Contents lists available at ScienceDirect

Energy & Buildings

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

A scalable approach for real-world implementation of deep reinforcement learning controllers in buildings based on online transfer learning: The HiLo case study

Davide Coraci [a], Alberto Silvestri [b], Giuseppe Razzano [a], Davide Fop [a], Silvio Brandi [a], Esther Borkowski [b,d], Tianzhen Hong [c], Arno Schlueter [b], Alfonso Capozzoli [a,*]

[a] Politecnico di Torino, Department of Energy, TEBE research group, BAEDA Lab, Corso Duca degli Abruzzi 24, Torino, 10129, Italy
[b] Architecture and Building Systems, ETH Zurich, Stefano-Franscini, Platz 5, Zurich, 8049, Switzerland
[c] Building Technology and Urban Systems Division, Lawrence Berkeley National Laboratory, One Cyclotron Road, Berkeley, CA 94720, USA
[d] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Richard Petersens Plads, Building 424, Kongens Lyngby, 2800, Denmark

ARTICLE INFO

Keywords:
Transfer learning
Real implementation
Deep reinforcement learning
Building HVAC control
Energy efficiency

ABSTRACT

In recent years, Transfer Learning (TL) has emerged as a promising solution to scale Deep Reinforcement Learning (DRL) controllers for building energy management, addressing challenges related to DRL implementation as high data requirements and reliance on surrogate models. Moreover, most TL applications are limited to simulations, not revealing their real performance in actual buildings. This paper explores the implementation of an online TL methodology combining imitation learning and fine-tuning to transfer a DRL controller between two real office environments.

Pre-trained in simulation using a calibrated digital twin, the DRL controller reduces energy consumption and improves indoor temperature control when managing the operation of a Thermally Activated Building System in one of the two offices both in simulation and in the real field. Afterwards, the DRL controller is transferred to the other office following the online TL methodology. The proposed approach outperforms a DRL controller implemented without pre-training, and Rule-Based and Proportional-Integral controllers, achieving energy savings between 6 and 40% and improving indoor temperature control between 30 and 50%. These findings underscore the efficacy of the online TL methodology as a viable solution to enhance the scalability of DRL controllers in real buildings.

Nomenclature

α	Boltzmann temperature coefficient
β	Temperature term weight of reward function
$\dot{Q}_{\mathrm{sol}}$	Solar radiation [W/m^2]
$\dot{Q}_{\mathrm{tabs}}$	Heating power delivered by TABS [kW]
γ	Discount factor
μ	DRL Learning rate
$\overline{T_{\mathrm{i}}}$	Upper limit of temperature comfort range [°C]
$\overline{T}_{\mathrm{viol,daily}}$	Mean value of the daily average temperature violation rate
θ	Reward scaling factor
$\underline{T_{\mathrm{i}}}$	Lower limit of temperature comfort range [°C]
b_{occ}	Occupancy boolean variable
E_{tabs}	Energy consumption associated with the TABS operation [kWh]
f_{occ}	Occupancy fraction over each control time step
$n_{\mathrm{viol,occ,daily}}$	Cumulative sum of daily occurrences of temperature violation
r	Reward function
r_E	Energy term of reward function
r_T	Temperature term of reward function
T_{i}	Indoor air temperature [°C]
T_{o}	Outdoor air temperature [°C]
$T_{\mathrm{viol,daily}}$	Cumulative sum of daily temperature violation [°C]

[*] Corresponding author.
E-mail address: alfonso.capozzoli@polito.it (A. Capozzoli).

https://doi.org/10.1016/j.enbuild.2024.115254
Received 9 September 2024; Received in revised form 14 December 2024; Accepted 27 December 2024

Available online 3 January 2025

u_t	Percentage opening of the valve

Acronyms

AI	Artificial Intelligence
BESS	Battery Energy Storage System
DRL	Deep Reinforcement Learning
DTW	Dynamic Time Warping
FDD	Fault Detection and Diagnosis
FMI	Functional Mock-up Interface
FMU	Functional Mock-up Unit
GA	Genetic Algorithm
HVAC	Heating, Ventilation and Air Conditioning
HVRF	Hybrid Variable Refrigerant Flow
IEA	International Energy Agency
IL	Imitation Learning
KPI	Key Performance Indicator
MAPE	Mean Absolute Percentage Error
ML	Machine Learning
MPC	Model Predictive Control
NEST	Next Evolution in Sustainable building Technologies
ODBC	Open DataBase Connectivity
PI	Proportional-Integral
PID	Proportional-Integral-Derivative
PIRs	Passive Infrared Sensors
PLC	Programmable Logic Controller
POMDP	Partially Observable Markov Decision Process
PV	Photovoltaic
RBC	Rule-Based Controller
RC	Resistance-Capacitance
RL	Reinforcement Learning
RMSE	Root Mean Squared Error
RTP	Real-Time Pricing
SAC	Soft Actor-Critic
TABS	Thermally Activated Building System
TL	Transfer Learning
TOU	Time-Of-Use
TPE	Tree-structured Parzen Estimator
VAV	Variable Air Volume

1. Introduction

The building sector is one of the major contributors to total world energy consumption considering it requires 40% of energy and contributes by nearly 25% to global CO_2 emissions [1]. The International Energy Agency (IEA) states that buildings are *"a source of enormous untapped efficiency potential"* [2] and *"this decade is crucial for implementing measures to achieve targets planned by 2030 to make energy systems smarter, more connected, efficient, and resilient"* [3]. In this context, building energy management measures emerge as a potential solution for enhancing the operation of energy systems to decrease energy costs related to building operation and to enhance indoor comfort conditions for occupants [4,5].

In buildings, Heating, Ventilation and Air Conditioning (HVAC) systems are the major contributors to energy consumption, demanding approximately 60% of the total energy [6]. In recent years, different measures have been proposed to enhance the efficiency of HVAC system, such as the introduction of energy-efficient energy systems coupled with the implementation of advanced control strategies [7].

Nowadays, HVAC systems in buildings are typically controlled by straightforward control strategies like Rule-Based Controller (RBC) or Proportional-Integral-Derivative (PID) controllers [8] that are often suboptimal since they can not optimise multi-objective control problems due to their reactive nature [9]. Moreover, RBC and PID controllers are unable to dynamically adjust their control policies to changing boundary conditions such as weather patterns, electricity costs and grid requirements [10].

In this regard, the adoption of advanced control approaches based on Artificial Intelligence (AI) can overcome the limitations of traditional control strategies [11], due to their predictive and adaptive capabilities allowing for increased energy flexibility in buildings, a key factor in optimally managing energy systems according to occupant needs and to the dynamic boundary conditions existing in real buildings [12].

Among advanced control strategies, Reinforcement Learning (RL) has emerged as a promising solution due to its capability to automatically enhance system operations, taking into account multiple objectives and adjusting to dynamic conditions autonomously with minimal human intervention, which are highly-requested features for controllers managing HVAC [13]. RL operates on a trial-and-error basis, where a control agent learns a near-optimal control policy through direct interaction with the environment, guided by a reward mechanism [14].

Deep Reinforcement Learning (DRL) is the most widely utilised control algorithm within the RL domain, effectively addressing complex control tasks, particularly in scenarios where multiple states and actions should be defined to properly represent the control problems, as in real-world buildings [15]. The implementation of DRL for HVAC systems demonstrates optimal performance when applied to manage the operation mode of thermal energy systems [16,17] and storage systems [18], supply water temperature [19] and mass flow rate [20], fan speed [21] and indoor air temperature setpoints [22,23].

The online and offline training methodologies are two different training strategies for enabling the interaction between the DRL-based agent and the controlled environment to retrieve the near-optimal control policy [24]. The online training strategy entails learning the optimal control policy while actively controlling the system [25]. However, this approach is not efficient due to the considerable training time needed for the DRL controller to interact with the environment to reach a near-optimal policy, primarily because of its initial poor performance. Therefore, the offline training approach is the most employed in building energy management DRL applications [25]. This training strategy is carried out in a simulation environment, and it requires the definition of a surrogate model that emulates the dynamics of the controlled environment. Although the offline training approach for DRL agents has shown impressive results, it faces significant challenges related to scalability and generalisability. Due to the unique features of each building, the design of data-driven or physics-based surrogate models becomes necessary. Developing data-driven surrogate models necessitates a minimum amount of monitoring data for the controlled building, while making physics-based models can be a time-intensive endeavour as it requires access to comprehensive building information (which may not always be available) and domain expertise [24].

To tackle these practical challenges, Transfer Learning (TL) appears as a promising solution to enhance the scalability of DRL controllers, enabling their implementation in real-world buildings. TL is a Machine Learning (ML) technique where a model initially trained to tackle a specific task (i.e., source task) in a particular domain is leveraged to address a new task (i.e., target task) [26]. This new task shares similarities with the original task, either within the same domain or across different domains [27]. In recent years, TL has been increasingly applied in building energy management across various domains, such as load prediction [28,29], occupancy detection and activity recognition [30,31], building dynamics [32,33], and Fault Detection and Diagnosis (FDD) [34,35].

The earliest instances of TL are recent compared to other DRL applications in the field of building energy management. Introducing a TL strategy for DRL controllers in buildings presents several benefits. These include boosting the direct deployment of these controllers in real buildings with adequate performance from the initial implementation stages, enhancing their scalability by eliminating the need to develop surrogate models and expanding their application in buildings with limited historical data. In addition to TL, other knowledge reuse approaches have been defined in the literature, such as Imitation Learning (IL). IL involves a control agent learning an optimal strategy for a specific task by observing the behaviour of an expert controller [36]. During this process, the control agent accesses the transitions generated by the expert, which detail the action selected by the expert controller and the resulting changes in state-spaces.

In recent years the number of contributions on TL for DRL controller applications has increased. In [37], TL was employed for transferring a DRL-based control policy between different buildings. The DRL controller managed the supply temperature setpoint of a chiller. This method achieved an average reduction of 40% in the duration of the warm-up period related to the implementation of DRL controllers while reducing on average by 50% the cumulative reward for target buildings. Moreover, Nweye et al. [38] introduced a TL approach incorporating IL to emulate the behaviour of a RBC considering measured data for five months. Thus, weight initialisation was carried out for training the DRL controllers before their deployment in an energy community. Results indicate that DRL control policy sharing within buildings in the energy community led to similar performance and reduced training time compared to the no-TL scenario.

The next section discusses the novelty and contributions of this paper, while other related works concerning the application of TL for DRL controllers in buildings are included in Appendix A.

1.1. Novelty and contributions of the paper

Several applications discussed in the literature consider a TL approach for DRL controllers with fine-tuning over multiple episodes. However, this does not find practical applications because it would require, depending on the type of application, several heating/cooling seasons (i.e., each epoch corresponds to a season in reality). In this framework, the TL applications developed so far for DRL controllers have only been evaluated in simulation environments, where building surrogate models were developed to emulate their real-world performance.

In this context, the online TL approach could be a viable solution to evaluate the real-world implementation of DRL control strategies since it allows for better performance from the early stages of implementation, significantly increasing their scalability and enabling more practical applications.

Implementing a TL strategy in real buildings must consider several aspects similar to those related to the DRL real implementation. These aspects include the need to ensure the continuous and proper functioning of the existing systems and the need to develop an adequate benchmarking method to quantify whether the performance of the transferred controller is better than that of traditionally implemented controllers in real buildings.

Following these considerations, this paper aims to demonstrate the practicality and effectiveness of applying online TL to DRL controllers in real-world scenarios, bridging the gap between theoretical research and real-world application while discussing the encountered implementation challenges.

Specifically, this paper evaluates the real implementation of the online TL methodology in a building located in Switzerland, where a DRL controller was transferred between two offices. The developed online TL approach was homogeneous and transductive, according to the classification defined in [39,40] and discussed in Appendix B, as the DRL controllers operating in the two offices had the same domain (i.e., action-space and state-space) and the same objective (i.e., reward function). For each office, a digital twin is developed in Modelica for pre-training and further benchmarking purposes.

The developed process involved transferring a pre-trained control policy based on the Soft Actor-Critic (SAC) algorithm. Theoretical foundations for DRL controllers are detailed in Appendix B. The aim of the implemented DRL controller is to minimise energy consumption while optimising indoor temperature conditions by regulating the opening percentage of the valve in the Thermally Activated Building System (TABS) supply circuit. Afterwards, the pre-trained DRL controller was deployed in the real source office to test its performance and allow it to continue learning according to real building dynamics and boundary conditions. A benchmarking process was carried out for the DRL controller to compare its performance with RBC and Proportional-Integral (PI)-based controllers.

Thus, the DRL controller implemented in the office was transferred to the real target office. The performance of the online TL implemented in the target office was benchmarked against that of a DRL-based controller implemented without any pre-training (i.e., online DRL), as well as against controllers based on traditional strategies such as RBC and PI.

Based on the previous considerations, the novelties of this paper can be summarised as follows:

- An online TL procedure was implemented to transfer a DRL controller pre-trained in simulation and later applied to the real source office. To the best of the authors' knowledge, the real implementation of TL for DRL controllers has not been explored in the literature before.
- An implementation framework ensuring a high level of interoperability with the building and its actuation and measurement systems has been developed to ensure the correct and continuous operation of the energy systems within the building. Furthermore, this methodology has been designed to be easily scalable across different buildings. The DRL controller implemented in the source office and transferred to the target office uses a limited number of variables easily measurable in real buildings.
- A benchmarking framework has been developed to compare the performance of the real controller implemented in the source office and the controller transferred to the target office with the performance of controllers belonging to three different families: RBC, PI and online DRL. In this case, a digital twin was developed for each office to implement the forementioned benchmark controllers and replicate the behaviour of the real implemented DRL-based controllers.

The paper follows this structure: Section 2 provides information about the case study, while Section 3 presents the methodological framework employed for training, implementing and transferring the DRL controller. Section 4 elaborates on the implementation details, covering digital twin development and building controller implementation. Results are discussed in Section 5, while Sections 6 and 7 delve into the implications of the research findings and suggest future directions.

2. Case study and control problem

The Next Evolution in Sustainable building Technologies (NEST) building, operated by the Swiss Federal Laboratories for Materials Science and Technology (EMPA), stands as a versatile research and innovation hub situated in Dübendorf, Switzerland [41]. It brings together collaborators from academic institutions, industry and the public sector. The architecture of NEST includes a main backbone supporting three platforms that host a variety of research and innovation modules.

Fig. 1 shows the HiLo (High Performance – Low Emissions) unit [42], a cutting-edge research space dedicated to the advancement and evaluation of building technologies. HiLo is a living laboratory enriched with an extensive sensor network, ensuring an ideal setting for the implementation of controllers.

The HiLo unit is built on two floors. The lower floor is divided into two similar office spaces, while an open-space area is on the upper floor. In this study, the two offices on the lower level are considered: *Office1* and *Office2*, both depicted in Fig. 1. *Office1* is positioned on the southwest corner, spanning an area of 22.94 m^2, while *Office2* lays on the southeast side with an area of 22.08 m^2. Both offices feature three different HVAC systems: a mechanical ventilation with heat recovery, a Hybrid Variable Refrigerant Flow (HVRF) system and a ceiling-mounted

Fig. 1. Case study representation.

integrated TABS. In this work, only the latter HVAC system, i.e., the TABS is considered, being the only one with full control access. The main difference between the HVAC systems of the two offices is that the ventilation system in *Office2* is integrated into the TABS. In addition, contrary to the conventional passive shading system in *Office1*, *Office2* features an adaptive solar facade that dynamically adjusts to modulate solar radiation, thereby facilitating local energy generation, passive heating and shading, and enhancing daylight penetration [43].

The data collected by the onsite sensors is stored in an MS-SQL database. In this study, only the indoor and outdoor temperature sensors, solar radiation sensors and occupancy detectors, a subset of the available data points, are used, reproducing typical data access constraints found in conventional buildings, as described in [44]. *Office1* is a single-person office which is occasionally used as a meeting space, while *Office2* is normally used by two people. Generally, both offices are occupied during working hours, from Monday to Saturday between 7:00 and 21:00. During these time ranges, the temperature is kept between 21°C and 23 °C in *Office1* and between 23°C and 25 °C in *Office2*. The temperature acceptability ranges of the two offices differed for two reasons concerning the indoor temperature preferences of the occupants and the system requirements. As defined above, in *Office2* the ventilation system is integrated with the TABS. This increases the temperature of the air supplied by the ventilation system to ensure indoor air quality from the CO_2 concentration perspective, compared to the temperature of the air supplied in *Office1* by the ventilation system. Therefore, the temperature acceptability range in *Office2* is defined as [23, 25] °C.

The valve is regulated at five-minute intervals. This control directly influenced the water flow rate through the TABS and the thermal power output in each office. Notably, the valve operated using a changeover mechanism, with the inlet temperature maintained constant and heated by the central heating water supply.

3. Methodology

This section delineates the methodological framework of this paper, structured into four stages as shown in Fig. 2.

3.1. Pre-training of DRL source controller in the HiLo digital twin

In the initial phase of the methodology, the DRL controller in the source office zone is pre-trained in simulation, but differently from [44] it is employed a digital twin based on a detailed Resistance-Capacitance (RC) model and developed in Modelica. The DRL agent is pre-trained to manage the opening percentage of the valve in the TABS circuit for the source office, employing a co-simulation environment integrating Modelica and Python that facilitated interaction between the controller and the digital twin. Throughout the pre-training process of the source DRL controller, an automated procedure is conducted to identify the best configuration of hyperparameters for the DRL controller. At this stage, the source DRL controller performance is compared with that of the baseline RBC controller implemented in the real building. Details about this methodological step can be found in Section 4.3.

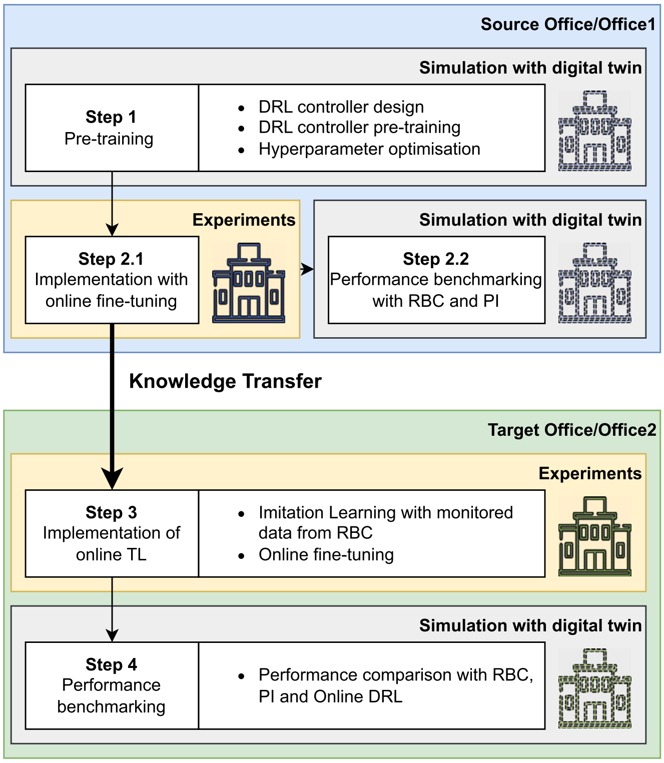
Source Office/Office1
Simulation with digital twin
Step 1
Pre-training
DRL controller design
DRL controller pre-training
Hyperparameter optimisation
Experiments
Step 2.1
Implementation with online fine-tuning
Simulation with digital twin
Step 2.2
Performance benchmarking with RBC and PI
Knowledge Transfer
Target Office/Office2
Experiments
Step 3
Implementation of online TL
Imitation Learning with monitored data from RBC
Online fine-tuning
Simulation with digital twin
Step 4
Performance benchmarking
Performance comparison with RBC, PI and Online DRL

Fig. 2. Methodological framework adopted in this work.

3.2. Implementation of DRL controller in source office

The second phase of the methodology involves the implementation of the pre-trained DRL controller in the source office.

The DRL controller is implemented in the source office, continually learning from ongoing interaction with the real building and iteratively updating and refining its control policy.

As in [44], the implementation adheres to the infrastructure provided by NEST, ensuring effective communication and control for the HVAC system of the building, including a fail-safe mechanism to ensure continuous and correct operation of the HVAC system.

Thus, a performance benchmark for the DRL controller is carried out by using the digital twin developed in Modelica for the source office with the RBCs and PI controllers.

3.3. Implementation of online transfer learning in target office

The third stage of the methodological framework entails the implementation of the online TL strategy, where the DRL control policy implemented in the source office is transferred to the target office. The online TL strategy includes imitation learning, weight-initialisation and fine-tuning.

The implementation of online TL in the target office integrates the same fail-safe mechanism developed for the source office. Further details about the online TL implementation are provided in Section 4.4.1.

3.4. Performance benchmarking of online transfer learning in target office digital twin

The last stage of the methodology benchmarks the performance of the online TL strategy against an online DRL controller, three PI controllers with different temperature setpoints, as well as two different RBCs. The benchmark for online TL with RBCs is provided to compare the performance of the former with conventional controllers commonly used in real buildings. Specifically, two different RBCs are evaluated, one of which emulated the operation of the baseline controller implemented in the real building. The benchmark controllers are implemented using the digital twin developed in Modelica for the target office, considering the same real-world conditions as those in which the online TL controller operated.

Further details on online DRL control strategy can be found in Section 4.4.

4. Implementation

4.1. Digital twins development

A detailed white-box model of the controlled system was developed to train the DRL controller and benchmark the performance of different controllers after the real implementation. The model was developed in Modelica [45] by using the Buildings library [46]. The model comprises both the dynamic characteristics of the adaptive solar facade and a detailed description of the HVAC system. The facade model was made of lumped parameter elements: a sub-model computed the mass and energy balance in the zone air volume, while facade opaque elements, namely the ceiling, the floor and the vertical partitions separating the zone with the adjacent zones and the external environment, were modelled as a series of thermal resistances and capacities, each accounting for a construction layer. The model included thermal bridges and a detailed description of the thermal gains due to the transparent envelope elements. The TABS model accounted for its water mass content, the thermal capacity and resistance of the concrete layers in which pipes were embedded, and the pressure drop of the pipes themselves, thus providing a realistic estimation of the heat exchange with the zone.

Model parameter values were inferred from available building descriptions and drawings; then, some of those parameters were selected for the calibration process, in which their value was fine-tuned to match the profile of the selected output variables with the actual measured profiles. The output variables of choice were the zone air temperature and the return water temperature of the TABS to ensure that the model properly described both the zone and TABS thermal dynamics. Internal gains due to people occupancy were estimated combining the available monitored occupancy fraction with properly selected values of convective and radiant gain per person, according to the expected office activity carried out by the occupants. As far as non HVAC-related appliances, a realistic usage schedule was inferred from the occupancy pattern.

The parameters selected to be tuned were the TABS, the internal air volume thermal capacities and the thermal resistance accounting for the thermal bridges of the building facade. In particular, the Modelica emulator modelled the internal air volume as lumped object on which to apply mass and energy balance equations. While the air mass was kept equal to the value estimated from the building geometry, a multiplier for the thermal capacity was introduced for the tuning process. These parameters were selected as being the most directly influencing the dynamic response of the building, and were tuned through a trial and error process. Fine-tuning these parameters allowed to adjust the time constant of both the building and the TABS, thus better fitting the output variables profile with the sensor measurements. In particular, the final value of the multiplying factor for scaling the sensible thermal mass of the volume was equal to 3. This value is often significantly larger than 1, as it has to account for internal thermal masses [47].

Table 1
Metrics for calibrated models of the two offices.

Zone	Indoor temperature RMSE [°C]	Energy consumption MAPE [%]
Office1	0.62	9.9
Office2	0.65	9.2

The calibrated model showed satisfactory performance, as can be seen in Fig. 3 for *Office1* for the period 15 to 28 November 2023. The upper plot compares the measured zone temperature to the simulation result. The second and third subplots compare respectively the thermal power delivered by the TABS and the return water temperature resulting from the simulation to that measured on the field. It must be noted that from the comparison between the real and simulated measurements for the thermal power delivered by the TABS that the two profiles were similar. Moreover, the return temperature reading was meaningful only when the heating system was in *ON* mode; indeed, the model performance with respect to the heating system dynamics was evaluated on the prediction error of the heating power consumption, which was null when the system is *OFF*. The lower plot shows the boundary conditions profiles during the considered time window, those variables being the outdoor air temperature, the adjacent zone temperature and the global horizontal solar radiation. Moreover, Table 1 shows the metrics employed to evaluate the performance of the model calibration process of the two offices. In detail, Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) are used to evaluate the performance related to indoor temperature calibration and energy consumption, respectively. The energy consumption was closely related to the return temperature, as the supply temperature and flow rate measured in the real field were used as inputs. The response of the models of both offices is acceptable, showing performance in the range of 0.6 °C of RMSE for indoor temperature and MAPE between 9 and 10% for energy consumption.

The Modelica model was exported as a Functional Mock-up Unit (FMU), according to the Functional Mock-up Interface (FMI) standard [48]. The standard allows to interface models built with different software tools with each other and with programming language code. In the present case, the model FMU was interfaced with a primary code written in the Python language, in order to allow control signals to be sent to the emulator, which would in turn send feedback measurements back to the primary code. The FMI standard allows such an interaction to be performed at each simulation timestep, enabling the simultaneous interaction between the emulator and the controller.

4.2. DRL controller design

This section describes the main features of the implemented DRL controller (i.e., state-space, action-space and reward function).

The formulation of the action-space and the reward function is the same as in the recent work developed by Silvestri et al. [44] for the same building, while the state space is modified to provide an improved version ensuring better performance for DRL controller.

The DRL controller was developed by employing the latest version of the Stable-Baselines library [49].

Since the SAC was employed as DRL algorithm, the action-space is continuous and defined as:

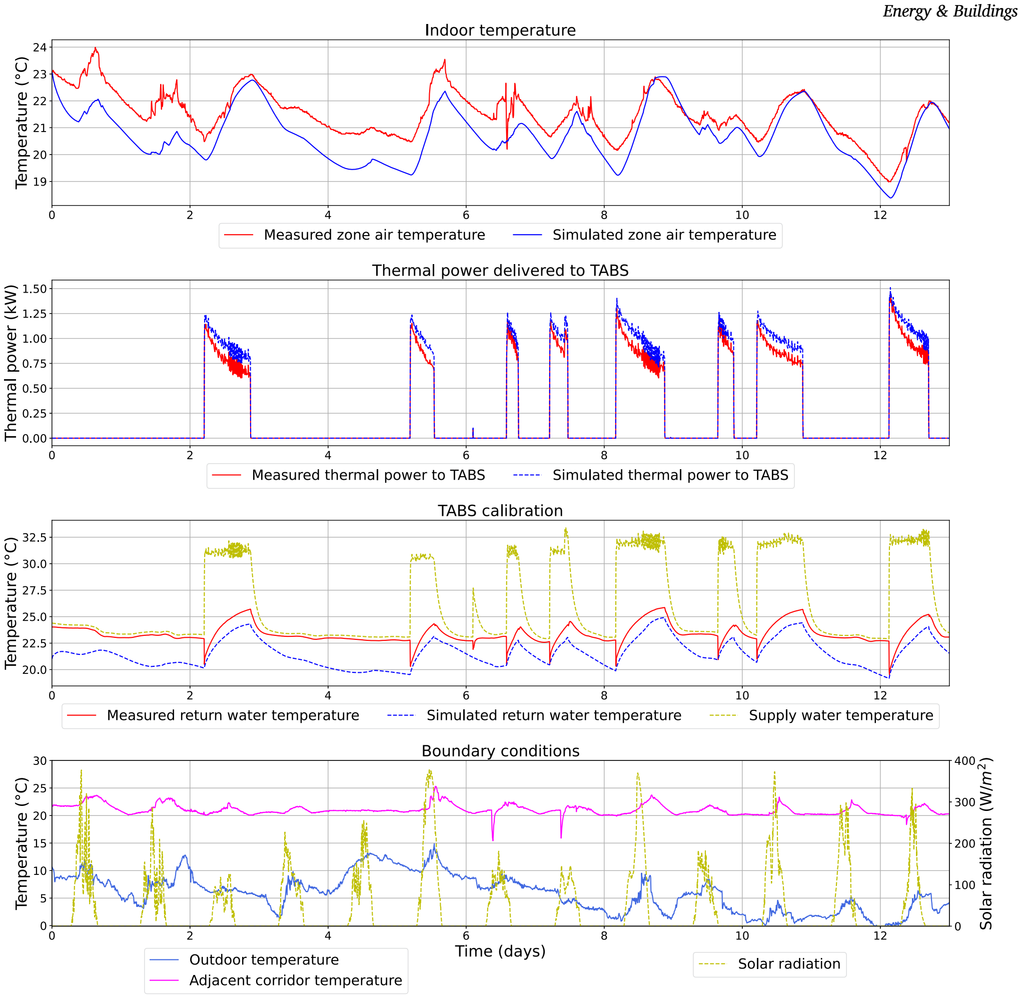

Fig. 3. Comparison for *Office1* of real and simulated indoor temperature, TABS thermal power and TABS return temperature after model calibration. (For interpretation of the colours in the figure(s), the reader is referred to the web version of this article.)

$$A : 0 \le u_t \le 1 \tag{1}$$

During each 5-minutes control time step t, the valve opening percentage u_t is chosen by the agent. This feature is proportional to the fraction of the nominal heating power $\dot{Q}_{\mathrm{tabs}}$ supplied by the TABS. When the control action u_t chosen by the agent is lower than 0.1, its value is set to 0, according to the operational characteristics of the TABS.

The state-space of the DRL agent included 23 variables that were measured in the case study building and defined in Table 2 with a detail on the evaluation time step and on their corresponding lower and upper bounds employed to re-scale the variables through min-max normalisation.

Similar to [44], *Outdoor Air Temperature* T_{o} and *Global Solar Radiation* $\dot{Q}_{\mathrm{sol}}$ were included within the state space as they are fundamental due to their influence on building heating energy consumption and indoor temperature. These variables were evaluated on the current time step t and for the previous 15 and 30 minutes. Moreover, hourly outdoor temperature predictions were provided for the next 6 hours. Differently from [44], the power supplied by the TABS to the environment $\dot{Q}_{\mathrm{tabs}}$ was included in the state-space considering only two historical measurements, 15 and 30 minutes before, as it was proportional to the control action u_t and this could lead to possible overfitting for the DRL controller. To ensure an adaptive definition, the information about *Indoor temperature* did not consider its absolute value but the temperature difference relative to the two temperature limits of the acceptability temperature range (i.e., $T_{\mathrm{i}} - \overline{T_{\mathrm{i}}}$ and $T_{\mathrm{i}} - \underline{T_{\mathrm{i}}}$). By combining these two variables, the DRL agent comprehended the status of indoor temperature compared to the temperature acceptability range. As per the other variables discussed so far, information concerning indoor air temperature was integrated into the state space at the present control time step t, along with two lagged values (15 minutes and 30 minutes before), enabling the evaluation of temperature evolution within the building over time and considering the impact of building thermal dynamics [50]. This approach augments the capacity of DRL controller to extract the dynamics of the environment from the state-space. To conclude, information regarding the presence of occupants was expressed through three distinct variables: the occupancy fraction over each five-minute control time step f_{occ}, time to occupancy start and time to occupancy end. f_{occ} was integrated to inform the DRL agent about the presence of the occupants, thereby indicating potential additional indoor gains that could reduce the energy demand for TABS. The remaining two variables denote the time until the next occupancy pattern change. When the building was unoccupied, *time to occupancy start* indicated the number of time steps remaining before the arrival time of occupants (equal to zero when the building is occupied). Conversely, during occupied periods, *time to occupancy end* denoted the number of time steps until the departure time of occupants (equal to zero during off-occupancy periods).

In this paper, the reward function was defined as the combination of two opposing terms, energy-related r_{E} and temperature-related r_{T}, weighted by employing the reward factor $\beta > 0$ and employing a reward scaling factor θ, similar to [44]. The reward formulation r follows a similar structure to the one employed in [44] and it is defined as follows:

$$r = -\theta \cdot (r_{\mathrm{E}} + \beta \cdot r_{\mathrm{T}}) \tag{2}$$

Table 2
Variables included in the state-space.

Variable	Min value	Max value	Unit	Timestep
T_o	261.15	293.15	K	t-30min, t-15min, t, t+1h, ..., t+6h
$\dot{Q}_{sol}$	0	800	$W/(m^2 \cdot K)$	t-30min, t-15min, t
$\dot{Q}_{tabs}$	0	0.9	kW	t-30min, t-15min
$T_i - \overline{T_i}$	-5	5	°C	t-30min, t-15min, t
$T_i - \underline{T_i}$	-5	5	°C	t-30min, t-15min, t
Time to occupancy start	0	407	-	t
Time to occupancy end	0	169	-	t
Occupancy fraction f_{occ}	0	1	-	t

The energy-related term r_E was equal to E_{tabs} [kWh], representing the energy consumption attributed to the operation of the TABS, which was directly proportional to the control action u_t. The temperature-related term r_T quantified the squared deviation of the zone temperature from the desired temperature limits, and its formulation depended on the presence of occupants during occupancy hours, indicated by b_{occ}:

$$r_T = \begin{cases} 0 & \text{if } b_{occ} = 0 \\ \max(0, T_i - \overline{T_i})^2 + \max(0, \underline{T_i} - T_i)^2 & \text{if } b_{occ} = 1 \end{cases} \tag{3}$$

During non-working hours, the value of r_T was equal to 0.

4.3. Setup of DRL controller pre-training process

This section delves into the setup of the DRL controller during the pre-training phase carried out by using the digital twin developed in Modelica to emulate the building dynamics. In this study, the latest version of the SAC algorithm developed in the Stable-Baselines library [49] was employed. The performance of DRL controllers was influenced by numerous hyperparameters that required adequate tuning [51]. Thus, an automated hyperparameter optimisation procedure is executed using the Python library Optuna [52]. The Tree-structured Parzen Estimator (TPE) algorithm [53] was chosen as the sampling optimisation method for Optuna. The optimisation process was carried out to retrieve the optimal configuration of hyperparameters that ensured the most favourable balance between decreasing the average daily energy consumption associated with TABS operation $\overline{E_{tabs}}$ and improving indoor temperature control for the DRL agent. The performance in terms of temperature control was evaluated by assessing the mean value of the daily average temperature violation rate $\overline{T_{viol,daily}}$, computed as:

$$\overline{T_{viol,daily}} = \frac{T_{viol,daily}}{n_{viol,occ,daily}} \tag{4}$$

where $T_{viol,daily}$ was the cumulative daily sum of temperature violations and $n_{viol,occ,daily}$ is the daily temperature violations occurrences. The cumulative daily sum of temperature violations $T_{viol,daily}$ [°C] was computed per day as defined in Equation (5), during the simulated testing phase $t \in [0, t_N]$, t_N corresponds to the number of daily timesteps (i.e., 288 steps).

$$T_{viol,daily} = \sum_{t=0}^{t_N} b_{occ,t} \cdot T_{viol,t} \tag{5}$$

where $b_{occ,t}$ was a Boolean variable equal to 1 when the thermal zone was occupied, while $T_{viol,t}$ indicated the temperature violation calculated per each control time step as follows:

$$T_{viol,t} = \begin{cases} \underline{T_i} - T_i & \text{if } T_i < \underline{T_i} \\ 0 & \text{if } \underline{T_i} \le T_i \le \overline{T_i} \\ T_i - \overline{T_i} & \text{if } T_i > \overline{T_i} \end{cases} \tag{6}$$

$\overline{T_i}$ and $\underline{T_i}$ represented respectively the upper and lower bounds of the temperature acceptability range.

Considering the multi-objective nature of the hyperparameter optimisation problem, multiple solutions exist in the optimal Pareto-front [54]. Therefore, designating a criterion for selecting the best solution among these optimal choices became necessary. The criterion adopted in this study was based on the minimum distance from the point with coordinates corresponding to the minimum values of both objective function terms, defined as the ideal point [55]. In this framework, the Euclidean distance between ideal and Pareto front points was calculated considering a plane with coordinates $[\overline{E_{tabs}}, \overline{T_{viol,daily}}]$. In the Euclidean distance calculation, no specific weighting was applied, since each term was considered equally.

Twenty agents trained for 20 episodes, where each episode corresponds to 90 days, from 1 December 2023 to 28 February 2024, were considered during the hyperparameters optimisation procedure.

Ideally, all possible combinations of the hyperparameters would be evaluated to identify the optimal configuration. However, this approach would be computationally prohibitive. Thus, twenty trials were selected to adequately explore the hyperparameter space while balancing computational costs. This approach aligns with the methodology used in [39] who adopted a similar number of trials in a comparable context.

The performance achieved per each trial at the end of the training phase by each set of DRL hyperparameters was computed by considering the Euclidean distance between the DRL performance the ideal point. The best set of hyperparameters was retrieved by considering the minimum Euclidean distance and by comparing the performance of each set against that of the $RBC1$ baseline controller implemented in the real *Office1* in HiLo, described in Section 4.4.2. Comparing each set of hyperparameters against the performance of $RBC1$ was conducted to ensure that each chosen configuration demonstrated an improvement over the baseline control strategy. This validation step was necessary because minimising Euclidean distance alone does not guarantee that each configuration performs better than the baseline controller. The values and range of DRL controller hyperparameters considered during the optimisation process are included in Table C.7 in Appendix C.

4.4. Benchmark control strategies

This section discusses the implementation details of the online TL and of the control strategies employed to benchmark the performance of DRL and online TL controllers.

4.4.1. DRL-based control strategies

The developed online TL strategy involves weight-initialisation, fine-tuning and imitation learning. Weight-initialisation enables knowledge transfer by sharing between the source and target DRL controllers the neural network parameters. Initially, the pre-trained source agent weights of Actor and Critic networks are employed to initialise those of target controllers. Afterwards, the neural network weights are updated during a fine-tuning process that enables the adjustment of the agent to the unknown conditions specific to the target office (e.g. variations in indoor temperature preferences of occupants). Before the fine-tuning process started, an IL phase take place, where transitions obtained from the RBC operation, implemented in the real office 2 for 14 days from 15 November to 28 November 2023, are employed to initialise the memory buffer of the online TL agent. This approach has proven effective in improving the ability of the online TL agent to learn the action-space-reward function relationship during the initial days of online TL real implementation, as the control problem is the same in source and target domains as well as due to the adaptive definition of the state space of the DRL controllers, as defined in Section 4.2.

In addition to the online TL strategy developed in this paper, another DRL-based control strategy was considered to evaluate online TL performance, i.e., online DRL controller. The online DRL control strategy involves the control agent learning the optimal policy while actively managing the system without prior knowledge of its dynamics [25]. The main advantage of the online DRL strategy is its model-free nature, which eliminates the need for a surrogate model of the environment to be controlled. However, during the early training stages, the limited knowledge of the agent increases the risk of suboptimal performance. In this context, the IL approach is employed as in online TL to initialise the memory buffer of the online DRL strategy. The online TL controller was implemented in real-time on *Office2*, while the online DRL agent was implemented in simulation employing the digital twin of the *Office2*. The automated optimisation of hyperparameters was not conducted for the DRL-based controllers, as it necessitated training the controllers over multiple episodes, such as in the source DRL agent [39]. However, this approach contrasted with the online DRL and online TL strategies developed in this study, which were implemented respectively for only one episode (aiming to represent direct implementation in a real building) and directly in the real office. Therefore, the hyperparameters θ and β were the same as those optimised in the source DRL controller for DRL-based controllers implemented in the target office. However, the values of *Batch size*, *Gradient steps*, *Train frequency* and the starting value of Boltzmann temperature coefficient α were modified.

Reducing the batch size to 64 for both online strategies was motivated by the limited data amount available for training the controllers in an online fashion and expediting the convergence process towards an optimal solution [56]. Increasing the number of gradient steps to 5 provided significant advantages to the controllers, speeding up the training process during the initial weeks of implementation when the agent had a limited amount of transition data stored in the memory buffer for proper training [39]. The initial value of α was reduced to 0.05 so that the agent would explore more carefully and exploit the control actions from the source DRL control policy [57].

Moreover, while initially impacting the performance of the DRL-based control strategies, a training frequency of 288 control time steps ensured that the control agent accumulated a larger number of transitions before proceeding to the next learning step, thus enhancing performance throughout the training period. The combination of increasing gradient steps to 5, reducing batch size to 64, and lowering learning rate to 0.00025 compared to the source controller safeguards against complete overwriting of the pre-trained source control strategy. Adjusting the learning rate enabled the optimisation of the control policy according to the varying boundary conditions in the target building while mitigating excessive exploration of the action space, which could lead to deviations from the optimal control policy learned during the initial training phase [58]. Hence, utilising a lower learning rate ensured that the prior knowledge from the source office was not completely discarded. Therefore, the learning rate μ was reduced for online TL by half compared to the value of the same parameter for the source DRL controller (i.e., 0.0005) as in [24] that demonstrated the effectiveness of the learning rate reduction by computing the Mahalanobis distance [59].

However, for the online DRL controller, the initial value of α and the value of learning rate μ were the same as source DRL controller since this control strategy was implemented without pre-training, and it was required to encourage exploration due to the absence of a pre-trained policy as in the case of online TL.

4.4.2. Traditional controllers

Similarly to the previous work by Silvestri et al. [44], two different types of traditional controllers were evaluated during the benchmarking phase: RBC and PI controllers.

Table 3
Time and indoor temperature conditions in RBCs for the pre-switch ON phase.

Combination	Time period	Indoor temperature
1	$t_{start} - 6 \leq t < t_{start} - 5$	$\underline{T_i} - T_i \geq 1\,°C$
2	$t_{start} - 5 \leq t < t_{start} - 3$	$\underline{T_i} - T_i \geq 0.5\,°C$
3	$t \geq t_{start} - 3$	$\underline{T_i} - T_i \geq 0\,°C$

Two different RBCs were used to evaluate the performance of the DRL controller implemented in the source office and the online TL controller in the target office. These RBCs operate on weekdays during occupied periods and according to an On-Off control strategy, fully opening or closing the valve based on indoor temperature and time of day conditions outlined in Table 3. These combinations were determined through a sensitivity analysis, during which various thresholds were tested to minimise temperature violations in the initial stages of the occupancy period [44]. The conditions related to the time period refer to the time of day, indicated as t in Table 3. The difference between t_{start} and the values listed under each time period combination is measured in hours. Therefore, the values reported in Table 3 (i.e., 6, 5 and 3) represent the time of the day in hours.

Following the pre-switch ON phase lasting at t_{start} = 7:00, the two RBCs managed the TABS to open the valve to supply heating power to the thermal zone. The two RBCs differed at this stage in the temperature limits below/above which the valve was opened or closed. In detail, the $RBC1$ represented the baseline RBC implemented in the real building, where the controller fully opened the valve to supply heating energy from TABS if the T_i fell below the lower-temperature acceptability threshold $\underline{T_i} = 21\,°C$ for *Office1* and $\underline{T_i} = 23\,°C$ for *Office2*. Conversely, if T_i exceeded the upper-temperature acceptability threshold $\overline{T_i} = 23\,°C$ for *Office1* or $\overline{T_i} = 25\,°C$ for *Office2*, $RBC1$ fully closed the valve to stop the heating energy supply from TABS.

A second implementation of the RBC, denoted as $RBC2$, was introduced to compare the performance of online TL against that of a more refined controller. $RBC2$ followed the same logic as $RBC1$, except its operation range, which has been restricted between [21, 22]°C for *Office1* and [23, 24]°C for *Office2*. This adjustment was made after observing that $RBC1$ often exceeded the temperature upper bound due to the thermal inertia of the TABS.

RBCs operated until occupants left the building at t_{end} = 21:00 or on Sundays when the TABS system was switched OFF.

The PI controllers operate according to the following relationship between the control output $u(t)$ and the reference error $e(t)$:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)\, d\tau \tag{7}$$

In this work, K_p and K_i were fine-tuned by employing a PID Tuner in MATLAB, configured for balanced performance. The values of the two constants resulting from the tuning process are respectively equal to $K_p = 0.3\ \frac{1}{°C}$ and $K_i = 1 \cdot 10^{-4}\ \frac{1}{s \cdot °C}$. The unit of measure of K_p and K_i were obtained considering that the PI controller operates respectively on the proportional and integral value of the error $e(t)$, measured in °C, while $u(t)$ is expressed as percentage. Moreover, the control action $u(t)$ was limited within the range [0,1] to be comparable with other controllers' action space, defined in (1). To prevent integral windup, an anti-windup scheme was integrated into the PI controller [60]. The PI controller was set to track an indoor temperature setpoint, and operated from Monday to Saturday from 1:00 to 21:00. The choice of 1:00 as the start time for the PI controllers was made to ensure consistency with the operation of RBC strategies, providing a fair comparison between the traditional controllers. However, details about the results of each tested start time are included in Table D.9 in Appendix D.

Different temperature setpoints and starting times were tested to find the PI controller with the best performance in terms of energy consump-

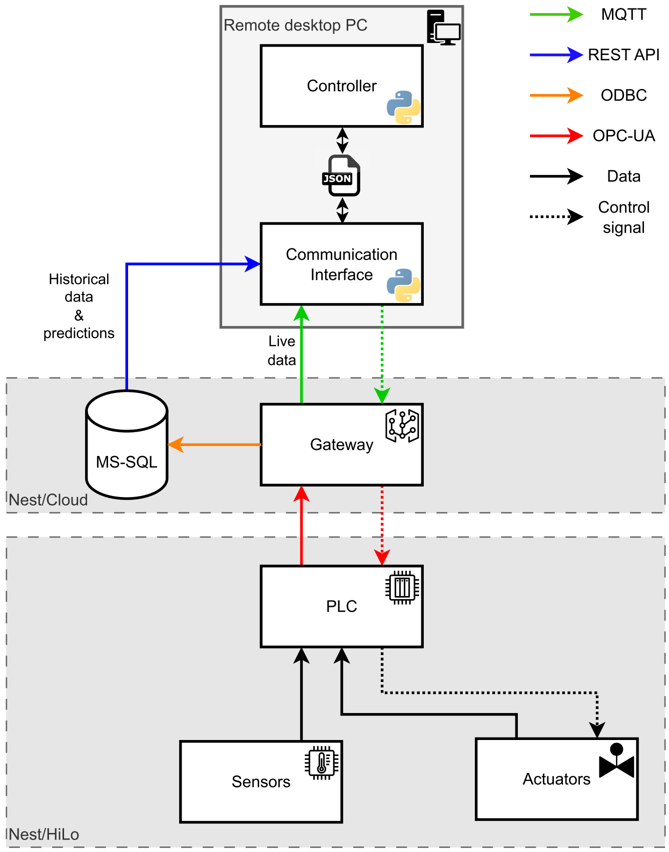

Fig. 4. The control systems infrastructure where the dotted lines represent the control signals, while the solid lines represent the data signals, adapted from [44].

tion and temperature violations. For *Office1* and *Office2*, three different indoor temperature setpoints were tested, considering values placed inside the acceptable temperature ranges per each office: 21 °C, 21.5 °C and 22 °C for *Office1*, 23 °C, 23.5 °C and 24 °C for *Office2*.

4.5. Real-world implementation

This section outlines the implementation of the controllers in the real systems, adhering to the methodology detailed in Section 3.

Fig. 4 shows the flow of data and signals through the communication infrastructure in NEST.

In the foundation tier of the system, the Programmable Logic Controller (PLC) situated in HiLo was responsible for gathering sensor data and dispatching control signals to actuators via various protocols, including Modbus RTU RS485 and conventional analog/digital signals (0-10V, 4-20mA, PT1000, DI/DO). This PLC communicated with a gateway housed on a virtual machine through the versatile, open-source OPC-UA protocol. Subsequently, this gateway transmitted the collected data to a MS-SQL historical database via Open DataBase Connectivity (ODBC). This database is hosted on a virtual machine within the NEST cloud and is accessible remotely through a Python-integrated REST API. Real-time data and control signals were reciprocally exchanged between the remote client and the gateway server utilising the MQTT protocol.

The remote PC, equipped with a 4-core CPU operating at 3.40 GHz and 16GB of RAM, ran two Python scripts in parallel within a Python virtual environment. The control logic was implemented in the *Controller* script, which read data and wrote control signals to a JSON file. The *Communication interface* script handled the transmission of information between the JSON file and the system. This structure separated the tasks of controlling and communicating, ensuring that communication remained active even if an error occurred in the *Controller* script.

Table 4
Average daily energy consumption and mean daily average temperature violation rate for DRL and benchmark controllers for *Office1* during pre-training phase.

Controller	$\overline{E_{\text{tabs}}}$ [kWh]	$\overline{T_{\text{viol,daily}}}$ [°C]
$RBC1$	7.28	0.139
$RBC2$	8.21	0.042
$PI21$	6.26	0.253
$PI21.5$	7.41	0.045
$PI22$	8.46	0.125
DRL	5.59	0.032

During the implementation, the DRL controller ran in *Office1* from 8 to 16 March 2024. Subsequently, the DRL controller was transferred to *Office2* and implemented from 3 to 26 April 2024.

5. Results

This section shows the outcomes of applying the methodological framework described in Section 3. The results are divided into two subsections, each focusing on a different office.

5.1. Pre-training and real-world implementation of DRL controller in *Office1*

As outlined in Section 4.3, the DRL controller was pre-trained offline using the digital twin developed for *Office1* before its real-world implementation. This pre-training phase involved optimising the hyperparameters of the source DRL controller, whose values are detailed in Table C.7. The results obtained for the DRL controller during the three-month pre-training phase in *Office1*, conducted using typical climatic data in Modelica for Dübendorf, and those of the benchmark controllers, are reported in Table 4.

Results in Table 4 indicate that the DRL controller outperformed the benchmark controllers implemented in *Office1* in simulation. Specifically, DRL reduced TABS average daily energy consumption between 23% and 32% compared to RBCs and between 11% and 34% in relation to PI controllers. In addition, the DRL provided better control of indoor temperature conditions, since it decreased $\overline{T_{\text{viol,daily}}}$ between 24% and 77% in comparison to RBCs and between 29% and 87% relative to PI controllers. Moreover, to provide a better overview of the performance for the pre-trained DRL controller, Table E.10 in Appendix E shows the performance comparison of the DRL controller and traditional controllers over another three-month period from a different year but for the same location (i.e., Dübendorf).

Subsequently, the DRL controller was implemented in learning mode on the source office, considering an implementation period of nine days. The implementation period was limited by connection issues with the database from which the DRL controller extracted the states required for its correct operation.

Fig. 5 provides a detailed performance comparison of the DRL agent in the real environment versus its behaviour in the digital twin of *Office1*. This comparison highlights the indoor temperature and energy consumption profiles in both settings, illustrating how the controller operates under real-world conditions compared to digital twin conditions. For the digital twin, the same boundary conditions of the analysed period (8-16 March 2024) were employed, as well as the same actions selected by DRL controller real implemented.

Fig. 5 shows that the DRL controller implemented in the digital twin provided identical performance in terms of energy consumption compared to real operation, while the temperature profiles were similar, with some differences due to sporadic temperature peaks that occurred in real operation and related to certain factors that were not modelled in the digital twin (e.g. other occupants that arrive in the office, doors/windows opening). However, a performance metrics comparison between

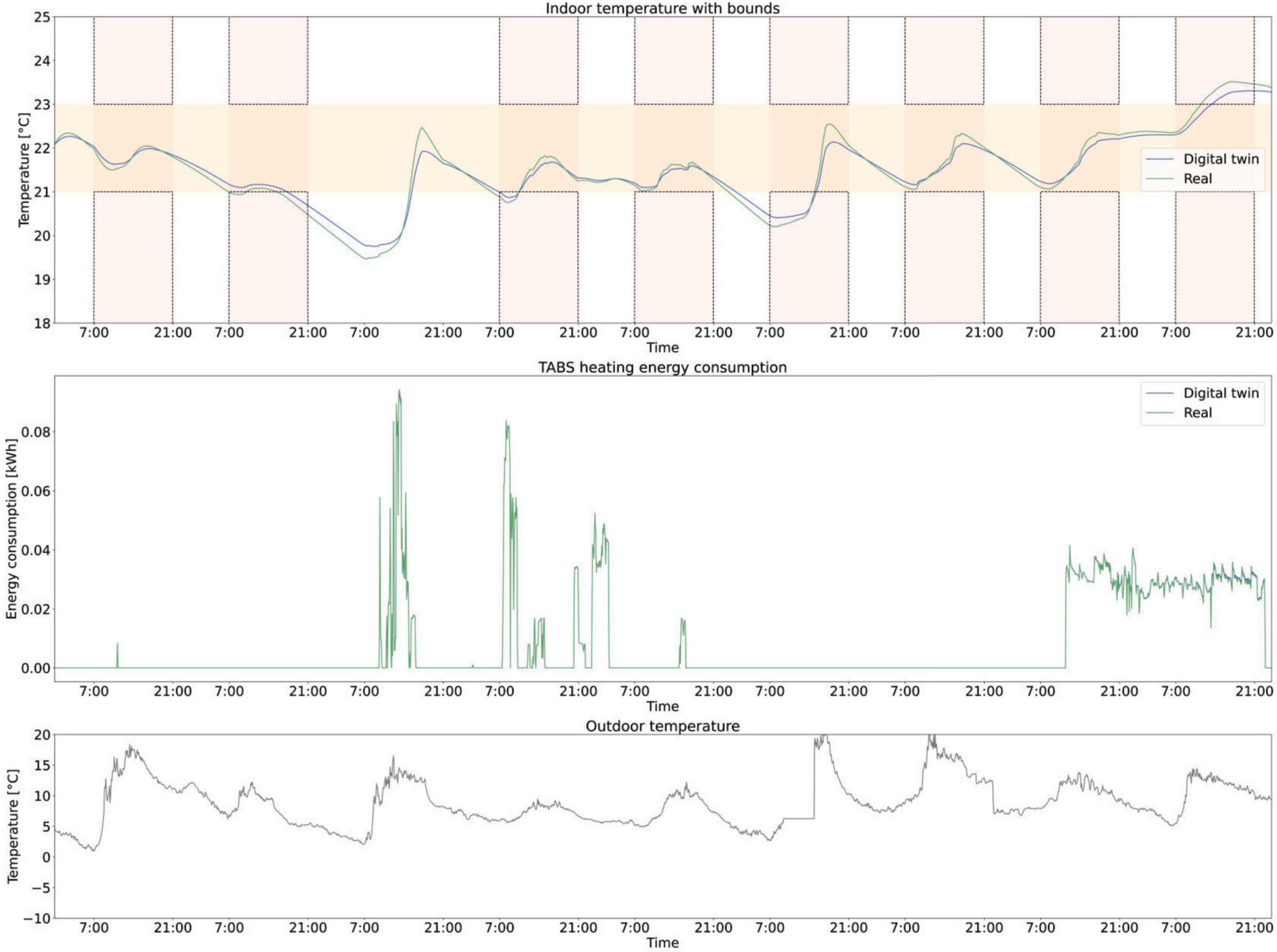

Fig. 5. Comparison of indoor temperature and energy consumption profiles between real and digital implementations for DRL in source office.

the real case and the digital twin showed that the consumption was similar ($E_{\text{tabs,real}_1} = 18.3\text{kWh}$ vs $E_{\text{tabs,twin}_1} = 18.9\text{kWh}$), while $\overline{T}_{\text{viol,daily}}$ differed by approximately 16%. The performance metrics for the comparison of the indoor temperature and energy consumption profiles of the DRL controller implemented in real building and the digital twin were respectively $RMSE_{T_\text{i}} = 0.61\,°\text{C}$ and $MAPE_{E_\text{tabs}} = 6.7\%$.

To provide a benchmark of the performance obtained by the DRL in *Office1* compared to that of RBCs and PI-based controllers, a comparison was made by implementing the controllers on the digital twin of *Office1* during the period 8-16 March 2024. The results are summarised in Table 5.

Table 5
Average daily energy consumption and mean daily average temperature violation rate for DRL and benchmark controllers implemented in *Office1* digital twin during the period 8-16 March 2024.

Controller	$\overline{E_{\text{tabs}}}$ [kWh]	$\overline{T_{\text{viol,daily}}}$ [°C]
$RBC1$	2.33	0.255
$RBC2$	2.13	0.190
$PI21$	2.04	0.182
$PI21.5$	2.37	0.182
$PI22$	3.38	0.296
DRL	2.1	0.108

As demonstrated by results in Table 5, the implementation of DRL controller in the digital twin of *Office1* led to similar TABS average daily energy consumption compared to $RBC2$ and $PI21$ and saved 10%, 11% and 38% of energy compared to $RBC1$, $PI21.5$ and $PI22$, respectively. On top of that, the DRL controller provided better control of indoor temperature conditions, reducing $\overline{T}_{\text{viol,daily}}$ up to 50% compared to RBCs and between 45% and 63% compared to PI controllers.

To conclude the description of results obtained in *Office1*, Fig. 6 compares for the period 8-16 March 2024 indoor temperature and energy consumption patterns, as well as the outdoor temperature trend, for $RBC2$, $PI21$ and DRL control strategies implemented on the digital twin.

Fig. 6 shows that DRL was the most effective algorithm for managing indoor temperature and optimising energy consumption. In detail, $RBC2$ showed wide and rapid oscillations in indoor temperature, often exceeding temperature limits. This behaviour generated high and discontinuous energy consumption peaks, which was indicative of less efficient control. $PI21$ followed a similar TABS management strategy as $RBC2$, while the DRL controller maintained the indoor temperature consistently within comfort limits with minimal variations. At the same time, the DRL controller reduced energy consumption peaks, demonstrating its capability to adapt the control policy to the boundary conditions experienced in the real building.

Although the DRL performed better than the analysed benchmark controllers, certain anomalous aspects could be highlighted. During unoccupied hours, as in the third and fifth day of the analysed period,

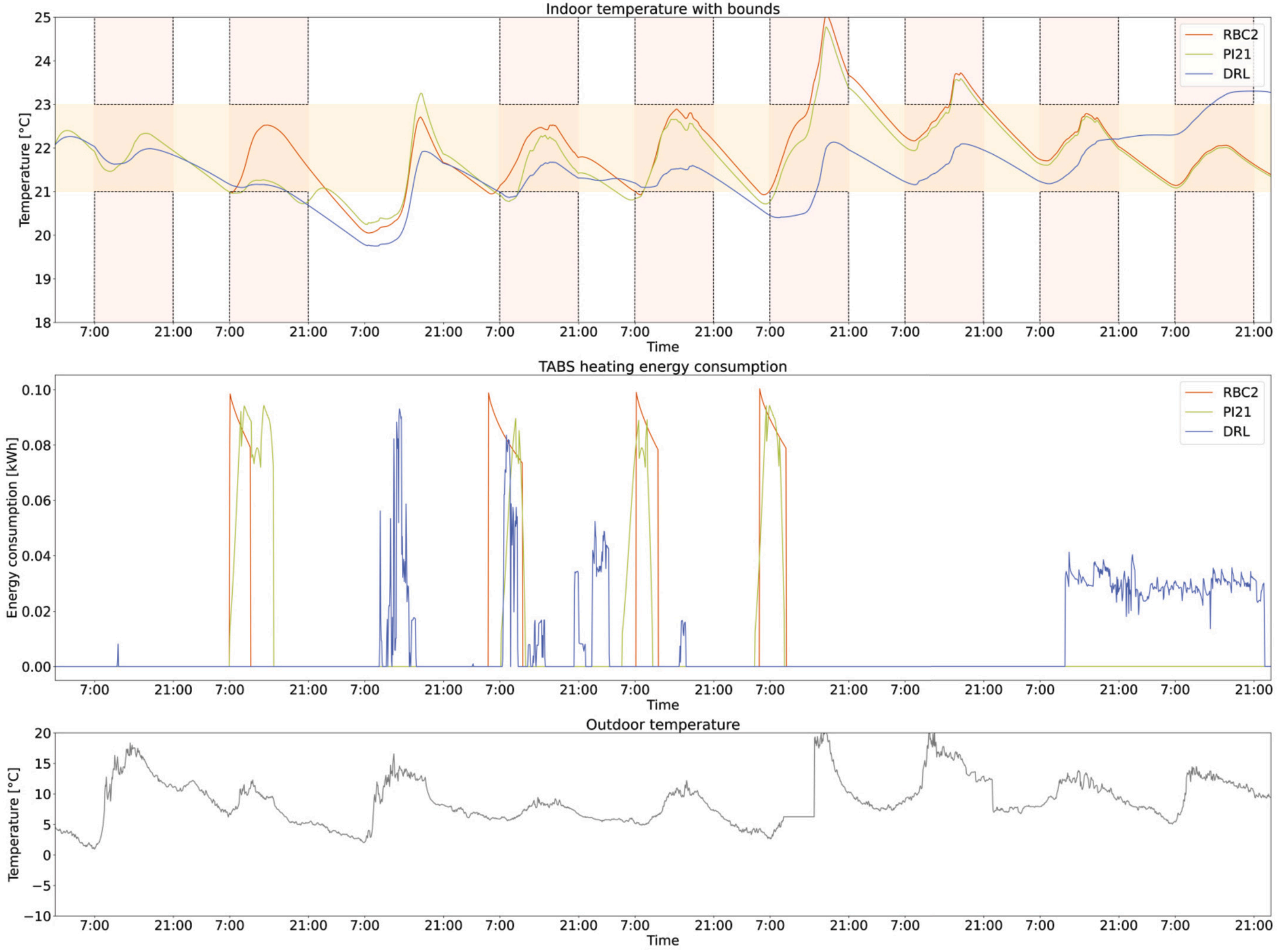

Fig. 6. Comparison of indoor temperature and energy consumption profiles between DRL, $RBC2$ and $PI21$ implemented in the *Office1* digital twin.

the DRL controller switched ON the TABS. This behaviour suggests that the DRL controller activated the TABS to offset for suboptimal decisions made the previous day (i.e. second day, where the TABS was not activated despite $T_{\rm i}$ is lower than $\underline{T_{\rm i}}$), highlighting a limitation in the long-term decision-making framework of DRL controllers. This observation aligns with the challenges encountered by DRL control agents in properly accounting for thermal inertia, in particular in highly dynamic environments such as office buildings.

Moreover, on the sixth day, the DRL controller did not switch ON the TABS although $T_{\rm i}$ was below $\underline{T_{\rm i}}$ during occupancy hours, while during the last two days of the analysed period the DRL controller failed to keep the TABS switched ON although $T_{\rm i}$ was above $\overline{T_{\rm i}}$ during occupancy hours. These behaviours could be attributed to the design of the reward structure, which prioritises long-term optimization objectives over immediate corrective actions, leading to delayed responses in some circumstances. Although the DRL controller integrated the measured indoor temperature in the state-space, it may not fully account for dynamic disturbances such as thermal exchanges with adjacent rooms or outdoor environments and air infiltration through open doors or windows. These events introduced temperature fluctuations that are challenging for the DRL controller to anticipate and mitigate effectively, given its current state-space representation. Including additional variables into the state-space (e.g. temperature of the nearest rooms or information about door and windows openings) could improve the ability of the DRL controller in refining its policy and enhance its responsiveness to such dynamic environmental changes. Moreover, this approach could help balance the limitations introduced by the reward structure, enabling the DRL controller to preemptively activate or deactivate the TABS at the appropriate time, thus minimising energy consumption while enhancing indoor temperature control.

5.2. Implementation of online transfer learning in Office2

This section summarises the results of implementing the online TL strategy in the target office (i.e., *Office2*). First, an overview of the outcomes from the real-world application of online TL is presented. Subsequently, the performance of the online TL strategy is benchmarked against RBCs, PI and online DRL approaches. These benchmark controllers were tested in the digital twin of *Office2* over the same period as the real-world implementation, from 3 April to 26 April 2024. No connection losses occurred during the implementation period of the online TL.

Fig. 7 shows the profiles from real measured data of indoor temperatures, real measured heating power provided by TABS and its supply/return water temperature, outdoor and nearest room temperatures and solar radiation over twelve days during the real implementation period of the online TL (from 3 April to 14 April 2024).

The online TL strategy aimed to minimise energy consumption, mainly during occupancy hours, by leveraging free thermal gains from occupants, appliances and solar radiation, as observed in the first two days of the analysed period. Moreover, from the third to the seventh day, the TABS was never switched on since the online TL had complete

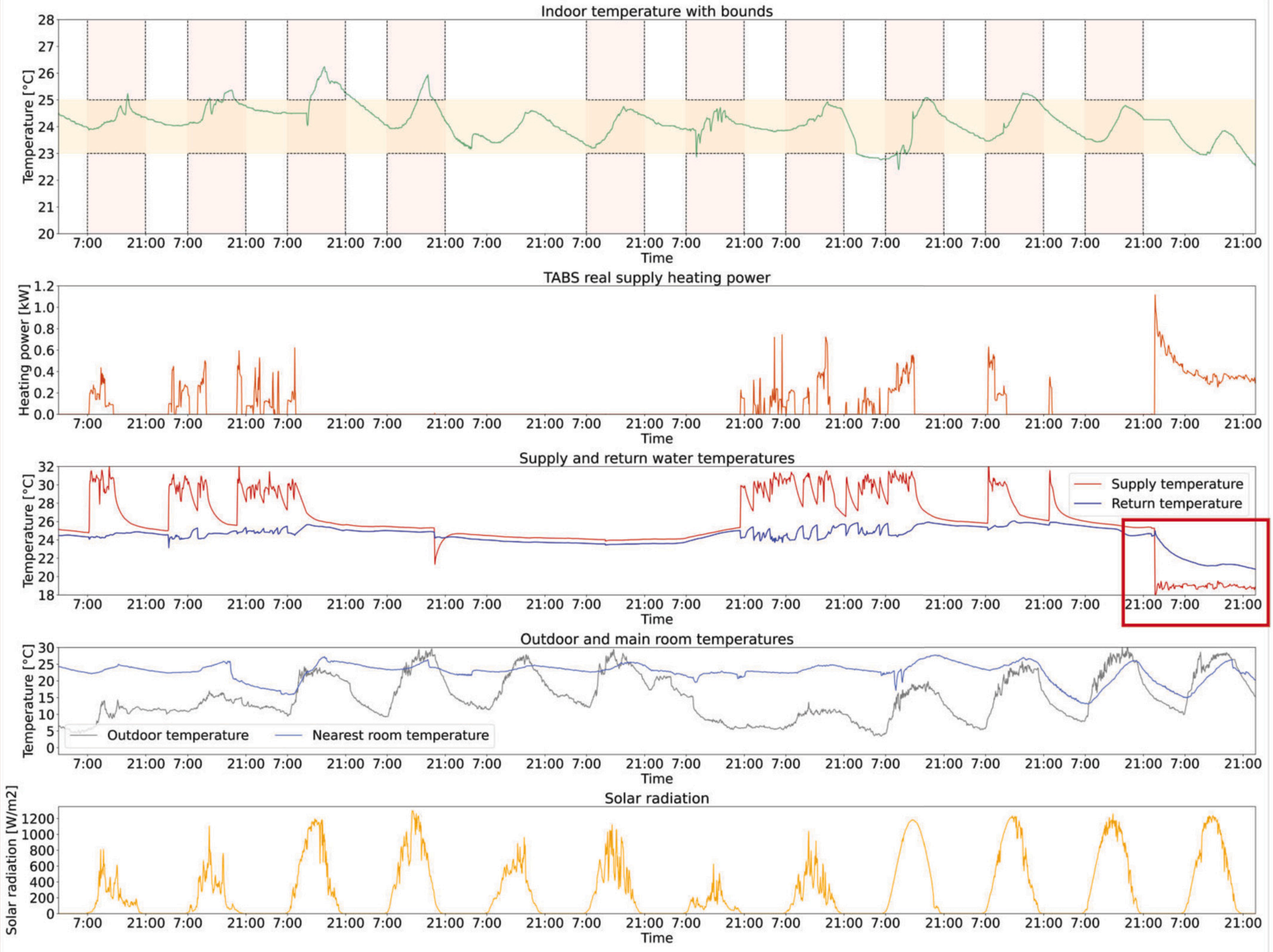

Fig. 7. Indoor temperature, heating power and TABS supply/return water temperature profiles from real implementation of online TL strategy in target office for the period 3 April - 14 April 2024.

knowledge of external climatic conditions due to the presence in the state-space of past and future information on outdoor temperature and solar radiation. In particular, during these days, climatic conditions were less severe compared to typical winter conditions (with peaks of 30 °C in some cases for T_o), influencing the building dynamics and allowing the online TL to avoid wasting energy while maintaining adequate indoor temperature conditions. However, during the last two days of the analysed period (13-14 April 2024), a technical issue emerged in the TABS system related to the presence of a threshold value on the average outdoor temperature over the past 24 hours. Specifically, during these two days, the operating mode of the TABS switched from heating to cooling mode (as can be seen from the supply and return temperature profiles in Fig. 7) due to the high value of outdoor temperature, which exceeded the limit for maintaining the TABS system in heating mode. However, from 15 April 2024, the 24-hour average of outdoor temperature was lower than the threshold limit, so the TABS system resumed its operation in heating mode. Due to this issue, 13 and 14 April 2024 had to be excluded from the analysis. As a result, the online TL implementation period was split into two parts: the first period from 3 to 12 April 2024, and the second period from 15 to 26 April 2024. This adjustment allowed for a more reliable performance benchmark for online TL when operating the TABS in heating conditions.

As for *Office1*, the performance of the online TL in the target office is compared to the performance when implemented in the digital twin of *Office2*. Therefore, a comparison of the indoor temperature and energy consumption profiles between the real implementation and the digital twin during the second period (15-26 April 2024) is shown in Fig. 8, using the same boundary conditions and the same actions selected by the online TL controller implemented in reality.

Fig. 8 shows that the digital twin was less accurate in emulating the real indoor temperature profile when compared to the previous case of *Office1*. The temperature profile from the digital twin frequently deviated from the real implementation, particularly in response to abrupt temperature changes. Several factors may have contributed to the discrepancies observed in the indoor temperature profiles, such as occupant behaviour that influenced the opening/closing of doors and windows and the effect of solar radiation in combination with the presence of blinds in the window, as described in Section 2. These aspects significantly influenced the indoor thermal dynamics, resulting in inability of the digital twin inability to capture the temperature variations accurately. These discrepancies were associated with not integrating the effect of the opening of doors and windows by the occupants in the digital twin of the target office. Moreover, although the effect of the presence of the blinds was integrated within the modelling through the change of the g-value of the window, it turned out not to be adequate with the results obtained during the analysed period, in contrast to the results from the model calibration phase described in Section 4.1. As a result, the value of $RMSE_{T_i}$ for the comparison of indoor temperature profiles between real measured data and digital twin of *Office2* was equal to 0.92 °C. Despite the discrepancies in the temperature pro-

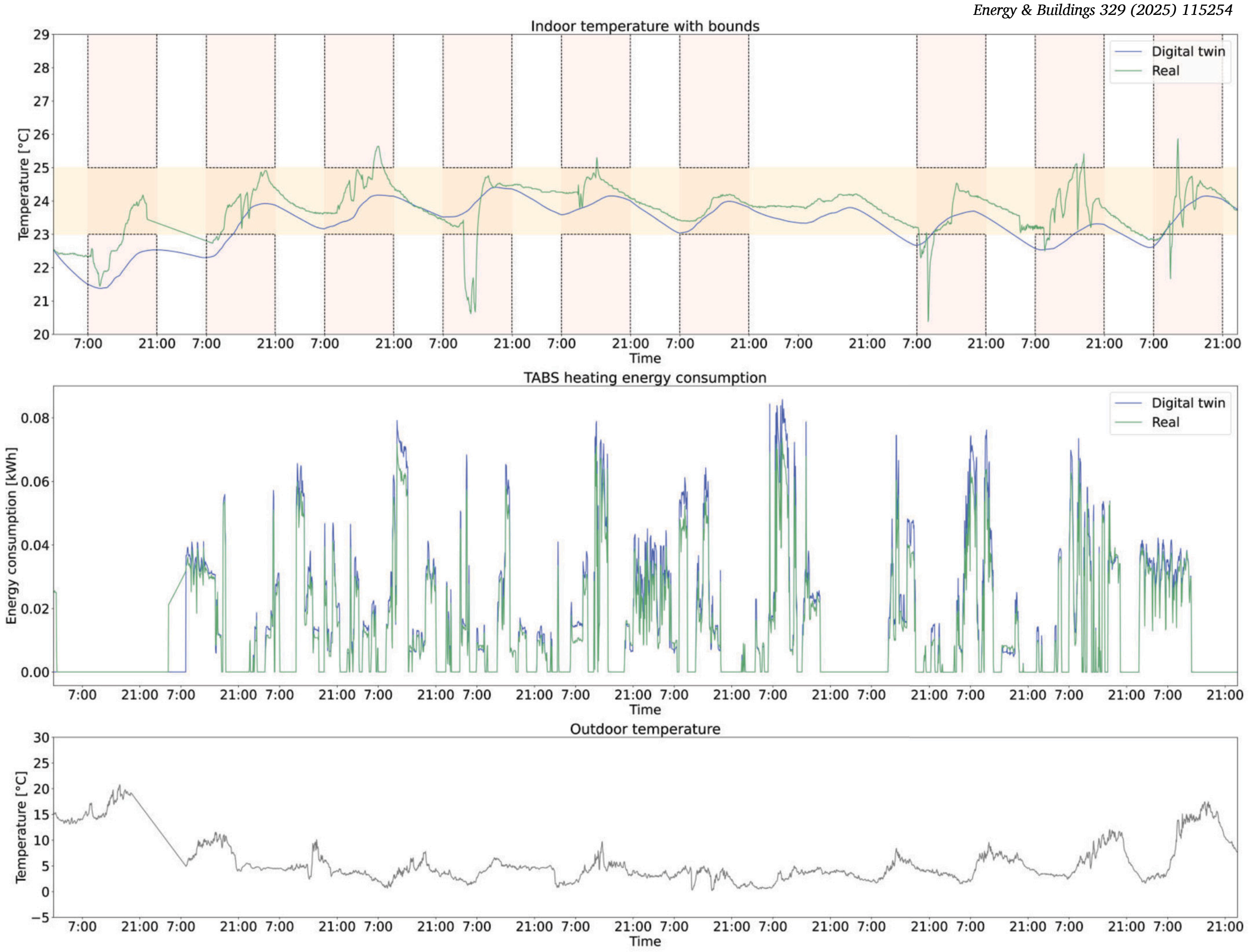

Fig. 8. Comparison of indoor temperature and energy consumption profiles between real and digital implementations for online TL in target office during the period 15-26 April 2024.

files, the energy consumption pattern of the digital twin was aligned with that of the real implementation. However, the peak values of energy consumed were higher for the digital twin. This suggests that while the temperature modelling faced challenges, the digital twin captured energy consumption dynamics more accurately. In this framework, the value of $MAPE_{E_{\text{TABS}}}$ was equal to 9.6%. In summary, while the digital twin demonstrated a reasonable approximation of the real energy consumption, it fell short of accurately modelling the indoor temperature profile for *Office2*. This discrepancy highlights the importance of including occupant interactions and behaviours in the digital twin models to enhance their predictive accuracy and reliability in real-world applications.

In conclusion, Table 6 offers an overview of the performance derived from implementing benchmark control strategies in the digital twin of the target office, while Fig. 9 provides a detailed view of indoor temperature and energy consumption profiles for the strategies providing the best performance per each control type.

Table 6
Average daily energy consumption and mean daily average temperature violation rate for online TL and benchmark controllers implemented in *Office2* digital twin.

Controller	$\overline{E_{\text{tabs}}}$ [kWh]	$\overline{T_{\text{viol,daily}}}$ [°C]
$RBC1$	3.55	0.371
$RBC2$	3.35	0.415
$PI23$	2.98	0.385
$PI23.5$	3.67	0.347
$PI24$	4.78	0.453
Online DRL	3.94	0.572
Online TL	2.81	0.247

The results in Table 6 refer to both implementation periods during 2024, 3-12 April and 15-26 April. The online TL controller had the lowest average daily energy consumption (i.e., 2.81 kWh), which is notably lower compared to all other controllers, saving around 20% of energy compared to RBCs, between 5% and 40% compared to PI controllers and up to 25% compared to the online DRL agent. Furthermore, the online TL controller achieved the lowest daily average temperature violation rate (i.e., 0.247 °C), reducing this metric by 33% and 40% compared to $RBC1$ and $RBC2$, up to 30% in comparison to PI controllers and by 57% compared to the online DRL strategy. Moreover, to demonstrate that the proposed online TL strategy is an effective solution to enhance the scalability of DRL controllers regardless of the transfer direction, Table E.11 in Appendix E shows the performance comparison for DRL-based controllers (i.e., offline DRL, online DRL and online TL) over a three-month period from a different year for Dübendorf. In detail, in Table E.11 the online TL process was carried out by transferring a DRL controller pre-trained in *Office2* (i.e., source) to *Office1* (i.e., target).

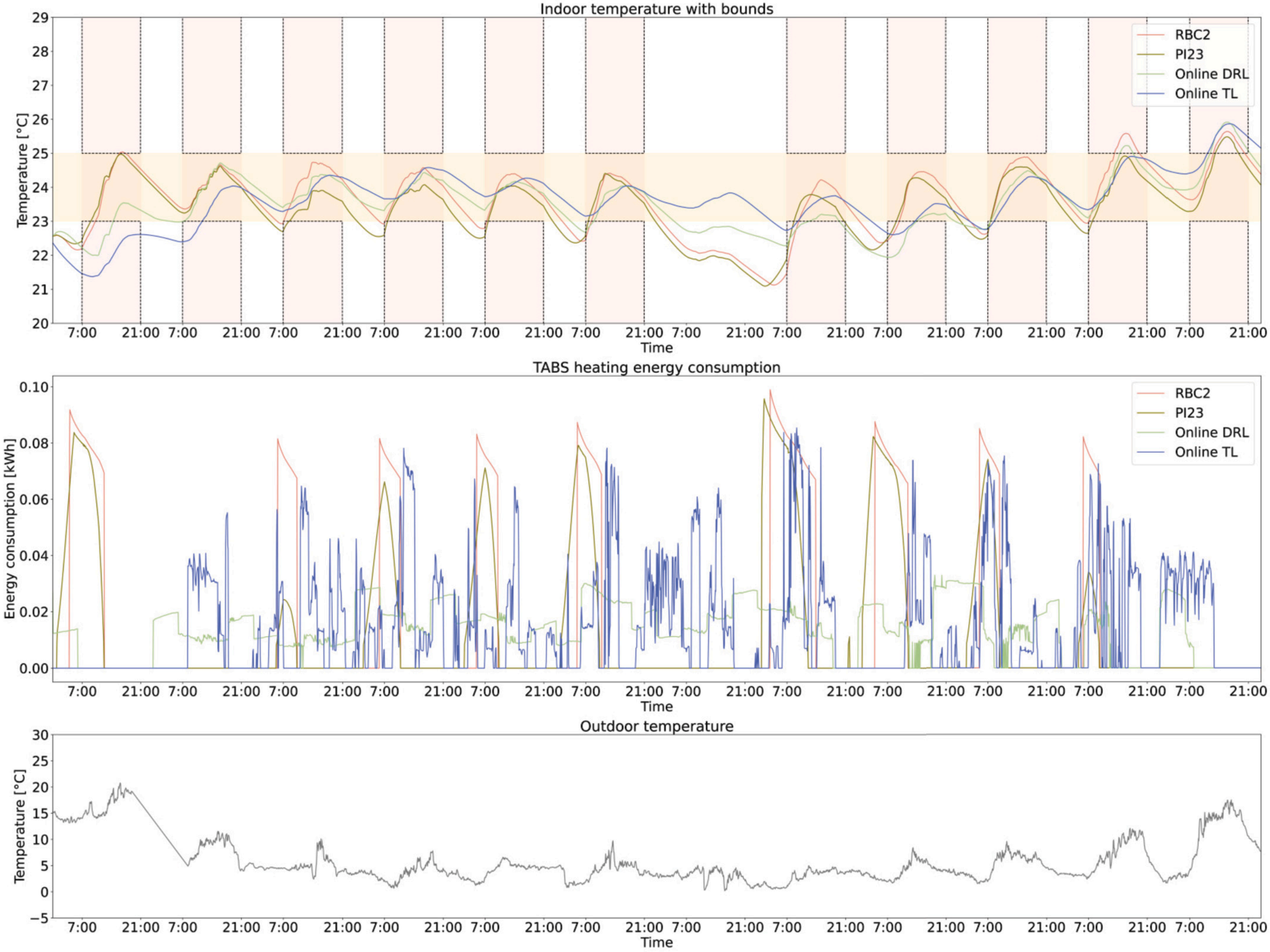

Fig. 9. Comparison of indoor temperature and energy consumption profiles between online TL, online DRL, $RBC2$ and $PI23$ implemented in the *Office2* digital twin during the period between 15 April and 26 April 2024.

Fig. 9 shows that the online TL controller outperformed the benchmark $RBC24$, $PI23$ and online DRL controllers in optimising indoor temperature profile and reducing energy consumption. The online TL controller was the most effective in mitigating temperature violations, particularly during the initial hours of occupancy, due to its ability to preemptively supply heating energy for maintaining indoor temperatures within the desired range more reliably than the other controllers. Moreover, the online TL controller reduced peaks in energy consumption compared to $RBC24$ and $PI23$ through a more efficient energy management strategy that adjusted the heating supply proactively to avoid large energy spikes. Compared to the online DRL controller, the online TL strategy demonstrated better management of the TABS system, achieving a more balanced trade-off between energy consumption and maintaining indoor temperature within comfort bounds. The online DRL and online TL controllers are designed to adapt to real-time conditions, which can result in large jumps in energy consumption as they respond to changing occupancy patterns and outdoor temperature influences. Nevertheless online TL led to better overall performance compared to other benchmark controllers, its control policy may pose challenges regarding component wear, as in this case study for the TABS valve. The maintenance of such components could incur additional costs that negate the benefits of energy savings. To address this issue, future implementations of the DRL-based controllers should consider the integration of a strategy to minimise abrupt transitions in valve operation. This integration could involve adding a term to the objective function that minimises the difference in the values of consecutive control actions or by including safety constraints for the actuating systems that override control actions when necessary.

6. Discussion

This paper assessed the effectiveness of an online TL strategy in homogeneous and transductive settings, implemented in a real building with two offices. By leveraging imitation learning and weight-initialisation, the online TL framework avoided the need for extensive pre-training, a significant limitation in real-world applications where DRL controllers must operate efficiently from the early stage of implementation.

The DRL controller implemented in the source office was directly transferred to the target office using the online TL methodology to evaluate its performance compared to traditional strategies like RBC and PI controllers, as well as an advanced DRL controller implemented without pre-training (i.e., online DRL). The developed online TL methodology ensured better performance than the benchmark controllers, avoiding a new pre-training phase of the controller on a new thermal zone. Such a pre-training phase is not compatible with the needs of real buildings, where the controller must ensure certain performance levels from the initial stages of implementation. Furthermore, the lack of adequate monitoring systems could limit the availability of monitored data to pre-train DRL controllers. In this context, avoiding the pre-training phase by using the developed online TL methodology makes the DRL implementation process in buildings more scalable.

One of the major strengths of the proposed TL method is its ability to adjust the control policy to different environmental conditions between source and target zones. The energy system implemented in the two office zones was the same and consisted of a lightweight TABS integrated into the ceiling. Moreover, both offices had the same geometry but differed in the number of occupants, heating load profiles, the acceptable indoor temperature range and differences in the layout of the ventilation system. In *Office2*, the ventilation system was integrated with the TABS, as a consequence the supply air temperature from ventilation and, consequently, indoor temperature increased compared to *Office1*, where the TABS and ventilation system were installed separately. Furthermore, the different layout of the ventilation system between the two offices leads to a variation of the thermal load profiles. In this regard, the online TL demonstrated excellent capabilities in adapting the transferred control policy to the indoor environment dynamics of the target office.

The results show that during the implementation period, the online DRL controller could not match the performance of the online TL, highlighting the advantages of the TL process considering Key Performance Indicator (KPI) defined in the literature for TL. From the comparison of online TL and online DRL controllers by using the TL KPI *Asymptotic performance* during the implementation period (3 April - 26 April 2024), it emerged that the online TL ensured better performance both in terms of energy consumption (i.e., -29%) and mean daily average violation rate (i.e., -57%). The online DRL controller failed in the short implementation period to guarantee the correct trade-off between reducing energy consumption and maintaining adequate indoor temperature conditions.

To benchmark the performance of the controllers implemented in both offices, a detailed RC-based digital twin model was developed for each office in Modelica. These models were developed according to the available technical specifications related to the thermophysical properties of the building envelope and the technical data of the TABS. Employing a digital twin guaranteed that the physical response of the model closely matched what would be observed if the same controller was implemented in a real-world setting. A digital twin was calibrated using a grid search approach by adjusting some model parameters and using real data measured in the offices during November 2023. However, the comparison of RMSE and MAPE obtained for the two offices during the real implementation periods revealed a limitation related to the calibration of the digital twin for the target office, which was less accurate than that for the source office. The main issue was the inadequate integration of occupant behaviour into the model, as they frequently changed the indoor temperature dynamic by opening doors and windows. To overcome this limitation, the possibility of air infiltration from adjacent spaces or the outside could be included in future modelling processes, allowing for a more accurate consideration of this phenomenon.

The widespread implementation of DRL-based controllers in real buildings is limited by potential equipment damage due to extreme indoor environmental conditions, connection and monitoring infrastructure issues [13]. Sensors significantly impact the quality of the control process, as the control agent uses the measured variables to select the optimal control action. If a sensor does not accurately measure the variable (e.g. indoor temperature), it would provide incorrect information to the controller, leading to a misinformed understanding of the actual conditions in the building. Therefore, it would be beneficial to quantify the effectiveness of the sensors through a sensitivity analysis [61].

The building employed as a case study was a living lab. While there are similarities between real buildings and living labs, there are also significant differences. For example, living labs like HiLo usually have comprehensive sensors and monitoring systems to collect detailed performance data. In contrast, real buildings often have more limited instrumentation. However, to ensure experiment replicability and scalability in real-world applications, this method offers a strong solution to typical DRL limitations, such as limited observation space and limited monitoring systems. The DRL controller had a limited and simplified observation space, including easy-to-measure variables monitored in real buildings with a limited amount of sensors (e.g., indoor temperature, solar radiation, occupancy data). Results demonstrated that the DRL controller can function effectively in standard building environments without needing extensive instrumentation.

For occupancy, Passive Infrared Sensors (PIRs) are frequently employed since they are low-cost sensors (typically ranging from $10 to $50) and easy to use [62]. PIRs mainly provide a binary variable indicating the presence of occupants, which is sufficient for several energy management applications. Their accuracy is generally good for detecting movement (around 5-10%), although they may not accurately count the number of occupants. For solar radiation, while dedicated sensors can be used, which may cost between $100 and $500, it is not necessary to install them. Solar radiation values can be obtained from online services like the Solcast API [63], which can provide reliable solar radiation data (accuracy less than 10%) at little cost, making it a practical solution for many applications.

The controller implementation setup ensured interoperability with the existing systems (as per the action implemented in the TABS valve) without including other sensors or actuators. Moreover, a fail-safe mechanism is included in the monitoring and implementation infrastructure as in [44] to avoid possible damages due to connection and infrastructure issues.

In this work, the controllers were implemented on a remote PC connected directly via a Python-developed communication interface with the NEST Cloud. However, this type of connection was subject to connection errors on the remote PC side. Therefore, a possible future improvement to limit connection issues could be the controller implementation directly on the devices in NEST, avoiding the development of a communication interface that leads to additional implementation issues. Moreover, to avoid extreme indoor temperature conditions that may cause discomfort for occupants, safety constraints related to indoor temperature values should be integrated into the fallback safety system. These constraints will automatically switch to the default controller if such conditions persist for extended periods.

Another limitation is linked to the case study, since the two thermal zones evaluated for the online TL process are part of the same building. Besides having similar geometry and energy systems, the two zones have the same boundary conditions (e.g. climatic conditions), except for the different range of acceptable indoor temperatures and the stochastic nature of the occupancy profile. It might be advantageous to implement the online TL between buildings located in different locations to evaluate if the climate conditions influence the effectiveness of the transfer process in real buildings. Furthermore, the benefits of implementing online TL should be considered among buildings with the same type of HVAC equipment (e.g. TABS as in this paper) but integrated within different and increasingly complex energy systems.

7. Conclusion

This paper proposes the implementation of an online TL strategy in a real building located in Switzerland to transfer a DRL-based controller between two offices inside this building. The DRL controller was implemented to manage the valve opening percentage of the TABS to reduce energy consumption and optimise indoor temperature conditions within an acceptable range.

The DRL controller was first pre-trained on the digital twin of the source office. Afterwards, the DRL controller was implemented in the real office and its performance was benchmarked using the digital twin of *Office1* against two RBCs and three PI controllers. In this phase, the DRL ensured similar performance in terms of average daily energy consumption compared to $RBC2$ and $PI21$, and 10 to 30% better performance compared to the other benchmark controllers. Furthermore, the DRL provided better performance in indoor temperature management, reducing the daily average temperature violation rate $\overline{T_{\text{viol,daily}}}$ by 50% compared to all benchmark controllers.

Thus, the source controller was implemented in the real *Office2* following the online TL strategy. The performance of online TL was benchmarked by implementing in the digital twin of the target office controllers of the same type as in *Office1* as well as a DRL-based controller implemented directly without pre-training (i.e., online DRL). Online TL was more effective than the other benchmark controllers, since it reduces the average daily energy consumption by 20%, between 5% and 40% and up to 25% when compared to RBCs, PI and online DRL controllers, respectively. Furthermore, online TL implementation led to a reduction in $\overline{T_{\text{viol,daily}}}$ of up to 30% compared to RBCs and PI controllers and of 57% compared to online DRL.

This study offers important insights into the practical implementation of online TL, effectively connecting simulation-based research with real-world implementation. The results highlight the potential and feasibility of using online TL for DRL controllers in real-world applications, proving that online TL can significantly enhance the scalability of DRL controllers in buildings.

Future works could be focused on the following aspects:

- Evaluation of the real performance for online TL implementation when the DRL manages both the TABS and other systems installed in the office, such as the HVRF. Including the HVRF ensures a faster response in maintaining appropriate indoor building temperature conditions. Furthermore, this implementation procedure could be extended to the cooling season.
- Improvement of the digital twin models developed for the two offices to account for other factors related to occupant behaviour that influences the indoor temperature dynamics, e.g. door and window opening. Moreover, the digital twins could be calibrated using specific algorithms, such as the Genetic Algorithm (GA), to ensure better performance in terms of RMSE and MAPE compared to those obtained in this work.
- Evaluation of the real performance of online TL when DRL-based controllers are transferred between different buildings with different boundary conditions (e.g. weather), different geometry and energy systems. Moreover, complex versions of the reward function could be evaluated, including factors such as electricity price (e.g., Time-Of-Use (TOU) or Real-Time Pricing (RTP)) or peak demand reduction.
- Extension of the performance benchmarking by considering other advanced model-based control strategies to better demonstrate the advantages of implementing online TL when implemented in real buildings.

CRediT authorship contribution statement

Davide Coraci: Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Alberto Silvestri:** Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Giuseppe Razzano:** Writing – original draft, Visualization, Software, Investigation, Formal analysis. **Davide Fop:** Writing – original draft, Visualization, Software, Investigation, Formal analysis. **Silvio Brandi:** Writing – review & editing, Validation, Methodology, Investigation, Conceptualization. **Esther Borkowski:** Writing – review & editing, Resources. **Tianzhen Hong:** Writing – review & editing, Validation, Methodology. **Arno Schlueter:** Writing – review & editing, Validation, Methodology, Conceptualization. **Alfonso Capozzoli:** Writing – review & editing, Validation, Supervision, Methodology, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Alberto Silvestri reports financial support was provided by Mitsubishi Electric R&D Centre Europe BV. Other authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

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 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).

The work of Alfonso Capozzoli and Giuseppe Razzano is carried out within the FAIR - Future Artificial Intelligence Research and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.3 – D.D. 1555 11/10/2022, PE00000013). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

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

Appendix A. Related works on transfer learning applications for DRL controllers in buildings

This section provides an overview about additional related works focused on the application of TL frameworks for DRL controllers.

Lissa et al. [64] introduced parallel TL to transfer the control policy among five distinct DRL agents throughout their training without the need to wait until the end of the training process. This method has been implemented in a microgrid consisting of five separate homes, resulting in a five-fold reduction in training time and a 10% decrease in energy consumption. This results was achieved by optimising the operation of a Photovoltaic (PV) system combined with a heat pump, compared to scenarios without knowledge transfer. An occupancy-based TL methodology leveraging the K-means algorithm and Dynamic Time Warping (DTW) was developed in [65] to match similar occupancy patterns in 26 residential units and improve control performance for the HVAC system. Thermal discomfort during the learning process was reduced and jumpstart and asymptotic performances were enhanced respectively by 25% and 5% compared to a model-free controller. Kadamala et al. [66] fine-tuned a DRL controller pre-trained on a source environment in two distinct simulation environments, one simulating the same building under different weather conditions while the other simulated a different building under the same weather conditions. The experiments included two different reward functions to assess their impact on TL. Results demonstrate that the transferred agents outperformed the RBC and reduce the total reward from 1% to 4% compared to DRL agents without pre-training.

Moreover, recent TL applications for DRL controllers have been introduced in the literature to ensure acceptable performance for DRL agents from the early stage of deployment in target buildings, emulating real-world applications. Liu et al. [67] proposed a generative adversarial IL approach to employ expert demonstration from a Model Predictive Control (MPC) to enhance the performance of a DRL controller managing a Variable Air Volume (VAV) system for cost reduction and load shifting purposes in a commercial building. The proposed approach outperformed a RBC and a DRL controller not employing IL by improving the cumulative reward respectively by 22% and 7%. Amasyali et al. [68] employed a TL strategy to transfer the knowledge from ten DRL controllers pre-trained for managing the cooling energy supply to a target building. The results demonstrated that the TL approach achieved a cumulative reward comparable to that of a DRL controller pre-trained for ten episodes while significantly outperforming both a DRL agent deployed with no pre-training and a fixed setpoint operation controller. To conclude, Coraci et al. [39,24] introduced an online TL approach aimed at simulating the real-world implementation of a TL process. The transferred DRL controller managed a cooling system to reduce electricity cost and to enhance indoor temperature conditions. In both applications, source and target buildings had the same spatial configuration but different envelope features, climate conditions and electricity price/occupancy schedules. In [39], the online TL approach reduced electricity cost and ensured an improvement in indoor temperature conditions when compared with RBC and a DRL agent directly implemented without pre-training. Although slightly less effective than the DRL controller deployed after an offline pre-training phase, the online TL approach offered the significant advantage of requiring no additional modelling effort and could be directly implemented in target buildings, making it highly suitable for real-world implementation. Similar results were obtained in [24] where online TL was evaluated for source and target buildings differed in terms of implemented energy systems due to the presence of PV systems and Battery Energy Storage System (BESS) in target buildings. Despite differences in the energy systems, the source and target controllers chose the same control actions, albeit with changes in the state-space design due to additional information related to PV and BESS operation.

Appendix B. Background on reinforcement learning and transfer learning

This section provides details about theoretical foundations related to RL and TL.

B.1. Reinforcement learning

Advanced controllers employing reinforcement learning algorithms are designed to discover the optimal control policy for a given problem through a process of trial and error. More specifically, RL can be framed as a Markov Decision Process (MDP), represented by a tuple that includes four key elements [14]:

- State (s): This is the mathematical representation of the controlled environment, used as input for the controller to determine the appropriate control action. The state may provide either a complete or partial description of the environment; in cases where only partial information is available, it constitutes a Partially Observable Markov Decision Process (POMDP). In building energy management, common state variables include parameters such as indoor temperature and occupancy status.
- Action (a): This refers to the control decision made by the controller at each control time step to optimize the control problem. In the context of building energy management, a typical action might involve setting the supply water temperature in heating or cooling systems.
- Transition Probabilities (P): Transition probability, denoted as $P(s_{t+1} = s'|s_t = s, a_t = a) = P : SXAXS'$ defines the likelihood of moving from a current state *s* to a subsequent state *s'* as a result of the action *a* chosen by the controller. In building energy management, these probabilities are often unknown, as deriving them requires a detailed model of the controlled environment.
- Reward (R): The reward quantifies the quality of transitioning from *s* to *s'* after taking action *a* and is defined by a function that aligns with specific control objectives. In building energy management, the reward function typically reflects a balance between minimising energy consumption and enhancing indoor temperature conditions.

The ultimate goal of an RL agent is to determine an optimal control policy π, which is a mapping between states and actions that maximizes the cumulative reward over a defined time horizon [14]. This is achieved through interaction with the controlled environment via a process of trial and error.

Within the RL framework, the problem is characterized by two core functions essential for determining the optimal control policy: the state-value function and the action-value function. The state-value function provides the expected cumulative reward when the agent begins in a given state *s* and follows the policy π thereafter [69].

$$\begin{aligned} v_\pi(s) &= \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a)[r+\gamma v_\pi(s')] \\ &= E[r_{t+1} + \gamma v_\pi(s')|S_t = s, S_{t+1} = s'] \end{aligned} \tag{B.1}$$

Here, r denotes the immediate reward that the agent receives as it transitions from state s to s' following the execution of action a. The parameter γ [0,1] represents the discount factor for future rewards [14]. When the agent sets $\gamma = 1$, it places higher emphasis on future rewards, while a value of $\gamma = 0$ reflects a preference for immediate rewards.

Similarly, the action-value function reflects the capability of the DRL agent to choose a specific control action a while following a control policy π from a given state s [70].

$$q_\pi(s,a) = E[r_{t+1} + \gamma q_\pi(s',a')|S_t = s, A_t = a] \tag{B.2}$$

Due to its simplicity Q-learning is the most widely used algorithm within the RL family. In Q-learning, the association between states and actions is represented using a tabular approach, where Q-values (i.e., state-action values) are iteratively updated by the agent through the Bellman equation [71].

$$Q(s,a) = Q(s,a) + \mu[r(s,a) + \gamma max_{a'} Q(s',a') - Q(s,a)] \tag{B.3}$$

where μ [0,1] represents the learning rate, i.e., the degree to which new knowledge overrides the old knowledge. When an agent uses a learning rate of 0, it does not acquire new knowledge and thus does not update its control policy. Conversely, setting the learning rate to 1 causes the agent to entirely overwrite any previously learned knowledge with new information.

Despite its benefits, a tabular approach can become impractical for real-world applications, due to the extensive state and action spaces that would need to be stored [72]. A viable solution to address this limitation in Q-learning is to employ Deep Neural Networks (DNNs) instead of conventional lookup tables. DNNs can approximate Q-values, thereby mapping the state-action relationship without relying on tables to identify the optimal control policy [73]. In this context, RL algorithms that incorporate neural networks fall under the category of DRL.

Different control algorithms based on DRL can be found in the literature. However, this chapter describes only the Soft Actor-Critic (SAC) algorithm used in this work.

Soft actor-critic The SAC algorithm, introduced by Haarnoja et al. [57], is an actor-critic-based approach that utilizes two distinct neural networks, known as the Actor and the Critic. The actor-critic structure is particularly beneficial for handling stochastic processes, such as HVAC control, as it enables direct learning of a stochastic policy [74]. Unlike other actor-critic methods, SAC is an off-policy DRL algorithm distinguished by its high performance in optimizing various control tasks and its ability to operate effectively in continuous action spaces.

In SAC, the Actor and Critic networks, parametrized as DNNs, are used to approximate the state-value and action-value functions, respectively. The Actor operates within both control and learning loops, selecting the optimal control action at each control time step (policy-based). The Critic, on the other hand, functions solely during the learning phase, assessing the effectiveness of the Actor's decisions (value-based). The control policy is further improved through the reuse of past experiences stored in replay memory, as prescribed by the off-policy framework.

Additionally, SAC follows the maximum entropy RL framework: by incorporating an entropy term into the objective function, alongside the expected return, the algorithm achieves greater stability and improved exploration [75].

$$\pi^* = argmax_\pi E_\pi[\sum_{t=0}^{\infty} \gamma^t (r(s_t, a_t) + \alpha H_t^\pi)] \tag{B.4}$$

Here, $r(s_t, a_t)$ denotes the reward for a given state-action pair, α is a regularisation term known as the temperature coefficient, and H represents the Shannon entropy term. The temperature coefficient α modulates the relative importance of the entropy term in relation to the reward; for traditional RL algorithms, α is set to 0. When α is low, the agent prioritizes reward maximisation over entropy maximisation. The Shannon entropy term, meanwhile, encourages the agent to maximize returns while exploring the action space as diversely as possible.

B.2. Transfer learning

Transfer learning is a machine learning approach that has emerged as an effective technique for leveraging knowledge gained from one task to enhance performance on a different but related problem [26]. Knowledge sharing occurs at the beginning of the learning process, enabling faster convergence of machine learning models compared to scenarios where learning is initiated from scratch without any prior knowledge. The mathematical framework of TL necessitates a clear understanding of the concepts of domain and task, as outlined by [27].

Domain: a domain D consists of two components, a feature space X and a marginal distribution probability $P(X)$, where $X = x_1, \dots, x_n \in X$.

Task: a task T consists of a label space Y and an objective predictive function $f(\cdot)$ denoted by $Y = Y, f(\cdot)$. This function is not directly observed but it is learned from the training data, represented by a pair (x_i, y_i), where $x_i \in X$ and $y_i \in Y$. $f(\cdot)$ is used to approximate the conditional probability $P(y|x)$ as well as to predict the label of new instances x.

TL occur between multiple domains, however research has focused on the case where knowledge sharing occurs between a source domain D_S = (x_{S_1}, y_{S_1}),..., ($x_{S_{n_S}}$, $y_{S_{n_S}}$) and a target domain D_T = (x_{T_1}, y_{T_1}),..., ($x_{T_{n_T}}$, $y_{T_{n_T}}$) [76]. Thus, according to [26,27], TL is defined as the process that improves the learning of the predictive function in the target domain D_T with learning task T_T, using the acquired knowledge in the source domain D_S with task T_S.

This paper investigates the knowledge-sharing among controllers based on DRL. Therefore, it is required to define a correspondence between the concepts of domain, label space, and task as defined in conventional ML contexts and the state-space, action-space, and reward function utilized in DRL [24]. Various studies, such as those by [40,77], provide valuable insights into the possible applications of TL for this control framework. In RL, the input feature space (i.e., domain) corresponds to the state-space, while the label space is aligned with the action-space. Overall, knowledge sharing in ML problems can occur when the source and target domains, tasks, and solutions differ or show similarities [39]. In this regard, [26] has classified different aspects based on the similarity of tasks (i.e., label classification), features and labels (i.e., space classification), and knowledge-sharing modalities (i.e., solution classification). This section provides only details on the elements relevant to each classification mentioned below.

Three categories are established for classifying TL approaches based on the similarity of tasks and the availability of labelled data in both the source and target domains.

Inductive TL occurs when labelled data is present in both the source and target domains, although the tasks differ. The main focus is on leveraging the labelled data from the source domain to enhance learning in the target domain, rather than emphasizing differences between the domains.

Transductive TL is characterized by the existence of different domains for the source and target, while the tasks remain the same. In this case, labelled data is only available for the source domain, with the goal of utilizing this labelled information to improve learning in the target domain.

Unsupervised TL applies when labelled data is not available in either the source or target domains. The domains may be the same or different, and the tasks in the source and target domains are distinct. The objective here is to exploit the shared information or structure between the domains to enhance learning in the target domain.

Furthermore, TL can also be categorized based on the differences in feature spaces and labels between the source and target domains. Homogeneous TL pertains to scenarios where the feature spaces and labels of the source and target domains are identical, indicating no variations in features or labels between them. Heterogeneous TL applies when there are discrepancies in feature spaces and/or labels between the source and target domains. This classification occurs when the feature spaces, labels, or both differ.

Additionally, TL can be further classified according to the method of knowledge sharing used in solution classification, which encompasses instance-based, feature representation-based, relational knowledge-based, and model parameter-based TL [26]. This paper emphasizes model parameter-based TL, which focuses on sharing specific parameters or their distributions, such as model weights, between the source and target tasks. This type of TL includes three sub-classifications based on the methods for sharing parameters: feature extraction, weight-initialisation, and relational knowledge-based [26]. In weight-initialisation, the target model weights are initialized using the pre-trained weights from the source task. This approach establishes a foundation for the target model to leverage the knowledge gained in the source domain. Following the initialisation, an additional fine-tuning process can be conducted. During this fine-tuning phase, the target model is further trained using data specific to the target task, allowing it to adjust its parameters according to the characteristics of the target domain [24].

Appendix C. Details on hyperparameters optimisation for the source DRL controller

This section provides details about the hyperparameter optimisation phase for the source DRL controller.

The first five rows of Table C.7 indicates the values of hyperparameters kept fixed during the optimisation process, while the last five rows report the optimised hyperparameters (i.e., Number of hidden layers, Number of neurons per hidden layer, Learning rate μ, θ and β) with their corresponding range and step value. Moreover, the Boltzmann temperature coefficient α was automatically optimised by employing the built-in function included in Stable-Baselines 3, considering a starting value of α equal to 1. From Table C.7 emerges that the best value from the optimisation of hyperparameters for β was equal to 2, indicating that the agent received an equivalent penalty for deviating by 0.5 kWh and for straying 1 °C beyond the comfort bounds [78,79].

Table C.8 reports the results for the twenty tested configuration of DRL hyperparameters for the source controller.

The 12th configuration resulted as the best, since it had the smallest Euclidean distance value from the ideal point, whose coordinates were $\overline{E_{\text{tabs,ideal}}} = 5.52$ kWh and $\overline{T_{\text{viol,daily,ideal}}} = 0.027\,°\text{C}$, as well as better performance than $RBC1$, equal to $\overline{E_{\text{tabs,RBC1}}} = 7.28$ kWh and $\overline{T_{\text{viol,daily,RBC1}}} = 0.139\,°\text{C}$. The best configuration of hyperparameters is highlighted in yellow.

Due to computational constraints, each trial was run only once in this study, since each trial consisted of 20 episodes, with each episode taking on average 10 minutes on a machine equipped with a 4-core CPU

Table C.7
Values and range of DRL controller hyperparameters.

Hyperparameters	Range	Step	Best value
Discount factor γ	-	-	0.99
Batch size	-	-	128
Training episodes	-	-	20
Gradient steps	-	-	1
Train frequency	-	-	1
# Hidden layers	[2,4]	2	4
# Neurons per hidden layer	[64, 128]	64	64
Learning rate μ	$[1\cdot10^{-4}, 1\cdot10^{-3}]$	$1\cdot10^{-4}$	$5\cdot10^{-4}$
Reward scaling factor θ	[1, 5]	1	1
Temperature-term reward weight β	[1, 10]	1	2

Table C.8
Configurations of DRL hyperparameters involved in the optimisation process for source controller, with the selected configuration highlighted in yellow. (For interpretation of the colours in the table(s), the reader is referred to the web version of this article.)

Configuration	# Layers	# Neurons	μ	θ	β	$\overline{E_{\text{tabs}}}$ [kWh]	$\overline{T_{\text{viol,daily}}}$ [°C]	$d_{\text{euclidean}}$
1	2	128	$8\cdot10^{-4}$	3	10	7.18	0.029	1.66
2	4	128	$4\cdot10^{-4}$	2	7	6.54	0.086	1.02
3	2	64	$1\cdot10^{-3}$	5	9	8.43	0.041	2.91
4	2	64	$7\cdot10^{-4}$	1	5	5.98	0.065	0.46
5	4	64	$3\cdot10^{-4}$	4	8	7.29	0.079	1.77
6	2	128	$1\cdot10^{-3}$	3	6	6.15	0.035	0.63
7	4	128	$5\cdot10^{-4}$	5	4	5.93	0.098	0.42
8	2	128	$7\cdot10^{-4}$	1	10	8.11	0.027	2.59
9	2	64	$9\cdot10^{-4}$	2	2	7.78	0.038	2.26
10	4	64	$4\cdot10^{-4}$	5	1	5.52	0.109	0.08
11	4	128	$2\cdot10^{-4}$	4	3	6.03	0.092	1.03
12	4	64	$5\cdot10^{-4}$	1	2	5.59	0.032	0.07
13	4	128	$6\cdot10^{-4}$	1	9	7.67	0.045	2.15
14	4	64	$1\cdot10^{-3}$	4	2	6.14	0.058	0.62
15	4	64	$8\cdot10^{-4}$	2	5	7.99	0.081	2.47
16	2	64	$2\cdot10^{-4}$	5	6	5.66	0.070	0.15
17	2	128	$5\cdot10^{-4}$	1	3	5.74	0.098	0.23
18	4	128	$7\cdot10^{-4}$	5	8	7.36	0.063	1.84
19	4	64	$1\cdot10^{-4}$	3	7	5.91	0.030	0.39
20	2	64	$3\cdot10^{-4}$	4	10	6.83	0.079	1.31

Table D.9
Results comparison for $PI21.5$ controller with different start time.

Start time	$\overline{E_{\text{tabs}}}$ [kWh]	$\overline{T_{\text{viol,daily}}}$ [°C]
1:00	7.41	0.045
2:00	7.35	0.056
3:00	7.28	0.086
4:00	7.19	0.149
5:00	7.06	0.224

operating at 3.40 GHz and 16GB of RAM. This resulted in approximately 3.5 hours per trial, leading to a total of 70 hours for the hyperparameter tuning process. However, the results indicated in Table C.8 does not fully capture variability due to random initialization or stochastic training processes of DRL controllers since results referred to a single run per each trial. Therefore, future work will aim to address this by running multiple trials for each hyperparameter configuration to ensure that the observed differences are statistically significant. This would allow for a more robust comparison between the results of each trial.

Appendix D. Details on PI tuning

This section provides further details regarding the tuning of PI controller.

Table D.9 summarises the results in terms of $\overline{E_{\text{tabs}}}$ and $\overline{T_{\text{viol,daily}}}$ for the $PI21.5$ controller in the source office considering five different start times.

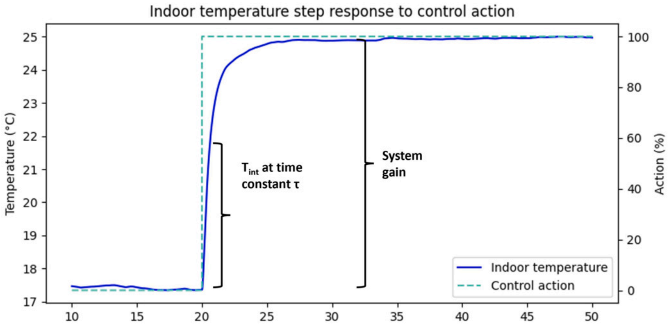

Fig. D.10. Indoor temperature and action provided as input to the *Office1* digital twin model to evaluate system parameters.

To provide estimated values of system gain, time constant, and dead time a simulation has been performed considering fixed values for inputs requested by *Office1* digital twin with the exception of the control action. The purpose of the experiment was to determine the requested step response parameters. Indeed, after a time period sufficient to reach steady state, the control input was increase to its maximum value (i.e., 100% of TABS valve opening) by means of a step function. Subsequently, the response of the controlled variable (i.e., office zone indoor air temperature) was used to compute system parameters. Fig. D.10 provides an overview of the indoor temperature trend during the experiment, from which results a time constant equal to 17 hours and a system gain equal to 7.7 °C. Moreover, the dead time value that corresponds to an increase of 0.1 °C in the indoor temperature value is equal to 1 hour. However,

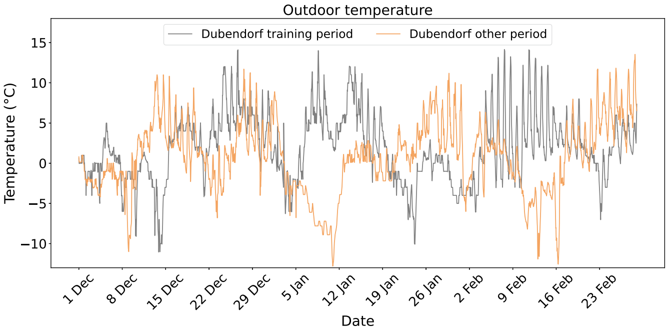

Fig. E.11. Comparison of outdoor temperature profiles for the two three-month periods evaluated for Dübendorf (Switzerland).

these parameters are defined for first-order systems, while the considered system is of higher order. Therefore, the values for the discussed parameters must be considered correct albeit approximated.

Appendix E. Supplementary validation analysis for DRL controller pre-training in *Office1* and for the online TL strategy

This section provides additional details on the analysis of supplementary results related to the pre-training process of the DRL controller on Office 1 and a benchmark of the online TL performances with those of offline DRL and online DRL.

For the DRL pre-training phase in *Office1*, the results in Table 4 have been extended by comparing the performance of DRL and traditional controllers (i.e., RBC and PI strategies). Here, the performance of DRL was evaluated over three months, from 1 December to 28 February, but from a different year, to evaluate the performance of the controllers in different climatic conditions. For the online TL strategy, additional experiments were carried out by reversing the roles of the two office zones, transferring a controller pre-trained in *Office2* to *Office1*.

To provide further insight for the climatic conditions during the new validation period, Fig. E.11 shows a comparison between the outdoor temperature profiles used for the results in Table 4 and those for the new validation period. During the original training period, outdoor temperatures generally ranged between approximately −5 °C and 10 °C. In the new validation period colder minimum temperatures can be observed on certain days. Moreover, Fig. E.11 highlights the variability between the two periods, demonstrating that the DRL controller was evaluated under slightly more severe climate conditions than those of the original training period.

The results obtained in *Office1* for the DRL controller during the new validation three-month period pre-training phase in *Office1* and those of the benchmark controllers, are reported in Table E.10. These experiments were conducted using the digital twin developed for *Office1*. Moreover, the DRL results are reported in Table E.10 as the average value ± standard deviation, retrieved from five separate trials each with a different seed, to account for variability in DRL controller performance. Moreover, the value of employed DRL hyperparameters are the same as those indicated in Table C.7.

Results in Table E.10 indicate that the DRL controller outperformed traditional controllers implemented in *Office1* in the new validation period in terms of energy consumption. Specifically, DRL reduced TABS average daily energy consumption between 16% and 26% compared to RBCs and between 4% and 27% in relation to PI controllers. In addition, the DRL provided better control of indoor temperature conditions compared to $RBC1$, $PI21$ and $PI22$, since it decreased $\overline{T_{\text{viol,daily}}}$ respectively by 69%, 87% and 39%. The temperature control performance of the DRL is worse than that of $RBC2$ and $PI21.5$. However, these traditional controllers can not optimize multi-objective functions. As a result, better temperature control performances are achieved with higher TABS energy consumption compared to the DRL.

Table E.10
Average daily energy consumption and mean daily average temperature violation rate for DRL and benchmark controllers for *Office1* during the new validation periods considered after pre-training phase.

Controller	$\overline{E_{\text{tabs}}}$ [kWh]	$\overline{T_{\text{viol,daily}}}$ [°C]
$RBC1$	8.74	0.0827
$RBC2$	7.65	0.0211
$PI21$	6.73	0.1905
$PI21.5$	7.88	0.0121
$PI22$	8.92	0.0417
DRL	6.45 ± 0.51	0.0253 ± 0.0129

Table E.11
Average daily energy consumption and mean daily average temperature violation rate for offline DRL, online DRL and online TL for *Office1* during the other three-month period for Dübendorf.

Controller	$\overline{E_{\text{tabs}}}$ [kWh]	$\overline{T_{\text{viol,daily}}}$ [°C]
Offline DRL	6.51 ± 0.34	0.0495 ± 0.0070
Online DRL	10.42 ± 0.17	0.1069 ± 0.0118
Online TL	6.84 ± 0.20	0.0691 ± 0.0176

Table E.11 provides an overview about the performances in terms of $\overline{E_{\text{tabs}}}$ and $\overline{T_{\text{viol,daily}}}$ for online TL when the role of the two office zones was reversed. Therefore, a DRL controller pre-trained on *Office2* (i.e., source zone) was transferred to *Office1* (i.e., target zone). The online TL performances were benchmarked with those of offline DRL and online DRL during a three-month period considering the same weather conditions as in Table E.10. These experiments were conducted using the digital twin developed for *Office1*. Moreover, the results are reported as the average value ± standard deviation, retrieved from five separate trials each with a different seed, to account for variability in DRL-based controllers.

Results in Table E.11 indicate that online TL outperformed online DRL reducing $\overline{E_{\text{tabs}}}$ by 34% and $\overline{T_{\text{viol,daily}}}$ by 35%. In contrast, online TL performances were lower by 5% for $\overline{E_{\text{tabs}}}$ and by 40% for $\overline{T_{\text{viol,daily}}}$ than those obtained by implementing the offline DRL agent. Despite offline DRL outperformed the online TL, its applicability is limited compared to the other DRL-based strategies since it requires to perform again the offline training process for several training episodes.

Data availability

The data that has been used is confidential.

References

[1] R. Yan, T. Zhao, Y. Rezgui, S. Kubicki, Y. Li, Transferability and robustness of a data-driven model built on a large number of buildings, J. Build. Eng. 80 (2023) 108127, https://doi.org/10.1016/j.jobe.2023.108127.

[2] IEA, Digitalisation and energy, https://www.iea.org/reports/digitalisation-and-energy, 2017. (Accessed 27 April 2024), IEA report: Digitalisation and energy, Paris, France.

[3] IEA, Buildings, https://www.https://www.iea.org/energy-system/buildings, 2023. (Accessed 21 June 2024), IEA report: Buildings, Paris, France.

[4] G. Li, Y. Wu, S. Yoon, X. Fang, Comprehensive transferability assessment of short-term cross-building-energy prediction using deep adversarial network transfer learning, Energy (2024) 131395, https://doi.org/10.1016/j.energy.2024.131395.

[5] 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.

[6] T. Lee, S. Yoon, K. Won, Delta-t-based operational signatures for operation pattern and fault diagnosis of building energy systems, Energy Build. 257 (2022) 111769, https://doi.org/10.1016/j.enbuild.2021.111769.

[7] G. Martinopoulos, K.T. Papakostas, A.M. Papadopoulos, A comparative review of heating systems in eu countries, based on efficiency and fuel cost, Renew. Sustain. Energy Rev. 90 (2018) 687–699, https://doi.org/10.1016/j.rser.2018.03.060.

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

[9] 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.

[10] 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, pp. 1–59.

[11] C. Miller, Z. Nagy, A. Schlueter, Automated daily pattern filtering of measured building performance data, Autom. Constr. 49 (2015) 1–17, https://doi.org/10.1016/j.autcon.2014.09.004.

[12] W. Liang, H. Li, S. Zhan, A. Chong, T. Hong, Energy flexibility quantification of a tropical net-zero office building using physically consistent neural network-based model predictive control, Adv. Appl. Energy 14 (2024) 100167, https://doi.org/10.1016/j.adapen.2024.100167, https://www.sciencedirect.com/science/article/pii/S2666792424000052.

[13] Z. Nagy, G. Henze, S. Dey, J. Arroyo, L. Helsen, X. Zhang, B. Chen, K. Amasyali, K. Kurte, A. Zamzam, H. Zandi, J. Drgoňa, 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.

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

[15] 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, https://doi.org/10.1038/nature14236.

[16] A. Riebel, J.M. Cardemil, E. López, Multi-objective deep reinforcement learning for a water heating system with solar energy and heat recovery, Energy 291 (2024) 130296, https://doi.org/10.1016/j.energy.2024.130296.

[17] G. Han, H.-J. Joo, H.-W. Lim, Y.-S. An, W.-J. Lee, K.-H. Lee, Data-driven heat pump operation strategy using rainbow deep reinforcement learning for significant reduction of electricity cost, Energy 270 (2023) 126913, https://doi.org/10.1016/j.energy.2023.126913.

[18] A. Crespo, D. Gibert, Álvaro de Gracia, C. Fernández, Optimal control of a solar-driven seasonal sorption storage system through deep reinforcement learning, Appl. Therm. Eng. 238 (2024) 121905, https://doi.org/10.1016/j.applthermaleng.2023.121905.

[19] Z. Zhang, A. Chong, Y. Pan, C. Zhang, K.P. Lam, Whole building energy model for hvac optimal control: a practical framework based on deep reinforcement learning, Energy Build. 199 (2019) 472–490, https://doi.org/10.1016/j.enbuild.2019.07.029.

[20] D. Wang, W. Zheng, Z. Wang, Y. Wang, X. Pang, W. Wang, Comparison of reinforcement learning and model predictive control for building energy system optimization, Appl. Therm. Eng. 228 (2023) 120430, https://doi.org/10.1016/j.applthermaleng.2023.120430.

[21] Z. Zou, X. Yu, S. Ergan, Towards optimal control of air handling units using deep reinforcement learning and recurrent neural network, Build. Environ. 168 (2020) 106535, https://doi.org/10.1016/j.buildenv.2019.106535.

[22] A.T. Nguyen, D.H. Pham, B.L. Oo, M. Santamouris, Y. Ahn, B.T. Lim, Modelling building hvac control strategies using a deep reinforcement learning approach, Energy Build. 310 (2024) 114065, https://doi.org/10.1016/j.enbuild.2024.114065.

[23] X. Fang, G. Gong, G. Li, L. Chun, P. Peng, W. Li, X. Shi, X. Chen, Deep reinforcement learning optimal control strategy for temperature setpoint real-time reset in multi-zone building hvac system, Appl. Therm. Eng. 212 (2022) 118552, https://doi.org/10.1016/j.applthermaleng.2022.118552.

[24] 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.

[25] 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.

[26] 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.

[27] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359, https://doi.org/10.1109/TKDE.2009.191.

[28] S.-M. Jung, S. Park, S.-W. Jung, E. Hwang, Monthly electric load forecasting using transfer learning for smart cities, Sustainability 12 (16) (2020), https://doi.org/10.3390/su12166364, https://www.mdpi.com/2071-1050/12/16/6364.

[29] Z. Xing, Y. Pan, Y. Yang, X. Yuan, Y. Liang, Z. Huang, Transfer learning integrating similarity analysis for short-term and long-term building energy consumption prediction, Appl. Energy 365 (2024) 123276, https://doi.org/10.1016/j.apenergy.2024.123276.

[30] E. Omeragic, O. Orhan, T. Uzunovic, E. Golubovic, Analysing transfer learning efficacy with different feature sets for occupancy detection, in: 2023 XXIX International Conference on Information, Communication and Automation Technologies (ICAT), 2023, pp. 1–6.

[31] Y.-T. Chiang, C.-H. Lu, J.Y.-J. Hsu, A feature-based knowledge transfer framework for cross-environment activity recognition toward smart home applications, IEEE Trans. Human-Mach. Syst. 47 (3) (2017) 310–322, https://doi.org/10.1109/THMS.2016.2641679.

[32] H. Li, G. Pinto, M.S. Piscitelli, A. Capozzoli, T. Hong, Building thermal dynamics modeling with deep transfer learning using a large residential smart thermostat dataset, Eng. Appl. Artif. Intell. 130 (2024) 107701, https://doi.org/10.1016/j.engappai.2023.107701.

[33] G. Pinto, R. Messina, H. Li, T. Hong, M.S. Piscitelli, A. Capozzoli, Sharing is caring: an extensive analysis of parameter-based transfer learning for the prediction of building thermal dynamics, Energy Build. (2022) 112530, https://doi.org/10.1016/j.enbuild.2022.112530.

[34] L. Chen, G. Li, J. Liu, L. Liu, C. Zhang, J. Gao, C. Xu, X. Fang, Z. Yao, Fault diagnosis for cross-building energy systems based on transfer learning and model interpretation, J. Build. Eng. (2024) 109424, https://doi.org/10.1016/j.jobe.2024.109424.

[35] G. Li, L. Chen, J. Liu, X. Fang, Comparative study on deep transfer learning strategies for cross-system and cross-operation-condition building energy systems fault diagnosis, Energy 263 (2023) 125943, https://doi.org/10.1016/j.energy.2022.125943.

[36] N. Gavenski, O. Rodrigues, M. Luck, Imitation learning: a survey of learning methods, environments and metrics, arXiv:2404.19456, https://arxiv.org/abs/2404.19456, 2024.

[37] M. Genkin, J. McArthur, A transfer learning approach to minimize reinforcement learning risks in energy optimization for automated and smart buildings, Energy Build. 303 (2024) 113760, https://doi.org/10.1016/j.enbuild.2023.113760.

[38] 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.

[39] 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.

[40] F.L. Da Silva, A.H.R. Costa, A survey on transfer learning for multiagent reinforcement learning systems, J. Artif. Intell. Res. 64 (1) (2019) 645–703, https://doi.org/10.1613/jair.1.11396.

[41] P. Richner, P. Heer, R. Largo, E. Marchesi, M. Zimmermann, NEST – a platform for the acceleration of innovation in buildings, Inf. Construc. 69 (548) (2018) 222, https://doi.org/10.3989/id.55380, http://informesdelaconstruccion.revistas.csic.es/index.php/informesdelaconstruccion/article/view/5879.

[42] 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, 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.

[43] B. Svetozarevic, M. Begle, P. Jayathissa, S. Caranovic, R.F. Shepherd, Z. Nagy, I. Hischier, J. Hofer, A. Schlueter, Dynamic photovoltaic building envelopes for adaptive energy and comfort management, Nat. Energy 4 (8) (2019) 671–682.

[44] 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.

[45] Modelica Association, Modelica$^{®}$ - a unified object-oriented language for physical systems modeling. Tutorial, http://www.modelica.org/documents/ModelicaTutorial14.pdf, Dec. 2000.

[46] 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.

[47] K.S. Cetin, M.H. Fathollahzadeh, N. Kunwar, H. Do, P.C. Tabares-Velasco, Development and validation of an hvac on/off controller in energyplus for energy simulation of residential and small commercial buildings, Energy Build. 183 (2019) 467–483, https://doi.org/10.1016/j.enbuild.2018.11.005.

[48] T. Blochwitz, M. Otter, M. Arnold, C. Bausch, C. Clauß, H. Elmqvist, A. Junghanns, J. Mauss, M. Monteiro, T. Neidhold, et al., The functional mockup interface for tool independent exchange of simulation models, in: Proceedings of the 8th International Modelica Conference, Linköping University Press, 2011, pp. 105–114.

[49] 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.

[50] S. Brandi, A. Gallo, A. Capozzoli, A predictive and adaptive control strategy to optimize the management of integrated energy systems in buildings, Energy Rep. 8 (2022) 1550–1567, https://doi.org/10.1016/j.egyr.2021.12.058.

[51] 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.

[52] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: a next-generation hyperparameter optimization framework, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 2623–2631.

[53] J. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, Algorithms for hyper-parameter optimization, in: Proceedings of the 24th International Conference on Neural Information Processing Systems, NIPS'11, Curran Associates Inc., Red Hook, NY, USA, 2011, pp. 2546–2554, https://proceedings.neurips.cc/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf.

[54] Q. Xin, 3 - optimization techniques in diesel engine system design, in: Q. Xin (Ed.), Diesel Engine System Design, Woodhead Publishing, 2013, pp. 203–296.

[55] M. Zelany, A concept of compromise solutions and the method of the displaced ideal, Comput. Oper. Res. 1 (3) (1974) 479–496, https://doi.org/10.1016/0305-0548(74)90064-1.

[56] S.L. Smith, P.-J. Kindermans, C. Ying, Q.V. Le, Don't decay the learning rate, increase the batch size, preprint, arXiv:1711.00489, 2017.

[57] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, S. Levine, Soft actor-critic algorithms and applications, arXiv:1812.05905, 2019.

[58] H. Li, P. Chaudhari, H. Yang, M. Lam, A. Ravichandran, R. Bhotika, S. Soatto, Rethinking the hyperparameters for fine-tuning, arXiv:2002.11770, 2020.

[59] M. Kaya, H.c. Bilge, Deep metric learning: a survey, Symmetry 11 (9) (2019), https://doi.org/10.3390/sym11091066, https://www.mdpi.com/2073-8994/11/9/1066.

[60] A. Visioli, Modified anti-windup scheme for pid controllers, IEE Proc., Control Theory Appl. 150 (2003) 49–54, https://doi.org/10.1049/ip-cta:20020769.

[61] 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.

[62] M. Verma, R. Kaler, M. Singh, Sensitivity enhancement of passive infrared (pir) sensor for motion detection, Optik 244 (2021) 167503, https://doi.org/10.1016/j.ijleo.2021.167503, https://www.sciencedirect.com/science/article/pii/S0030402621011256.

[63] IEA, Solcast API, https://www.solcast.com/, 2024. (Accessed 29 October 2024), Solcast: Solar API and weather forecast.

[64] 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.

[65] M. Esrafilian-Najafabadi, F. Haghighat, Transfer learning for occupancy-based hvac control: a data-driven approach using unsupervised learning of occupancy profiles and deep reinforcement learning, Energy Build. 300 (2023) 113637, https://doi.org/10.1016/j.enbuild.2023.113637.

[66] K. Kadamala, D. Chambers, E. Barrett, Enhancing hvac control systems through transfer learning with deep reinforcement learning agents, Smart Energy 13 (2024) 100131, https://doi.org/10.1016/j.segy.2024.100131.

[67] 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.

[68] 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.

[69] V. Gullapalli, A stochastic reinforcement learning algorithm for learning real-valued functions, Neural Netw. 3 (6) (1990) 671–692, https://doi.org/10.1016/0893-6080(90)90056-Q.

[70] D. Azuatalam, W.-L. Lee, F. de Nijs, A. Liebman, Reinforcement learning for whole-building hvac control and demand response, Energy AI 2 (2020) 100020, https://doi.org/10.1016/j.egyai.2020.100020.

[71] R. Bellman, Dynamic programming, Science 153 (3731) (1966) 34–37, https://doi.org/10.1126/science.153.3731.34, https://science.sciencemag.org/content/153/3731/34.full.pdf.

[72] 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.

[73] 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.

[74] S.P. Singh, T.S. Jaakkola, M.I. Jordan, Learning without state-estimation in partially observable Markovian decision processes, in: Proceedings of the Eleventh International Conference on International Conference on Machine Learning, ICML'94, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1994, pp. 284–292.

[75] G. Pinto, M.S. Piscitelli, J.R. Vázquez-Canteli, Z. Nagy, A. Capozzoli, Coordinated energy management for a cluster of buildings through deep reinforcement learning, Energy 229 (2021) 120725, https://doi.org/10.1016/j.energy.2021.120725.

[76] Y. Himeur, M. Elnour, F. Fadli, N. Meskin, I. Petri, Y. Rezgui, F. Bensaali, A. Amira, Next-generation energy systems for sustainable smart cities: roles of transfer learning, Sustain. Cities Soc. 85 (2022) 104059, https://doi.org/10.1016/j.scs.2022.104059.

[77] M.E. Taylor, P. Stone, Transfer learning for reinforcement learning domains: a survey, J. Mach. Learn. Res. 10 (56) (2009) 1633–1685, http://jmlr.org/papers/v10/taylor09a.html.

[78] 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.

[79] L. Di Natale, B. Svetozarevic, P. Heer, C. Jones, Physically consistent neural networks for building thermal modeling: theory and analysis, Appl. Energy 325 (2022) 119806, https://doi.org/10.1016/j.apenergy.2022.119806.