lower PPD values, does so at a significant energy cost, making it less suitable for applications where energy efficiency is a priority or where a balance between energy consumption and thermal comfort is desired. The inherent design of DQN, particularly its approach to discretizing the action space, might be a contributing factor to its performance characteristics. Non-continuous discretization can limit the algorithm's

ability to fine-tune its actions for optimal performance, potentially leading to higher energy consumption and only marginal improvements in thermal comfort.

4.3. Overall Comparison and Implications

The analysis reveals a diverse range of responses from each RL algorithm to the prioritization of energy reduction versus thermal comfort. PPO and A2C exhibit a more balanced approach across different weights, suggesting their suitability for scenarios where both energy efficiency and comfort are equally prioritized. SAC and A2C are the most energy-efficient but they heavily compromise comfort in the w = 0.1 scenario. On the other hand, these two algorithms produce the desired performance on the other two weighting scenarios in both objective metrics. DDPG and DQN appear more inclined towards optimizing comfort, especially at higher weights, leading to degraded models.

However, it is essential to identify the best-performing algorithm for each thermal comfort category based on the results to provide clear guidance on which RL algorithms are most suitable for different levels of thermal comfort expectations, from the most strict (Category I) to the least (Category IV). Thus, the best-performing RL algorithms for each thermal comfort category, considering both energy consumption and PPD values, as follows:

Category I (High Expectation, $PPD < 6\%$**)**: For this category, where a high level of thermal comfort is expected, the algorithm that maintains PPD closest to 6% with the lowest energy consumption is ideal. In our results, PPO and A2C stand out as the most suitable choices, balancing energy efficiency while maintaining a high level of comfort. More specifically, the lowest achieved PPD values are 6.5% and 6.92% for PPO and A2C, respectively, under weight case w = 0.9, providing energy reduction of 26.83% and 29.81%. Both PPO and A2C demonstrate their capability to operate effectively in scenarios demanding stringent comfort requirements, as defined by Category I. Their performances suggest that they can achieve near-optimal thermal comfort levels while also contributing to energy savings, making them well-suited for applications where occupant comfort is a paramount concern, but energy efficiency cannot be overlooked.

Category II (Normal Expectation, $PPD < 10\%$**)**: Here, the acceptable level of discomfort is slightly higher but still lies within the normal level of expectation. Thus, algorithms that can maintain PPD below 10% while optimizing energy consumption are preferred. SAC demonstrates a commendable balance between energy efficiency and thermal comfort. In our results, under the weight case w = 0.9, SAC achieved a PPD of 8.68%, which is within the acceptable range for Category II. Moreover, it managed to reduce energy consumption by nearly 40% (39.88%), indicating its effectiveness in optimizing energy usage while maintaining a reasonable level of occupant comfort. This performance makes SAC particularly suitable for environments that require normal levels of thermal comfort. Its ability to achieve a relatively low PPD while also providing substantial energy savings exemplifies its applicability in scenarios where both comfort and energy efficiency are important, but a perfect balance is not critical.

Category III (Moderate Expectation, $PPD < 15\%$**)**: Based on the produced results, no case lies within the PPD range of [10%, 15%), so the selected algorithm would still be SAC. We should select an algorithm that keeps the PPD within this threshold while optimizing energy consumption. Looking at the data, SAC with weight 0.9 could be a better fit for Category III, as it has a PPD of 8.68%, which is within the threshold, and offers a reasonable energy reduction of 39.88%.

Category IV (Low Expectation, $PPD < 25\%$**)**: This category allows for a higher level of discomfort in favor of energy savings, but the PPD still needs to be below 25%. DDPG with weight 0.1 might be a suitable option for Category IV, as it has a PPD of 24.4%, which is within the threshold, and an average HVAC power of 191.3231 Wh, indicating high energy efficiency (55.53% consumption reduction with respect to classic controller).

Figure 4 provides a visualized representation of Table 4 to easily compare the trade-offs between energy savings and thermal comfort provided by each algorithm under different preference weightings, giving a clear picture of which algorithms are more suitable for

certain comfort categories and energy efficiency objectives. The data points are color-coded and shape-coded for each RL algorithm across three different weight conditions (w = 0.1, w = 0.5, and w = 0.9). The lines with arrows are intended to depict the trajectory from energy reduction towards thermal comfort for each RL algorithm as the weighting factor changes from w = 0.1 to w = 0.9. This way, the arrows represent the direction of increasing weight on the PPD in the reward function, moving from right to left on the graph.

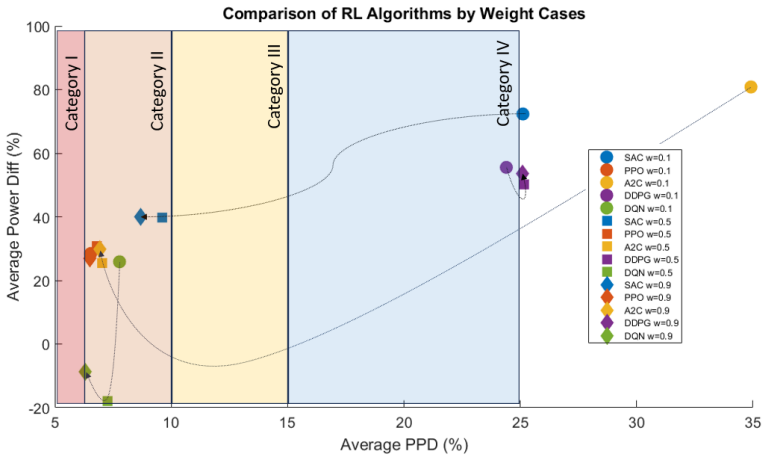


Figure 4. Visualized version of Table 4 incorporating thermal comfort categories. Dotted lines with arrows show the direction of each algorithm from energy saving to thermal comfort case based on reward function weights.

This categorization approach allows for a targeted selection of RL algorithms based on specific thermal comfort requirements and energy efficiency goals. It enables the implementation of more nuanced and effective building management strategies, catering to the varying needs of building occupants and operational efficiency mandates. In the broader context of multi-objective optimization in building management, these insights are critical. They not only inform the selection of appropriate algorithms for specific building environments but also highlight the inherent trade-offs between energy efficiency and occupant comfort. This understanding is pivotal for developing intelligent and sustainable building management systems that align with various occupant needs and environmental sustainability goals.

5. Conclusions and Future Work

The current study presents a comprehensive analysis of five prominent RL algorithms—PPO, DDPG, DQN, A2C, and SAC—in the context of residential energy management, with a focus on balancing energy efficiency and occupant comfort. The research stands out for its in-depth evaluation of these algorithms' performance in maintaining energy efficiency while ensuring thermal comfort, taking into account different occupant comfort expectations and energy efficiency goals. It should be noted that the study does not merely advance the perception of different RL applications in residential energy and comfort management but also serves as a guide for implementing RL algorithms in real-world scenarios. It underscores the potential of these algorithms to create more energy-efficient and comfortable living environments, while also emphasizing the importance of aligning algorithm selection with specific user preferences and comfort requirements

The current study quantified thermal comfort using the PPD, aligned with international standards, categorizing levels of thermal comfort expectations into four categories based on the PMV range. The results demonstrated that SAC and A2C are particularly effective in scenarios emphasizing energy savings, presenting minimal deviations in thermal comfort from the ideal thermal comfort category. PPO maintained a balanced performance in energy efficiency and thermal comfort irrespective of the weighting factors in the reward function. DDPG provided a lower level of occupant thermal comfort, leading to a

degraded performance, whereas DQN offered an adequate energy reduction with a small penalty on thermal comfort. However, DQN's tendency to increase energy consumption when prioritizing thermal comfort was evident. This analysis underscored the nuanced capabilities and limitations of each algorithm, suggesting that the optimal choice is highly dependent on specific energy and comfort goals. To this end, the study highlighted the importance of tailored algorithm selection in intelligent building management systems and offered insights for future applications aimed at harmonizing energy conservation with occupant comfort.

The future work generated from the current study is primarily focused on the real-life implementation of RL algorithms in residential energy management. This will provide invaluable data on their performance and robustness in diverse real-world environments, where variables such as varying weather conditions, different architectural designs, and fluctuating occupant behaviors play significant roles. Additionally, integrating user feedback mechanisms to refine the algorithms' responsiveness to dynamic comfort preferences portrays another essential aspect for the continuation of the work. Moreover, the integration of renewable energy sources (RESs) and the algorithms' adaptability to smart grid technologies may also significantly enhance their applicability and efficiency, aligning with broader sustainability goals. Such real-world application and continuous refinement will validate the research findings in real life, while also contributing to the evolution of more intelligent, adaptive, and user-centric home energy management systems (BEMS).

Author Contributions: Conceptualization, all authors; methodology, C.R.L. and I.M.; software, C.R.L., I.M. and G.K.; validation, all authors; formal analysis, P.M.; investigation, P.M.; resources, all authors; writing—original draft preparation, C.R.L., G.K., and P.M.; writing—review and editing, G.K. and P.M.; visualization, P.M.; supervision, I.M and E.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by MASTERPIECE project funded by the European Union's Horizon Europe programme, Grant Agreement ID. 101096836 (https://masterpiece-horizon.eu/) and also financed through SMART2B Horizon 2020 programme of the European Union, Grant Agreement ID. 101023666 (https://smart2b-project.eu/).

Data Availability Statement: Data are contained within the article.

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

Abbreviations

A2C	Advantage Actor-Critic
A3C	Asynchronous Advantage Actor-Critic
BCNN	Bayesian Convolutional Neural Network
DDPG	Deep Deterministic Policy Gradients
DQN	Deep Q-Network
HVAC	Heating, Ventilation, and Air Conditioning
LSTM	Long Short-Term Memory Neural Networks
PAR	Peak-to-Average Ratio
PMV	Predicted Mean Vote
PPD	Predicted Percentage of Dissatisfied
PPO	Proximal Policy Optimization
RBC	Rule-Based Control
SAC	Soft-actor-critic

Appendix A. Reinforcement Learning Algorithm Background

Appendix A.1. DQN Conceptual Background

The DQN methodology [50] was introduced in 2015, igniting the field of deep RL and replacing the need for a table to store the Q-values. Deep neural networks are used to approximate the Q-function for each state–action pair in a given environment, by minimizing the mean squared error between actual and predicted Q-values. The contributions of DQN

are as follows. (i) Policies can be learned directly utilizing a design of an end-to-end RL approach. (ii) The training of action value function approximation is stabilized with the adoption of deep neural networks that use the core ideas of experience replay (removes the correlations in the observation sequence and smoothes over changes in the data distribution) and target network. (iii) It has an iterative update that adjusts the action values (Q) towards target values that are only periodically updated, thereby reducing correlations with the target. (iv) A flexible network is trained using the same algorithm, architecture, and hyperparameters, performing well in diverse applications. The pseudo-code of DQN is presented in Algorithm A1.

Algorithm A1 DQN [50]

Initialize replay memory D
Initialize action-value function Q with random weights θ
Initialize target action-value function $\hat{Q}$ with weights $\theta^- = \theta$
for $episode = 1$ to M **do**
  Initialize sequence $s_1 = \{x_1\}$ and preprocessed sequence $\phi_1 = \phi(s_1)$
  for $t = 1$ to T **do**
    Following ϵ-greedy policy, select $a_t = \begin{cases} \text{a random action}, & \text{with probability } \epsilon \\ \arg\max_a Q(\phi(s_t), a; \theta), & \text{otherwise} \end{cases}$
    Execute action a_t in emulator and observe reward r_t and image x_{t+1}
    Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$
    Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in D ▷ Store the experience in replay buffer
    Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from D
    Set $y_j = \begin{cases} r_j, & \text{if episode terminates at step } j+1 \\ r_j + \gamma \max_{a'} \hat{Q}(\phi_{j+1}, a'; \theta^-), & \text{otherwise} \end{cases}$
    Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ with respect to the network parameters θ ▷ Update the Q-network by minimizing the loss
    Every C steps reset $\hat{Q} = Q$, i.e., set $\theta^- = \theta$ ▷ Periodic update of target Q-network
  end for
end for

Appendix A.2. DDPG Conceptual Background

Deep Deterministic Policy Gradient (DDPG) [51] provides a combination of DQN and deterministic policy gradient (DPG) [52] in an actor-critic and model-free approach for continuous action spaces. In contrast with DQN, which tries to predict the Q-values for each state–action pair at every time step, obtaining a greedy policy, DDPG is an actor-critic method. DDPG adopts the ideas of experience replay (store past transitions and off-policy learning) and separate target network (stabilize learning) from DQN. Another issue for DDPG is that it seldom performs exploration for actions. Additionally, in the DDPG implementation, the update frequency of the target networks is modified, keeping a slower track of the trained networks compared with DQN. Thus, the updates in the target network weight parameters are being updated after each update of the trained network using a sliding average for both the actor and the critic; thus, $\theta : \theta' \leftarrow \tau\theta + (1\text{-}\tau)\theta'$ with $\tau \ll 1$. Using this update rule, the target networks are always "late" concerning the trained networks, providing more stability to the learning of Q-values. The next-state Q-values are calculated with the target value network and target policy network. The key idea borrowed from DPG is the policy gradient for the actor. The critic is learned using regular Q-learning and target networks minimizing the mean squared loss between the updated Q-value and the original Q-value (the original Q-value is calculated with the value network, not the target value network). For the actor, to calculate the policy loss, the derivative of the objective function concerning the policy parameter is taken. The actor network in DDPG simply uses the negative average Q-value generated by the critic model as the loss for it.

This way, the actor network learns to generate actions to maximize the Q-value in each state. The general pseudo-code of the DDPG algorithm is represented in Algorithm A2.

Algorithm A2 DDPG [51]

Randomly initialize critic network $Q(s, a \mid \theta^Q)$ and actor $\mu(s \mid \theta^\mu)$ with weights θ^Q and θ^μ

Initialize target network Q' and μ' with weights $\theta^{Q'} \leftarrow \theta^Q$, $\theta^{\mu'} \leftarrow \theta^\mu$

Initialize replay buffer R

for $episode = 1$ to M **do**

  Initialize a random process N for action exploration sequence

  Receive initial observation state s_1

  for $t = 1$ to T **do**

    Select action $a_t = \mu(s_t \mid \theta^\mu) + N_t$ according to the current policy and exploration noise

    Execute action a_t and observe reward r_t and observe new state s_{t+1}

    Store transition (s_t, a_t, r_t, s_{t+1}) in R ▷ Store the experience in replay buffer

    Sample a random minibatch of N transitions (s_i, a_i, r_i, s_{i+1}) from R

    Set $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1} \mid \theta^{\mu'}) \mid \theta^{Q'})$

    Update critic by minimizing the loss: $L = \frac{1}{N}\sum_i (y_i - Q(s_i, a_i \mid \theta^Q))^2$

    Update the actor policy using the sampled policy gradient:

$$\nabla_{\theta^\mu} J(\theta) \approx \frac{1}{N}\sum_i [\nabla_a Q(s, a \mid \theta^Q) \mid_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu} \mu(s \mid \theta^\mu) \mid_{s=s_i}]$$

    Update the target networks:

$$\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}$$
$$\theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}$$

  end for

end for

Appendix A.3. PPO Conceptual Background

The PPO methodology [53] has gained prominence for its stability and efficiency in training agents to perform tasks within diverse environments. PPO is classified as a policy optimization method and operates as an on-policy algorithm. The algorithm's distinguishing feature is its utilization of a trust region approach, which constrains policy changes to avoid large deviations from the existing policy. This trust region is enforced through a clipped objective function, which curbs excessive policy adjustments. PPO often combines value function estimation to reduce variance, performs multiple optimization epochs, and employs exploration strategies for efficient learning. With its versatility and proven track record, PPO has been widely applied to tackle complex tasks across a variety of domains, making it a valuable tool in the field of RL. The PPO consists of two neural networks belonging to the actor-critic family of approaches. The policy network is represented by the actor determining the policy function $\pi_\theta(s, a)$. The critic part provides the evaluation of the selected policy utilizing the estimation of state value function $\hat{V}_\phi^\pi(s)$ or $\hat{R}_t$. The parameters of the actor (θ) and critic (ϕ) are optimized in a separate way using mini-batch stochastic gradient descent. The critic parameters are updated utilizing a value loss function, while the policies from the policy network side are clipped using a hyperparameter ϵ, so that the probability ratio $r(\theta) = \frac{\pi_\theta(a|s)}{\pi_{\theta_{old}}(a|s)}$ is constrained within the interval $(1 - \epsilon, 1 + \epsilon)$. The latter means that the policy function is restricted from potentially large policy updates, providing enhanced stability during the training phase. The pseudo-code of PPO is given in Algorithm A3.

Algorithm A3 PPO [53]

Initialize policy parameters θ_0 and value function parameters ϕ_0

1: **for** $k = 0, 1, 2, \ldots$ **do**

2: Collect set of trajectories $D_k = \{\tau_i\}$ by running policy $\pi_k = \pi(\theta_k)$ in the environment

3: Compute rewards-to-go $\hat{R}_t$

4: Compute advantage estimates, $\hat{A}_t$ based on the current value function V_{ϕ_k}

5: Update the policy by maximizing the PPO-Clip objective:

$$\theta_{k+1} = \arg\max_{\theta} \frac{1}{|D_k|T} \sum_{\tau \in D_k} \sum_{t=0}^{T} \min \left(\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_k}(a_t \mid s_t)} A^{\pi_{\theta_k}}(s_t, a_t),\ g(\epsilon, A^{\pi_{\theta_k}}(s_t, a_t)) \right)$$

6: Fit value function by regression on mean squared error:

$$\phi_{k+1} = \arg\min_{\phi} \frac{1}{|D_k|T} \sum_{\tau \in D_k} \sum_{t=0}^{T} \left(V_\phi(s_t) - \hat{R}_t \right)^2$$

7: **end for**

Appendix A.4. SAC Conceptual Background

Soft Actor-Critic (SAC) [54] is an off-policy maximum entropy actor-critic algorithm known for its effectiveness in training agents to perform tasks in challenging and continuous-action environments. SAC is capable of reusing collected data efficiently for training. It leverages an entropy-regularized objective function to encourage exploration while optimizing the policy. One notable feature of SAC is its ability to balance between maximizing expected cumulative rewards and maximizing entropy, promoting both efficiency and exploration. SAC also incorporates a critic network to estimate the value function, reducing variance in policy updates and enhancing stability. The algorithm typically involves a target value network, and it employs a form of the soft Bellman backup, which helps maintain smooth policy updates. SAC is highly regarded for its robust performance and adaptability across a wide range of tasks, making it a valuable asset in the field of RL.

Soft Actor-Critic (SAC) distinguishes itself from conventional actor-critic methods by emphasizing the maximization of information entropy in addition to cumulative rewards. SAC favors stochastic policies, achieved by augmenting the objective function with an extra component representing the expected entropy of the policy. An adaptive temperature parameter (α) is introduced to regulate the trade-off between entropy and expected return. This parameter allows the agent to automatically adjust exploration based on the task's difficulty. More specifically, the temperature or trade-off coefficient is tuned automatically through minimizing the $J(\alpha)$ throughout the training (in every step), with $J(\alpha) = \mathbb{E}_{a_t \sim \pi_t}[-\alpha \ln(\pi_t(a_t \mid s_t)) - \alpha H_0]$ leading to $\alpha \leftarrow \alpha - \lambda \widehat{\nabla}_\alpha J(\alpha)$, where H_0 equals $-\dim(A_1)$ and A_1 is the dimensions of action. The significance of entropy maximization in SAC is rooted in its ability to promote policies that exhibit substantial exploration, thereby capturing multiple modes of near-optimal strategies. Furthermore, the augmentation of entropy acts as a protective measure against the premature convergence of policies to undesirable local minima. The Q-function parameters are updated using $\phi_i \leftarrow \phi_i - \lambda_Q \hat{\nabla}_{\phi_i} J_Q(\phi_i)$ for $i \in \{1, 2\}$ while for the policy weights as $\theta \leftarrow \theta - \lambda_\pi \hat{\nabla}_\theta J_\pi(\theta)$ and finally the target network parameters are updated using $\phi_i' \leftarrow \tau\phi_i + (1 - \tau)\phi_i'$ for $i \in \{1, 2\}$. The pseudo-code of the SAC algorithm is presented in Algorithm A4.

Algorithm A4 Soft-Actor-Critic (SAC) [54]

Initialize policy parameters θ, Q-function parameters ϕ_1, ϕ_2 and empty replay buffer D
Set target parameters equal to main parameters $\phi'_1 \leftarrow \phi_1, \phi'_2 \leftarrow \phi_2$

repeat
  Observe state s and select action $a \sim \pi_\theta(\cdot \mid s)$
  Execute a in the environment
  Observe next state s', reward r, and done signal d to indicate whether s' is terminal
  Store (s, a, r, s', d) in replay buffer D
  if it is time to update **then**
    for j in range (however many updates) **do**
      Randomly sample a batch of transitions, $B = \{(s, a, r, s', d)\}$ from D
      Compute targets for the Q functions:

$$y(r, s', d) = r + \gamma(1 - d)\left(\min_{i=1,2} Q_{\phi'_i}(s', \tilde{a}') - \alpha \log \pi_\theta(\tilde{a}' \mid s')\right), \; \tilde{a}' \sim \pi_\theta(\cdot \mid s')$$

      Update Q-functions by one step of gradient descent using

$$\nabla_{\phi_i} \frac{1}{|B|} \sum_{(s,a,r,s',d) \in B} \left(Q_{\phi_i}(s, a) - y(r, s', d)\right)^2 \text{ for } i = 1, 2$$

      Update policy by one step of gradient ascent using:

$$\nabla_\theta \frac{1}{|B|} \sum_{s \in B} \left(\min_{i=1,2} Q_{\phi_i}(s, \tilde{a}_\theta(s)) - \alpha \log \pi_\theta(\tilde{a}_\theta(s) \mid s)\right)$$

      where $\tilde{a}_\theta(s)$ is a sample from $\pi_\theta(\cdot \mid s)$ which is differentiable wrt θ via the reparametrization trick
      Update target networks with:

$$\phi'_i \leftarrow \tau\phi_i + (1 - \tau)\phi'_i \text{ for } i = 1, 2$$

    end for
  end if
until convergence

Appendix A.5. A2C Conceptual Background

The A2C method is a derivative of the more general actor-critic methods, which have been studied for several decades in RL. The specific formulation of A2C, especially in the context of DRL, was popularized by its asynchronous version, the Asynchronous Advantage Actor-Critic (A3C) [55]. Such techniques utilize two neural networks: the actor, which proposes actions based on the current environment state, and the critic, which evaluates these actions. To this end, while traditional Q-learning focuses on learning the Q-values directly, in the A2C method, the critic is trained to learn the advantage values instead of the Q-values. By emphasizing the advantage, the algorithm assesses actions not just by their intrinsic value, but also by their superiority relative to other possible actions. As a consequence, the algorithm reduces the high variance often seen in policy networks, leading to a more stable model.

In the HVAC framework, A2C illustrates a promising approach to curb energy wastage. Balancing energy efficiency with occupant comfort is a challenging problem, making it an ideal candidate for sophisticated RL techniques like A2C. By continuously learning and adapting to changing conditions, A2C is adequate to dynamically adjust HVAC parameters, such as temperature setpoints or airflow rates, aiming for an optimal trade-off between energy conservation and user comfort. Algorithm A5 illustrates the pseudo-code of A2C.

Algorithm A5 Advantage Actor-Critic

Initialize Actor network with random weights θ
Initialize Critic network with random weights ϕ
Initialize empty experience buffer D
for $episode = 1$ to M **do**
    Initialize sequence s_1
    for $t = 1$ to T **do**
        Use Actor to get policy $\pi(a|s_t;\theta)$
        Sample action a_t from π
        Execute action a_t in environment and observe reward r_t and next state s_{t+1}
        Store transition (s_t, a_t, r_t, s_{t+1}) in D
        Sample random minibatch of transitions (s_j, a_j, r_j, s_{j+1}) from D
        Use Critic to get value estimates $V(s_j;\phi)$ and $V(s_{j+1};\phi)$
        Compute advantage $A(s_j, a_j) = r_j + \gamma V(s_{j+1};\phi) - V(s_j;\phi)$
        Update Actor using gradient ascent on $\log \pi(a_j|s_j;\theta) \times A(s_j, a_j)$
        Update Critic by minimizing $(r_j + \gamma V(s_{j+1};\phi) - V(s_j;\phi))^2$
    end for
end for

Appendix B. Architecture and Hyperparameter Configuration of RL Algorithms

This appendix is dedicated to ensuring the reproducibility of the results presented in this study. It details the specific hyperparameter configurations for each RL algorithm evaluated. Providing this information is essential for transparency and allows other researchers and practitioners to replicate the experiments, verify the findings, and extend the work if desired. For each RL algorithm—A2C, DDPG, DQN, SAC, and PPO—we include tables that list the parameters utilized along with their default values. These parameters encompass learning rates, neural network architectures, discount factors, and other key settings that significantly influence algorithm performance (see Tables A1–A5).

Table A1. A2C hyperparameter configuration.

Parameter	Default Value
Learning Rate	7×10^{-4}
Number of Steps per Rollout	5
Discount Factor (γ)	0.99
Neural Network Architecture	2 layers, 64 neurons each
Entropy Coefficient	0.01

Table A2. DDPG hyperparameter configuration.

Parameter	Default Value
Learning Rate	1×10^{-3}
Batch Size	100
Discount Factor (γ)	0.99
Neural Network Architecture	2 layers (400, 300 neurons)
Replay Buffer Size	1,000,000
Polyak Coefficient (Tau)	0.005
Exploration Noise (Std Dev)	0.2
Noise Clip	0.5

Table A3. DQN hyperparameter configuration.

Parameter	Default Value
Learning Rate	1×10^{-4}
Batch Size	32
Discount Factor (γ)	0.99
Neural Network Architecture	2 layers, 64 neurons each
Replay Buffer Size	1,000,000
Exploration Strategy (Epsilon)	Start: 1.0, End: 0.05
Target Network Update Frequency	1000 steps
Learning Starts	5000 steps

Table A4. SAC hyperparameter configuration.

Parameter	Default Value
Learning Rate	3×10^{-4}
Batch Size	256
Discount Factor (γ)	0.99
Neural Network Architecture	2 layers, 256 neurons each
Polyak Coefficient (Tau)	0.005

Table A5. PPO hyperparameter configuration.

Parameter	Default Value
Learning Rate	3×10^{-4}
Batch Size	64
Discount Factor (γ)	0.99
Neural Network Architecture	2 layers, 64 neurons each
Number of Epochs	10
Clip Range	0.2
GAE Lambda	0.95
Value Function Coefficient	0.5
Entropy Coefficient	0.01
Number of Steps per Rollout	2048

Appendix C. Analytical Results for Each RL Algorithm

This appendix provides a visual compendium of the simulation results for various RL algorithms applied within the scope of residential building energy management (see Figures A1–A15). The following visual data are presented for each algorithm under multiple weight scenarios:

- *Predicted Percentage of Dissatisfied Measurements*: Illustrations of the PPD across different building zones, reflecting the occupant comfort levels attained during the simulations.
- *Zone Temperature Measurements*: Temperature readings for each building zone, demonstrating the algorithms' effectiveness in maintaining thermal conditions.
- *HVAC Energy Consumption Profiles*: Graphs depicting the energy consumed by the HVAC system throughout the day, showcasing the algorithms' energy performance.

Each of the 45 images within this appendix is integral to the comprehensive evaluation of the algorithms' performance, providing an empirical basis for the analysis discussed in the main body of the paper. These images offer readers an opportunity to visually assess the impact of RL-driven control on both the micro-scale (zone-specific comfort and temperature) and macro-scale (overall energy consumption) aspects of building management.

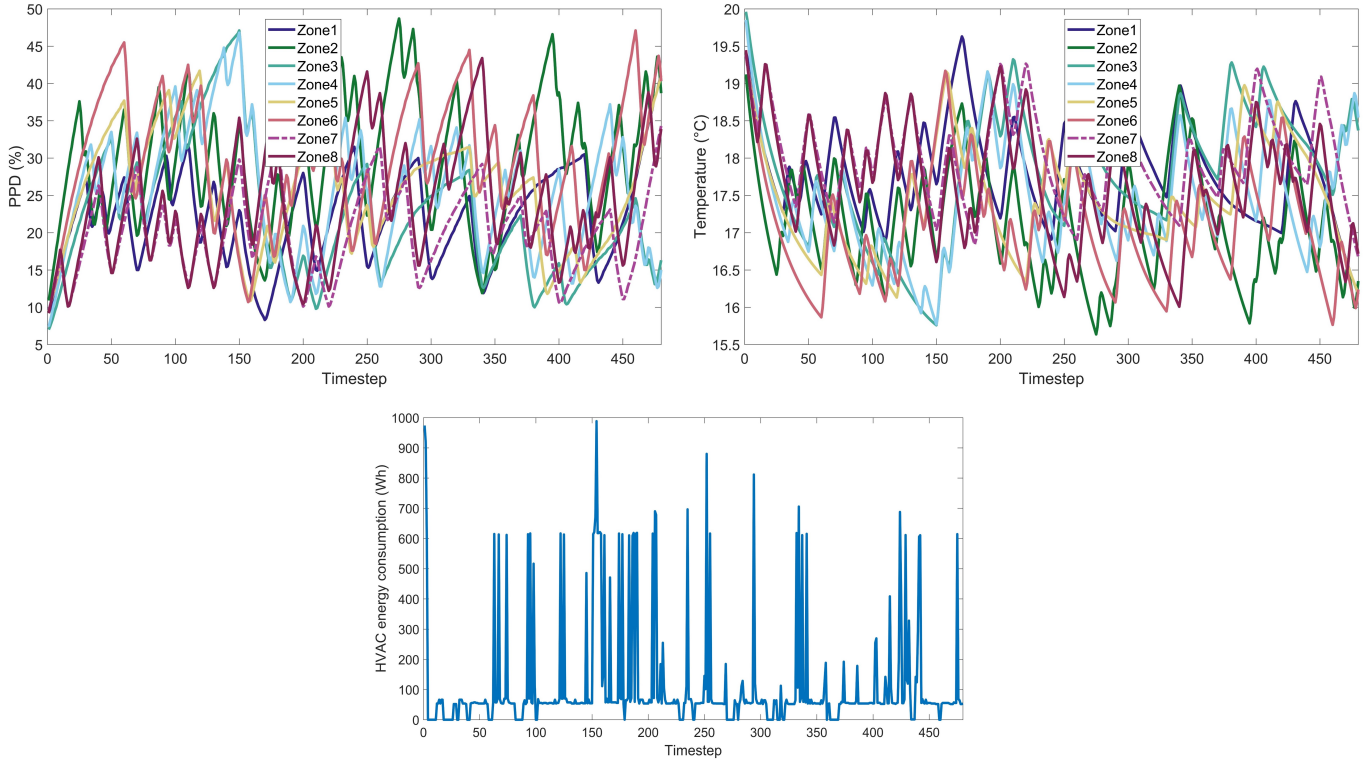


Figure A1. SAC under weight case w = 0.1. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

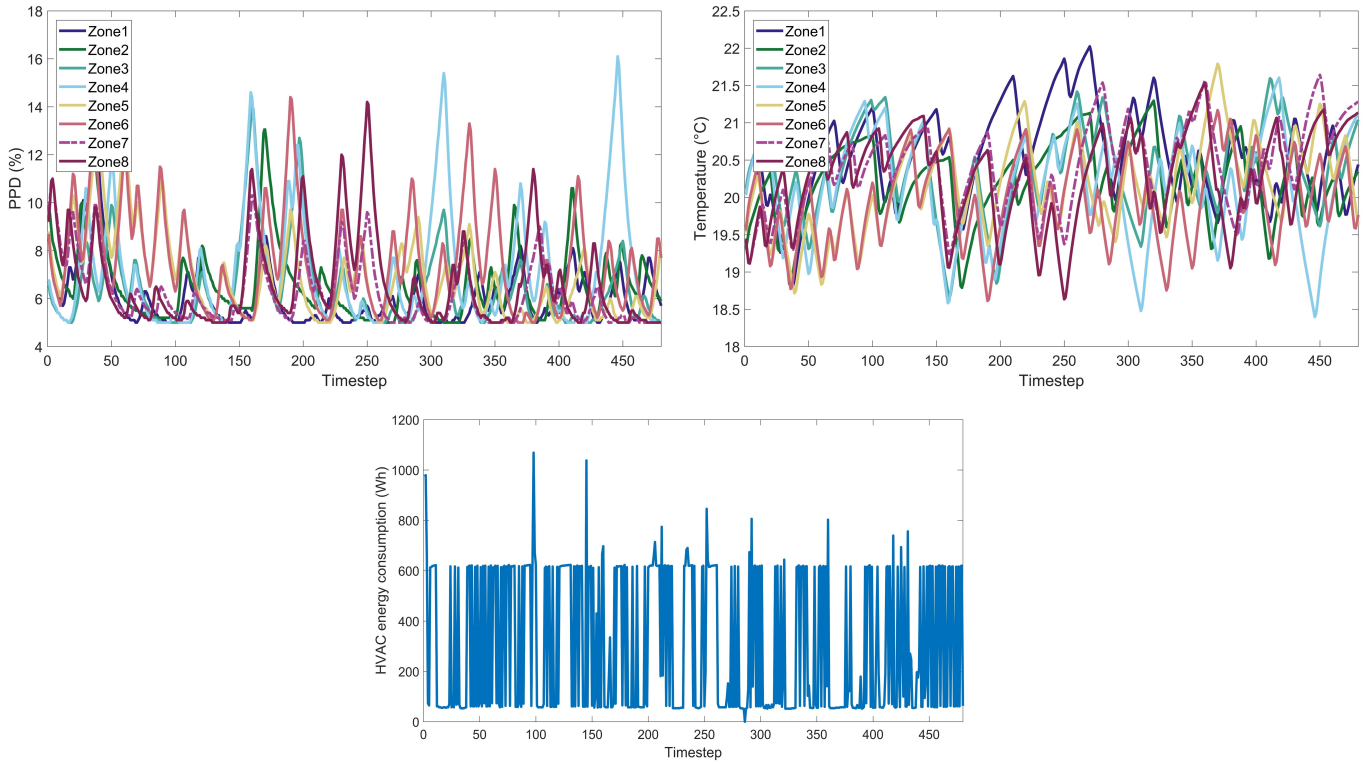


Figure A2. PPO under weight case w = 0.1. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

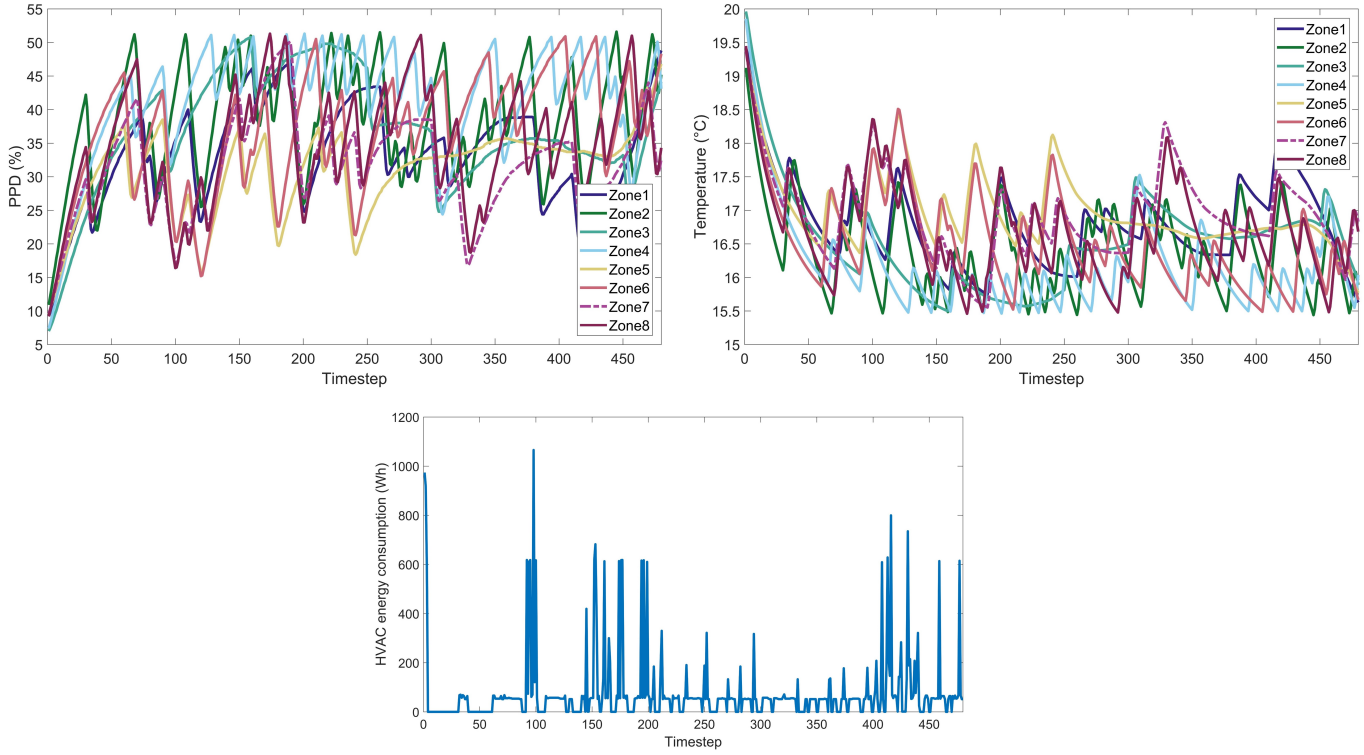


Figure A3. A2C under weight case w = 0.1. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

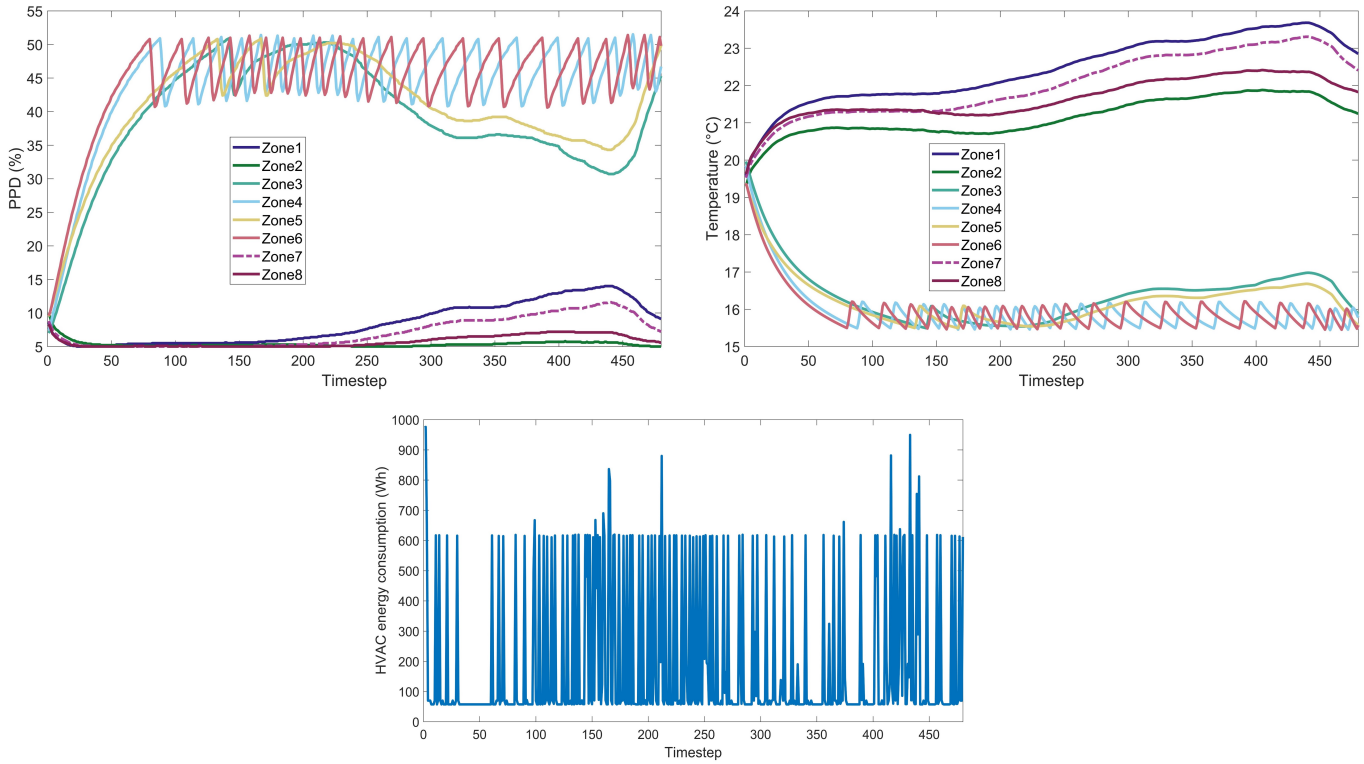


Figure A4. DDPG under weight case w = 0.1. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

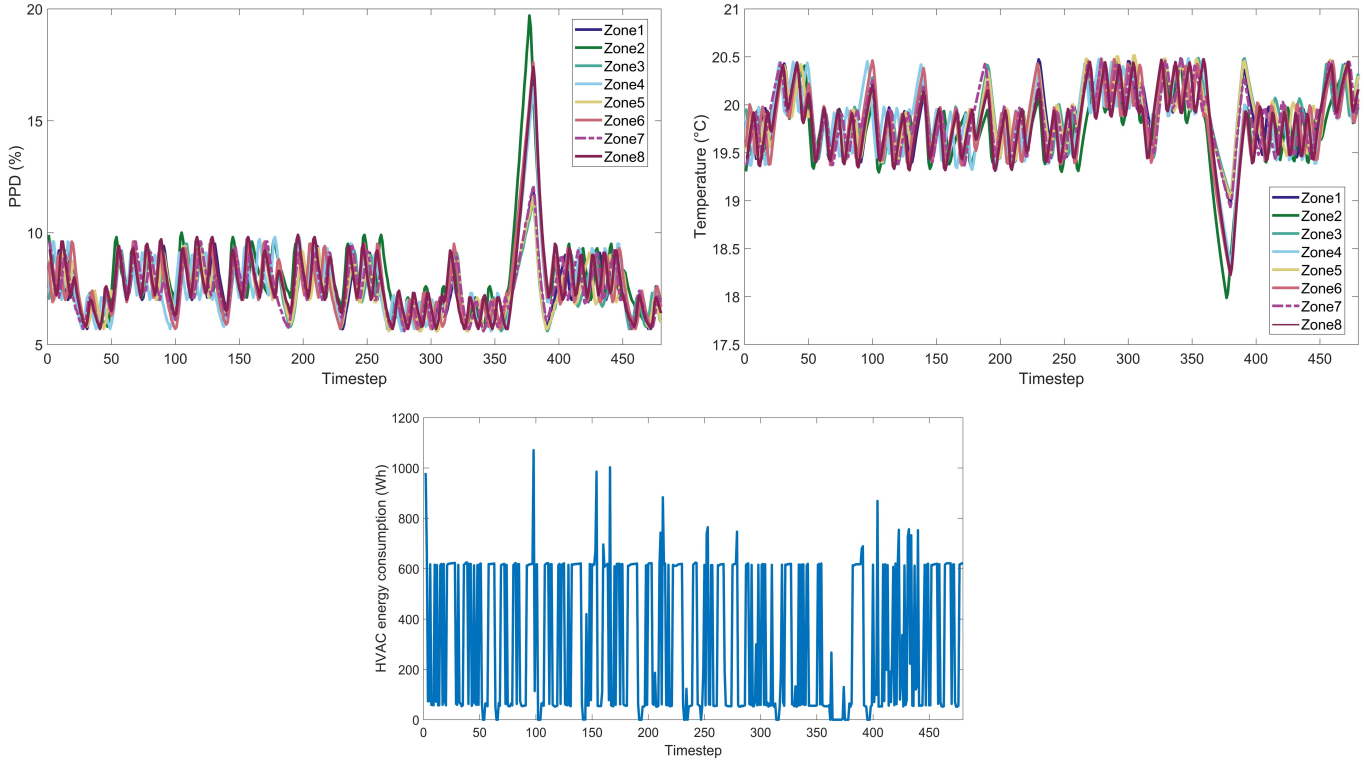


Figure A5. DQN under weight case w = 0.1. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

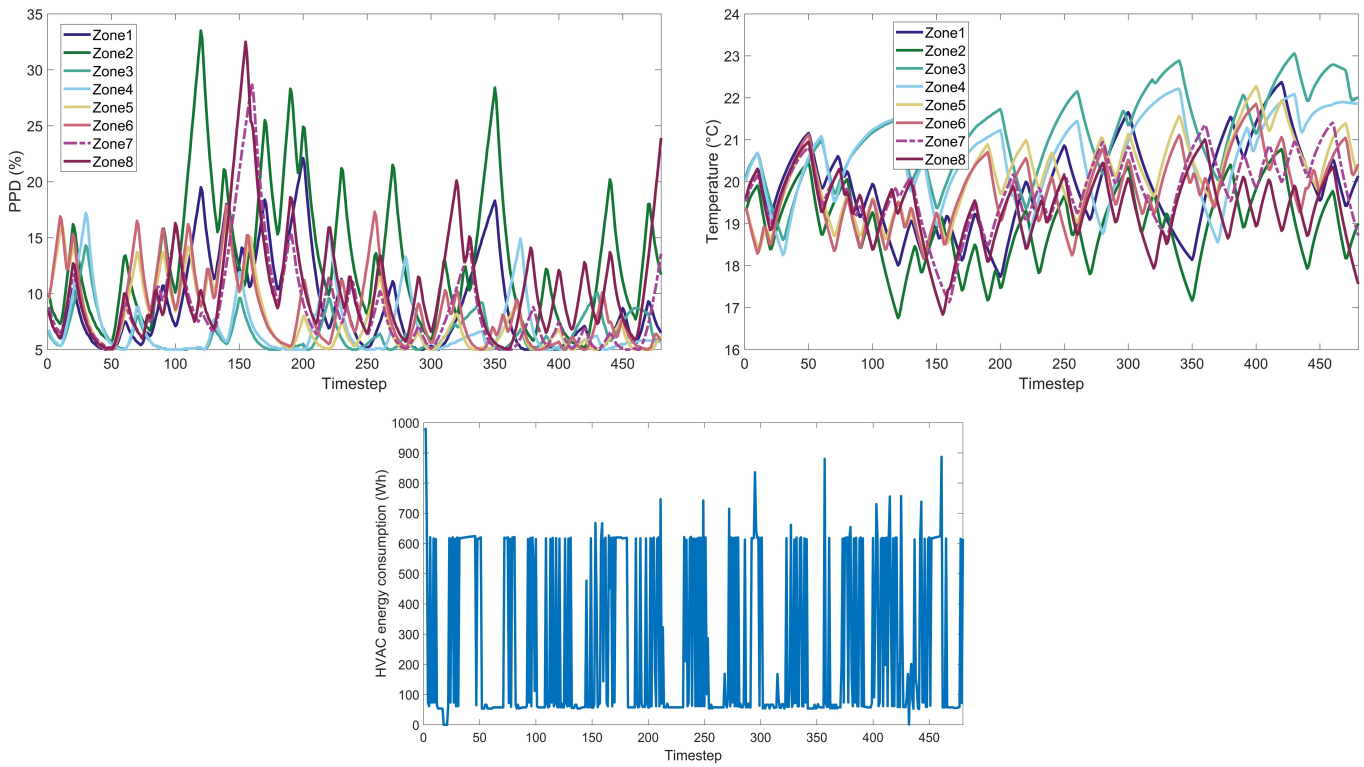


Figure A6. SAC under weight case w = 0.5. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

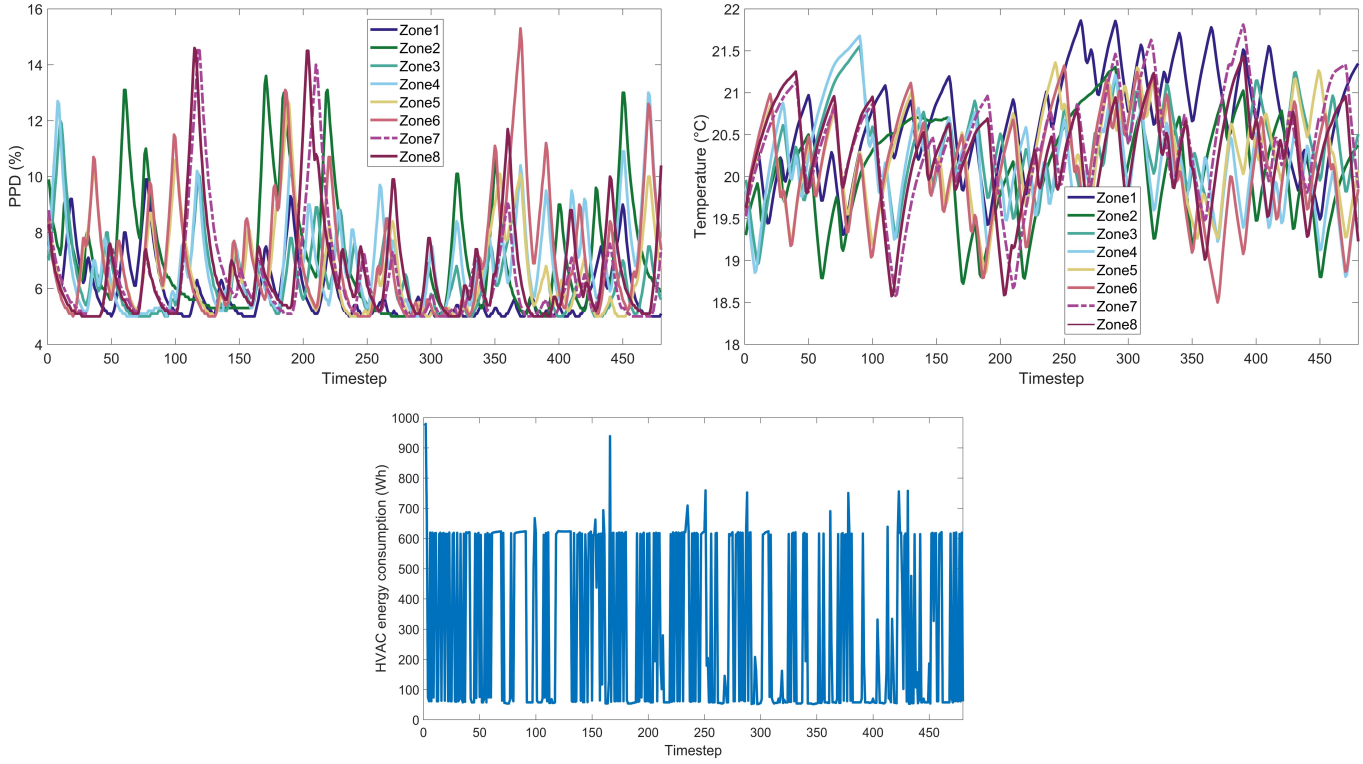


Figure A7. PPO under weight case w = 0.5. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

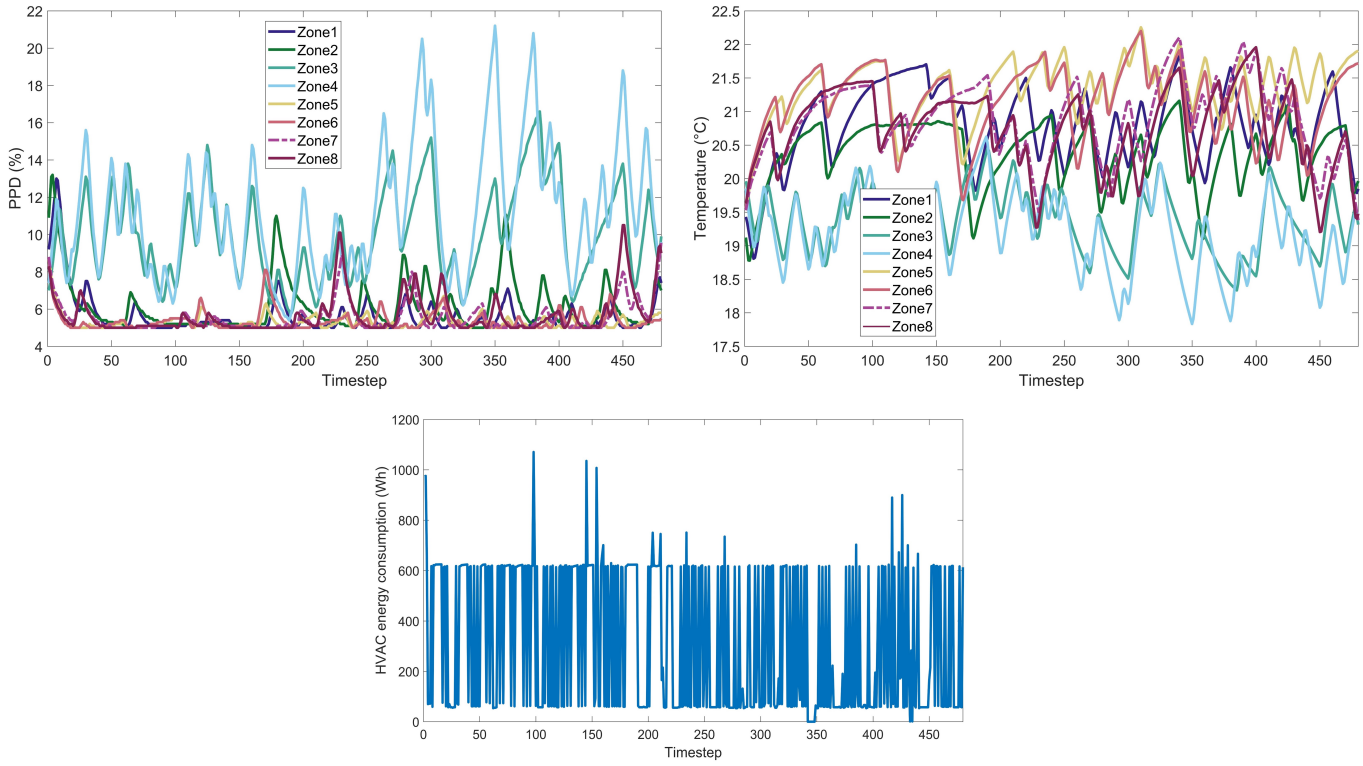


Figure A8. A2C under weight case w = 0.5. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

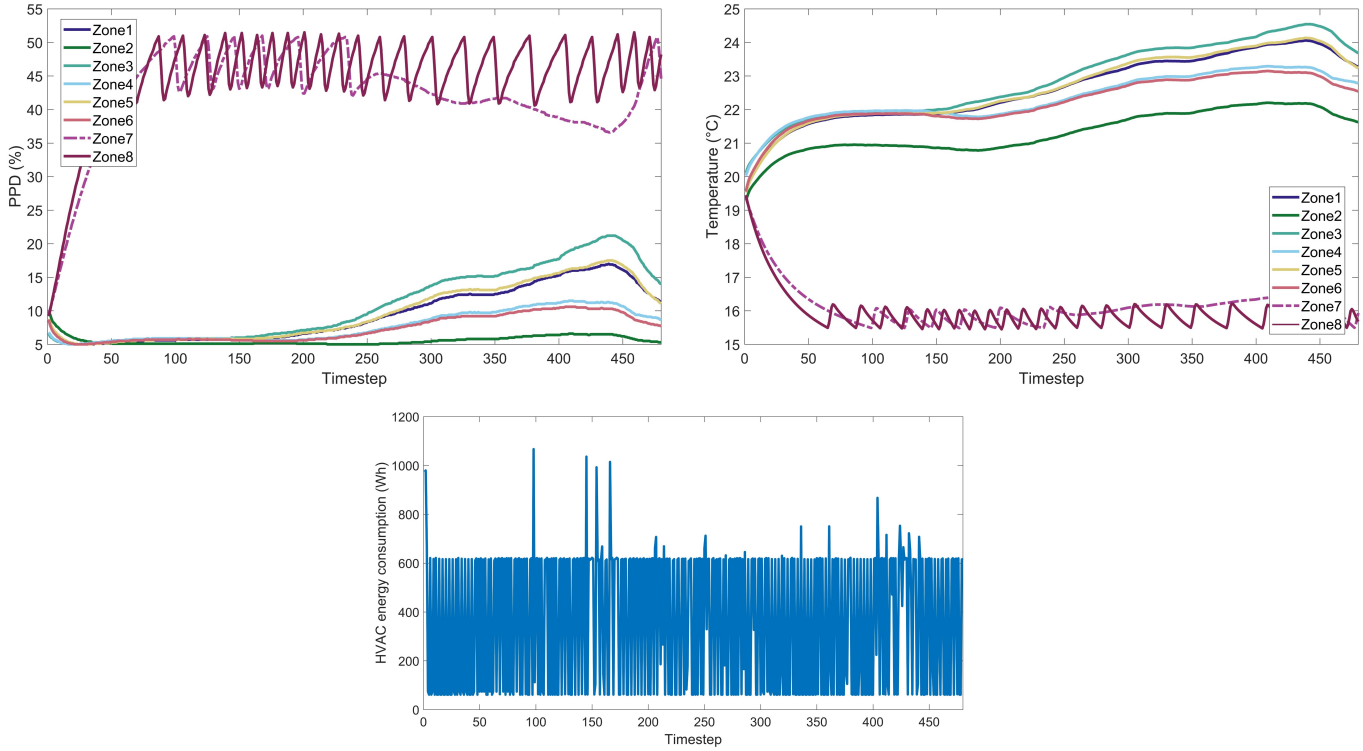


Figure A9. DDPG under weight case w = 0.5. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

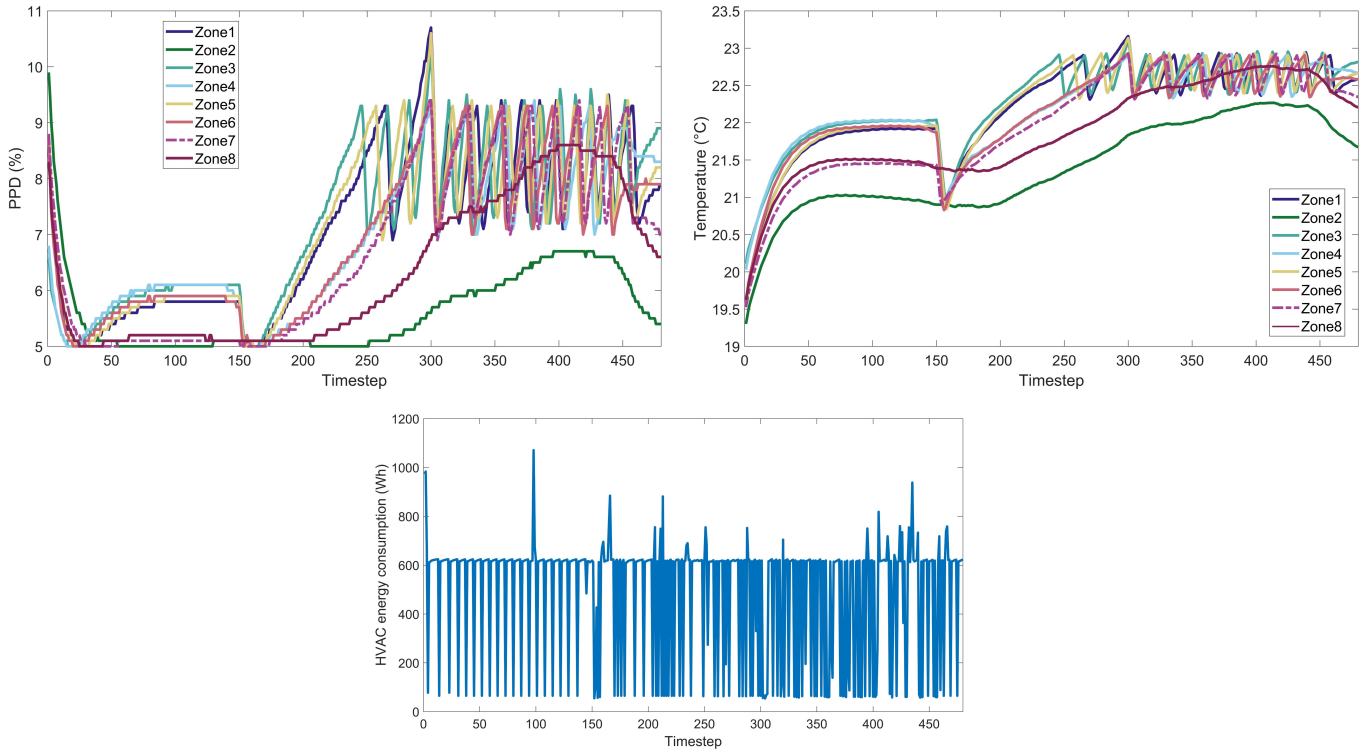


Figure A10. DQN under weight case w = 0.5. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

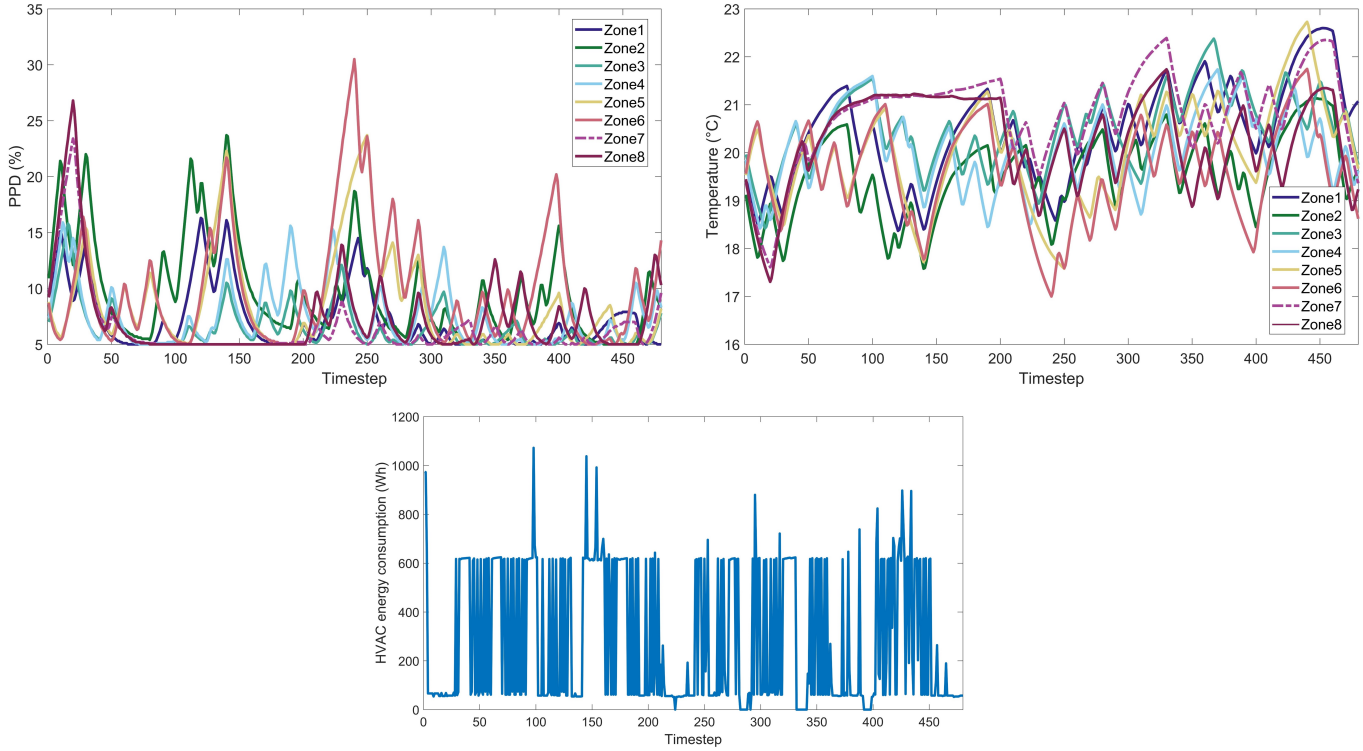


Figure A11. SAC under weight case w = 0.9. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

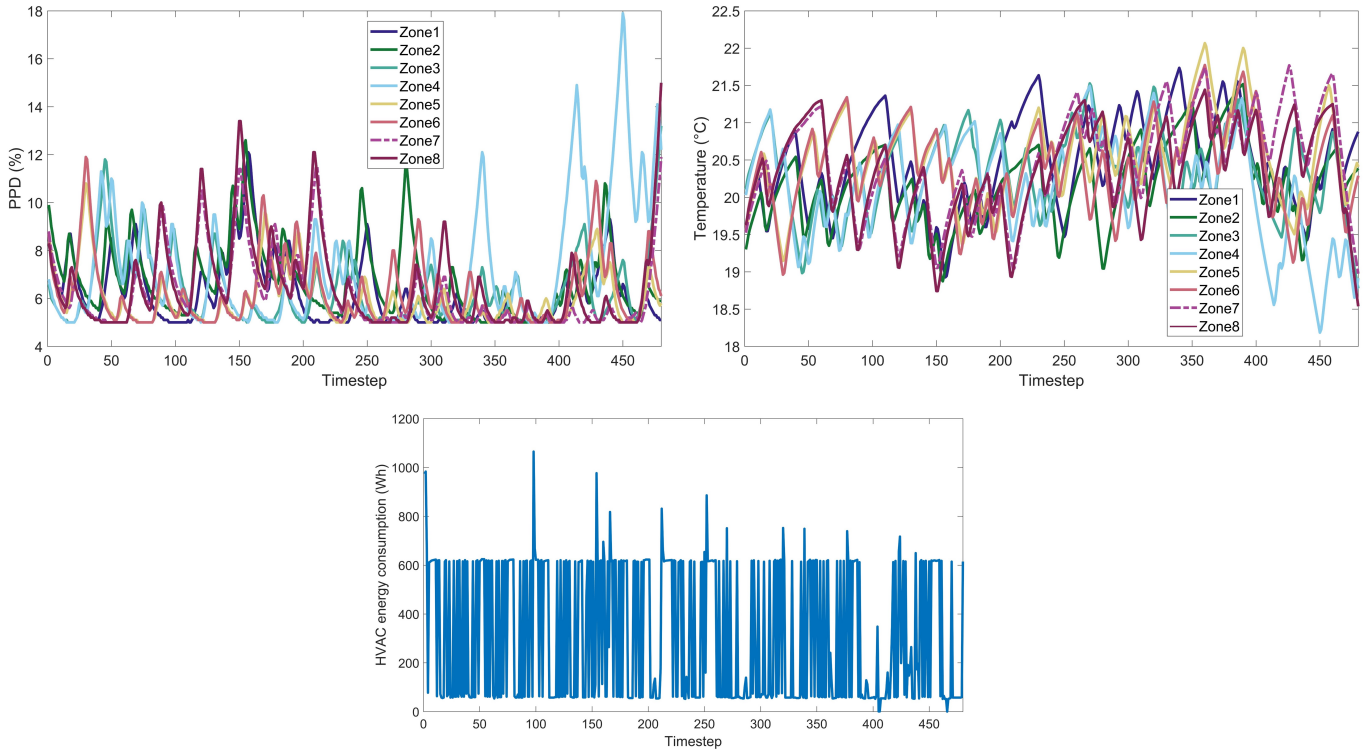


Figure A12. PPO under weight case w = 0.9. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

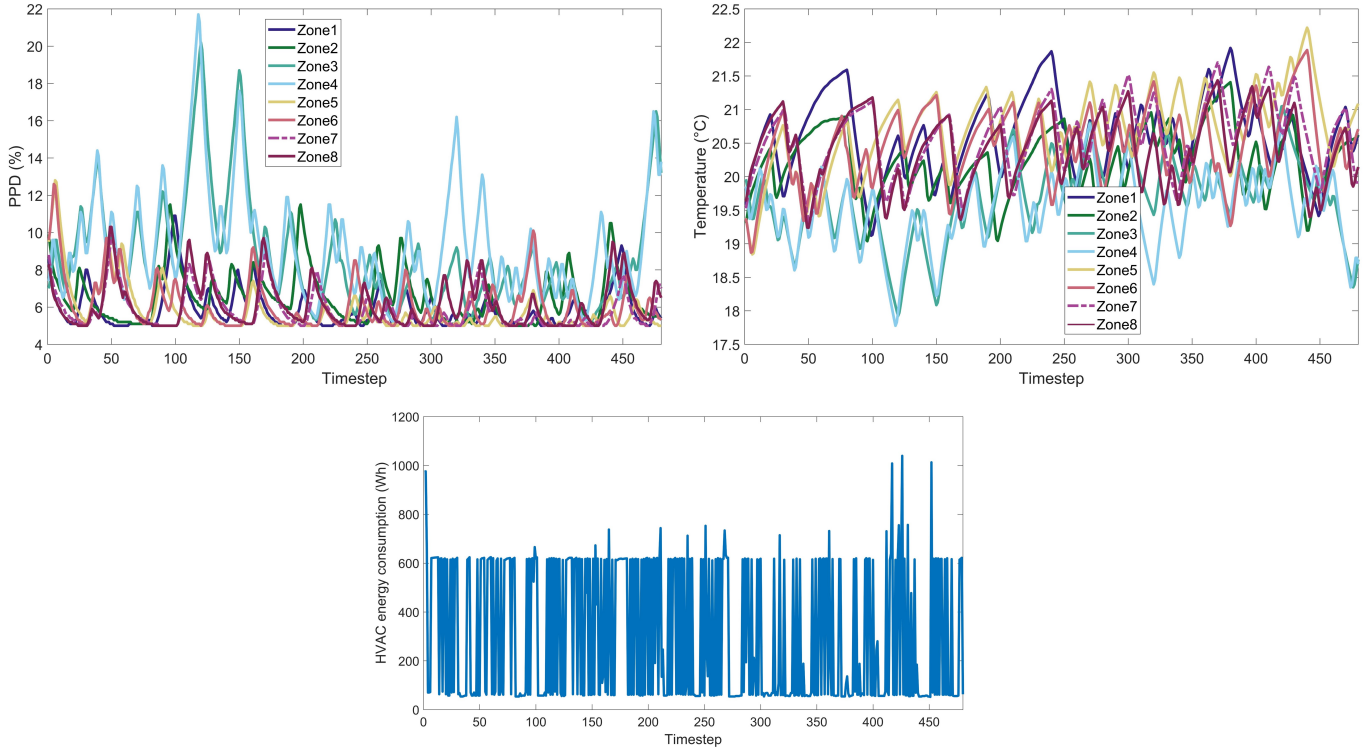


Figure A13. A2C under weight case w = 0.9. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

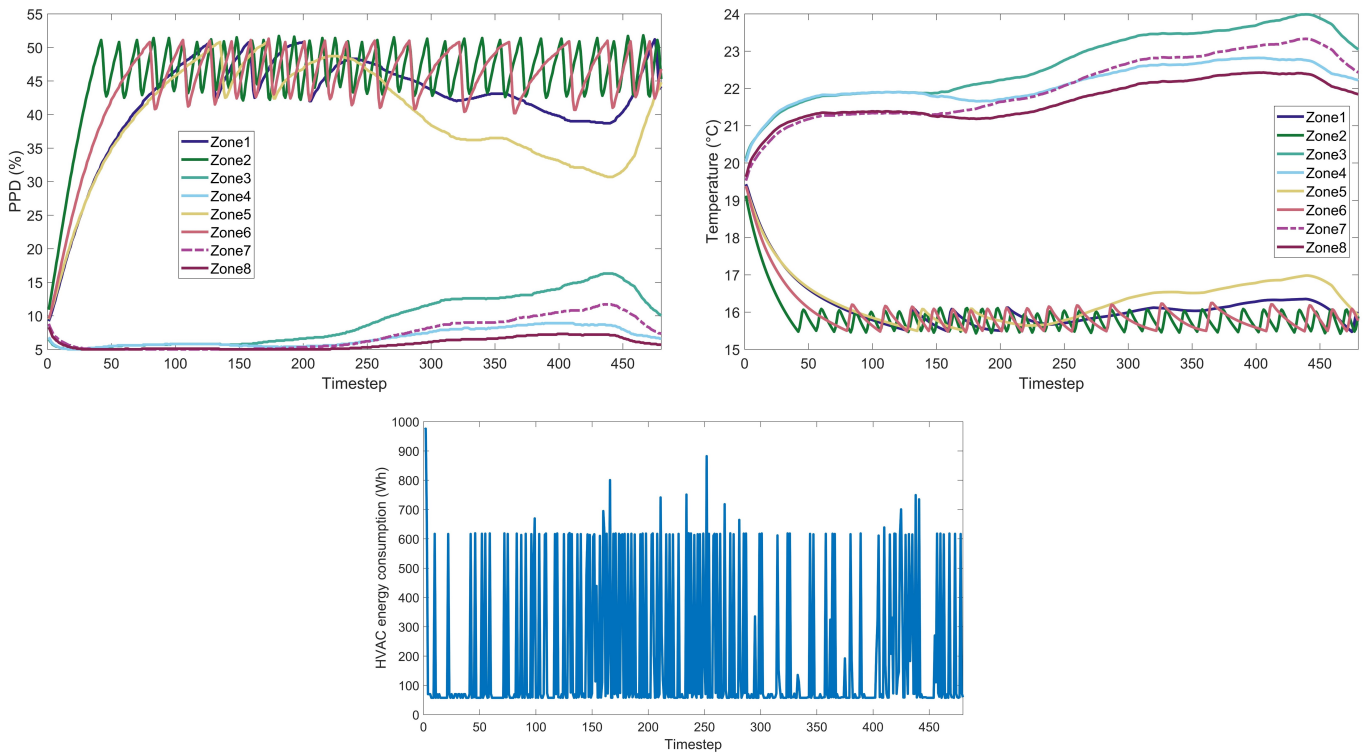


Figure A14. DDPG under weight case w = 0.9. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

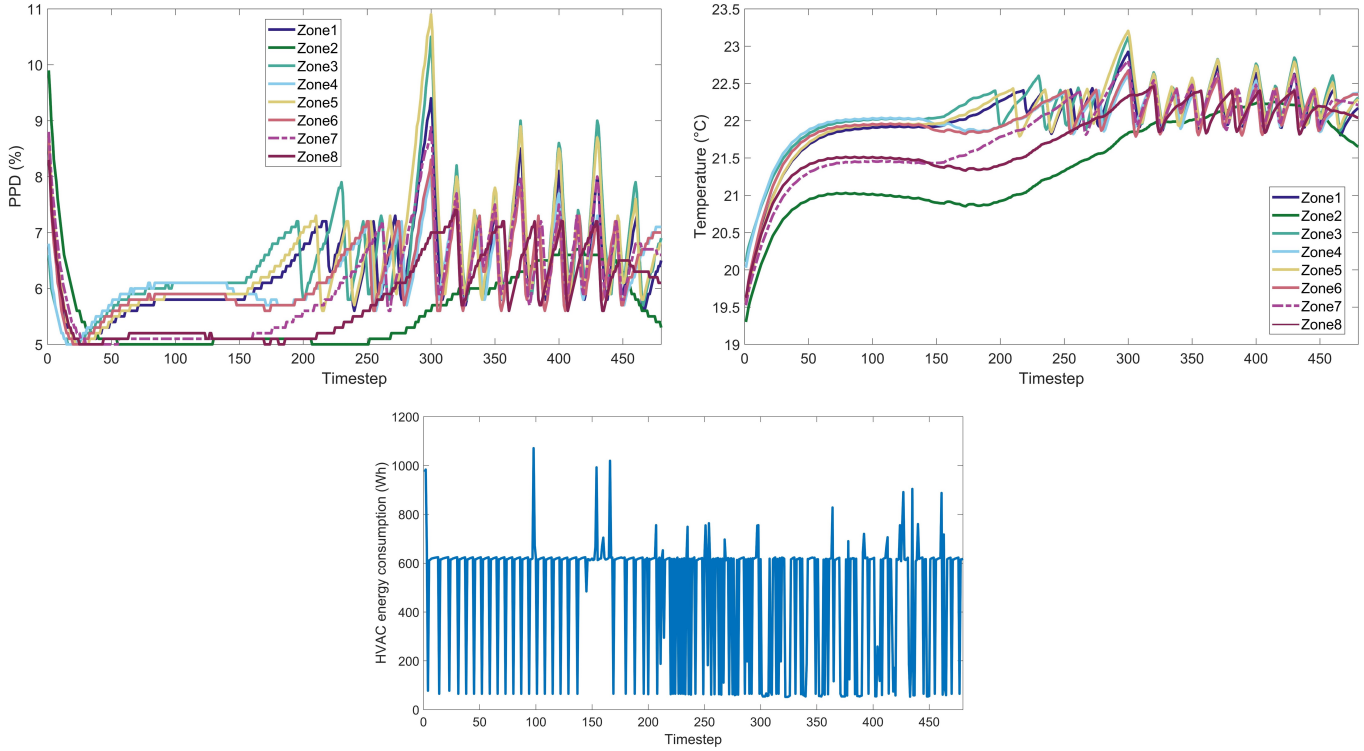


Figure A15. DQN under weight case w = 0.9. (**Upper left**): PPD for each building zone; (**Upper right**): Temperature for each building zone; (**Lower**): HVAC energy consumption within day.

References

1. Marikyan, D.; Papagiannidis, S.; Alamanos, E. A systematic review of the smart home literature: A user perspective. *Technol. Forecast. Soc. Chang.* **2019**, *138*, 139–154. [CrossRef]
2. Dimara, A.; Anagnostopoulos, C.N.; Kotis, K.; Krinidis, S.; Tzovaras, D. BEMS in the Era of Internet of Energy: A Review. In *Proceedings of the Artificial Intelligence Applications and Innovations, Proceedings of the 17th IFIP WG 12.5 International Conference, AIAI 2021, Hersonissos, Crete, Greece, 25–27 June 2021*; Proceedings 17; Springer: Berlin/Heidelberg, Germany, 2021; pp. 465–476.
3. Alaa, M.; Zaidan, A.A.; Zaidan, B.B.; Talal, M.; Kiah, M.L.M. A review of smart home applications based on Internet of Things. *J. Netw. Comput. Appl.* **2017**, *97*, 48–65. [CrossRef]
4. Michailidis, P.; Michailidis, I.; Vamvakas, D.; Kosmatopoulos, E. Model-Free HVAC Control in Buildings: A Review. *Energies* **2023**, *16*, 7124. [CrossRef]
5. Keroglou, C.; Kansizoglou, I.; Michailidis, P.; Oikonomou, K.M.; Papapetros, I.T.; Dragkola, P.; Michailidis, I.T.; Gasteratos, A.; Kosmatopoulos, E.B.; Sirakoulis, G.C. A Survey on Technical Challenges of Assistive Robotics for Elder People in Domestic Environments: The ASPiDA Concept. *IEEE Trans. Med. Robot. Bionics* **2023**, *5*, 196–205. [CrossRef]
6. Miko, R.; Thompson, S. The Environmental Impact of Housing: Local and Global Ecological Footprint of a House. In *Future as Fairness*; Brill: New York, NY, USA, 2004.
7. Goldstein, B.; Gounaridis, D.; Newell, J.P. The carbon footprint of household energy use in the United States. *Proc. Natl. Acad. Sci. USA* **2020**, *117*, 19122–19130. [CrossRef]
8. Lettenmeier, M.; Laakso, S.; Toivio, V. Future Households: Smaller Footprint, Better Life. In *Boosting Resource Productivity by Adopting the Circular Economy*; Paul Scherrer Institute: Villigen, Switzerland, 2017; pp. 293–297.
9. Xue, F.; Zhao, J. Building thermal comfort research based on energy-saving concept. *Adv. Mater. Sci. Eng.* **2021**, *2021*, 7132437. [CrossRef]
10. Ma, Z.; Zhao, D.; She, C.; Yang, Y.; Yang, R. Personal thermal management techniques for thermal comfort and building energy saving. *Mater. Today Phys.* **2021**, *20*, 100465. [CrossRef]
11. Cottafava, D.; Magariello, S.; Ariano, R.; Arrobbio, O.; Baricco, M.; Barthelmes, V.; Baruzzo, G.; Bonansone, M.; Console, L.; Contin, L.; et al. Crowdsensing for a sustainable comfort and for energy saving. *Energy Build.* **2019**, *186*, 208–220. [CrossRef]
12. Michailidis, I.T.; Sangi, R.; Michailidis, P.; Schild, T.; Fuetterer, J.; Mueller, D.; Kosmatopoulos, E.B. Balancing energy efficiency with indoor comfort using smart control agents: A simulative case study. *Energies* **2020**, *13*, 6228. [CrossRef]
13. Xu, Y.; Peet, Y.T. Effect of an on/off HVAC control on indoor temperature distribution and cycle variability in a single-floor residential building. *Energy Build.* **2021**, *251*, 111289. [CrossRef]

14. Chinnakani, K.; Krishnamurthy, A.; Moyne, J.; Gu, F. Comparison of energy consumption in HVAC systems using simple ON-OFF, intelligent ON-OFF and optimal controllers. In Proceedings of the 2011 IEEE Power and Energy Society General Meeting, Detroit, MI, USA, 24–28 July 2011; pp. 1–6.
15. Oldewurtel, F.; Sturzenegger, D.; Morari, M. Importance of occupancy information for building climate control. *Appl. Energy* **2013**, *101*, 521–532. [CrossRef]
16. Michailidis, P.; Pelitaris, P.; Korkas, C.; Michailidis, I.; Baldi, S.; Kosmatopoulos, E. Enabling optimal energy management with minimal IoT requirements: A legacy A/C case study. *Energies* **2021**, *14*, 7910. [CrossRef]
17. Michailidis, I.T.; Schild, T.; Sangi, R.; Michailidis, P.; Korkas, C.; Fütterer, J.; Müller, D.; Kosmatopoulos, E.B. Energy-efficient HVAC management using cooperative, self-trained, control agents: A real-life German building case study. *Appl. Energy* **2018**, *211*, 113–125. [CrossRef]
18. Lu, X.; Fu, Y.; O'Neill, Z. Benchmarking high performance HVAC Rule-Based controls with advanced intelligent Controllers: A case study in a Multi-Zone system in Modelica. *Energy Build.* **2023**, *284*, 112854. [CrossRef]
19. Merabet, G.H.; Essaaidi, M.; Haddou, M.B.; Qolomany, B.; Qadir, J.; Anan, M.; Al-Fuqaha, A.; Abid, M.R.; Benhaddou, D. Intelligent building control systems for thermal comfort and energy-efficiency: A systematic review of artificial intelligence-assisted techniques. *Renew. Sustain. Energy Rev.* **2021**, *144*, 110969. [CrossRef]
20. Shaikh, P.H.; Nor, N.B.M.; Nallagownden, P.; Elamvazuthi, I.; Ibrahim, T. A review on optimized control systems for building energy and comfort management of smart sustainable buildings. *Renew. Sustain. Energy Rev.* **2014**, *34*, 409–429. [CrossRef]
21. Khajenasiri, I.; Estebsari, A.; Verhelst, M.; Gielen, G. A review on Internet of Things solutions for intelligent energy control in buildings for smart city applications. *Energy Procedia* **2017**, *111*, 770–779. [CrossRef]
22. Panchalingam, R.; Chan, K.C. A state-of-the-art review on artificial intelligence for Smart Buildings. *Intell. Build. Int.* **2021**, *13*, 203–226. [CrossRef]
23. Vamvakas, D.; Michailidis, P.; Korkas, C.; Kosmatopoulos, E. Review and Evaluation of Reinforcement Learning Frameworks on Smart Grid Applications. *Energies* **2023**, *16*, 5326. [CrossRef]
24. Fu, Q.; Han, Z.; Chen, J.; Lu, Y.; Wu, H.; Wang, Y. Applications of reinforcement learning for building energy efficiency control: A review. *J. Build. Eng.* **2022**, *50*, 104165. [CrossRef]
25. Fang, X.; Gong, G.; Li, G.; Chun, L.; Peng, P.; Li, W.; Shi, X.; Chen, X. Deep reinforcement learning optimal control strategy for temperature setpoint real-time reset in multi-zone building HVAC system. *Appl. Therm. Eng.* **2022**, *212*, 118552. [CrossRef]
26. Blad, C.; Bøgh, S.; Kallesøe, C.S. Data-driven offline reinforcement learning for HVAC-systems. *Energy* **2022**, *261*, 125290. [CrossRef]
27. Lissa, P.; Schukat, M.; Barrett, E. Transfer learning applied to reinforcement learning-based hvac control. *SN Comput. Sci.* **2020**, *1*, 127. [CrossRef]
28. Du, G.; Zou, Y.; Zhang, X.; Liu, T.; Wu, J.; He, D. Deep reinforcement learning based energy management for a hybrid electric vehicle. *Energy* **2020**, *201*, 117591. [CrossRef]
29. Zhang, S.; Nandakumar, S.; Pan, Q.; Yang, E.; Migne, R.; Subramanian, L. Benchmarking Reinforcement Learning Algorithms on Island Microgrid Energy Management. In Proceedings of the 2021 IEEE PES Innovative Smart Grid Technologies-Asia (ISGT Asia), Brisbane, Australia, 5–8 December 2021; pp. 1–5.
30. Zhou, Y.; Ma, Z.; Zhang, J.; Zou, S. Data-driven stochastic energy management of multi energy system using deep reinforcement learning. *Energy* **2022**, *261*, 125187. [CrossRef]
31. Gupta, A.; Badr, Y.; Negahban, A.; Qiu, R.G. Energy-efficient heating control for smart buildings with deep reinforcement learning. *J. Build. Eng.* **2021**, *34*, 101739. [CrossRef]
32. Lork, C.; Li, W.T.; Qin, Y.; Zhou, Y.; Yuen, C.; Tushar, W.; Saha, T.K. An uncertainty-aware deep reinforcement learning framework for residential air conditioning energy management. *Appl. Energy* **2020**, *276*, 115426. [CrossRef]
33. Yu, L.; Xie, W.; Xie, D.; Zou, Y.; Zhang, D.; Sun, Z.; Zhang, L.; Zhang, Y.; Jiang, T. Deep reinforcement learning for smart home energy management. *IEEE Internet Things J.* **2019**, *7*, 2751–2762. [CrossRef]
34. Liu, B.; Akcakaya, M.; McDermott, T.E. Automated control of transactive hvacs in energy distribution systems. *IEEE Trans. Smart Grid* **2020**, *12*, 2462–2471. [CrossRef]
35. 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]
36. Azuatalam, D.; Lee, W.L.; de Nijs, F.; Liebman, A. Reinforcement learning for whole-building HVAC control and demand response. *Energy AI* **2020**, *2*, 100020. [CrossRef]
37. Lee, J.Y.; Rahman, A.; Huang, S.; Smith, A.D.; Katipamula, S. On-policy learning-based deep reinforcement learning assessment for building control efficiency and stability. *Sci. Technol. Built Environ.* **2022**, *28*, 1150–1165. [CrossRef]
38. Pinto, G.; Deltetto, D.; Capozzoli, A. Data-driven district energy management with surrogate models and deep reinforcement learning. *Appl. Energy* **2021**, *304*, 117642. [CrossRef]
39. Pinto, G.; Piscitelli, M.S.; Vázquez-Canteli, J.R.; Nagy, Z.; Capozzoli, A. Coordinated energy management for a cluster of buildings through deep reinforcement learning. *Energy* **2021**, *229*, 120725. [CrossRef]
40. Zhou, S.; Shah, A.; Leung, P.; Zhu, X.; Liao, Q. A Comprehensive Review of the Applications of Machine Learning for HVAC. *DeCarbon* **2023**, 100023. [CrossRef]

41. Jia, R.; Jin, M.; Sun, K.; Hong, T.; Spanos, C. Advanced building control via deep reinforcement learning. *Energy Procedia* **2019**, *158*, 6158–6163. [CrossRef]
42. Scharnhorst, P.; Schubnel, B.; Fernández Bandera, C.; Salom, J.; Taddeo, P.; Boegli, M.; Gorecki, T.; Stauffer, Y.; Peppas, A.; Politi, C. Energym: A building model library for controller benchmarking. *Appl. Sci.* **2021**, *11*, 3518. [CrossRef]
43. Raffin, A.; Hill, A.; Ernestus, M.; Gleave, A.; Kanervisto, A.; Dormann, N. Stable-baselines3: Reliable reinforcement learning implementations. *J. Mach. Learn. Res.* **2021**, *22*, 12348–12355.
44. Taleghani, M.; Tenpierik, M.; Kurvers, S.; Van Den Dobbelsteen, A. A review into thermal comfort in buildings. *Renew. Sustain. Energy Rev.* **2013**, *26*, 201–215. [CrossRef]
45. *ASHRAE 55*; Thermal Environmental Conditions for Human Occupancy. ASHRAE Standards: Peachtree Corners, GA, USA, 2004.
46. *ISO 7730*; Ergonomics of the Thermal Environment—Analytical Determination and Interpretation of Thermal Comfort Using Calculation of the Pmv and Ppd Indices and Local Thermal Comfort Criteria. ISO (International Organization for Standardization): Geneva, Switzerland, 2005.
47. *EN 16798*; Ventilation for Non-Residential Buildings-Performance Requirements for Ventilation and Room-Conditioning Systems. European Committee for Standardization Brussels: Brussels, Belgium, 2007.
48. Nicol, J.F.; Humphreys, M.A. A stochastic approach to thermal comfort-occupant behavior and energy use in buildings/discussion. *ASHRAE Trans.* **2004**, *110*, 554.
49. Markov, D. Practical evaluation of the thermal comfort parameters. In *Annual International Course: Ventilation and Indoor Climate, Avangard, Sofia*; Technical University of Sofia: Sofia, Bulgaria, 2002; pp. 158–170.
50. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [CrossRef] [PubMed]
51. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. *arXiv* **2015**, arXiv:1509.02971.
52. Silver, D.; Lever, G.; Heess, N.; Degris, T.; Wierstra, D.; Riedmiller, M. Deterministic policy gradient algorithms. In Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 21–26 June 2014; pp. 387–395.
53. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal policy optimization algorithms. *arXiv* **2017**, arXiv:1707.06347.
54. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 1861–1870.
55. Mnih, V.; Badia, A.P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 1928–1937.