POLITECNICO DI TORINO
Repository ISTITUZIONALE

Practical deployment of reinforcement learning for building controls using an imitation learning approach

*Original*
Practical deployment of reinforcement learning for building controls using an imitation learning approach / Silvestri, A., Coraci, D., Brandi, S., Capozzoli, A., Schlueter, A.. - In: ENERGY AND BUILDINGS. - ISSN 0378-7788. - 335:(2025). [10.1016/j.enbuild.2025.115511]

*Availability:*
This version is available at: 11583/2998194 since: 2025-03-30T16:17:32Z

*Publisher:*
Elsevier

*Published*
DOI:10.1016/j.enbuild.2025.115511

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

(Article begins on next page)

16 November 2026

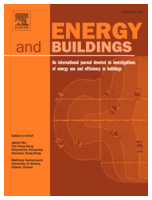
Energy & Buildings 335 (2025) 115511



Contents lists available at ScienceDirect

Energy & Buildings

journal homepage: www.elsevier.com/locate/enbuild



# Practical deployment of reinforcement learning for building controls using an imitation learning approach

Alberto Silvestri [a,*], Davide Coraci [b], Silvio Brandi [b], Alfonso Capozzoli [b], Arno Schlueter [a]

[a] Architecture and Building Systems, ETH Zurich, Stefano-Franscini, Platz 5, Zurich, 8049, Switzerland
[b] Politecnico di Torino, Department of Energy, TEBE research group, BAEDA Lab, Corso Duca degli Abruzzi 24, Torino, 10129, Italy

ARTICLE INFO

*Keywords:*
Imitation learning
Behavioural cloning
Deep reinforcement learning
Building HVAC control
Energy efficiency
Real implementation

ABSTRACT

This paper addresses the critical need for more efficient and adaptive building control systems to maximise occupant comfort while reducing energy consumption. Our objective is to explore the practical application of model-free Deep Reinforcement Learning (DRL) in real-world building environments by developing a system that learns and adapts to changing conditions, beginning its operation by imitating an existing Rule-Based Control (RBC) system. This approach ensures initial reliability and performance while setting the stage for advanced learning capabilities. The methodology involves two distinct phases. Initially, the DRL controller mimics the behaviour of the RBC system, using imitation learning with behavioural cloning as a safe and efficient strategy to achieve baseline operational efficiency. Subsequently, the controller is implemented within a real building in an online learning setting. In this phase, the controller utilises real-time data to continuously refine its control policy, responding adaptively to occupant behaviours and external environmental conditions. To validate our approach, we conducted a comprehensive analysis, comparing the performance of our DRL controller against the baseline RBC controller, another RBC, and a PI (Proportional-Integral) controller implemented in a digital twin model of the real office environment. Energy consumption and temperature violations related to a temperature acceptability range are considered as metrics, providing a robust framework for assessing the effectiveness of our system. The results indicate that our DRL controller, supported by imitation learning, outperforms the two RBCs by reducing energy consumption by 40 % while reducing the cumulative sum of temperature violations by 43 % and 13 % with respect to the two RBCs. Although the PI controller ensures better performance in terms of temperature violations compared to DRL, it requires 45 % more energy than the proposed DRL controller due to its inherent inability to deal with multi-objective control problems. In conclusion, this paper demonstrates the feasibility and advantages of implementing advanced DRL techniques in real-world building control scenarios. Integrating imitation learning with a DRL controller offers a novel and effective way to enhance the scalability of DRL systems, expanding their application in buildings and driving significant improvements in energy efficiency.

## 1. Introduction

Buildings represent a substantial fraction of worldwide energy consumption, accounting for approximately 40 %, and are responsible for about 30 % of global greenhouse gas emissions [1]. In response to this challenge, researchers have focused on enhancing building energy efficiency [2]. Heating, Ventilation and Air Conditioning (HVAC) systems stand out as the most energy-intensive systems in buildings [3]. Consequently, improving their operations through advanced energy management strategies emerges as a viable solution to reduce their energy cost and enhance indoor comfort conditions for occupants in buildings [4]. At present, HVAC systems are mainly operated using Rule-Based Control (RBC) [5]. While these controllers are designed by building control specialists, often based on ASHRAE Guidelines 36 [6], they may exhibit suboptimal performance, as they are reactive systems that struggle with multi-objective control challenges [7,8], lacking the capacity to dynamically modify their control strategies in anticipation of external factors, such as weather changes, that affect energy consumption and indoor comfort [9].

In this scenario, the increased access to historical building data, driven by the widespread integration of Internet of Things (IoT) devices and Information and Communication Technologies (ICT), offers significant advantages [10,11]. This wealth of data enables the design of more advanced control strategies that can predict and estimate building

* Corresponding author.
*E-mail address:* silvestri@arch.ethz.ch (A. Silvestri).

https://doi.org/10.1016/j.enbuild.2025.115511
Received 17 December 2024; Received in revised form 6 February 2025; Accepted 21 February 2025
Available online 26 February 2025
0378-7788/© 2025 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

**Nomenclature**

| | |
|---|---|
| $\alpha$ | Boltzmann temperature coefficient |
| $\beta$ | Temperature term weight of reward function |
| $\dot{Q}_{\text{sol}}$ | Solar radiation [W/m$^2$] |
| $\dot{Q}_{\text{tabs}}$ | Heating power delivered by TABS [kW] |
| $\epsilon$ | Clipping parameter of the surrogate objective function |
| $\gamma$ | Discount factor |
| $\hat{\nabla}_\theta J(\theta)$ | Gradient of the expected return objective function |
| $\mu$ | DRL Learning rate |
| $\overline{T_{\text{i}}}$ | Upper limit of temperature comfort range [°C] |
| $\overline{T_{\text{viol,daily}}}$ | Mean value of the daily average temperature violation rate [°C] |
| $\pi_{\theta_{old}}$ | Previous control policy |
| $\pi_\theta$ | Current control policy |
| $\theta$ | Parameters of the control policy |
| $\underline{T_{\text{i}}}$ | Lower limit of temperature comfort range [°C] |
| $b_{\text{occ}}$ | Occupancy boolean variable |
| $E_{\text{tabs}}$ | Energy consumption associated with the TABS operation [kWh] |
| $f_{occ}$ | Occupancy fraction over each control time step |
| $J(\theta)$ | Expected return objective function |
| $L^{CLIP}(\theta)$ | Clipped surrogate objective function |
| $r$ | Reward function |
| $r(\theta)$ | Probability ratio between the new and the old policy |
| $r_E$ | Energy term of reward function |
| $r_T$ | Temperature term of reward function |
| $T_{\text{i}}$ | Indoor air temperature [°C] |
| $T_{\text{n}}$ | Neighboring room temperature [°C] |
| $T_{\text{o}}$ | Outdoor air temperature [°C] |
| $T_{\text{viol,daily}}$ | Cumulative sum of daily temperature violation [°C] |
| $T_{\text{viol}}$ | Cumulative sum of temperature violation [°C] |
| $u_t$ | Percentage opening of the valve |

**Acronyms**

| | |
|---|---|
| AHUs | Air Handling Units |
| BC | Behavioural Cloning |
| BESS | Battery Energy Storage System |
| DNNs | Deep Neural Networks |
| DRL | Deep Reinforcement Learning |
| FMI | Functional Mock-up Interface |
| FMU | Functional Mock-up Unit |
| HVAC | Heating, Ventilation and Air Conditioning |
| HVRF | Hybrid Variable Refrigerant Flow |
| ICT | Information and Communication Technologies |
| IL | Imitation Learning |
| IoT | Internet of Things |
| KL | Kullback–Leibler |
| MAPE | Mean Absolute Percentage Error |
| MDP | Markov Decision Process |
| MPC | Model Predictive Control |
| ODBC | Open DataBase Connectivity |
| PI | Proportional-Integral |
| PIRs | Passive Infrared Sensors |
| PPO | Proximal Policy Optimisation |
| PV | Photovoltaic |
| RBC | Rule-Based Control |
| RC | Resistance-Capacitance |
| RL | Reinforcement Learning |
| RMSE | Root Mean Squared Error |
| TABS | Thermally Activated Building System |
| TES | Thermal Energy Storage |
| TL | Transfer Learning |
| TRPO | Trust Region Policy Optimisation |
| VAV | Variable Air Volume |

conditions and energy system behaviours, both in real-time and in the future, thus addressing the shortcomings of conventional HVAC control techniques [12].

These advanced control strategies can be divided into two main categories: model-based and model-free methods. Model-based approaches involve the use of a detailed model that represents the system being controlled. The model is then used in an optimal control problem to find the optimal control policy. On the other hand, model-free methods do not require an explicit model of the environment, as they can find a near-optimal control policy through a trial-and-error learning process.

Model Predictive Control (MPC) has become one of the most extensively studied model-based control approaches in recent literature, particularly for HVAC systems, due to its capability to predict future system dynamics and adjust to varying boundary conditions [13,14]. The optimisation features of MPC have gathered significant attention within the building sector [15], as it effectively balances indoor comfort with energy efficiency, reducing energy consumption while enhancing occupant conditions [16,17].

Despite its benefits, the widespread implementation of MPC is still limited, primarily due to its model-dependent nature. Adopting MPC necessitates a comprehensive and precise model of the specific building and energy systems involved, which requires considerable time and effort to develop [4,18].

To address these limitations, researchers have explored implementing model-free control strategies, such as Reinforcement Learning (RL) [19]. RL employs a trial-and-error interaction with the controlled environment to learn a near-optimal control policy $\pi^*$, which maps the relationship between states and actions to maximise the cumulative sum of future rewards [20].

Within RL domain, Q-learning is recognised as an effective algorithm for addressing Markov Decision Process (MDP) without requiring prior knowledge of the environment. As the most commonly used method in the RL family, it operates within a tabular framework to estimate state-action values, also known as Q-values, derived from interaction with the environment [21]. However, when Q-learning is applied to environments with large state and action spaces, typical of real-world scenarios, it faces challenges related to scalability and memory efficiency [22]. To overcome these limitations, Deep Reinforcement Learning (DRL) provides a promising alternative by leveraging neural networks to approximate Q-values. Given the complexity and non-linearity inherent in building control problems, Deep Neural Networks (DNNs) are frequently employed in RL algorithms, thereby leading to a DRL approach [23].

DRL has been applied to enhance energy management in buildings, optimising indoor temperature and comfort conditions for occupants while optimising the operation of energy systems according to grid requirements. The analysis of the current state-of-the-art reveals successful applications of RL in simulated environments in the management of supply water temperature [24,25] and mass flow rate in generation systems [26,27], management of supply air temperature and mass flow rate in Air Handling Units (AHUs) [28], optimisation of the indoor temperature setpoint [29], and of the operation modes of generation systems [30], management of the charging and discharging process for Thermal Energy Storage (TES) [31] and Battery Energy Storage System (BESS) [32].

Training a RL agent to interact with a controlled environment and derive an optimal control policy can be accomplished through two primary approaches: online DRL and offline DRL. The online method reflects an ideal scenario where a model-free DRL agent is implemented directly within a real building [11]. In this context, the agent progressively learns the optimal control policy via trial and error while managing the system in real-time, without any prior offline training [30].

However, this approach tends to be inefficient, as it necessitates numerous interactions to reach convergence and may involve navigating extreme conditions within the controlled environment, potentially resulting in suboptimal performance, especially during the early

stages of deployment [33]. Consequently, the direct application of a DRL agent in a real building is often impractical due to economic and safety concerns.

To address these issues, a common strategy in the literature is to conduct offline pre-training of DRL controllers using surrogate models of building and energy systems before deployment. This approach enables the DRL controller to be trained beforehand, significantly reducing the risks associated with real-world implementation. Researchers frequently employ detailed engineering models created with energy modelling software (i.e. EnergyPlus [34]) and programming language, such as Modelica [35].

While the offline training method for DRL agents has produced impressive results, it faces considerable challenges in terms of scalability and generalisability. The distinctive characteristics of each building necessitate the development of either data-driven or physics-based surrogate models. Developing data-driven surrogate models requires a minimum threshold of monitoring data from the controlled building. In contrast, developing physics-based models can be a labour-intensive process, as it requires access to detailed building information—which may not always be readily available—and specialised expertise in the relevant domain [36].

To overcome these practical challenges, knowledge-reuse strategies like Transfer Learning (TL) and Imitation Learning (IL) present promising solutions to improve the scalability of DRL controllers, thereby facilitating their application in real-world buildings.

Transfer learning is a machine learning approach where a model trained on a specific task (the source task) in a particular domain is adapted to tackle a new, related task (the target task) [37]. This new task may share characteristics with the original task, either within the same domain or across different domains [38]. Imitation learning, on the other hand, aims to replicate the behaviour of an expert agent that performs well on a specific task. This approach fundamentally involves learning to associate observations with actions, which simplifies the teaching process by demonstrating the actions necessary to achieve a given objective [39]. Among the various IL strategies, Behavioural Cloning (BC) stands out as a straightforward yet powerful form of IL, particularly when applied to DRL. BC is performed offline and consists of pre-training a policy network on a dataset of expert demonstrations. This phase uses conventional supervised learning techniques to replicate the expert's actions given the observed states. The objective is to minimise the difference between the agent's and expert's actions, often measured by loss functions like mean squared error or cross-entropy loss. Next, the agent interacts with the environment and is further trained using the standard reinforcement learning workflow. This online training allows the agent to imitate the expert initially and successively improve the control policy by exploring and optimising the reward function.

The next section reviews literature studies that focus on the application of TL and IL in the framework of DRL controllers applications for building energy management purposes. Afterwards, the research gaps and contributions of this paper are discussed.

### 1.1. Related works on knowledge-sharing approaches for DRL controllers

The implementation of TL and IL strategies for smart building control has recently gained attention, as its applications in this field are relatively new compared to other areas within machine learning. Implementing these knowledge-reusing strategies for DRL controllers could enable the direct deployment of controllers in real-world buildings with strong initial performance.

Lissa et al. [40] introduced an intra-transfer learning method called parallel transfer learning, which facilitates knowledge sharing among five different agents during their training process, eliminating the need to wait until training is complete. This transfer approach was applied in a microgrid consisting of five homes, each equipped with its energy system that included a Photovoltaic (PV) system and a heat pump. Each home had its own DRL controller responsible for managing the heat pump to minimise energy costs. As a result, training time was reduced by a factor of five, and energy savings of 10 % were achieved compared to the scenario without transfer. Coraci et al. [41,36] proposed an online TL approach that utilised two knowledge-sharing techniques: weight initialisation and imitation learning. Their studies evaluated the performance of the transferred DRL controller in terms of electricity cost savings and reduction of indoor temperature violations. The online TL method was designed to mimic real-world implementation. The DRL controller managed a cooling system consisting of a chiller and cold thermal storage and was tested across various target buildings to evaluate the effectiveness of the transfer strategy. Performance was benchmarked against a RBC and two DRL-based control policies: one deployed online without pre-training and another deployed following an offline pre-training phase. In both cases, source and target buildings shared the same spatial layout but differed in weather conditions, electricity pricing, occupancy schedules, and building envelope characteristics. Results in [41] showed that the online TL approach improved indoor temperature conditions by 50 % and 80 % compared to the RBC and the non-pre-trained DRL agent, respectively. Similar findings were reported in [36], where the online TL approach was evaluated for source and target buildings that differed in their energy systems, with the target buildings equipped with PV systems and BESS. Nweye et al. [42] presented a TL approach that integrates IL to replicate the behavior of a RBC using five months of measured data. This process involved weight initialisation to pre-train the DRL controllers before deployment in an energy community where buildings were equipped with appliances, PV panels, and BESS. The DRL controllers independently managed each BESS to minimise electricity costs and carbon emissions from grid electricity. The results showed that transferring the DRL control policy between buildings within the energy community achieved similar performance while significantly reducing training time.

Imitation learning applications have been explored more recently in the framework of energy management in buildings. Hou et al. [43] developed an integrated TL approach that constructs an optimal source domain dataset from multiple source buildings. Parameters from this optimal multi-source dataset are then transferred to the target DRL controller, significantly reducing training time and improving performance by 20 % compared to training a DRL model from scratch. Additionally, this method reduces the average temperature deviation by up to 14 %. Liu et al. [44] introduced a generative adversarial IL approach that leverages expert demonstrations from a MPC to enhance the performance of a DRL controller managing a Variable Air Volume (VAV) system for cost reduction and load shifting in a commercial building. This approach achieved better performance than a RBC by increasing the cumulative reward by 22 % and by 7 % compared to a DRL controller without imitation learning. Dey et al. [39] implemented an IL strategy for a DRL agent controlling the indoor temperature setpoints of a five-zone office building to reduce energy consumption and minimise thermal discomfort. The implementation of IL allowed to reduce by 6 % the average cost and to improve the average score by 7 % during the testing phase compared to a rule-based heuristic policy. Amasyali et al. [45] employed a student-teacher distillation process to transfer knowledge from ten pre-trained DRL controllers used for managing cooling energy supply to a target building. The approach achieved a cumulative reward comparable to that of a DRL controller pre-trained over ten episodes, while significantly outperforming both a DRL agent without pre-training and a fixed setpoint controller. To conclude, Kadamala et al. [46] proposed a hybrid approach combining imitation learning for pre-training with RL for fine-tuning. The authors utilised a behavioural cloning technique to select heating and cooling setpoints in a five-zone residential setup. The results showed a 10 % improvement in total reward compared to a DRL agent trained from scratch, indicating enhanced overall performance.

### 1.2. Research gaps and novelties of the proposed contribution

Despite promising advancements, most DRL applications for controlling energy systems in buildings are still predominantly confined to simulated environments. Real-world deployment of DRL controllers poses significant challenges due to economic constraints and safety concerns linked to the trial-and-error learning process of model-free approaches [11]. Theoretically, a DRL agent could be trained directly in a real building, progressively improving its control policy through continuous interaction with the environment [30]. However, this results in extended times to converge towards a near-optimal control policy, which conflicts with occupant comfort requirements and may lead to suboptimal performance, especially during initial deployment phases [33].

A common approach researchers explore to overcome this limitation consists of pre-training the DRL controllers offline using detailed engineering models. However, creating engineering models of the buildings is a time-consuming activity that requires domain expertise and comprehensive technical data (e.g., thermophysical properties, energy systems details) of the buildings[41]. These requirements limit the scalability of DRL-based solutions, as each deployment phase in different buildings requires the development of a customised surrogate model.

Based on these limitations, the present paper proposed the development of a model-free approach based on imitation learning and its further real-world implementation in a real testbed to enhance the scalability of DRL controllers in buildings.

To the best of the authors' knowledge, imitation learning strategies have never been applied in real-world settings, as previous studies have confined their use to simulated applications. This paper addresses this gap by proposing a model-free approach based on imitation learning for the real-world deployment of DRL controllers in buildings.

Specifically, a DRL controller using the Proximal Policy Optimisation (PPO) algorithm has been implemented in a real office building located in Dübendorf, Switzerland. Theoretical aspects regarding PPO are provided in Section 2, while general foundations about DRL controllers can be found in [20,47]. The developed methodology employs an imitation learning framework in which the controller's policy is pre-trained to replicate an existing RBC strategy using historical data collected from the building. The controller regulates the water flow supplied to a Thermally Activated Building System (TABS) by adjusting the valve opening percentage, to reduce energy consumption while maintaining optimal indoor temperature conditions.

The main contributions of this paper are as follows:

- A behavioural cloning-based imitation learning technique is employed to pre-train through a supervised learning approach the Actor network of the PPO controller, using data collected during the operation in the real building of a RBC strategy. This approach allows the controller to perform effectively from the early stages of deployment, reducing the duration of the learning process and improving the scalability of DRL controllers across different buildings without requiring complex engineering models.
- While prior research has explored transfer learning and fine-tuning to improve DRL controllers in simulated environments, this paper pioneers the practical application of knowledge reuse through imitation learning in a real-world setting. By pre-training the DRL controller using behavioural cloning, this approach overcomes the limitations of extended fine-tuning over multiple heating/cooling seasons, bridging the gap between simulation-based research and practical DRL applications in energy management.
- Unlike conventional DRL applications that focus mainly on simulation, this study demonstrates the real-world deployment of the PPO controller with an online learning mechanism, enabling continuous adaptation of the control policy. This facilitates dynamic responses to occupant behaviours and changing environmental conditions, thereby ensuring sustained energy efficiency and comfort.
- A detailed thermal Resistance-Capacitance (RC) model of the building was developed in Modelica, serving as a digital twin that closely mimics the actual building's behaviour. This model was fine-tuned with real-world data to provide a robust comparison of the PPO controllers performance against traditional RBC and Proportional-Integral (PI) strategies, evaluating energy savings and indoor temperature control performance.

The structure of the paper is as follows: Section 2 describes the methods applied in the study, and Section 3 outlines the framework used for implementing the imitation learning process and deploying the DRL controller in the real building. Section 4 provides details on the implementation process. The results are presented in Section 5, with Section 6 exploring the implications of the findings. Lastly, Section 7 offers suggestions for future research directions.

## 2. Methods

### 2.1. Proximal policy optimisation

PPO is a DRL-based algorithm derived from the policy gradient method Trust Region Policy Optimisation (TRPO) [48]. As a member of the policy gradient methods family, PPO optimises the policy function directly to maximise cumulative rewards. Being an on-policy algorithm, PPO relies solely on data collected from the current policy. On-policy methods generally offer greater stability and are less sensitive to hyperparameter adjustments.

PPO combines both value-based and policy-based approaches, demonstrating good performance across a wide range of tasks [49], as well as stability and robustness to variations in hyperparameters and network architectures [50]. Policy-based methods focus on learning a control policy $\pi_\theta(a|s)$ that maximises the expected return $J(\theta)$, typically expressed as $\mathbb{E}_{\pi_\theta}(G_t)$, where $\theta$ denotes the policy parameters. A PPO agent estimates the gradient $\hat{\nabla}_\theta J(\theta)$ stochastically from previously collected experience trajectories and applies a gradient ascent update during training, as defined below:

$$\theta_{t+1} = \theta_t + \mu \cdot \hat{\nabla}_\theta J(\theta) \tag{1}$$

where $\theta_t$ and $\theta_{t+1}$ represents the parameters of the policy at time $t$ and $t+1$, $\mu$ is the learning rate and $\hat{\nabla}_\theta J(\theta)$ is the gradient of the expected return objective $J(\theta)$.

PPO seeks to optimise the expected return of the current policy while constraining how much it can differ from the previous policy. This is achieved through a surrogate objective function that balances the expected return of the new policy with a restriction on its deviation from the old one. Instead of using a Kullback–Leibler (KL) divergence constraint as in TRPO [51], PPO adopts a simpler approach by applying a clipped surrogate objective, $L^{CLIP}(\theta)$ which directly limits how far the new policy can deviate from the previous policy [39]:

$$L^{CLIP}(\theta) = \mathbb{E}\left[min\left(r(\theta)\cdot \hat{A}_\theta(s,a), clip(r(\theta), 1-\epsilon, 1+\epsilon)\cdot \hat{A}_\theta(s,a)\right)\right] \tag{2}$$

where $\hat{A}_\theta$ represents the advantage function , corresponding to the difference between the expected return of a specific action and the value of the state, $\epsilon$ is the clipping parameter tht limits how the new policy is alloweD to deviate from the old policy during training. Moreover, the term $clip(r(\theta), 1-\epsilon, 1+\epsilon)$ is a clipping mechanism that limits the magnitude of the probability ratio $r(\theta)$, thereby preventing excessive updates to the policy. $r(\theta)$ refers to the ratio of probabilities of taking action $a$ in state $s$ respectively under the new policy $\pi_\theta$ and the old policy $\pi_{\theta_{old}}$, defined as follows:

$$r(\theta) = \frac{\pi_\theta(s|a)}{\pi_{\theta_{old}}(s|a)} \tag{3}$$

PPO utilises an actor–critic architecture, where the Actor is responsible for maximising the clipped objective along with an entropy term,

while the Critic minimises the value function loss to enhance value estimates. These value estimates are then used to compute the advantage function $\hat{A}_\theta$ [52].

### 2.2. Imitation learning

IL is a strategy that allows an agent to learn to perform tasks by mimicking expert behaviour rather than through trial-and-error interactions with the environment [53,54]. The agent learns a control policy that approximates the expert's performance without requiring extensive exploration or interaction with the environment. In IL framework, an agent is defined as an entity that operates independently within an environment to accomplish a specific objective [55]. IL differs from RL, as it does not involve the agent learning solely from its own experiences. Instead, IL utilises the actions demonstrated by a teacher to inform the agent's behaviour [56]. In this process, the target agent accesses trajectories provided by the expert, which encompass a series of states or state-action pairs [57]. Additionally, the agent can save these observed transitions in a buffer for later use. The IL paradigm is based on supervised learning, where the agent learns a mapping from states to actions using a dataset of expert demonstrations. This approach is particularly beneficial in scenarios where the agent is required to rapidly acquire competence by leveraging the knowledge of an expert.

Gavenski et al. [56] provide a recent classification of IL methods, dividing them into behaviour cloning, dynamics model methods, adversarial methods, hybrid methods, and online methods.

- **Behavioural cloning** represents the foundational approach in imitation learning, employing a methodology akin to supervised learning [58] to guide the agent in replicating the trajectory demonstrated by the teacher. In this framework, the agent learns to predict the most probable action corresponding to a given state using the dataset derived from the teacher agents implementation. By utilising state-action pairs, the agent acquires the ability to emulate the teacher based on the data observed. However, this technique can become resource-intensive in more intricate scenarios, as it necessitates numerous samples and a comprehensive understanding of how actions influence the environment [56].
- **Dynamics model methods**, also known as model-based imitation learning, refers to a learning strategy that derives teacher behaviour not from direct action information but through interactions with dynamic models. The development of dynamics models typically involves some level of online interaction, wherein the agent actively engages with the environment while focusing on a particular task. These models can be classified into two categories: *inverse* dynamics models, which estimate the probability of each action based on state transitions without requiring labelled state-action pairs, and *forward* dynamics models, which capture the environment's dynamics by predicting subsequent states based on previously experienced conditions.
- **Adversial methods** involve the creation of an artificial reward function that incentivises the agent by rewarding behaviours that closely resemble those of the teacher. These methods fall under the category of model-free approaches, as they do not model the dynamics of the environment; instead, the policy is developed through a trial-and-error process during interactions with the environment.
- **Hybrid methods** produce more robust policies by integrating dynamics models with adversarial methods. This combination provides a temporal signal that narrows the gap between the teacher and the student while incorporating intrinsic knowledge about the dynamics of the environment [59].
- **Online methods** enable the agent to learn the teacher's behaviour by accessing information in real-time rather than relying on pre-recorded demonstrations. This approach is particularly beneficial when real-time feedback from the teacher is accessible or when the agent must adapt to rapidly changing conditions. However, online imitation learning can present challenges in designing a reward function or modelling a policy function, especially if these aspects are complex to define.

## 3. Methodology

This section delineates the methodological framework employed in this study, comprising three primary stages, as shown in Fig. 1. The first step involves collecting data from the baseline controller, which is used to pre-train the control policy with BC. Next, the DRL controller is deployed in the real building, improving its policy while controlling the system. Finally, a digital twin of the system is used to compare the performance of the considered controllers.

**Imitation learning**. The first step of the methodological framework involves generating the dataset used for imitation learning, followed by the application of behavioural cloning. In this stage, the baseline controller is implemented for 15 days to manage the TABS system in a real building, collecting data on the controller's actions and the corresponding system states. This dataset is then used to train a neural network to approximate the RBC control policy by employing a standard supervised learning approach, where the network learns to match the baseline controller's actions based on the observed states. The performance of the DRL controllers is influenced by various hyperparameters that require careful tuning. Consequently, the most critical hyperparameter values are optimised through an automated procedure.

**Real-world implementation**. The second step of the methodological framework aims to implement the DRL controller in the real testbed. This implementation phase lasted approximately 14 days and included the implementation and testing processes of the control strategy. The real-world implementation is carried out in alignment with the infrastructure of the real testbed, ensuring efficient communication and control of the building's HVAC systems. Initially, the DRL control policy mimics the RBC strategy, but being implemented in an online learning setting, it improves its policy while interacting with the environment to be controlled.

**Performance benchmarking**. In the final step of the methodological process, the performance of the DRL controller implemented in the real office is compared to that of the baseline controller, as well as a different RBC and a PI controller, in terms of energy consumption and indoor temperature control. This comparison is conducted using a digital twin developed in Modelica. The operation of the baseline controller, the alternative RBC, and the PI controller are all simulated under the same real-world conditions in which the DRL controller was tested.

## 4. Implementation

### 4.1. Case study description

Similarly to [21], the proposed imitation learning approach was implemented on the HiLo (High Performance – Low Emissions) unit, a module included in the NEST building [60], represented in Fig. 2(a). The NEST building is a modular research and innovation facility [61] located in Dübendorf (Switzerland) and part of the Swiss Federal Laboratories for Materials Science and Technology (EMPA). Featuring a central backbone and three open platforms, the building accommodates various research and innovation modules. This research environment represents an ideal case study for our controller. HiLo is a living lab with a large availability of sensors and data sources to monitor physical variables, with the collected data systematically archived in a dedicated database.

The HiLo unit is occupied from Monday to Saturday from 7:00 to 21:00, and it includes two floors. The lower level accommodates two office spaces, while the upper floor features an open-space area. For our investigation, we selected the office shown in Fig. 2(b) as a case study, situated on the southwest side and covering an area of 22.94 $m^2$. This office is equipped with three HVAC systems: a ceiling-mounted integrated TABS, a Hybrid Variable Refrigerant Flow (HVRF) system, and

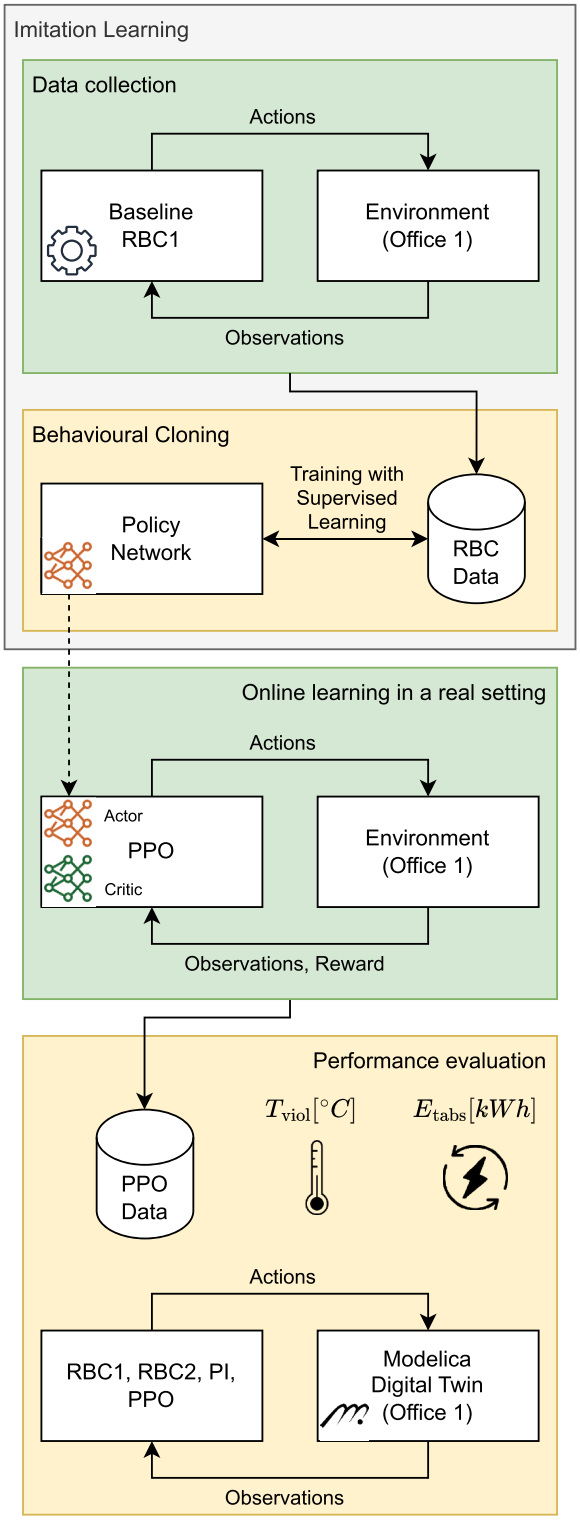



**Fig. 1.** Methodological framework adopted in this paper.

a conventional ventilation system. Due to constraints regarding access to low-level controls, our study exclusively focused on the TABS system to manage the water flow rate of the TABS every 5 minutes (i.e., control time step $k$) by determining the percentage of valve opening $u_{valve} \in [0, 1]$. The valve is of the changeover type, and the inlet temperature is maintained at a constant value, heated by hot water supplied by the backbone.

**Table 1**
Time and indoor temperature conditions for the RBC preheating phase.

| Combination | Time condition | Temperature condition |
|---|---|---|
| 1 | $3{:}00 \le t < 4{:}00$ | $\underline{T_i} - T_i \ge 1\,°C$ |
| 2 | $4{:}00 \le t < 5{:}00$ | $\underline{T_i} - T_i \ge 0.5\,°C$ |
| 3 | $t \ge 5{:}00$ | $\underline{T_i} - T_i \ge 0\,°C$ |

### 4.2. Baseline controller

The baseline controller relies on the bang-bang control principle, toggling between fully closing (i.e., $u_{valve} = 0$) or opening (i.e., $u_{valve} = 1$) the valve [21]. This RBC controller (called RBC1) consists of a pre-heating phase and a standard heating phase, which starts at $t = 7{:}00$. The pre-heating phase starts at $t = 3{:}00$ and activates the TABS based on indoor air temperature readings and the time of day, as outlined in Table 1. Following the pre-heating phase, the controller manages the TABS to open the valve when the indoor temperature falls below the lower acceptable temperature threshold $\underline{T_i}$ (i.e., 21 °C) and closes it when the temperature exceeds the upper threshold $\overline{T_i}$ (i.e., 23 °C). This strategy is active until occupants leave the building at $t = 21{:}00$, ensuring the valve remains fully closed on Sundays to save energy since the office is not occupied.

Another version of this controller, referred to as RBC2, has been considered in this work. The only difference from the RBC1 control logic is that RBC2 employs a narrower temperature range, from 21 °C to 22 °C.

### 4.3. PI controller

In this work, a PI controller was also implemented as a benchmark to compare its performance with that of the DRL controller. The PI controller adjusts the control input based on the error between the measured indoor temperature $T_i$ and the desired setpoint $T_{sp}$ of 22 °C, using both proportional and integral actions.

The control law of the PI controller is given by the following equation:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)\, d\tau \tag{4}$$

where: $u(t)$ is the control output (i.e., the valve opening percentage), $K_p$ is the proportional gain, $K_i$ is the integral gain, $e(t) = T_{sp} - T_i$ is the error at time $t$, defined as the difference between the setpoint temperature $T_{sp} = 22$ °C and the measured indoor temperature $T_i$.

The proportional term $K_p e(t)$ responds to the current error by adjusting the control output proportionally to the magnitude of the error. The integral term $K_i \int_0^t e(\tau)\, d\tau$ takes into account the accumulated error over time, helping to eliminate any steady-state offsets by integrating past errors.

The tuning of the proportional and integral gains, $K_p$ and $K_i$, was carried out to ensure a balance between fast response and setpoint tracking without excessive oscillations.

### 4.4. Imitation learning process

The state and action pairs data collected by the baseline RBC has been stored in a dataset in the form of tuples. The dataset has been divided into training and testing sets, with a proportion of 80 % and 20 %, respectively. Next, an optimisation with Optuna [62] has been performed to find the best hyperparameters for the neural network representing the policy. The network was trained on the training dataset using an Adam Optimiser and evaluated on the testing data using the coefficient of determination $R^2$ metric. Table 2 shows the hyperparameters considered in the optimisation, with their ranges and final optimal value.

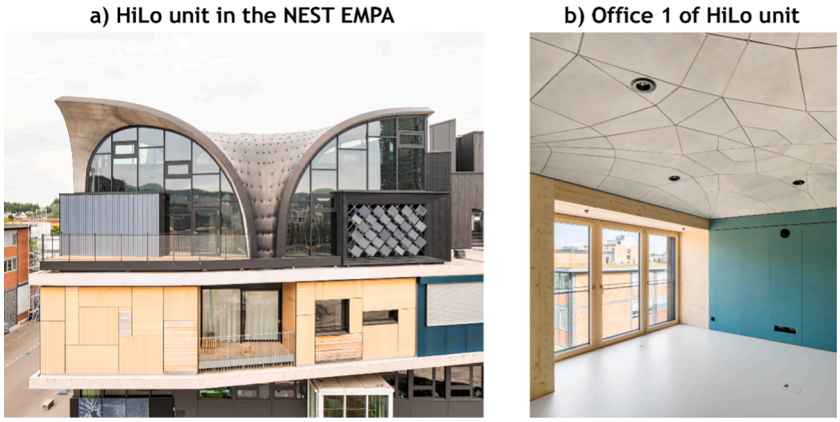



**Fig. 2.** (a) HiLo unit in the NEST building of EMPA and (b) Office 1 of HiLo employed as a case study in this paper.

**Table 2**
Values and range of hyperparameters optimised with Optuna during the imitation learning process.

| Hyperparameter | Value | Step | Best value |
|---|---|---|---|
| Learning rate | $[10^{-6}, 10^{-2}]$ | $10^{-6}$ | $6 \cdot 10^{-4}$ |
| Number of hidden layers | [1, 4] | 1 | 3 |
| Number of neurons per layer | [32, 128] | 32 | 64 |

### 4.5. Design of DRL controller

The implementation of DRL controllers requires defining their main features, such as action space, state space, and reward function. In this work, the action-space $A$ is continuous and defined as $A_k$ at each control time step $k$, equal to 5 minutes. At each step, the agent chooses the valve opening percentage, which is linearly correlated to the fraction of the nominal heating power (i.e., 0.9 kW) supplied by the TABS.

$$A : 0 \leq u_k \leq 1 \tag{5}$$

In this study, the state-space of the DRL agent includes the 23 features detailed in Table 3, along with their respective lower and upper bounds used for rescaling the state space via min-max normalisation. *Outdoor Air Temperature* $T_o$ and *Global solar radiation* $\dot{Q}_{sol}$ are included in the state-space due to their significant impact on building heating energy consumption and indoor temperature. The 6-hour time horizon for $T_o$ ensures that the DRL agent receives sufficient information to anticipate future changes in outdoor conditions and providing an effective control strategy while maintaining a manageable state-space complexity. Information about *Indoor air temperature* $T_i$ is included by defining the temperature difference relative to the two temperature limits (i.e., $T_i - \overline{T_i}$ and $T_i - \underline{T_i}$) specified by the acceptable temperature range, ensuring an adaptive definition. By combining these two variables, the DRL agent gains knowledge of the indoor temperature in relation to the temperature acceptability range. The features outlined so far are integrated into the state-space at the current time step $k$ and over the previous 15 and 30 minutes, alongside the information on the power supply in the environment by the TABS $\dot{Q}_{tabs}$. This approach enhances the DRL controller's capability to understand the dynamics of the environment. Moreover, hourly predictions for the outdoor temperature have been provided for the next 6 hours. To conclude, the information about occupant presence is provided by three different variables: the occupancy fraction over each 5-min control time step $f_{occ}$, time to occupancy start and time to occupancy end. The $f_{occ}$ variable was integrated to notify the DRL agent about the presence of occupants and, thereby, the possible occurrence of additional indoor gains that may reduce the energy demand for TABS. The other two variables indicate the time remaining until the following modification in the occupancy pattern. When the building is unoccupied, time to occupancy start indicates the number of time steps left before occupants' arrival time (equal to zero when the building is occupied), while during occupied periods, time to occupancy end denotes the number of time steps until occupants' departure time (equal to zero during off-occupancy periods).

Moreover, Table 3 indicates in the *Data origin* column whether each variable was directly measured or calculated. In detail, $T_o$ and $\dot{Q}_{sol}$ were directly measured using sensors installed on-site to measure outdoor environmental conditions, while outdoor air temperature forecast was retrieved from an external service (i.e. Solcast). $\dot{Q}_{tabs}$ was directly measured as it was provided by the controlled valve, which provided the thermal power supplied to the office, while $f_{occ}$ was measured using a sensor that provided real-time occupancy information for each 5-minute timestep $k$. Otherwise, the temperature differences from indoor temperature (i.e., $T_i - \overline{T_i}$ and $T_i - \underline{T_i}$) were calculated using the measured indoor temperature $T_i$ obtained from sensors and the predefined lower and upper bounds of the temperature acceptability range. To conclude, time to occupancy start and time to occupancy end were calculated by combining the known occupancy schedule of the building with the current time of day.

The reward function is defined as the weighted combination of the energy consumption associated with the TABS operation Etabs and comfort violations, which quantify the squared deviation of the zone temperature from the desired temperature limits. The reward function r is defined as follows:

$$r = -(\lambda \cdot E_{tabs} + b_{occ} \cdot [\max(0, T_i - \overline{T_i})^2 + \max(0, \underline{T_i} - T_i)^2] \tag{6}$$

where $b_{occ}$ is a Boolean variable equal to 1 during working hours and 0 otherwise, while $\lambda$ is a weighting factor set to 1 to equally penalise a 1 kWh energy consumption or a 1 °CC deviation from the temperature acceptability range. In this paper, we implemented the PPO from Stable Baselines 3 [63] with hyperparameters chosen based on experience and according to the practical implementation framework of the algorithm. In detail, the number of hidden layers (i.e., 3) and the number of neurons per hidden layer (i.e., 64) are equal to those found during the optimisation carried out by means of Optuna during the imitation learning process, while the values of the other PPO hyperparameters are reported in Table 4.

### 4.6. Simulation environment and digital twin development

A co-simulation environment was established to integrate Modelica [35] with a Python interface utilising the OpenAI Gym framework [64]. This setup aimed to evaluate the performance of various controllers implemented within the developed digital twin during a benchmarking

**Table 3**
Variables included in the state-space.

| Variable | Min value | Max value | Unit | Timestep | Data origin |
|---|---|---|---|---|---|
| $T_o$ | 261.15 | 293.15 | K | $k-30$ min, $k-15$ min, $k$, $k+1$ h, …, $k+6$ h | Measured |
| $\dot{Q}_{sol}$ | 0 | 800 | W/m$^2$ | $k-30$ min, $k-15$ min, $k$ | Measured |
| $\dot{Q}_{tabs}$ | 0 | 0.9 | kW | $k-30$ min, $k-15$ min | Measured |
| $T_i - \overline{T_i}$ | −5 | 5 | °C | $k-30$ min, $k-15$ min, $k$ | Calculated |
| $T_i - \underline{T_i}$ | −5 | 5 | °C | $k-30$ min, $k-15$ min, $k$ | Calculated |
| Time to occupancy start | 0 | 407 | – | $k$ | Calculated |
| Time to occupancy end | 0 | 169 | – | $k$ | Calculated |
| $f_{occ}$ | 0 | 1 | – | $k$ | Measured |

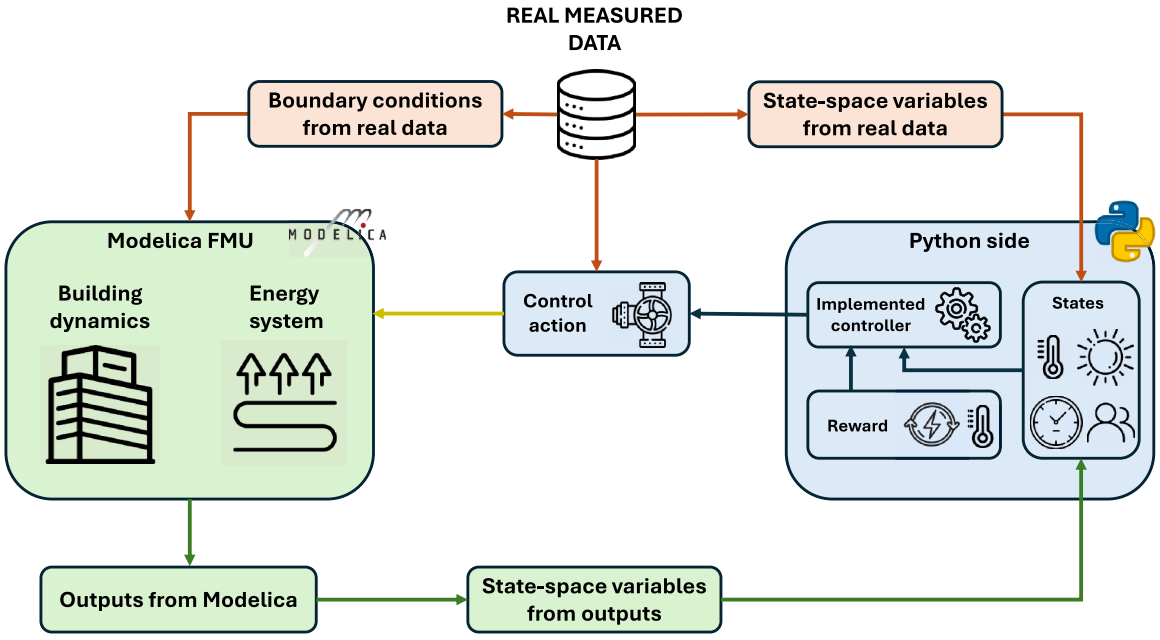



**Fig. 3.** Simulation environment architecture enabling the interaction between Modelica and Python.

**Table 4**
Values of PPO agent hyperparameters. Note that *# Optimisation epochs* in this table refers to the number of passes over each mini-batch during a single PPO update step.

| Hyperparameters | Value |
|---|---|
| Discount factor | 0.95 |
| Batch size | 64 |
| Train frequency | 576 |
| GAE lambda | 0.85 |
| Clip range | 0.1 |
| # Hidden layers | 3 |
| # Neurons per hidden layer | 64 |
| Learning rate $\mu$ | $10^{-5}$ |
| # Optimisation epochs | 2 |

phase. Fig. 3 displays the configuration of the simulation environment. The building and energy systems were modelled in Modelica on the OpenModelica platform [65], employing version 10.0.0 of the Buildings library [66]. Python managed the co-simulation process by utilising the Functional Mock-up Interface (FMI) 2.0 standard [67] and the *pyfmi* package [68].

The FMI standard facilitates the standardised packaging and exchange of simulation models through Functional Mock-up Unit (FMU)s. Python was employed to handle the loading, execution, and real-time interaction of the FMU, which encompasses components related to the building and HVAC systems.

At each simulation step, the Modelica-based building model receives control actions from the Python interface as input. However, for benchmarking purposes and to ensure a fair comparison between PPO and the RBCs and PI controllers, the performance of PPO was evaluated in the digital twin. This approach ensured that any modelling errors associated with the development of the digital twin were consistently considered for the controllers evaluated during the benchmarking phase. Therefore, when PPO was tested in the digital twin the control actions were not generated dynamically by the PPO controller developed in Python. Instead, the control actions chosen by PPO during its real-world operation were collected and provided directly to the Modelica-based digital twin model as input.

Additionally, real data—including outdoor temperature conditions, solar radiation, and occupancy information—was supplied as inputs to the Modelica FMU. These inputs are also integrated into the state space of the controller, as discussed in the next subsection, which focuses on the design of the DRL-based controller. Key outputs from the Modelica FMU include critical data such as the energy consumption of the TABS, indoor environmental conditions, and other relevant information necessary for defining the state space of the implemented controller (e.g., time of day). The interaction between the FMU generated in Modelica and Python is dynamic, occurring at each simulation time step. In this paper, the simulation time step is set to 5 minutes, which is in line with the control time step utilised in the real building. To ensure an adequate representation of the system dynamics, the Modelica model is simulated with a time resolution of 1 minute. At this resolution, the solver exchanges information with the FMU at each simulation step, effectively

capturing the thermal and hydraulic dynamics of the system. However, interactions with the controller are sampled every 5 minutes, which corresponds to the control time step adopted in the study. This approach allows the dynamics of the system to be resolved with high fidelity while maintaining computational efficiency in the decision-making processes of the controllers. While a finer control time step might yield slightly different results, it would also increase computational demands without providing a significant improvement in the accuracy of the evaluated scenarios. This sampling strategy ensures a balance between a detailed representation of the system and practical computational feasibility.

A detailed white-box model of the controlled system was developed to train the DRL controller and evaluate the performance of various controllers following real-world deployment. Built in Modelica [35] using the Buildings library [66], the model captured both the dynamic behaviour of the building and and a detailed representation of the HVAC system.

The facade was modelled using lumped parameter elements: a submodel computed the mass and energy balance for the zone air volume, while the opaque facade components—such as the ceiling, floor, and vertical partitions separating the zone from adjacent spaces or the external environment—were represented as thermal resistance and capacitance layers corresponding to construction materials. The model also included thermal bridges and a comprehensive representation of thermal gains from transparent envelope elements.

The TABS model incorporated its water mass content, the thermal resistance and capacity of the concrete layers embedding the pipes, and the pipes pressure drop, providing a realistic representation of heat exchange with the zone.

Parameter values for the model were initially derived from building descriptions and drawings. Certain parameters were then selected for calibration to align simulated output profiles with measured data. The calibration focused on the zone air temperature and the return water temperature of the TABS, ensuring the model accurately reflected both zone and TABS thermal dynamics. Internal gains from occupancy were estimated by combining monitored occupancy data with suitable convective and radiant heat gain values per person, based on typical office activities. Non-HVAC appliances were modelled using realistic schedules inferred from occupancy patterns.

The calibration process targeted parameters such as the thermal capacities of the TABS and internal air volume, as well as the thermal resistance representing facade thermal bridges. The internal air volume was modelled in the emulator as a lumped object governed by mass and energy balance equations. While the air mass was fixed based on geometric calculations, a multiplier was introduced for thermal capacity during calibration. These parameters, which strongly influenced the dynamic behaviour of the building, were refined iteratively. Fine-tuning improved the alignment of simulated outputs with sensor measurements by adjusting the time constants of both the building and TABS.

Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) were used to assess indoor temperature accuracy and energy consumption, respectively. Since energy consumption was closely linked to return temperature, the measured supply temperature and flow rate were used as model inputs. The calibrated model showed satisfactory results, achieving RMSE equal to 0.62 °C for indoor temperature and MAPE values equal to 9.9 % for energy consumption.

### 4.7. Real-world implementation

The baseline controller and the DRL control agent obtained from the imitation learning process operated within a Python virtual environment and were deployed in the real testbed by means of a remote desktop PC, equipped with a 4-core CPU running at 3.40 GHz and 16GB of RAM, serving as the central hub for the control logic. Data from sensors are gathered by a Programmable Logic Controller (PLC) situated in HiLo, which furthermore sends the control signals to actuators using the Modbus RTU RS485 protocol and standard analogue/digital signals. The PLC communicates via the multiplatform, open-source OPC-UA protocol with a gateway hosted on a virtual machine. This gateway transmits the collected data from the lower level to an MS-SQL historical database using the Open DataBase Connectivity (ODBC) protocol. A virtual machine accessible remotely through a REST API and integrated in Python hosts the database in the NEST cloud. Real-time data and control signals are exchanged between the remote client and the gateway server using the OPC-UA protocol. Due to the specific control architecture of the NEST units, the control signal must include additional overhead. This includes a signal requesting remote controllability of the system and a square wave watchdog signal that alternates between true and false states every thirty seconds to maintain remote control of the system.

The real-world implementation phase is divided into two main parts: from 15 to 30 November 2023, the baseline controller was implemented to control the TABS and collect data used in the imitation learning phase to extract the initial control policy for the DRL controller. Afterwards, the DRL controller was implemented from 23 February to 7 March 2024, in an online learning setting so that the controller could update its control policy by interacting with the real building. The performance of the DRL was compared with that of the RBC implemented in a digital twin developed in Modelica for our case study and calibrated employing data measured in the real building between 15 and 30 November 2023. The digital twin consists of a detailed RC model that includes modelling the real energy system and thermal zone. The RBC controller was implemented considering the same outdoor conditions related to the real-implementation phase of the DRL controller. A second implementation of the RBC, called RBC2, has been considered in order to compare the performance against that of a better controller. RBC2 employs the same logic as RBC1, except the operation range which has been reduced to 21 and 22 °C. These values were defined after seeing that RBC1 frequently overshooted the upper bound of 23 °C due to the system's thermal inertia. The performance benchmarking was carried out in terms of the energy consumption $E_{\text{tabs}}$ associated with the operation of the TABS, measured in kWh, and of the cumulative temperature violations $T_{\text{viol}}$ measured in °C, during the whole implementation period, defined as follows:

$$T_{\text{viol}} = \sum_{k=0}^{N} b_{\text{occ},k} \cdot T_{\text{viol},k} \tag{7}$$

$T_{\text{viol},k}$ is the temperature violation computed at each timestep $k$ as $|T_i - \underline{T_i}|$ if $T_i < \underline{T_i}$ (i.e., 21 °C), or $|T_i - \overline{T_i}|$ if $T_i > \overline{T_i}$ (i.e., 23 °C), otherwise $T_{\text{viol},k} = 0$ [36].

## 5. Results

This section summarises the results obtained from the implementation of the behavioural cloning strategy in *Office1*. Initially, the outcomes from the training process of the Actor of the PPO controller are described. Then, details about the real-world deployment of PPO controller and the comparison with its performance in the digital twin are discussed. In conclusion, the performance of the proposed imitation learning strategy is compared to that of three reference controllers: RBC1, RBC2, and PI. These controllers were tested in the digital twin over the same period as the real-world application, from 23 February to 7 March 2024.

Fig. 5 presents two key performance metrics, RMSE and the coefficient of determination $R^2$, to evaluate the model's performance on both the training and test datasets over 1000 epochs. In the top panel, RMSE curves for the training (blue) and test (green) datasets show a general downward trend, indicating improved performance as training progresses. Initially, both RMSE values are relatively high but decrease rapidly in the first 200 epochs before gradually declining further. While the test RMSE remains slightly higher than the training RMSE throughout, the small difference suggests normal generalisation rather than overfitting. Both curves stabilise around 600 epochs, signalling convergence. The bottom panel presents the $R^2$ curves for the training and test

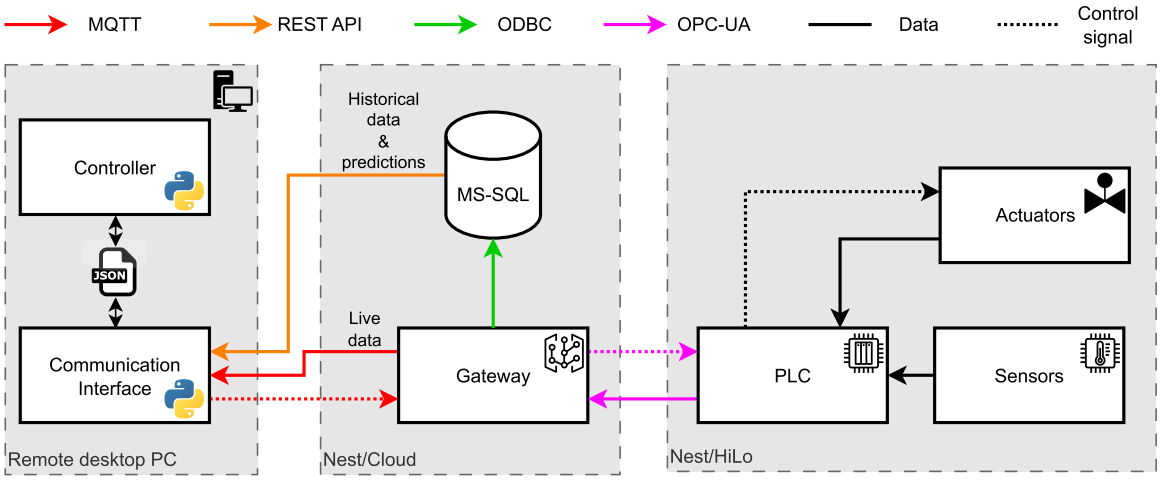



**Fig. 4.** The system physical layout. Adapted from [21].

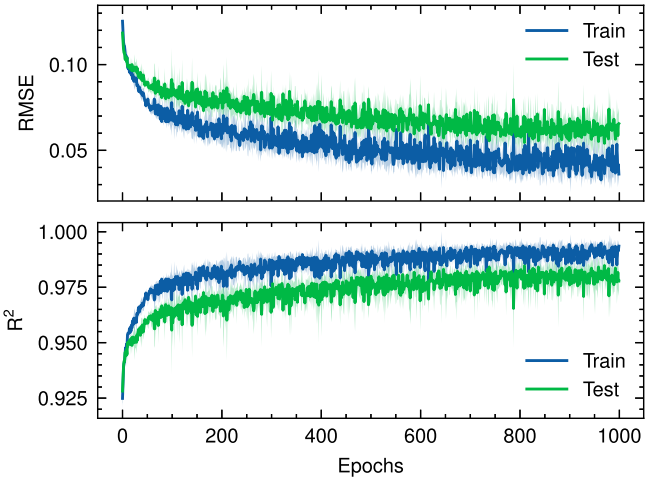



**Fig. 5.** Performance of the model over 1000 epochs, i.e. full pass over the training dataset. The top panel shows the RMSE, and the bottom panel shows the $R^2$ for both the training and test datasets. The solid lines represent the mean values, and the shaded areas indicate $\pm 1$ standard deviation across 5 random seeds.

datasets, which demonstrate a steady improvement. The training $R^2$ approaches values close to 1, indicating an excellent model fit, while the test $R^2$ stabilises near 0.97. Despite the small gap between training and test performance, both metrics show sustained improvement with continued training, suggesting effective generalisation with no significant overfitting.

Fig. 6 depicts the real-world data collected during the application of the behavioural cloning strategy. It includes the indoor temperature $T_i$, the measured heating power supplied by the TABS $\dot{Q}_{tabs}$, the selected control action $u_{valve}$, the supply and return water temperatures $T_s$ and $T_r$, as well as the outdoor temperature $T_o$, the temperature of the nearest room $T_n$, and the solar radiation $\dot{Q}_{sol}$ during the implementation period.

The PPO controller, applied after the behavioural cloning process, successfully keeps the indoor temperature within the acceptable range during the analysis period. However, exceptions occur on the second, seventh, and thirteenth days, when sudden door or window openings cause temperature drops in the early hours of occupancy (around 7:00). In these cases, the controller responds by fully opening the valve to boost heating power and restore the temperature to acceptable levels. Furthermore, the proposed strategy helps reduce energy consumption during occupancy by alternating between two energy management approaches. On some days, it closely replicates the behaviour of the RBC, while on others, it reduces the operating time of the TABS by utilising free thermal gains from occupants, appliances, and solar radiation. Moreover, as the action is directly proportional to the heating power supplied by the TABS, the trends of heating power and control action are generally very similar. However, in certain cases, slight differences may appear due to variations in the temperature difference between the supply and return water in the TABS. These variations can influence the heating power delivered despite the control action being consistent, reflecting the dynamic thermal response of the system.

Fig. 7 compares the performance of the PPO agent in the real-world setting with its behaviour in the digital twin of *Office1* during the second week of the analysis period (i.e., 1–7 March 2024). The figure illustrates both indoor temperature trends and energy consumption profiles, demonstrating how the controller performs in real conditions compared to the digital twin. In the digital twin, the boundary conditions were set to match those of the real-world scenario, and the PPO controller executed the same actions as in the real-world implementation.

Fig. 7 indicates that the PPO controller implemented within the digital twin demonstrated energy consumption performance closely aligned with that of the real building operation. While the temperature profiles were largely consistent, minor discrepancies were observed due to occasional temperature spikes during real-world operations, attributed to factors not represented in the digital twin model, such as doors or windows being opened. A comparative analysis of the energy consumption between the real and the digital twin implementation for the period from 23 February to 7 March 2024 showed nearly identical values ($E_{tabs,real_1} = 37.9$kWh vs $E_{tabs,twin_1} = 37.4$kWh), with a variation of about 2 % in $T_{viol}$. The performance metrics for assessing indoor temperature and energy consumption profiles for the DRL controller in both the real building and the digital twin were $RMSE_{T_i} = 0.53\,°CC$ for temperature and $MAPE_{E_{tabs}} = 7.4\,\%$ for energy consumption.

Fig. 8 shows the performance of the RBC1, RBC2, PI and PPO controllers over a two-week experimental period, focusing on energy consumption and cumulative sum of temperature violations. In Fig. 8, the *Score* term is used to represent the values of the two evaluated metrics, TABS energy consumption ($E_{tabs}$) and the cumulative sum of temperature violations ($T_{viol}$). The x-axis indicates the respective metric being evaluated, while the bar heights correspond to the results obtained for each controller (RBC1, RBC2, PI, and PPO). $T_{viol}$ is expressed in °C, as it represents the cumulative sum of temperature violations from the acceptable range, according to the definition of this metric as indicated in Eq. (7) in Section 4.7. As described in Section 4.6, the benchmarked controllers were implemented using a calibrated Modelica model of the building that functioned as a digital twin, adhering to the

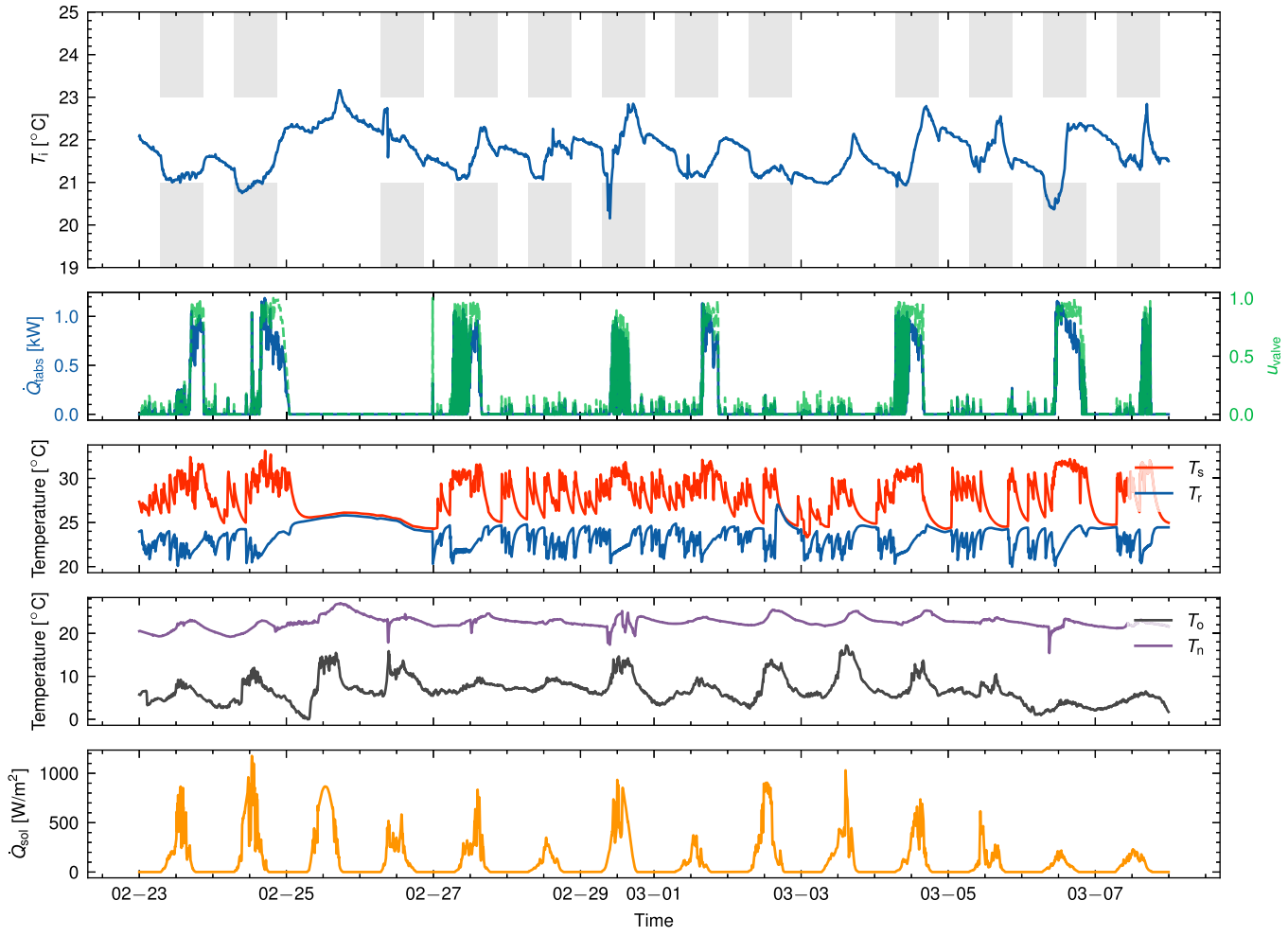



**Fig. 6.** The top plot shows the indoor temperature $T_i$ with the respective bounds, the heating power $\dot{Q}_{tabs}$ and the control action $u_{valve}$ are depicted in the second subplot. The TABS supply/return water temperature profiles $T_s$ and $T_r$ are shown in the third subplot. The bottom plots show the outdoor temperature $T_o$, the temperature of the nearest room $T_n$, and the solar radiation $\dot{Q}_{sol}$. The data have been collected during the implementation of PPO controller in the target office for the period 23 February - 7 March 2024.

same boundary conditions as the real-world implementation of the PPO controller.

Although the PPO controller was trained on data obtained under the RBC1 policy, it reduced energy consumption by 41 % than RBC1 and by 38 % than RBC2. Furthermore, the PPO controller achieved a 43 % reduction in $T_{viol}$ compared to RBC1, and a 13 % reduction compared to RBC2. These improvements are largely due to the online learning capabilities of the PPO, which enabled it to continually optimise its control policy, resulting in enhanced energy efficiency and superior indoor temperature management relative to the two RBC controllers. Comparing the performance of PPO with that of the PI controller, it emerges that the PI controller ensures better temperature control performance, reducing $T_{viol}$ by 58 % compared to PPO, but at the cost of consuming about 45 % more energy. In this case, the PPO is able to manage the TABS better to meet both control objectives, overcoming the limitations of traditional controllers like PI, which cannot handle multi-objective control problems.

In conclusion, Fig. 9 illustrates the indoor temperature with the corresponding temperature ranges (in yellow) for the RBC1, RBC2, PI, and PPO controllers. The second subplot displays the outdoor temperature, while the third subplot presents the control actions of each controller. The final subplot shows the solar radiation.

Fig. 9 shows that the control policy of the PPO closely represents that of the RBC it was trained on. Specifically, while the control actions may vary continuously, the PPO tends to behave similarly to an ON-OFF controller. However, the PPO agent also explores new actions, which allows for the refinement of its policy over time. Towards the end of the experimental period, the PPO learned to switch off earlier than the RBC which it was trained on, and successfully adapted its policy to the system dynamics, maintaining indoor temperatures closer to the lower bound compared to the two RBCs. This adaptation enabled effective energy savings while keeping indoor temperatures within the desired comfort range. While the PPO operates within a continuous action space, allowing for finer adjustments to the valve position, its exploration mechanism results in frequent variations in the action value at each control step $k$. This dynamic adaptability enhances its ability to refine its policy and adapt to changing conditions. However, compared to the RBC1 and RBC2 controllers, which operate in a discrete action space, and the PI controller, which also utilises a continuous action space but with smoother adjustments, the frequent actuation of the PPO controller may increase stress on the valve and reduce its operational lifespan. Future work could address this limitation by introducing a regularisation term or reward penalty to encourage smoother action trajectories, balancing system performance with reduced wear on the actuator. On the other hand, the PI controller aims to maintain the indoor temperature around the setpoint value of 22 °C, effectively avoiding temperature violations on the lower limit (i.e., 21 °C) of the acceptability range and only recording violations when exceeding the upper limit (i.e., 23 °C), as

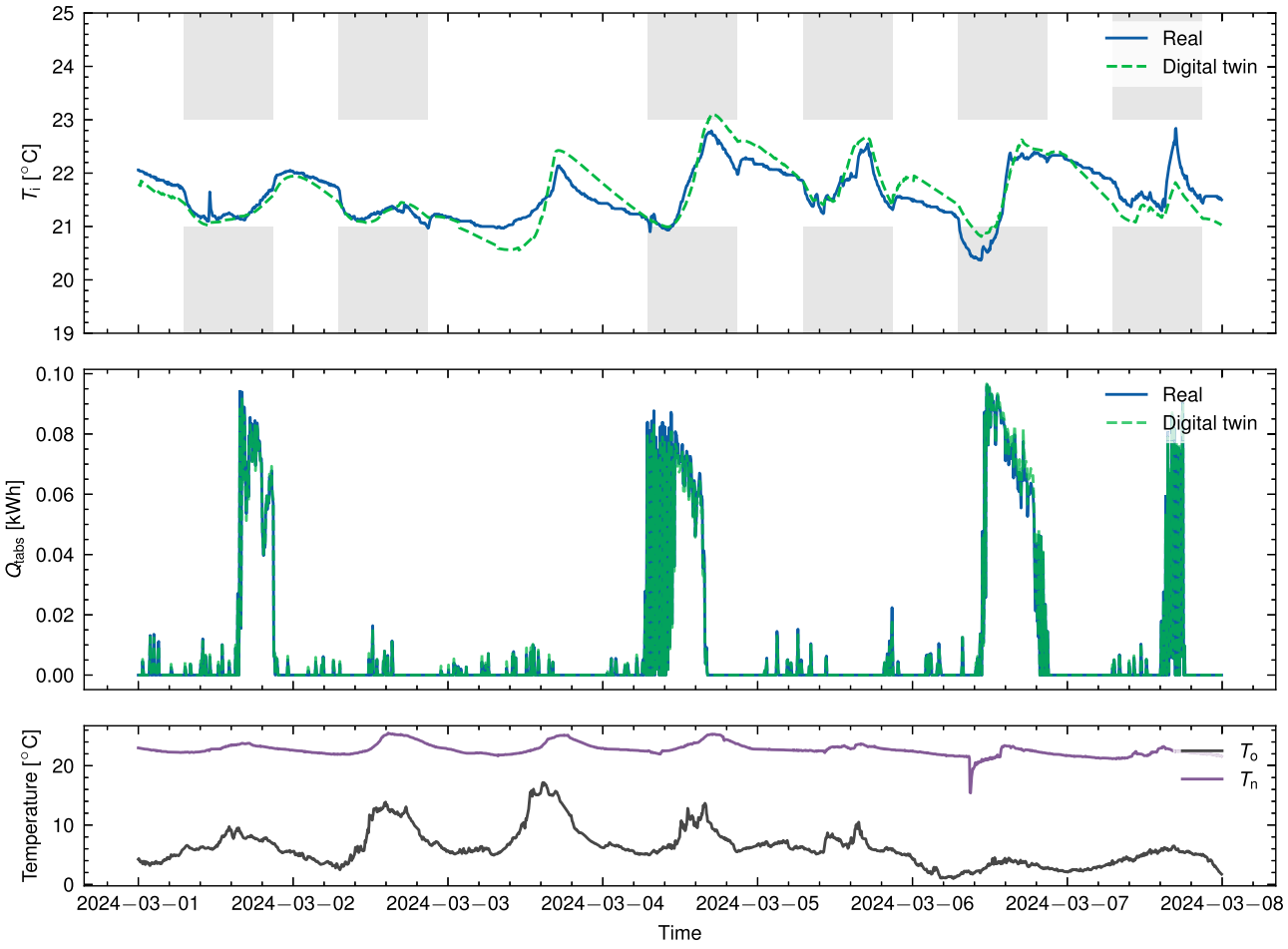



**Fig. 7.** Comparison of indoor temperature and energy consumption profiles between real and digital twin implementations for PPO in *Office1* during the period 1–7 March 2024.

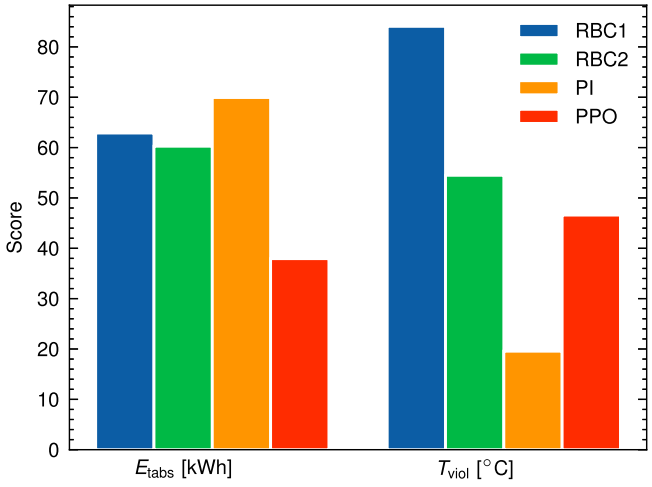



**Fig. 8.** Overall results obtained during the experimental period from the implementation of RBC1, RBC2, PI and PPO controllers in the *Office1* digital twin.

occurs only on the seventh day of the analysis period. In this case, the PI controller keeps the valve closed (i.e., $u_{valve} = 0$) but records violations due to unexpected conditions in *Office1*, such as the sudden opening of the door or an increased number of occupants entering the room compared to typical working days. Additionally, compared to PPO, the PI controller manages to limit the valve opening to around a maximum of 80 %. However, the management of TABS by the PI controller leads to an increase in energy consumption compared to other benchmark controllers, as also discussed in Fig. 8. Although the PPO controller outperformed the benchmark controllers analyzed, an anomalous behavior was observed. Specifically, between the sixth and seventh day of the analysed period, the PPO controller failed to activate the TABS even though the indoor temperature was below the lower limit of the acceptable range, $\underline{T_i}$, during occupancy hours. This behavior appears to be linked to the reliance of PPO controller on the occupancy fraction as a key parameter in its decision-making process. In this context, the PPO controller erroneously anticipated that endogenous heat gains from occupancy would suffice to restore the indoor temperature within the comfort range, thereby avoiding TABS activation when it was actually necessary.

## 6. Discussion

This study presents an imitation learning-based approach to enhance the scalability of a DRL controller for real-world building applications. The proposed method addresses the challenge of deploying DRL controllers by significantly reducing not only the pre-training phase but mainly avoiding the need for a surrogate building model to pre-train the agent. By employing behavioural cloning during the imitation learning stage, the DRL controller was initialised with a policy derived from a baseline rule-based controller. This approach allowed the controller to start from a known behavior and adapt its policy during the deployment phase, where it operated in an online learning mode, continuously adjusting to changes in occupancy and external conditions.

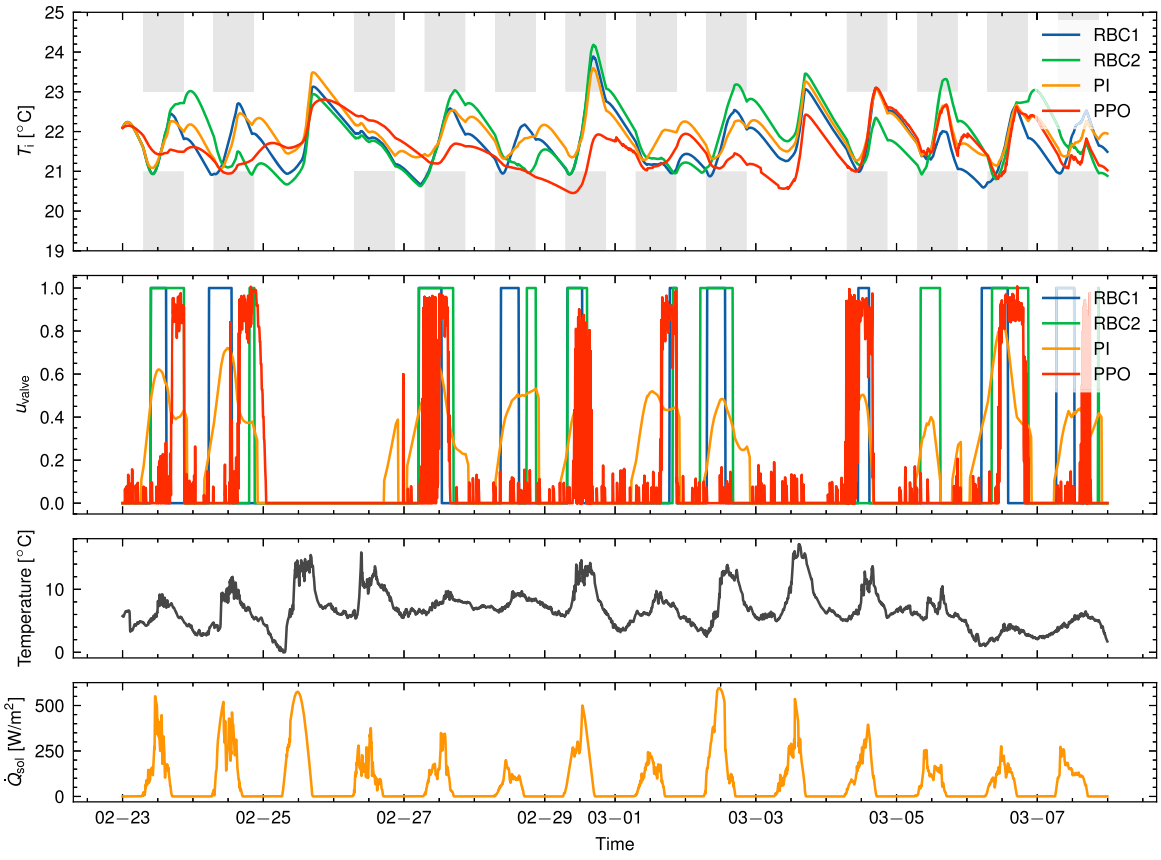



**Fig. 9.** Comparison of indoor temperature profiles and action selection for the analysed RBCs, PI and PPO controllers implemented in the *Office1* digital twin during the experimental period.

Although the method avoids the need for a surrogate model during pre-training, it is still necessary to develop a model for benchmarking purposes. In this study, a well-calibrated digital twin was used to evaluate the DRL controllers performance against two rule-based strategies (RBC1 and RBC2) and a PI controller. The digital twin was calibrated with real building data to closely match real conditions, ensuring that the benchmarking accurately reflected the real-world performance of the controllers. The results showed that the DRL controller implementing the behavioural cloning strategy, achieved higher energy efficiency while ensuring better indoor temperature conditions than RBCs. In contrast, the PI controller demonstrated better temperature regulation due to its consistent control logic, but it was the least efficient controller in terms of energy consumption. This indicates that while the PI controller may stabilise indoor temperatures more effectively during rapid changes, it does so at the cost of higher energy use, making it less suitable for energy-efficient control.

Extending the experiment to a year or another season would require more extensive data for behavioural cloning (BC) to ensure the policy generalises across all conditions. Seasonal changes alter the state space, especially the normalisation of power values, as heating and cooling needs shift. After the initial training with BC, RL-based controllers can adapt dynamically through online learning, continuously refining their policy based on real-time interactions. This makes RL well-suited to handle long-term and seasonal variations, even though the stability of the learning process and the speed of adaptation are critical factors to consider when evaluating long-term performance.

Despite the proposed approach ensuring good performances, several factors need to be addressed when considering the imitation learning approach. The quality of the data used during the behavioural cloning process is of paramount importance, as it directly influences the initial performance of the DRL controller. Poor-quality data could limit the effectiveness of the imitation learning phase, making it difficult for the controller to adapt effectively during the online phase. While the approach simplifies deployment by eliminating the need for detailed pre-training models, the quality of the monitoring infrastructure in real buildings may still present a challenge, as it is often less comprehensive than in controlled environments such as living labs. Data quality can also be influenced by the accuracy of the sensors. If sensors do not measure the variables correctly, they may provide inaccurate information to the controller, leading to suboptimal decisions. Therefore, it is important to assess the effectiveness of the sensors through a sensitivity analysis, which can help identify the impact of sensor inaccuracies on the control strategy [69].

The proposed behavioural cloning strategy differs from transfer learning approaches, where control policies are transferred from one building to another. Instead, the behavioural cloning strategy focuses on acquiring knowledge within the same environment, which avoids the need to adapt the controller across different settings. This simplifies deployment but could limit scalability when applying the approach to buildings with diverse characteristics. Future research could explore hybrid methods combining behavioural cloning with TL, allowing the controller to start with initial knowledge transfer and fine-tune its performance across different environments.

The experiments were carried out for a building equipped with a TABS system, which may not fully represent other HVAC configurations or building types. Further studies could explore the applicability of the approach in various climates, different HVAC systems, and more integrated energy systems, including PV and BESS.

Another limitation is linked to the need for a safety mechanism for unsafe conditions that could be experienced by occupants. While the existing fallback system includes a mechanism that switches to a default control strategy if the DRL encounters malfunctioning or connection losses, it does not fully account for scenarios where conditions may be unsafe for both occupants and the energy systems. Enhancing the safety

mechanism to detect not only occupant discomfort but also potentially harmful situations for the energy infrastructure of the building would improve the robustness of the fallback system. This could involve incorporating thresholds for vital parameters related to HVAC operation or occupants' conditions, triggering the fallback mode when these thresholds are exceeded.

The expert effort required in the different phases of the methodology has been quantified using an approximate percentage of the overall implementation time. Data cleaning accounted for approximately 10 %, communication infrastructure development 20 %, and setting up and tuning the PPO controller 30 %. The remaining 40 % was allocated to validation, monitoring, and system refinement. While these values provide a general idea of the workload distribution, they may vary depending on the specific case and the complexity of the building systems.

Moreover, an adequate monitoring and control infrastructure is a fundamental prerequisite for applying the proposed approach in real buildings. Installing such systems in buildings without them may involve costs ranging from €5000 to €20,000 per thermal zone, depending on the building's complexity and the desired level of instrumentation. The installation time may vary from one to three months, accounting for hardware procurement, installation, and integration with existing systems. For occupancy data, low-cost Passive Infrared Sensors (PIRs) [70], typically ranging between €10 and €50, can be employed. These sensors provide a binary signal indicating the presence of occupants, which is sufficient for many energy management applications. Their accuracy is generally high, with an error margin of 5–10 %, but they cannot provide detailed information about the number of occupants. Regarding solar radiation, it is not always necessary to install dedicated sensors, which may cost between €100 and €500. Reliable data can be obtained from online services like the Solcast API [71], which offers solar radiation data with less than 10 % error, representing a practical and cost-effective solution.

This combination of simplified observation spaces, affordable sensors, and external data sources enhances the scalability and applicability of the methodology in real-world building contexts. Nevertheless, the quality of the monitoring infrastructure in real buildings may still present challenges, as it is often less comprehensive than in controlled environments such as living labs. Poor sensor accuracy or incomplete data could lead to controllers making suboptimal decisions. A sensitivity analysis could be used to assess the robustness of the proposed method under varying levels of data accuracy.

Another aspect of this study was the choice of baseline controllers for benchmarking the proposed DRL-based methodology. The comparison was conducted against RBCs and PI controllers, which are widely used and practically implemented in real-world building environments. This approach was adopted to ensure that the evaluation reflects realistic and practical building control practices.

While advanced control strategies, such as those presented in [72,73] are valuable contributions to the field of building control and similar to the approach proposed in this work, their inclusion in this study was deemed beyond its scope. Such controllers, while demonstrating the potential of hybrid or advanced DRL-based methodologies in research contexts, are not broadly implemented in operational buildings and thus do not represent a standard benchmark in practice. The focus of this study was to highlight the advantages of the proposed DRL-based controller in comparison to control strategies that are currently employed in real-world scenarios, offering a practical perspective on its benefits in terms of energy efficiency and temperature management.

Future work could explore broader comparisons, including advanced DRL implementations and hybrid strategies, to position the proposed approach within the broader landscape of innovative control methodologies. Such comparisons could provide valuable insights into the adaptability and scalability of DRL-based systems when assessed against hybrid methodologies or strategies specifically designed to address barriers to DRL adoption.

In conclusion, this study demonstrates that imitation learning can improve the scalability and feasibility of DRL controllers in real building environments by reducing the need for complex model development and pre-training phases. The findings suggest that this approach can effectively bridge the gap between advanced control methods and practical implementation, opening new opportunities for applying machine learning techniques to optimise smart building operations while ensuring stable and efficient operation from the early stages of deployment.

## 7. Conclusion

This study introduces an imitation learning-based approach to enhance the scalability and practicality of DRL controllers in real-world building applications. By employing behavioural cloning during the imitation learning phase, the DRL controller was initialised with a policy derived from a baseline rule-based controller (RBC1), thereby eliminating the need for detailed pre-training models or surrogate building models commonly required in traditional DRL implementations.

The proposed DRL controller, utilising the PPO algorithm, was implemented in a real office building equipped with TABS. Its performance was compared against three reference controllers: RBC1, RBC2, and a PI controller. These controllers were tested over a two-week period, from 23 February to 7 March 2024, using a calibrated digital twin of the building to ensure accurate benchmarking under the same boundary conditions.

The results demonstrated that the DRL controller significantly improved energy efficiency while maintaining acceptable indoor temperature conditions. Specifically, the DRL controller reduced energy consumption by 41 % compared to RBC1 and by 38 % compared to RBC2. In terms of indoor temperature control, the DRL controller reduced the cumulative sum of temperature violations $T_{\text{viol}}$ by 43 % compared to RBC1 and by 13 % compared to RBC2. These improvements are attributed to the online learning capabilities of the DRL controller, which allowed it to continuously optimise its control policy in response to real-time data and changing conditions.

Compared to the PI controller, the DRL controller achieved substantial energy savings but with a trade-off in temperature control precision. The PI controller ensured better temperature regulation, reducing $T_{\text{viol}}$ by 58 % compared to the DRL controller, but consumed approximately 45 % more energy. This indicates that while the PI controller may track the temperature setpoint better, it does so at the cost of higher energy use, making it less suitable for energy-efficient control.

While the proposed approach simplifies deployment by eliminating the need for complex pre-training models, it relies on high-quality data during the behavioural cloning process. The accuracy of sensors and the comprehensiveness of the monitoring infrastructure are critical factors that can influence the effectiveness of the controller. Additionally, the study focused on a building equipped with TABS, and further research is needed to assess the applicability of the approach to other HVAC configurations and building types.

In conclusion, the findings demonstrate that behavioural cloning can effectively enhance the scalability of DRL controllers in real building environments, achieving significant energy savings and improved indoor temperature control compared to traditional RBCs. The approach bridges the gap between advanced control methods and practical implementation, opening new opportunities for applying machine learning techniques to optimise smart building operations while ensuring stable and efficient operation from the early stages of deployment.

Future work will focus on extending the methodology to more complex HVAC systems, exploring interactions with photovoltaic and storage systems, and incorporating occupant feedback into the control problem. Additionally, testing other DRL-based control algorithms and implementing alternative control strategies, such as MPC, could further improve performance. Exploring transfer learning could also assess whether the control policy developed in this study can be successfully transferred to other buildings or environments.

## Declaration of competing interest

Alberto Silvestri reports financial support was provided by Mitsubishi Electric R&D Centre Europe BV. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## CRediT authorship contribution statement

**Alberto Silvestri:** Writing – original draft, Visualization, Software, Methodology, Formal analysis, Data curation, Conceptualization; **Davide Coraci:** Writing – original draft, Visualization, Software, Methodology, Formal analysis, Data curation, Conceptualization; **Silvio Brandi:** Writing – review & editing, Supervision, Methodology, Conceptualization; **Alfonso Capozzoli:** Writing – review & editing, Validation, Supervision, Conceptualization; **Arno Schlueter:** Writing – review & editing, Validation, Supervision, Conceptualization

## Data availability

The authors do not have permission to share data.

## Acknowledgments

The measured data used in this study were collected and made available by Empa with the support of the Swiss Federal Office of Energy and the Swiss National Science Foundation. The experiments in HiLo were carried out with the support of Sascha Stoller and Reto Fricker.

The work of Alfonso Capozzoli and Silvio Brandi is funded by the project NODES which has received funding from the MUR — M4C2 1.5 of PNRR funded by the European Union — NextGenerationEU (Grant agreement no. ECS00000036).

This study was partly financed by Mitsubishi Electric R&D Centre Europe B.V. [contract number ETH ID No 23035].

## References

[1] Ş. Kılkış, G. Krajačić, N. Duić, M.A. Rosen, M. Ahmad Al-Nimr, Accelerating mitigation of climate change with sustainable development of energy, water and environment systems, Energy Convers. Manag. 245 (2021) 114606. https://doi.org/10.1016/j.enconman.2021.114606.

[2] M.S. Piscitelli, S. Brandi, A. Capozzoli, F. Xiao, A data analytics-based tool for the detection and diagnosis of anomalous daily energy patterns in buildings, Build. Simul. 14(1) (2021) 131–147. https://doi.org/10.1007/s12273-020-0650-1.

[3] U.N. UNEP, 2022 global status report for buildings and construction, 2019. (Accessed 19 January 2025.

[4] D. Coraci, S. Brandi, M.S. Piscitelli, A. Capozzoli, Online implementation of a soft actor-critic agent to enhance indoor temperature control and energy efficiency in buildings, Energies 14(4) (2021). https://doi.org/10.3390/en14040997.

[5] M. Dorokhova, C. Ballif, N. Wyrsch, Rule-based scheduling of air conditioning using occupancy forecasting, Energy AI 2 (2020) 100022. https://doi.org/10.1016/j.egyai.2020.100022.

[6] G. ASHRAE, 36: High performance sequences of operation for HVAC systems, American Society of Heating, Refrigerating and Air-Conditioning Engineers, Atlanta (2021). https://www.ashrae.org/news/ashraejournal/guideline-36-2021-what-s-new-and-why-it-s-important.

[7] Z. Wang, T. Hong, Reinforcement learning for building controls: the opportunities and challenges, Appl. Energy 269 (2020) 115036. https://doi.org/10.1016/j.apenergy.2020.115036.

[8] C. Finck, P. Beagon, J. Clauß, T. Péan, P. Vogler-Finck, K. Zhang, H. Kazmi, Review of applied and tested control possibilities for energy flexibility in buildings, Technical report from IEA EBC Annex 67 – Energy Flexible Buildings (2017) 1–59. https://doi.org/10.13140/RG.2.2.28740.73609.

[9] T.I. Salsbury, A survey of control technologies in the building automation industry, IFAC Proc. 38(1) (2005) 90–100. 16th IFAC World Congress. https://doi.org/10.3182/20050703-6-CZ-1902.01397.

[10] A. Capozzoli, M.S. Piscitelli, S. Brandi, D. Grassi, G. Chicco, Automated load pattern learning and anomaly detection for enhancing energy management in smart buildings, Energy 157 (2018) 336–352. https://doi.org/10.1016/j.energy.2018.05.127.

[11] D. Coraci, S. Brandi, A. Capozzoli, Effective pre-training of a deep reinforcement learning agent by means of long short-term memory models for thermal energy management in buildings, Energy Convers. Manag. 291 (2023) 117303. https://doi.org/10.1016/j.enconman.2023.117303.

[12] M. Molina-Solana, M. Ros, M.D. Ruiz, J. Gómez-Romero, M.J. Martin-Bautista, Data science for building energy management: a review, Renew. Sustain. Energy Rev. 70 (2017) 598–609. https://doi.org/10.1016/j.rser.2016.11.132.

[13] G. Serale, M. Fiorentini, A. Capozzoli, D. Bernardini, A. Bemporad, Model predictive control (MPC) for enhancing building and HVAC system energy efficiency: problem formulation, applications and opportunities, Energies 11(3) (2018). https://doi.org/10.3390/en11030631.

[14] G. Serale, M. Fiorentini, A. Capozzoli, P. Cooper, M. Perino, Formulation of a model predictive control algorithm to enhance the performance of a latent heat solar thermal system, Energy Convers. Manage. 173 (2018) 438–449. https://doi.org/10.1016/j.enconman.2018.07.099.

[15] J. Drgoa, J. Arroyo, I. Cupeiro Figueroa, D. Blum, K. Arendt, D. Kim, E.P. Ollé, J. Oravec, M. Wetter, D.L. Vrabie, L. Helsen, All you need to know about model predictive control for buildings, Annu. Rev. Control (2020). https://doi.org/10.1016/j.arcontrol.2020.09.001.

[16] F. Oldewurtel, A. Parisio, C.N. Jones, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, M. Morari, Use of model predictive control and weather forecasts for energy efficient building climate control, Energy Build. 45 (2012) 15–27. https://doi.org/10.1016/j.enbuild.2011.09.022.

[17] G.P. Henze, R.H. Dodier, M. Krarti, Development of a predictive optimal controller for thermal energy storage systems, HVAC&R Res. 3(3) (1997) 233–264. https://doi.org/10.1080/10789669.1997.10391376.

[18] G.D. Kontes, G.I. Giannakis, V. Sánchez, P. De Agustin-Camacho, A. Romero-Amorrortu, N. Panagiotidou, D.V. Rovas, S. Steiger, C. Mutschler, G. Gruen, Simulation-based evaluation and optimization of control strategies in buildings, Energies 11(12) (2018). https://doi.org/10.3390/en11123376.

[19] Z. Nagy, G. Henze, S. Dey, J. Arroyo, L. Helsen, X. Zhang, B. Chen, K. Amasyali, K. Kurte, A. Zamzam, H. Zandi, J. Drgoa, M. Quintana, S. McCullogh, J.Y. Park, H. Li, T. Hong, S. Brandi, G. Pinto, A. Capozzoli, D. Vrabie, M. Bergés, K. Nweye, T. Marzullo, A. Bernstein, Ten questions concerning reinforcement learning for building energy management, Build. Environ. 241 (2023) 110435. https://doi.org/10.1016/j.buildenv.2023.110435.

[20] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, The MIT Press, second ed., 2018. http://incompleteideas.net/book/the-book-2nd.html.

[21] A. Silvestri, D. Coraci, S. Brandi, A. Capozzoli, E. Borkowski, J. Köhler, D. Wu, M.N. Zeilinger, A. Schlueter, Real building implementation of a deep reinforcement learning controller to enhance energy efficiency and indoor temperature control, Appl. Energy 368 (2024) 123447. https://doi.org/10.1016/j.apenergy.2024.123447.

[22] G. Pinto, D. Deltetto, A. Capozzoli, Data-driven district energy management with surrogate models and deep reinforcement learning, Appl. Energy 304 (2021) 117642. https://doi.org/10.1016/j.apenergy.2021.117642.

[23] V. Mnih, K. Kavukcuoglu, D. Silver, A.A. Rusu, J. Veness, M.G. Bellemare, A. Graves, M. Riedmiller, A.K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, D. Hassabis, Human-level control through deep reinforcement learning, Nature 518(7540) (2015) 529–533. http://dx.doi.org/10.1038/nature14236.

[24] A. Kathirgamanathan, E. Mangina, D.P. Finn, Development of a soft actor critic deep reinforcement learning approach for harnessing energy flexibility in a large office building, Energy AI 5 (2021) 100101. https://doi.org/10.1016/j.egyai.2021.100101.

[25] K. He, Q. Fu, Y. Lu, Y. Wang, J. Luo, H. Wu, J. Chen, Predictive control optimization of chiller plants based on deep reinforcement learning, J. Build. Eng. 76 (2023) 107158. https://doi.org/10.1016/j.jobe.2023.107158.

[26] T. Schreiber, S. Eschweiler, M. Baranski, D. Müller, Application of two promising reinforcement learning algorithms for load shifting in a cooling supply system, Energy Build. 229 (2020) 110490. https://doi.org/10.1016/j.enbuild.2020.110490.

[27] A. Silvestri, D. Coraci, D. Wu, E. Borkowski, A. Schlueter, Comparison of two deep reinforcement learning algorithms towards an optimal policy for smart building thermal control, J. Phys. Conf. Ser. 2600(7) (2023) 072011. https://doi.org/10.1088/1742-6596/2600/7/072011.

[28] Y. Gao, S. Shi, S. Miyata, Y. Akashi, Successful application of predictive information in deep reinforcement learning control: a case study based on an office building HVAC system, Energy 291 (2024) 130344. https://doi.org/10.1016/j.energy.2024.130344.

[29] M. Wang, B. Lin, Mf̂: model-free reinforcement learning for modeling-free building hvac control with data-driven environment construction in a residential building, Build. Environ. 244 (2023) 110816. https://doi.org/10.1016/j.buildenv.2023.110816.

[30] S. Brandi, M. Fiorentini, A. Capozzoli, Comparison of online and offline deep reinforcement learning with model predictive control for thermal energy management, Autom. Constr. 135 (2022) 104128. https://doi.org/10.1016/j.autcon.2022.104128.

[31] X. Wang, X. Kang, J. An, H. Chen, D. Yan, Reinforcement learning approach for optimal control of ice-based thermal energy storage (TES) systems in commercial buildings, Energy Build. 301 (2023) 113696. https://doi.org/10.1016/j.enbuild.2023.113696.

[32] A. Hussain, P. Musilek, Energy management of buildings with energy storage and solar photovoltaic: a diversity in experience approach for deep reinforcement learning agents, Energy AI 15 (2024) 100313. https://doi.org/10.1016/j.egyai.2023.100313.

[33] G.T. Costanzo, S. Iacovella, F. Ruelens, T. Leurs, B.J. Claessens, Experimental analysis of data-driven control for a building heating system, Sustain. Energy Grids Netw. 6 (2016) 81–90. https://doi.org/10.1016/j.segan.2016.02.002.

[34] D.B. Crawley, L.K. Lawrie, F.C. Winkelmann, W.F. Buhl, Y.J. Huang, C.O. Pedersen, R.K. Strand, R.J. Liesen, D.E. Fisher, M.J. Witte, J. Glazer, Energyplus: cre-

ating a new-generation building energy simulation program, Energy Build. 33(4) (2001) 319–331. Special Issue: BUILDING SIMULATION'99. https://doi.org/10.1016/S0378-7788(00)00114-6.
[35] M. Association, Modelica® – a unified object-oriented language for physical systems modeling. Tutorial, 2000. http://www.modelica.org/documents/ModelicaTutorial14.pdf.
[36] D. Coraci, S. Brandi, T. Hong, A. Capozzoli, An innovative heterogeneous transfer learning framework to enhance the scalability of deep reinforcement learning controllers in buildings with integrated energy systems, Build. Simul. 17 (2024) 739–770. https://doi.org/10.1007/s12273-024-1109-6.
[37] G. Pinto, Z. Wang, A. Roy, T. Hong, A. Capozzoli, Transfer learning for smart buildings: a critical review of algorithms, applications, and future perspectives, Adv. Appl. Energy 5 (2022) 100084. https://doi.org/10.1016/j.adapen.2022.100084.
[38] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22(10) (2010) 13451359. https://doi.org/10.1109/TKDE.2009.191.
[39] S. Dey, T. Marzullo, X. Zhang, G. Henze, Reinforcement learning building control approach harnessing imitation learning, Energy AI 14 (2023) 100255. https://doi.org/10.1016/j.egyai.2023.100255.
[40] P. Lissa, M. Schukat, M. Keane, E. Barrett, Transfer learning applied to DRL-based heat pump control to leverage microgrid energy efficiency, Smart Energy 3 (2021) 100044. https://doi.org/10.1016/j.segy.2021.100044.
[41] D. Coraci, S. Brandi, T. Hong, A. Capozzoli, Online transfer learning strategy for enhancing the scalability and deployment of deep reinforcement learning control in smart buildings, Appl. Energy 333 (2023) 120598. https://doi.org/10.1016/j.apenergy.2022.120598.
[42] K. Nweye, S. Sankaranarayanan, Z. Nagy, Merlin: multi-agent offline and transfer learning for occupant-centric operation of grid-interactive communities, Appl. Energy 346 (2023) 121323. https://doi.org/10.1016/j.apenergy.2023.121323.
[43] F. Hou, J.C.P. Cheng, H.H.L. Kwok, J. Ma, Multi-source transfer learning method for enhancing the deployment of deep reinforcement learning in multi-zone building HVAC control, Energy Build. 322 (2024) 114696. https://doi.org/10.1016/j.enbuild.2024.114696.
[44] M. Liu, M. Guo, Y. Fu, Z. O'Neill, Y. Gao, Expert-guided imitation learning for energy management: evaluating GAIL's performance in building control applications, Appl. Energy 372 (2024) 123753. https://doi.org/10.1016/j.apenergy.2024.123753.
[45] K. Amasyali, Y. Liu, H. Zandi, A Transfer Learning Strategy for Improving the Data Efficiency of Deep Reinforcement Learning Control in Smart Buildings, in: 2024 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), 2024, pp. 1–5. https://doi.org/10.1109/ISGT59692.2024.10454120.
[46] K. Kadamala, D. Chambers, E. Barrett, Enhancing HVAC control efficiency: a hybrid approach using Imitation and reinforcement learning, in: A. Bifet, T. Krilavičius, I. Miliou, S. Nowaczyk (Eds.), Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track, Springer Nature Switzerland, Cham, 2024, pp. 256–270.
[47] R. Bellman, Dynamic programming, Science 153(3731) (1966) 34–37. https://doi.org/10.1126/science.153.3731.34.
[48] J. Schulman, S. Levine, P. Moritz, M.I. Jordan, P. Abbeel, Trust region policy optimization, 2017. https://arxiv.org/abs/1502.05477.
[49] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms, 2017. https://arxiv.org/abs/1707.06347.
[50] Z. Zhang, K.P. Lam, Practical Implementation and Evaluation of Deep Reinforcement Learning Control for a Radiant Heating System, in: Proceedings of the 5th Conference on Systems for Built Environments, BuildSys '18, Association for Computing Machinery, New York, NY, USA, 2018, p. 148157. https://doi.org/10.1145/3276774.3276775.
[51] T.M. Cover, J.A. Thomas, Elements of Information Theory, Wiley, 2012. https://books.google.it/books?id=VWq5GG6ycxMC.
[52] S. Dey, T. Marzullo, G. Henze, Inverse reinforcement learning control for building energy management, Energy Build. 286 (2023) 112941. https://doi.org/10.1016/j.enbuild.2023.112941.
[53] A. Gleave, M. Taufeeque, J. Rocamonde, E. Jenner, S.H. Wang, S. Toyer, M. Ernestus, N. Belrose, S. Emmons, S. Russell, imitation: clean imitation learning implementations, 2022. https://arxiv.org/abs/2211.11972.
[54] A. Hussein, M.M. Gaber, E. Elyan, C. Jayne, Imitation learning: a survey of learning methods, ACM Comput. Surv. 50(2) (2017). https://doi.org/10.1145/3054912.
[55] S.J. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, Pearson, 2016.
[56] N. Gavenski, O. Rodrigues, M. Luck, Imitation learning: a survey of learning methods, environments and metrics, 2024. https://arxiv.org/abs/2404.19456.
[57] B. Zheng, S. Verma, J. Zhou, I.W. Tsang, F. Chen, Imitation learning: progress, taxonomies and challenges, IEEE Trans. Neural Netw. Learn. Syst. 35(5) (2024) 6322–6337. https://doi.org/10.1109/TNNLS.2022.3213246.
[58] M. Bain, C. Sammut, A framework for behavioural cloning, in: Machine Intelligence 15, 1995, pp. 103–129.
[59] Z. Zhu, K. Lin, B. Dai, J. Zhou, Off-policy imitation learning from observations, Adv. Neural Inf. Process. Syst. 33 (2020) 12402–12413.
[60] P. Block, A. Schlueter, D. Veenendaal, J. Bakker, M. Begle, I. Hischier, J. Hofer, P. Jayathissa, I. Maxwell, T.M. Echenagucia, Z. Nagy, D. Pigram, B. Svetozarevic, R. Torsing, J. Verbeek, A. Willmann, G.P. Lydon, et al., NEST HiLo: investigating lightweight construction and adaptive energy systems, J. Build. Eng. 12 (2017) 332–341. https://doi.org/10.1016/j.jobe.2017.06.013.
[61] P. Richner, P. Heer, R. Largo, E. Marchesi, M. Zimmermann, et al., NEST - a platform for the acceleration of innovation in buildings, Inf. Constr. 69(548) (2018) 222. https://doi.org/10.3989/id.55380.
[62] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: a next-generation hyperparameter optimization framework, in: KDD '19: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, New York, NY, USA, 2019, p. 26232631. https://doi.org/10.1145/3292500.3330701.
[63] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, N. Dormann, Stable-baselines3: reliable reinforcement learning implementations, J. Mach. Learn. Res. 22(268) (2021) 1–8. http://jmlr.org/papers/v22/20-1364.html.
[64] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, W. Zaremba, OpenAI gym, 2016.
[65] P. Fritzson, A. Pop, P. Aronsson, H. Lundvall, K. Nyström, L. Saldamli, D. Broman, The openmodelica modeling, simulation, and development environment, 2005. https://www.researchgate.net/publication/252264811.
[66] M. Wetter, W. Zuo, T.S. Nouidui, X. Pang, Modelica buildings library, J. Build. Perform. Simul. 7(4) (2014) 253–270. https://doi.org/10.1080/19401493.2013.765506.
[67] T. Blochwitz, M. Otter, J. Åkesson, M. Arnold, C. Clauss, H. Elmqvist, M. Friedrich, A. Junghanns, J. Mauss, D. Neumerkel, H. Olsson, A. Viel, Functional mockup interface 2.0: the standard for tool independent exchange of simulation models, in: Proceedings of the 9th International Modelica Conference, The Modelica Association, 2012, pp. 173–184. Key=blo+12mc project=LCCC-modeling; 9th International Modelica Conference ; Conference date: 03-09-2012. https://doi.org/10.3384/ecp12076173.
[68] C. Andersson, J. Åkesson, C. Führer, PyFMI: A Python Package for Simulation of Coupled Dynamic Models with the Functional Mock-up Interface, volume LUTFNA-5008-2016 of Technical Report in Mathematical Sciences, Centre for Mathematical Sciences, Lund University, 2016.
[69] Y. Bae, S. Bhattacharya, B. Cui, S. Lee, Y. Li, L. Zhang, P. Im, V. Adetola, D. Vrabie, M. Leach, T. Kuruganti, Sensor impacts on building and HVAC controls: a critical review for building energy performance, Adv. Appl. Energy 4 (2021) 100068. https://doi.org/10.1016/j.adapen.2021.100068.
[70] D. Coraci, A. Silvestri, G. Razzano, D. Fop, S. Brandi, E. Borkowski, T. Hong, A. Schlueter, A. Capozzoli, A scalable approach for real-world implementation of deep reinforcement learning controllers in buildings based on online transfer learning: the HiLo case study, Energy Build. 329 (2025) 115254. https://doi.org/10.1016/j.enbuild.2024.115254.
[71] Solcast, Solcast API, 2025, (https://www.solcast.com/). Solcast: Solar API and weather forecast. Accessed 22 January 2025.
[72] B. Chen, Z. Cai, M. Bergés, Gnu-RL: a precocial reinforcement Learning solution for building HVAC control using a differentiable MPC policy, in: Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, Association for Computing Machinery, New York, NY, USA, 2019, p. 316325. https://doi.org/10.1145/3360322.3360849.
[73] A. Krishna G. S. , T. Zhang, O. Ardakanian, M.E. Taylor, Mitigating an adoption barrier of reinforcement learning-based control strategies in buildings, Energy Build. 285 (2023) 112878. https://doi.org/10.1016/j.enbuild.2023.112878.