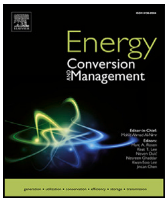
Effective pre-training of a deep reinforcement learning agent by means of long short-term memory models for thermal energy management in buildings

Contents lists available at ScienceDirect

Energy Conversion and Management

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



Effective pre-training of a deep reinforcement learning agent by means of long short-term memory models for thermal energy management in buildings

Davide Coraci, Silvio Brandi, Alfonso Capozzoli [*]

Politecnico di Torino, Department of Energy, TEBE research group, BAEDA Lab, Corso Duca degli Abruzzi 24, Torino, 10129, Italy

ARTICLE INFO

Keywords:
Deep reinforcement learning
LSTM neural network
Data-driven building modelling
Real-time deployment
Heating system
Energy management

ABSTRACT

Recently, deep reinforcement learning has emerged as a popular approach for enhancing thermal energy management in buildings due to its flexibility and model-free nature. However, the time-consuming convergence of deep reinforcement learning poses a challenge. To address this, offline pre-training of deep reinforcement learning controllers using physics-based simulation environments has been commonly employed. However, developing these models requires significant effort and expertise. Alternatively, data-driven models offer a promising solution by emulating building dynamics, but they struggle to predict previously unseen patterns. Therefore, this paper introduces a strategy to effectively train and deploy a deep reinforcement learning controller by means of long short-term memory neural networks. The experiments were carried out using an EnergyPlus simulation environment as a proxy of a real building. An automatic and recursive procedure is designed to determine the minimum amount of historical data required to train a robust data-driven model which mimics building dynamics. The trained deep reinforcement learning agent meets safety requirements in the simulation environment after two and a half months of training. Additionally, it reduces indoor temperature violations by 80% while consuming the same amount of energy as a baseline rule-based controller.

1. Introduction

The building sector is responsible for approximately 40% of total energy consumption worldwide [1]. In this framework, Renewable Energy Sources (RES) are globally experiencing a significant penetration, in particular solar Photovoltaic (PV) [2] and wind energy [3]. At building level, the introduction of various incentive programs has supported the penetration of PV systems, Thermal Energy Storage (TES) and batteries in integrated Heating, Ventilation and Air Conditioning (HVAC) systems. As a result, building energy management has been recognised as a crucial factor to optimise the operation of energy systems [4] but it has become a challenging task as energy systems in buildings are more complex and integrated [5]. Considering that HVAC systems represent the most energy-intensive building use, significant improvements have been implemented to enhance their energy efficiency through better energy management [6] and to reduce the energy cost associated with its operation, in particular during the current period where price volatility for raw materials involved in electricity production led to an increase in the average electricity price [7].

Nowadays, building energy systems are typically managed through sub-optimal strategies which are not the result of optimisation processes [8]. In particular, the control of HVAC systems is typically handled through controllers based on rules exploiting pre-determined schedules based on Proportional-Integrative-Derivative (PID) or ON–OFF logic to track the desired setpoints [5].

In this context, the increasing penetration of Internet of Things (IoT) devices and Information and Communication Technologies (ICT) has opened the door to a great availability of building-related data. The information and knowledge stored on building historical data can be leveraged by advanced control strategies based on Artificial Intelligence (AI) [9] to characterise the present and expected future states of buildings and their energy systems [10,11].

Among advanced control strategies, Model Predictive Control (MPC) has gained wide attention in the building industry for its capability to optimise the operation of energy systems over a certain receding time horizon, accounting for current system behaviour as well as its possible evolution [12,13]. The current state of the art concerning MPC applications in literature proves its excellent capabilities in optimising the operation of Integrated Energy Systems (IES) to reduce building energy consumption [14,15] and enhance the efficiency of energy systems [16]. MPC controllers have demonstrated their ability to handle PV electricity generation predictions as well as the energy exchange with the electrical grid according to price signals [17,18]. Moreover,

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

https://doi.org/10.1016/j.enconman.2023.117303

Available online 24 June 2023

Nomenclature

α	Boltzmann temperature coefficient
β	Temperature-term weight of reward function
δ	Power term-weight of reward function
γ	Discount factor
μ	DRL controller learning rate
θ	Temperature-term prize of reward function
$E_{DRL,week}$	Weekly DRL energy consumption [kWh]
$E_{RBC_{model},daily}$	Daily RBC predicted energy consumption [kWh]
$E_{RBC_{model},week}$	Weekly RBC predicted energy consumption [kWh]
$P_{max,heating}$	Maximum supplied heating power [kW]
r	Reward function
SP_{INT}	Indoor air temperature setpoint [°C]
$SP_{T_{SUPP}}$	Supply water temperature setpoint [°C]
T_{INT}	Indoor air temperature [°C]
T_{LOW}	Lower threshold limit of temperature comfort range [°C]
T_{RET}	Return water temperature [°C]
T_{UPP}	Upper threshold limit of temperature comfort range [°C]
T_{viol}	Cumulated sum of temperature violations [°C]
$x_{power}(t)$	Fraction of the nominal heating power

Acronyms

AHUs	Air Handling Units
AI	Artificial Intelligence
BESS	Battery Energy Storage System
BCVTB	Building Control Virtual Test Bed
DDPG	Deep Deterministic Policy Gradient
DNNs	Deep Neural Networks
DQN	Deep Q-Network
DRL	Deep Reinforcement Learning
HVAC	Heating, Ventilation and Air Conditioning
ICT	Information and Communication Technologies
IES	Integrated Energy Systems
IoT	Internet of Things
LSTM	Long Short-Term Memory
MAPE	Mean Absolute Percentage Error
MPC	Model Predictive Control
PID	Proportional-Integrative-Derivative
PPD	Predicted Percentage of Dissatisfied
PV	Photovoltaic
RBC	Rule-Based Controller
RES	Renewable Energy Sources
RL	Reinforcement Learning
RMSE	Root Mean Squared Error
SAC	Soft Actor-Critic
TES	Thermal Energy Storage
TL	Transfer Learning
TPE	Tree-structured Parzen Estimator

the implementation of MPC proved to be effective in enabling the sharing of energy in a community of buildings [19]. However, the real-world deployment of MPC is limited by its model-based nature, since it requires an accurate characterisation of the building and energy system to be optimised [20,21]. This modelling effort could be challenging since each building represents a singular entity requiring the definition of a proper description to be employed during the control strategy optimisation [18]. Consequently, MPC controllers have not been widely adopted in the building industry despite promising results [6,22].

In this context, Reinforcement Learning (RL) has emerged as an alternative approach to MPC to revolutionise the implementation of advanced controllers in buildings [23]. The interest in RL-based control strategies has increased especially because it follows a model-free approach, where an agent directly learns the optimal control policy by interacting with the system through a trial-and-error approach [24].

Among the RL-based control strategies, the most frequently implemented refers to the Q-Learning algorithm [25]. However, control applications in buildings are characterised by a high number of states and actions together with a high complexity inherent to the exploration of non-linear relationships in buildings [26]. Therefore, to make the application of advanced controllers in buildings effective, a variant of RL is introduced, named Deep Reinforcement Learning (DRL), in which the control policy is approximated employing Deep Neural Networks (DNNs) [27]. In the next subsection, relevant applications of DRL controllers are discussed before defining the motivation and novelties of this paper.

1.1. Related works on reinforcement learning control strategies implemented in buildings

In the context of building energy management, DRL controllers have been adopted to manage the supply water temperature of generation systems [28,29], supply water mass flow rate [30], thermal storage temperature setpoint [31,32], chiller operation [33], thermal storage charging and discharging [18,34], Battery Energy Storage System (BESS) operation [35], indoor temperature setpoint [36,37], fan speed [38], valve position [30] and lighting devices [39].

In [6,28] a DRL control agent was trained and deployed on an EnergyPlus model, to optimise the supply water temperature of a heating system for an office building. The results showed energy savings ranging between 5 and 12% with respect to a Rule-Based Controller (RBC), with enhanced performance in indoor temperature control.

Schreiber et al. [30] developed a Modelica environment to train and compare two RL algorithms, a Deep Q-Network (DQN) agent and a Deep Deterministic Policy Gradient (DDPG) agent, to control the supply water temperature and the mass flow rate of a chiller plant, as well as valve positions. Both algorithms showed better performances when compared to a demand-oriented baseline controller.

The same strategies based on DRL were compared in [40] for managing the indoor temperature setpoint in a multi-zone residential HVAC system. The authors demonstrated that the control agent based on DDPG in a simulated building environment can reduce energy consumption by 15% compared to DQN and ensure 79% and 98% reduction in temperature violations when compared respectively to DQN and RBC.

In [31,41] the authors achieved an energy saving between 4 and 10% while maintaining desired indoor temperature conditions by managing the TES water temperature with a DRL controller during the cooling season.

Brandi et al. [18] compared the performance of DRL and MPC controllers in managing the charging/discharging process of a TES in a cooling system serving an office building to minimise electricity cost. In particular, two different DRL training strategies were implemented, named online DRL and offline DRL. The online DRL control strategy achieved similar performance compared to MPC after approximately 4 weeks, representing a possible solution to enhance the scalability of advanced controllers as the model-based or the offline DRL, since it requires the definition of a surrogate model of the building to be controlled.

Moreover, DRL control agents have shown excellent capabilities in providing services to the grid (i.e., peak shaving and load shifting) by optimising the operation of energy systems in a district of buildings [42,43] or the operation of integrated electricity and natural gas systems [44]. Zhang et al. [44] effectively implemented DRL to coordinate the operation of a power-to-gas unit and generators in an integrated electricity and natural gas system to increase economic profit while shaving electricity peaks in the net load demand curve. The implementation of a DRL control agent allowed the adaptive determination of the conversion ratio of wind power, power-to-gas and gas turbine operations adjusting the energy system operation according to dynamic factors such as wind power production, wholesale gas price and power demand uncertainties.

The analysis of the current scientific literature suggests that DRL is a promising technique since it employs a trial-and-error approach while interacting with the energy system to learn an optimal control policy. Moreover, DRL implementation constitutes a valuable asset to optimise the operation of IES addressing a number of objectives while handling high-dimensional control problems (e.g., real-world problems) also characterised by stochastic behaviour.

Despite its proven effectiveness, some limitations related to the scalability of DRL agents in real-world implementations emerge from the literature. In addition, DRL controllers require a considerable amount of time to converge to near-optimal solutions which stability is not always guaranteed. In ideal conditions, a model-free DRL agent should be directly implemented in a real building gradually learning the optimal control policy through a trial and error approach [18]. However, the convergence process requires several interactions and may lead to the exploration of extreme conditions of the controlled environment resulting in poor control performance, especially during the first period of deployment [45]. As a consequence, the direct deployment of a DRL agent in a real building is unfeasible due to economic and safety reasons. To overcome this limitation, a common approach explored in literature leverages surrogate models of building and energy systems to offline pre-train the DRL controller. In this context, the majority of researchers developed detailed engineering models employing software such as Modelica [46] and EnergyPlus [47]. However, the definition of a building surrogate engineering model represents a time-consuming task that also requires expert knowledge and detailed information.

A different approach involves the use of data-driven architectures trained on historical monitoring data as surrogate models of building and system dynamics. Following this approach, Zou et al. [38] demonstrated how a DRL agent could learn an optimal control policy by interacting with Long Short-Term Memory (LSTM) models that emulate the building dynamics. The authors implemented this approach for a building served by a HVAC system which consists of three Air Handling Units (AHUs), achieving a 30% energy saving and maintaining the Predicted Percentage of Dissatisfied (PPD) at 10% when compared to a baseline controller. Although the authors in [38] effectively managed to employ a data-driven model to perform offline the training of a DRL controller, the proposed approach can be further extended. One limitation of this study was the use of a LSTM model for both training and testing the DRL agent. While the authors employed as first LSTM models trained on real building data to train DRL algorithms, the generalisability of the approach proposed is limited by the fact that the trained agent was only evaluated on a LSTM model. In the present paper, the performance of the DRL trained on LSTM is tested on an EnergyPlus simulation environment conceived as a proxy of a real building to fully understand the potential benefits and limitations of using data-driven models to train DRL agents for building control.

1.2. Novelty and contributions of the paper

The development of data-driven models of building dynamics is a topic widely explored in the current scientific literature [48,49]. However, models specifically built for training offline DRL agents present particular challenges worth to be investigated. In fact, the values of the output variable of data-driven models of building dynamics are strongly dependent on the control action being implemented [50].

Since during training DRL agents explore different control trajectories, the data-driven model should be capable to provide a physically robust emulation of the behaviour of the building system in different conditions. However, one of the major drawbacks of data-driven models is their inability to correctly represent patterns that are not present in the training data. For example, a model trained on data collected in a building where traditional logic is implemented may no longer be effective in properly emulating building dynamics when an advanced control strategy such as DRL is implemented. Consequently, the control policy learned by the DRL agent may be sub-optimal. However, to effectively verify the performance of a control policy learned through a data-driven model, it is fundamental to implement the DRL controller on the original system from which the data-driven model was built. Despite its limitation, it is possible to determine if through this approach the data-driven model is effective in emulating the behaviour of the physical system enabling the DRL to converge to a near-optimal solution.

A further key aspect to consider in the development of data-driven model for the estimation of building dynamics is the amount of historical data to build a robust model. The amount of data in terms of both number of variables and time period necessary to build data-driven model of building dynamics can affect the scalability of DRL agents trained through this approach.

The present paper introduces a novel approach to pre-train a DRL agent for building energy management by means of data-driven models of building dynamics. The proposed approach is conceived considering the requirements and limitations of a real-world context such as the necessity to rapidly deploy advanced control strategies in buildings with limited availability of historical data.

The proposed approach leverages a LSTM neural network-based model as a surrogate for the building dynamics to pre-train in an offline fashion a DRL agent based on Soft Actor-Critic (SAC) algorithm. Conceptual details for LSTM can be found in [51,52], while theoretical foundation regarding advanced controller can be found in [24,53] for DRL controller and in [42,54] for SAC algorithm. The data employed to train this surrogate model are synthetically generated from a building model implementing first a traditional control logic and then directly the DRL-based control strategy. The proposed approach includes periodic re-training of both the surrogate model and the DRL agent, coupled with a safety control strategy to evaluate whether the implemented DRL logic is able to achieve acceptable performance. The model of building dynamics and the DRL controller are trained to gradually converge as long as new data are made available. Eventually, hyperparameters optimisation routines are included at different steps of the proposed approach in order to ensure adequate performances of the machine learning model implemented.

Thus, the present paper aims to demonstrate how LSTM models can be exploited in an effective way to train and deploy DRL agents for building energy management. In this context, the comparison of the selected DRL controller with other advanced control strategies is not performed since it deviates from the scope of the present work.

Despite the proposed approach is evaluated in a simulation environment, the experiments are carried out to effectively emulate the implementation within a physical system. In this framework, the main barriers related to the implementation of the proposed approach in a real-world system are identified and potentially addressed.

2. Method of the proposed approach

The core of the proposed controller relies not on its formulation but rather on the approach applied to perform its training and deployment. The approach introduced in the present paper aims to investigate the applicability of data-driven models to perform offline training of the

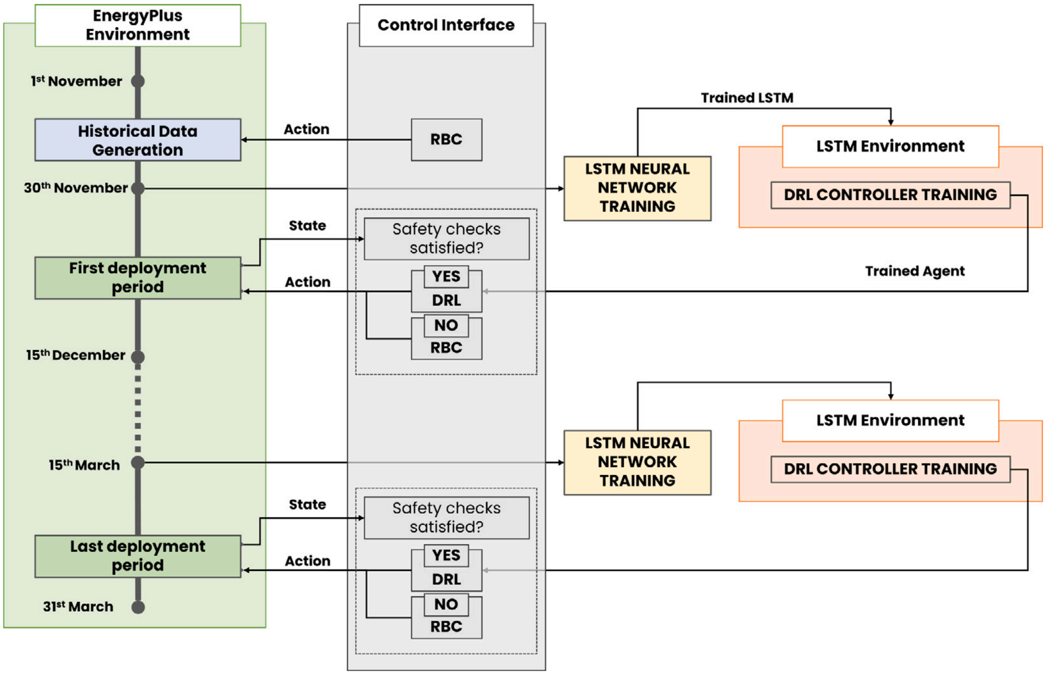


Fig. 1. Framework of the proposed approach for DRL training and deployment.

DRL agent. In a real-world context, the data-driven model should be trained on existing historical data collected in-field. However, since in a real-world context access to historical data may be limited, the proposed approach is conceived to be effective even when a limited amount of data is initially available.

The experiments are carried out in a simulation environment, composed of three main elements. The first element is represented by a simulation environment (identified as EnergyPlus environment) based on OpenAI Gym [55] combining EnergyPlus [47] and Python through Building Control Virtual Test Bed (BCVTB) [56]. Python and EnergyPlus dynamically exchange information considering a simulation timestep of 15 min. The EnergyPlus environment is conceived as a proxy of a real building managed through a traditional rule-based control strategy and equipped with a monitoring infrastructure collecting energy-related data. As a consequence, differently from traditional application of DRL controllers in simulation environments [9], the EnergyPlus environment is not employed to perform multiple episodes to train the DRL agent while the simulation outputs, resulting from the implementation of a control action, are treated as monitoring data collected in real field. The second element is a different simulation environment (identified as LSTM environment) based on OpenAI Gym including a LSTM model developed with the PyTorch library [57]. LSTM is employed as data-driven model to emulate the dynamics of the building simulated in EnergyPlus in the first environment. The third element is a control interface that interacts with the two previously described environments. The interface is developed in Python and includes the implemented baseline RBC and DRL control strategy. The proposed DRL control agent is developed employing TensorFlow [58] and Stable Baselines libraries [59].

The proposed approach for DRL training and deployment unfolds through three main steps as shown in Fig. 1: (i) LSTM neural network training, (ii) DRL controller training and (iii) DRL controller deployment.

The first step of the proposed approach involves the training of the data-driven model of the building dynamics based on LSTM architecture. In a real-world context, the data required to perform this step would be collected from available measurements. However, since the present experiment is carried out exclusively in a simulation environment, those data are generated by simulating the behaviour of the building in the EnergyPlus environment while implementing the baseline rule-based control strategy for a period of one month. The baseline RBC strategy is chosen since it represents the most common control approach implemented in real buildings and that can be upgraded through the introduction of advanced strategies like DRL. This procedure can be considered a preliminary step and in the next sections it is identified as *Historical data generation* process.

The trained model is then used in the second step to train offline the DRL agent for multiple episodes in the LSTM environment. In the third step, the controller is deployed for a specific period of time defined as deployment period in the EnergyPlus environment to emulate the implementation of the DRL agent in a real building. During the deployment period, safety checks are implemented to ensure that the performance of the controller is satisfactory. A deployment period is set with a length of 15 days and starts at the end of the training period. At the end of the deployment period, the data collected from the proxy of the real building (i.e., EnergyPlus environment) with the DRL are added to those obtained during the first month of RBC implementation to re-train the data-driven model. This updating process (except for *Historical data generation*) is repeated recursively after each deployment period (i.e., 15 days) until the end of the heating season. Thanks to this approach the data-driven model of the building dynamics is constantly updated to better emulate the behaviour of the controlled building. Section 3 describes in detail the different steps of the proposed approach. Eventually, a simulation of the whole heating season is carried out in the EnergyPlus environment implementing only the baseline RBC strategy. These results are employed to demonstrate the improvement achievable by implementing an advanced controller trained and deployed according to the proposed framework. A comparison with other advanced control strategies is not performed since it deviates from the scope of the present experiment.

The proposed approach is implemented for an office building located in Torino, Italy. The building consists of five heated floors with a net building heated surface of about $9300\,\mathrm{m}^2$. The average transmittance values for the opaque and transparent components of the envelope are $1.084\,\mathrm{W/(m^2\,K)}$ and $2.921\,\mathrm{W/(m^2\,K)}$ respectively. The aspect ratio (i.e., the ratio between the heat transfer surface to gross volume) is 0.25 m^{-1}. The occupancy schedule is defined based on the actual office opening and closing times. Every weekday, except

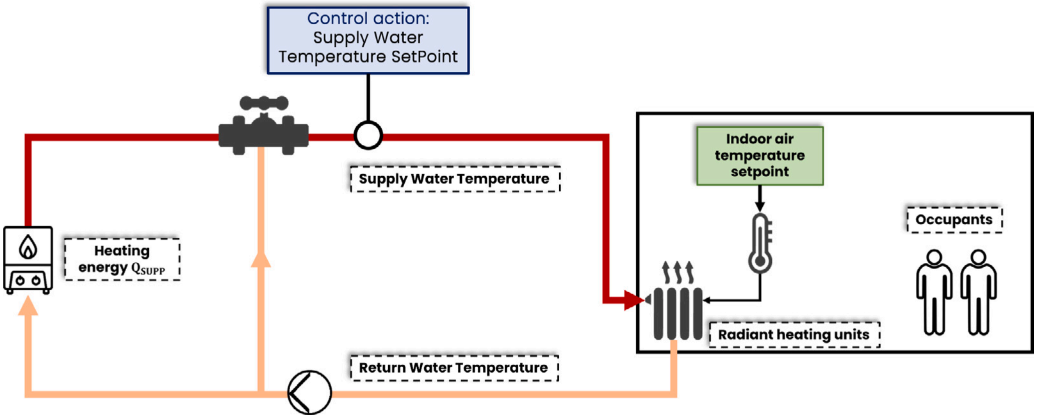


Fig. 2. Schematic of the heating system analysed.

Table 1
Start time window and indoor temperature conditions employed to switch ON the heating system.

Combination	Time period	Indoor temperature
1	$0:00 \leq t < 1:00$	$T_{INT} - T_{UPP} \geq 8$ °C
2	$1:00 \leq t < 2:00$	$T_{INT} - T_{UPP} \geq 6$ °C
3	$2:00 \leq t < 3:00$	$T_{INT} - T_{UPP} \geq 5$ °C
4	$3:00 \leq t < 4:00$	$T_{INT} - T_{UPP} \geq 3$ °C
5	$4:00 \leq t < 5:00$	$T_{INT} - T_{UPP} \geq 2$ °C
6	$t \geq 5:00$	$T_{INT} - T_{UPP} \geq 0$ °C

Sundays, the office is occupied from 7:00 to 19:00. Fig. 2 introduces a simplified scheme of the analysed heating system. The heating system consists of a single hot water loop that includes a 470 kW nominal power gas-fired boiler. The indoor environment is heated through radiators.

The controllers have the ability to manipulate the supply water temperature setpoint while maintaining a constant hot-water mass flow rate. The objective of the implemented DRL controller is to minimise the amount of thermal energy supplied to the water while ensuring that the indoor air temperature remains within the desired acceptability range during occupancy periods. The acceptability range is defined as [−1, +1] °C from the indoor temperature setpoint of 21 °C.

The RBC strategy combining rule-based and climatic-based logic for the modulation of the supply water temperature is employed as baseline. The supply water temperature can vary linearly from 40 °C to 70 °C with outdoor temperature ranging between 12 °C and −5 °C. This strategy is employed until one hour before occupants leave the building when the heating system is turned off to exploit the heat stored in the thermal mass of the building. The starting time of the heating system is determined according to the indoor air temperature value and the time period. Table 1 reports the time periods with the corresponding conditions on indoor temperature related to this control logic. The six combinations of start time window and indoor temperature conditions are determined through a sensitivity analysis to reduce the temperature violations during the early stages of the occupancy period. Additionally, during occupancy periods the heating system is turned off when the indoor temperature reaches the upper threshold of the acceptability range T_{UPP} (i.e., 22 °C) and is turned on if the temperature falls below the lower threshold T_{LOW} (i.e., 20 °C). The heating system is turned off on Sundays.

3. Implementation of the method

This section describes in detail the steps of the proposed approach characterising the training and deployment of a DRL agent by means of data-driven models of building dynamics. Experiments are carried out for a heating season lasting 5 months (i.e., from November to March) considering the reference weather data available in EnergyPlus for Torino, Italy.

The preliminary step to the application of the proposed approach is defined *Historical data generation*. According to this procedure, a simulation is carried out in the EnergyPlus environment for the first month of the heating season (i.e., November) implementing the baseline RBC strategy. This step is conceived to emulate the collection of a limited amount of monitored data from a real-world building.

3.1. Long short-term memory neural network training

The first step of the proposed approach aims to perform the training of the LSTM model of building dynamics. LSTM networks employ memory blocks that substitute conventional hidden layer neurons enhancing their capability to handle long-term and short-term dependency problems [60] while detecting hidden features and invariant structures [61]. The goal of the present LSTM model is to estimate the evolution of indoor air temperature between two successive control steps given a set of influencing variables and the implemented control action (i.e., supply heating power), as shown in Fig. 3.

The input variables are arranged within 48 lookback sequences. Fig. 3 and Table 2 list the variables included in each sequence. These variables are selected considering their ease of being monitored and collected in field. Time-related variables, such as the *Hour of the day* and *Day of the week*, are required to inform the model about the usage profiles of the building, governing endogenous loads. *Outdoor air temperature* and *Direct solar radiation* are included as the major influencing meteorological factors influencing exogenous loads. Previous *Indoor air temperature* values are given as input to ensure that the data-driven model receives information concerning the indoor conditions of the controlled environment. Eventually, *Supplied heating power* provides key information about the operation of the heating system. The direct metering of the heating power supplied to a thermal zone is a non-trivial task in a real building. However, for the present application, this variable can be derived by the water mass flow rate delivered to zone terminals and supply and return water temperatures of the heating loop. Time variables are encoded using sine and cosine transformation while the other input variables are re-scaled through a min–max normalisation.

After the training process, the performance of the model is tested in a closed-loop configuration as shown in Fig. 3. According to this procedure, the LSTM is tested receiving as input the estimated previous temperature and the other input variables reported in Fig. 3 (e.g., weather and supplied heating power). Through this approach it is possible to effectively evaluate the capability of the model in simulating the behaviour of the analysed system.

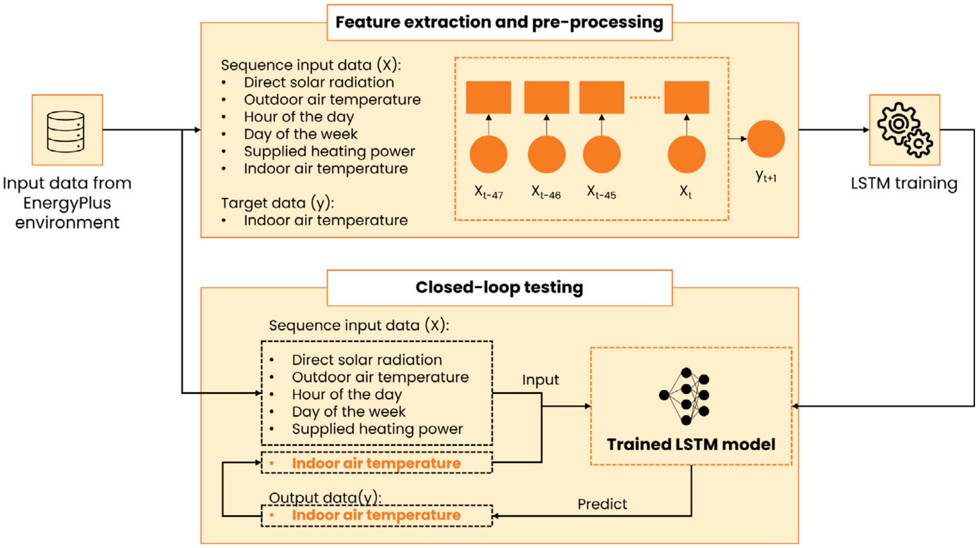


Fig. 3. Development of the LSTM model and details about the selected input variables.

Table 2
Variables included in input sequences of the LSTM model.

Variable	Min value	Max value	Unit
Outdoor air temperature	−10.4	17.6	°C
Direct solar radiation	0	714	W/m^2
Hour of the day	0	23	h
Day of the week	1	7	–
Supplied heating power	0	470	kW
Indoor air temperature	13.0	25.0	°C

Table 3
Values and range of fixed/optimised LSTM model hyperparameters.

LSTM hyperparameter	Value	Step
Batch size	[80, 120]	1
Learning rate	[0.0001, 0.01]	0.0001
Number of hidden layers	[2, 4]	1
Number of neurons per layer	[16, 32]	1
Lookback	48	–
Training epochs	30	–
Optimiser	Adam	–

LSTM models are characterised by several hyperparameters which require appropriate tuning. Therefore, during the LSTM training phase the values of the most important hyperparameters have been optimised through an automated procedure by employing the Optuna library [62]. Optuna is an open-source Python library that automates the search for optimal hyperparameters configuration in machine learning-based models. In particular, Optuna requires in input the set of hyperparameters to be optimised and their acceptability ranges, the objective function to be minimised or maximised and the sampling algorithm employed in the optimisation process.

The optimisation of LSTM hyperparameters is carried out on batch size, LSTM learning rate, number of hidden layers, and number of neurons per hidden layer. Table 3 reports for each hyperparameter the range and the incremental step employed in the optimisation process. Moreover, Table 3 shows the values of the hyperparameters not involved in the optimisation. The hyperparameter optimisation is carried out to minimise the Root Mean Squared Error (RMSE) and employs the Tree-structured Parzen Estimator (TPE) as Optuna sampling algorithm [63].

The best hyperparameters configuration of the LSTM is determined by evaluating the objective function in a closed-loop configuration. According to the closed-loop strategy, the LSTM model is tested employing its own predictions of indoor air temperature value as inputs for subsequent timesteps. The values of the other variables provided as inputs to the LSTM model are taken from the original training dataset. As a consequence, the weather and the control action are exactly the same as observed by the network during training, while the indoor air temperature values evolves according to network predictions. This approach allows for the evaluation of potential deviations in the model estimations by comparing the predictions made by the model and the actual data considering the same boundary conditions.

According to the proposed approach, the LSTM training is performed multiple times. Initially, the model is trained only considering the data collected through the *Historical data generation* process. Successively, the training size of the training dataset is constantly increased to include data generated during DRL deployment periods. One hundred different sets of LSTM hyperparameters are compared to determine the best configuration by means of an automated hyperparameter optimisation procedure that is carried out each time the LSTM model is re-trained.

3.2. Deep reinforcement learning controller training

The second step of the proposed approach aims at training the DRL controller by employing the trained LSTM model, as shown in Fig. 4.

The training process of DRL agent is implemented offline in the LSTM environment by repeating a training episode multiple times to promote control policy refinement. In the present application, a training episode does not have a fixed length. According to the proposed approach, the DRL training process is performed multiple times. Initially, it is performed after the *Historical data generation* process. Successively, it is performed after each deployment period. At each re-training the length of a training episode is increased to include the new information collected during a deployment period in the EnergyPlus environment which is treated as a real-world building in this study. As a consequence, a new training episode is defined at each re-training to include an increasing period of the heating season being considered.

The proposed DRL controller can be defined through its main features namely action-space, state-space and reward function. The action space includes the set of possible control actions that can be performed by the agent. In this work, the action picked by the agent consists in the percentage fraction of the nominal heating power (i.e., 470 kW) supplied to the building between two control timesteps. This action is selected instead of supply water temperature setpoint since the LSTM model developed in the previous step employed the supplied heating power to predict the evolution of indoor air temperature. Since SAC is

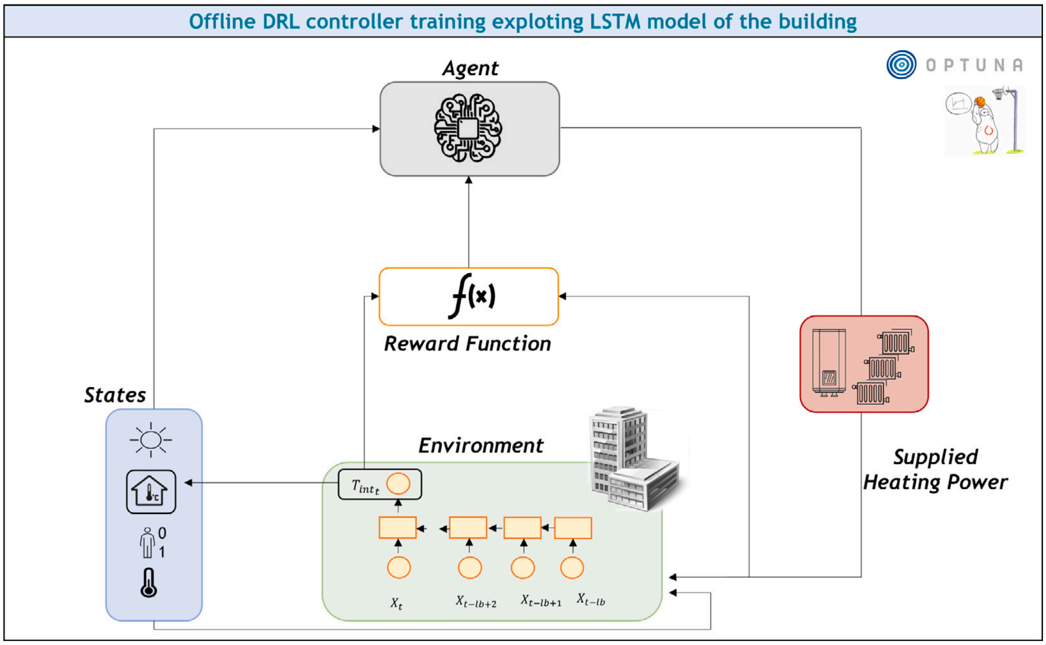


Fig. 4. DRL training process exploiting LSTM model of the building dynamics.

Table 4
Variables included in the DRL state-space.

Variable	Min value	Max value	Unit	Timestep
Outdoor air temperature	−10.4	17.6	°C	t
Direct solar radiation	0	714	W/m^2	t
$SP_{INT} - T_{INT}$	−5	3	°C	t, t−1, t−2, t−3
Time to occupancy start	0	144	–	t
Time to occupancy end	0	48	–	t

selected as DRL algorithm, the action space is continuous and limited between 0 (i.e., 0% of the nominal heating power) and 1 (i.e., 100% of the nominal heating power). Moreover, a filter is applied to the action picked by the DRL agent to enforce 0% of the nominal heating power for each action lower than a minimum threshold.

The state-space comprises a series of observations provided as inputs to the agent. In this work, the DRL state-space includes eight features, reported in Table 4, together with their lower and upper bounds used to re-scale the state space through a min–max normalisation before providing the variables to DNNs.

Outdoor Air Temperature and *Direct Solar Radiation* are included in the state-space, as they are the most influencing ambient variables affecting building heating energy consumption and indoor temperature. The information related to the *Indoor temperature* is linked with the formulation of the reward function since it is expressed as the difference between the indoor temperature setpoint SP_{INT} and real indoor temperature T_{INT}. These values are included in the state-space at the current control time step t and for 3 lagged values in the past (15, 30 and 45-min lag respectively). Moreover, information on the presence of occupants in the building is provided through two different variables, *Time to occupancy start* and *Time to occupancy end*. These two variables define the time left for the subsequent change in the occupancy pattern. When the building is not occupied, *Time to occupancy start* represents the number of timesteps left before occupants' arrival time. During occupancy periods, the value of this variable is zero. Conversely, when the building is occupied, *Time to occupancy end* represents the number of timesteps to occupants' leaving time. During off-occupancy periods this variable is equal to zero.

The reward function measures the performance of the controller after selecting an action at each time step. In this case, the reward includes two terms, a power-related term and a temperature-related term, since the agent aims to minimise energy consumption while maintaining the indoor temperature within the comfort range. Two coefficients (δ and β, respectively) have been introduced to weigh the importance of the two terms of the reward function.

$$R = -\delta * P(t) + \beta * r_T \tag{1}$$

The first term is proportional to the heating power $P(t)$ supplied to the building and it is introduced to minimise energy consumption. On the other hand, the temperature term is introduced to maintain the indoor air temperature within an acceptability range of ±1 °C from the desired setpoint of 21 °C during occupancy periods. The temperature term is evaluated only when the building is occupied and has three different formulations depending on the indoor temperature values, as reported in Eq. (2).

$$r_T = \begin{cases} -(|SP_{INT} - T_{INT}|)^3 & \text{if } T_{INT} < T_{LOW} \text{ or } T_{INT} > T_{UPP} \\ -(T_{INT} - SP_{INT}) & \text{if } SP_{INT} < T_{INT} < T_{UPP} \\ \theta & \text{if } T_{LOW} \leq T_{INT} \leq SP_{INT} \end{cases} \tag{2}$$

The temperature term is conceived to encourage the controller to maintain indoor temperature values as close as possible to the lower band of the acceptability range to reduce heating energy consumption. The reward function has a positive value (i.e., temperature-term prize, indicated as θ) when the temperature value falls in the range [20, 21] °C to promote the exploration of this condition. The values of θ and of the two reward weights are obtained from the hyperparameters optimisation procedure.

DRL controller performances are influenced by numerous hyperparameters (e.g., DRL learning rate μ, discount factor γ) that require adequate tuning. For this reason, the automated hyperparameter optimisation procedure is carried out by means of the Optuna library similarly to the LSTM training phase. During the DRL training phase, the optimisation of the following hyperparameters is performed: reward weights (δ, β, θ), DRL learning rate (μ) and discount factor (γ). The hyperparameters optimisation is carried out to identify the best configuration leading to an agent capable to identify the best trade-off between energy consumption E_{cons}, measured in MWh, and the cumulated sum of temperature violation T_{viol}, measured in °C. In particular, a temperature violation is determined by calculating the

Table 5
Values and range of fixed/optimised DRL controller hyperparameters.

DRL hyperparameter	Value	Step
δ	[0.002, 0.01]	0.0005
β	[2, 8]	0.5
θ	[0.01, 0.05]	0.005
Discount factor γ	[0.9, 0.95, 0.99]	–
Learning rate μ	[0.001, 0.005]	0.001
Boltzmann temperature coefficient α	0.1	–
Batch size	128	–
Number of hidden layers	4	–
Number of neurons per hidden layer	64	–
Training episodes	20	–

absolute difference between the indoor temperature T_{INT} and the lower T_{LOW} or upper limit T_{UPP} of the acceptable temperature range [20, 22] °C when the indoor air temperature exceeded these boundaries during the occupancy period.

Since the hyperparameter optimisation is multi-objective, it results in Pareto-optimal solutions [64]. As a consequence, it is necessary to establish a criterion to choose the best solution among the optimal ones. The criterion of the minimum distance from the so-called ideal point [65] (i.e., the point whose coordinates correspond to the minimum of both objective function terms) is adopted. In this framework, the Euclidean distance between points corresponding to Pareto front solutions and the ideal point is computed in the plane with coordinates $[E_{cons}, T_{viol}]$.

The first five rows of Table 5 reports the hyperparameters subjected to optimisation with the relative range of variation and the incremental step. Other hyperparameters are kept fixed due to computational constraints. The last five rows of Table 5 include the values of these latter hyperparameters (i.e., *Boltzmann temperature coefficient* α, *Batch size*, *Number of hidden layers*, *Number of neurons per hidden layer* and *Training episodes*).

As mentioned in Section 3.2, the initial training period includes the month of November, then it is gradually extended to include every two weeks the data resulting from the deployment period of the agent for the subsequent re-training of the LSTM model and DRL controller. Twenty different sets of hyperparameters are considered during the automated optimisation procedure carried out during the DRL controller training phase. The best DRL control agent is chosen by considering (1) the Euclidean distance between the ideal point and the performance achieved for each DRL controller, and (2) the performance of the RBC implemented during the same period on the EnergyPlus environment in terms of total energy consumption and cumulated sum of temperature violations. Therefore, the best DRL solution is the one with the smallest Euclidean distance and the greatest performance improvement over the RBC.

3.3. Deep reinforcement learning controller deployment

In the last step of the proposed approach, the best DRL agent resulting from the previous step is deployed for a deployment period (i.e., 15 days) in the EnergyPlus environment as shown in Fig. 5. The goal of this step is to assess if the DRL agent trained on a surrogate data-driven model is capable to control effectively the original system implemented in the EnergyPlus environment with whom it has never interacted before. The DRL controller is deployed statically, as the control policy is not updated. Since the action selected by the DRL controller is the heating power supplied to the building, a supply water temperature calculator is developed to define the supply water temperature setpoint to be implemented in the EnergyPlus environment. The calculator is defined as a piecewise function of the fraction of nominal heating power. In detail, the supply water temperature setpoint is defined in Eq. (3) as follows:

$$SP_{T_{SUPP}}(t)\ [°\mathrm{C}] = \begin{cases} 20 & \text{if } x_{power}(t) < 0.3 \\ \frac{x_{power}(t)*P_{max,heating}}{C} & \\ \quad +T_{RET}(t-1) & \text{if } 0.3 \le x_{power}(t) \le 0.95 \\ 70 & \text{if } x_{power}(t) > 0.95 \end{cases} \tag{3}$$

where $P_{max,heating}$ [kW] is the maximum supplied heating power to the building (equal to 470 kW) and C is a constant that depends on the value of the control action and it is expressed in [kW/°C]. $T_{RET}(t-1)$ [°C] stands as the return water from the thermal zone at the previous time-step. As defined in Section 2, the water mass flow rate in the heating loop is constant.

Throughout the deployment period, safety checks are defined on energy consumption and temperature violations. During the deployment of the DRL, as per Fig. 5, the safety constraints are checked to determine which controller is implemented according to the prescribed requirements. Non-compliance with safety checks results in the switching from DRL to RBC strategy until the end of the deployment period. Fig. 6 depicts an overview of the deployment strategy and outlines the requirements for the operation of the DRL controller and any potential switch to the RBC strategy.

Safety checks are conceived to ensure that the DRL controller performances during building operation are at least comparable with that of the RBC. In fact, during first periods of deployment, the DRL could exhibit poor performance considering the limited amount of data on which the LSTM model is trained and consequently considering the limited length of a training episode. The safety checks are defined on both temperature violations and energy consumption and are employed to determine whether or not to switch back to the RBC baseline strategy.

The safety check on energy consumption is performed weekly. To this scope, a multivariate regression model is developed to estimate the energy consumption of the RBC under the same boundary conditions occurring during the DRL deployment. The DRL control policy has to achieve an energy demand ($E_{DRL,week}$) which does not exceed the 5% of the energy demand that the RBC would have achieved ($E_{RBC_{model},week}$) if implemented considering the same boundary conditions. To this purpose, the weekly RBC energy consumption $E_{RBC_{model},week}$ is calculated as the sum of the daily consumption $E_{RBC_{model},daily}$ obtained from multivariate regression considering outdoor air temperature and solar radiation as input variables.

$$E_{RBC_{model},daily} = c_0 + c_{T_{ext}} * T_{ext} + c_{Q_{sol,rad}} * Q_{sol,rad} \tag{4}$$

with c_0, $c_{T_{ext}}$ and $c_{Q_{sol,rad}}$ coefficients of the multivariate regression. The multivariate regression is updated each time the RBC is implemented during the experiment as a consequence of a low-performing DRL agent. Moreover, this model can be employed as a baseline for energy consumption in order to estimate during operation potential savings achieved by DRL agent implemented in the building with respect to the RBC strategy.

The safety check on indoor temperature control is performed daily. The average daily cumulated sum of temperature violations obtained by the RBC strategy while implemented in the building is employed as threshold to determine the goodness of the implemented DRL control policy. This threshold is calculated by averaging the cumulated sum of temperature violations obtained by RBC during its deployment. Moreover, the threshold is evaluated separately for Mondays and the rest of the days of the week (i.e., Tuesday–Saturday) since during Sundays the building is not occupied. This approach is deemed appropriate due to the likelihood of higher deviations occurring at the beginning of the week since it is more probable that the heating system is inactive on Sundays. If the cumulated sum of temperature violations is greater than the threshold the check is considered not passed and the deployed controller is switched back to the RBC strategy. The temperature violation threshold value, as well as the coefficients of the multivariate regression

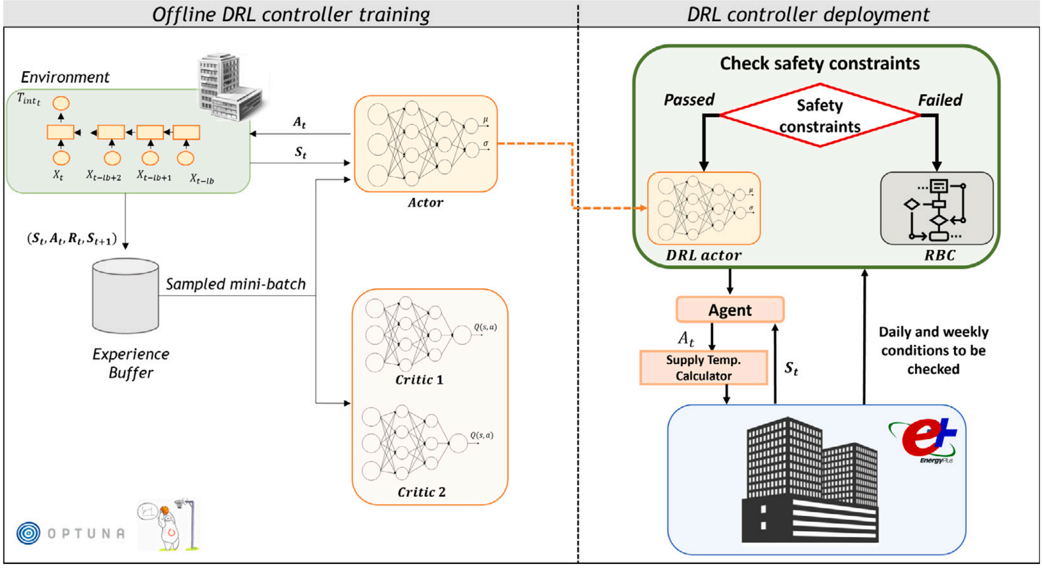


Fig. 5. DRL controller deployment in the EnergyPlus environment.

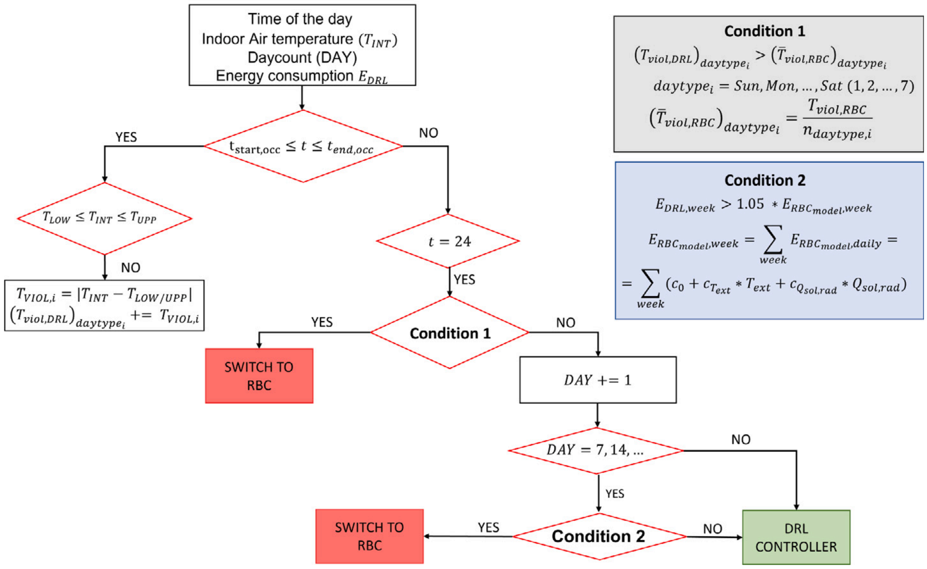

Time of the day
Indoor Air temperature (T_{INT})
Daycount (DAY)
Energy consumption E_{DRL}
YES
$t_{start,occ} \leq t \leq t_{end,occ}$
NO
$T_{LOW} \leq T_{INT} \leq T_{UPP}$
NO
$T_{VIOL,i} = |T_{INT} - T_{LOW/UPP}|$
$(T_{viol,DRL})_{daytype_i}$ += $T_{VIOL,i}$
$t = 24$
YES
YES
Condition 1
NO
SWITCH TO RBC
DAY += 1
$DAY = 7, 14, \ldots$
NO
YES
YES
Condition 2
NO
SWITCH TO RBC
DRL CONTROLLER
Condition 1
$(T_{viol,DRL})_{daytype_i} > (\overline{T}_{viol,RBC})_{daytype_i}$
$daytype_i = Sun, Mon, \ldots, Sat\ (1, 2, \ldots, 7)$
$(\overline{T}_{viol,RBC})_{daytype_i} = \frac{T_{viol,RBC}}{n_{daytype,i}}$
Condition 2
$E_{DRL,week} > 1.05 * E_{RBC_{model},week}$
$E_{RBC_{model},week} = \sum_{week} E_{RBC_{model},daily} =$
$= \sum_{week} (c_0 + c_{T_{ext}} * T_{ext} + c_{Q_{sol,rad}} * Q_{sol,rad})$

Fig. 6. Flowchart of the deployment strategy of the DRL controller on the EnergyPlus environment with details on the daily and weekly safety checks to be respected.

for the daily energy consumption prediction, are updated whenever one of these safety checks fails and the control agent switched from DRL to RBC.

4. Results

This section presents the results obtained by implementing the proposed approach. The outcomes of the LSTM model of building dynamics and the DRL controller training phases are firstly introduced in Section 4.1. Successively, the results of the DRL deployment in the EnergyPlus environment are summarised in Section 4.2.

4.1. Training of long short-term memory neural network and deep reinforcement learning controller

An automated optimisation procedure was carried out using Optuna to identify the best configurations of LSTM hyperparameters which minimised the indoor air temperature RMSE in closed-loop configuration. Table 6 shows the best configurations of LSTM hyperparameters identified during each training phase performed at the beginning of the deployment and after each deployment period.

The Mean Absolute Percentage Error (MAPE) and RMSE values obtained in the closed-loop configuration for all trained LSTM models indicate excellent predictive capability for indoor temperature. Notably, the shortest training period exhibits the lowest values, while the longest training period showed the highest values among the analysed metrics. Additionally, both MAPE and RMSE increase proportionally with the volume of data utilised for LSTM training and testing.

Two subplots in Fig. 7 illustrate the distributions of prediction errors respectively for (a) open-loop and (b) closed-loop configurations. This comparison was conducted to demonstrate the impact of input on the performance of the LSTM model in predicting indoor temperature. The input considered in this comparison includes actual indoor temperature data from the training dataset (i.e., open-loop) and indoor temperature data predicted at the previous time step by the LSTM model itself

Table 6
LSTM models performance in closed loop testing resulting from optimisation procedure during the incremental training period.

Training period	Batch size	Learning rate	# Hidden layers	# Neurons	MAPE [%]	RMSE [°C]
1/11–30/11	107	2.35e−3	4	19	0.955	0.2194
1/11–15/12	98	1.04e−3	4	26	1.159	0.2672
1/11–31/12	80	2.37e−3	3	18	1.282	0.2889
1/11–15/01	115	4.01e−3	3	24	1.394	0.3077
1/11–31/01	104	7.43e−4	3	31	1.406	0.3120
1/11–14/02	94	5.52e−4	4	21	1.378	0.3116
1/11–28/02	113	3.41e−3	3	18	1.441	0.3233
1/11–15/03	84	1.12e−3	3	28	1.46	0.3301

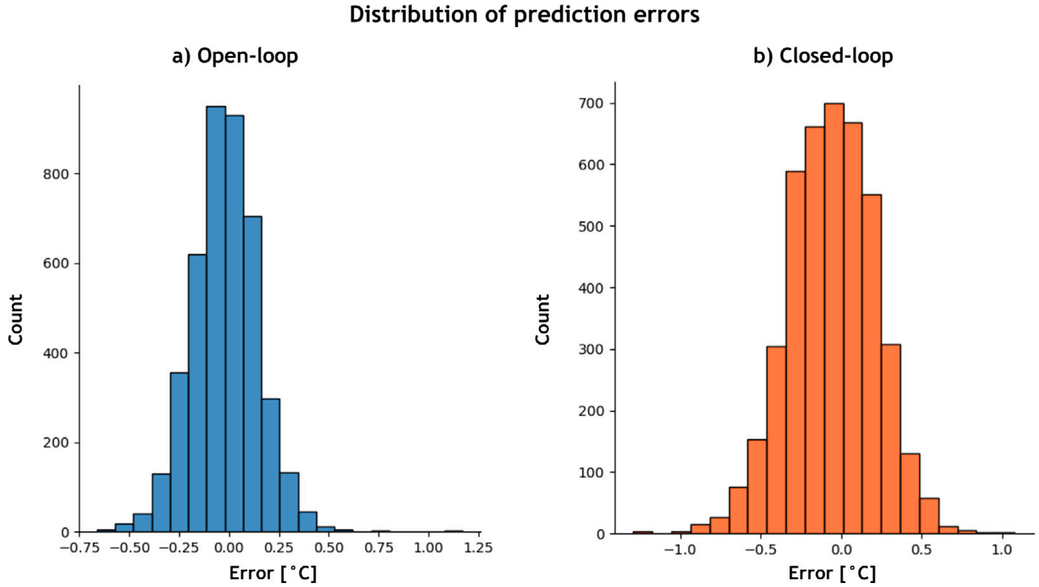


Fig. 7. Error distribution of the LSTM network trained for 1.5 months and implementing the best configuration of hyperparameters for (a) open-loop and (b) closed-loop conditions.

(i.e., closed-loop). The errors were computed based on the comparison between the LSTM predictions and the indoor air temperature values in the training dataset.

Fig. 7 shows that the error distribution in the open-loop case is more confined than in the closed-loop case. In absolute terms, the open-loop led to an error distribution with a peak value around 0.75 °C, in contrast to the closed-loop whose peak value was around 1.5 °C. This outcome was expected since in closed-loop the LSTM model used its own predictions to estimate subsequent values potentially propagating errors. However, as reported in Table 6, MAPE values below 1.5% and RMSE values between 0.2 and 0.35 °C with respect to the training dataset proved the robustness of the training process.

Similarly, the optimisation procedure was conducted during the training phase of DRL using Optuna. However, as the optimisation task involved two conflicting objectives (i.e., minimisation of energy consumption and temperature violations), multiple optimal solutions were identified among the twenty configurations analysed by Optuna in each training period. These solutions are reported in a Pareto front, showcasing the trade-off between energy consumption and temperature violations. In detail, Fig. 8 represents the Pareto front in which each point corresponds to the performance in terms of energy consumption E_{cons} and cumulated sum of temperature violations T_{viol} obtained per each of the twenty sets of hyperparameters explored. This Pareto front refers to the hyperparameters optimisation procedure carried out during the DRL controller training lasting two months (i.e., 1 November–31 December).

The best configuration of hyperparameters was identified in the Pareto front employing the criterion of the minimum distance from the ideal point, whose coordinates correspond to the minimum energy consumption E_{cons} and cumulated sum of temperature violations T_{viol} among the analysed solutions.

Table 7 shows the optimised hyperparameter configurations for each training period and the respective performances in terms of energy consumption and cumulated sum of temperature violations. These configurations were chosen based on the performance of the DRL control agent when it was tested in the environment exploiting the LSTM network as a model of the building dynamics. All configurations were trained for 20 episodes with a Boltzmann temperature coefficient α of 0.1. Differently from that observed in most DRL applications in the literature, the optimal discount factor α was found to be 0.95 instead of 0.99 for all training periods evaluated in this application (except for the first one in which 0.9 was selected as best value). One potential explanation for this could be that the LSTM model of the building dynamics used to train the DRL agent was less accurate compared to engineering models typically used for this purpose. As a result, it was effective to prioritise more immediate rewards (i.e., decreasing the discount factor) because they can be more accurately estimated. Furthermore, the results in terms of cumulated sum of temperature violations in all training periods demonstrate how the DRL controller was able to refine the optimal control policy as the duration of the training period increased.

When the DRL controller was trained for a period longer than three months (indicated by the last three rows in Table 7), it was able to improve the indoor temperature conditions by reducing the cumulated sum of temperature violations compared to when it was trained over shorter periods (e.g., 1 November 1 to 31 January). There are two reasons that might justify this behaviour. First, the DRL controller has a larger amount of data available for training. Second, the LSTM model

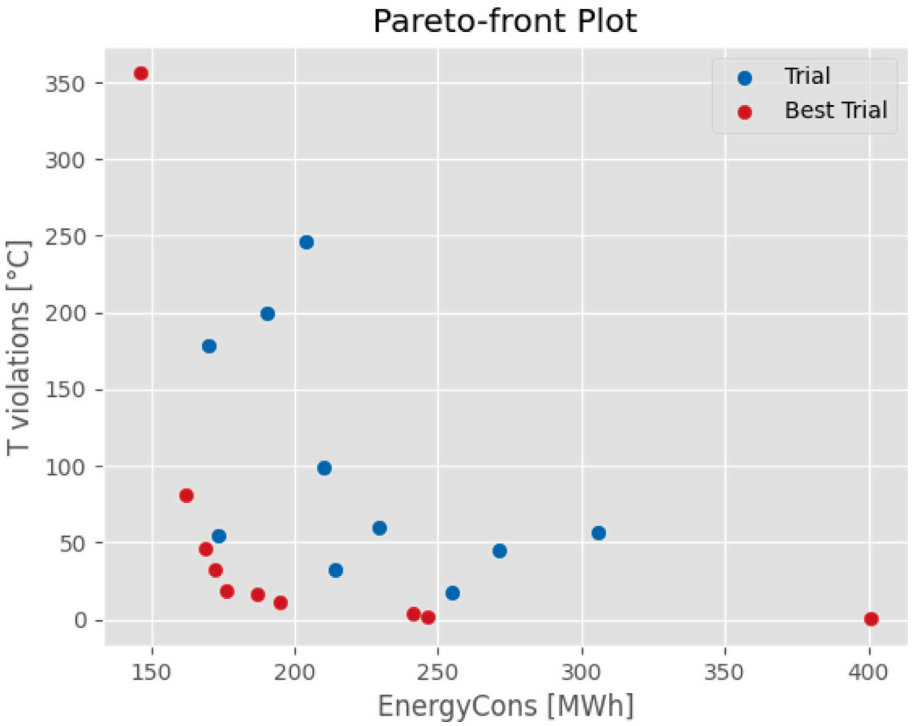


Fig. 8. Pareto-front plot from Optuna of DRL configurations explored during optimisation procedure over 2 training months.

Table 7
DRL controller performances resulting from optimisation procedure during the incremental training period.

Training period	δ	β	θ	Discount factor γ	Learning rate μ	E_{cons} [MWh]	T_{viol} [°C]
1/11–30/11	0.0085	5	0.05	0.9	0.001	59.0	2.50
1/11–15/12	0.0045	4.5	0.03	0.95	0.001	91.1	2.05
1/11–31/12	0.006	4.5	0.045	0.95	0.003	176.4	18.3
1/11–15/01	0.005	6.5	0.04	0.95	0.005	276.7	54.0
1/11–31/01	0.006	6	0.045	0.95	0.001	376.0	79.6
1/11–14/02	0.0055	7	0.01	0.95	0.003	435.4	27.9
1/11–28/02	0.0065	3	0.04	0.95	0.005	451.5	34.8
1/11–15/03	0.009	6.5	0.05	0.95	0.001	464.8	35.1

Table 8
Cumulated sum of temperature violations and total energy consumption resulting from the proposed approach.

Deployment period	Implemented controller	E_{cons} [MWh]	T_{viol} [°C]
1/11–30/11	RBC	96.0	50.1
1/12	DRL	3.1	2.7
2/12–15/12	RBC	53.0	29.9
16/12–18/12	DRL	13.4	37.4
19/12–31/12	RBC	71.2	79.8
1/01	DRL	6.9	14.2
2/01–15/01	RBC	78.6	175.8
16/01–31/01	DRL	82.5	11.7
1/02–14/02	DRL	77.9	5.5
15/02–28/02	DRL	52.9	10.4
1/03–15/03	DRL	36.4	2.2
16/03–31/03	DRL	36.4	5.0
TOTAL	**–**	**608.4**	**424.7**

of the building dynamics becomes more precise in predicting indoor temperature as the size of the training dataset increases.

4.2. Deployment of deep reinforcement learning controller

The deployment phase represents the core of the proposed approach determining whether the DRL control agent resulting from the training process can be effectively implemented in the building to optimise the management during operation. Moreover, according to the proposed approach the results of this phase can suggest the minimum amount of data required for the present case study to train a data-driven model of building dynamics which can be effectively employed to train a DRL agent.

Table 8 summarises the performances achieved in terms of energy consumption and cumulated sum of temperature violations during the implementation of the developed deployment strategy on the EnergyPlus environment. Furthermore, details about the type of controller implemented in each deployment period are provided to give indications regarding any switches from DRL to RBC related to the non-satisfaction with safety checks. As can be seen from Table 8, during the analysed heating season there were three switches (i.e., 1 December, 18 December and 1 January) from DRL controller to RBC since the daily safety check related to temperature violations was not satisfied. However, after a period of two and a half months (i.e., 76 days) the DRL control agent met the daily and weekly safety checks on temperature violations and energy consumption. Therefore, from 16 January to 31 March the DRL controller was successfully deployed during the whole deployment period. Overall, the proposed deployment strategy resulted in total energy consumption of 608.4 MWh and a cumulated sum of temperature violations of 424.7 °C. As previously stated in Section 3.3, safety checks were updated whenever the RBC

Table 9
Values of cumulated sum of temperature violations thresholds, coefficients and performance metrics of the multivariate regression during different deployment periods.

Deployment period	T_{viol} thresholds [°C]		Multivariate regression					
	Mon	Tue–Sat	c_0	$c_{T_{ext}}$	$c_{Q_{sol,rad}}$	R^2 [%]	MAPE [%]	RMSE [kW]
01/12–15/12	3.61	1.41	237.2	−14.6	−0.02	80.9	6.5	9.6
16/12–31/12	4.78	1.61	232.9	−13.6	0.02	83.6	5.2	8.4
01/01–15/01	7.33	2.59	245.9	−15.4	−0.06	86.2	4.6	7.9
16/01–31/03	12.66	4.27	243.3	−14.7	−0.05	90.7	4.3	7.2

Table 10
Results for DRL deployment periods meeting safety checks.

Deployment period	E_{cons} [MWh]			T_{viol} [°C]	
	RBC_{model}	RBC	DRL	RBC	DRL
16/01–31/01	75.2	79.2	82.5	65.3	11.7
1/02–14/02	72.1	72.2	77.9	81.1	5.5
15/02–28/02	53.4	55.2	52.9	40.8	10.4
1/03–15/03	43.2	39.7	36.4	6.9	2.2
16/03–31/03	46.0	41.5	36.4	5.9	5.0
TOTAL	**289.9**	**287.8**	**286.2**	**200.0**	**34.8**

was implemented. RBC was implemented during the first month of the heating season (i.e., November) or whenever the deployed DRL controller failed a safety check.

Table 9 displays the daily threshold employed for the cumulated sum of temperature violations in each deployment period. This threshold was set separately for Mondays and the other weekdays. In fact, Mondays were the most critical day since, at the beginning of the day, the indoor temperature was generally lower than the average of the other weekdays leading to more frequent temperature violations of higher magnitude. This behaviour can be attributed to the heating system being inactive on Sundays, corresponding to days without the presence of occupants.

The threshold values employed for the daily check on temperature violations increased over the heating season (e.g., from 1.41 °C in the first deployment period to 4.27 °C in the last period for weekdays from Tuesday to Saturday). This result could be attributed to the increasing reduction of outdoor temperature from November to January, which results in a higher magnitude of temperature violations. Moreover, Table 9 provides details regarding the coefficients and metrics of the multivariate regression model used to estimate the daily RBC energy consumption (i.e., RBC_{model}). This information was employed to compute the weekly threshold for energy consumption. The performance metrics reported in the table for the multivariate regression model (i.e., R^2, MAPE, and RMSE) indicate that the robustness of the model increased as it proceeds through the indicated deployment periods. In fact, the RBC was implemented following the switch from the DRL during the first three deployment periods, and both the intercept c_0 and the coefficients $c_{T_{ext}}$ and $c_{Q_{sol,rad}}$ were updated, improving the accuracy of the regression model which was trained on a larger dataset. As a result, the R^2 value increased from 80.9% to 90.7% while MAPE and RMSE decreased respectively from 6.5% and 9.6 kW to 4.3% and 7.2 kW.

Fig. 9 reports the indoor temperature profile over the 16–31 December deployment period, after a training phase of 1.5 months (i.e., from 1 November to 15 December). Here, the DRL control agent was implemented for only 3 days, since on the third day (i.e., Monday) the safety check imposed on temperature violations was not met ($(T_{viol,DRL})_{Mon18Dec}$ = 36.0 °C vs. $(T_{viol,RBC})_{Mon}$ = 4.78 °C). In that case, the RBC threshold for Monday was exceeded. Therefore, as indicated in Fig. 9, after the black dashed line the controller automatically switched to the RBC algorithm until the end of the deployment period (i.e., 31 December) to effectively manage the system. This result could be explained considering the limited information included in a DRL training episode until that moment. Since the DRL was not trained on episodes ranging for a whole heating season it was probable that it had not the chance to learn a control policy behaving optimally for each condition of the controlled environment. For example, in this case the agent could not have the chance to learn how to optimally behave in colder climate conditions with respect to the training episode (such as the ones present in the weather between December and January). The DRL control policy was unable to adequately map the control action with the current and past indoor temperature condition of the building and its current and future occupancy status (i.e., time to occupancy start/end).

Table 10 summarises the results obtained in terms of cumulated sum of temperature violations and total energy consumption over each deployment period where both safety checks were met. The performance achieved by the DRL controller in terms of total energy consumption was compared to both the output of the multivariate regression model (RBC_{model}) expressed according to Eq. (4) and to the results obtained by implementing the RBC strategy (RBC) in the EnergyPlus environment for the same period. This latter scenario cannot be obtained in a real-world experiment. However, it was included to better justify the quality of the results obtained by both the proposed DRL controller and the multivariate regression model. The results obtained by the proposed DRL controller in terms of cumulated sum temperature violations were compared to those obtained by the baseline RBC strategy implemented in the EnergyPlus environment for the same simulation period.

Over the entire period (i.e., 2.5 months, from 16 January to 31 March) the energy consumption achieved by the DRL ($E_{cons,DRL}$ = 286.2 MWh) was slightly lower than both the energy consumption achieved by the RBC (i.e., 1.6 MWh less, $E_{cons,RBC}$ = 287.8 MWh) and the energy consumption estimated by the multivariate regression model RBC_{model} (i.e., 3.7 MWh less, $E_{cons,RBC_{model}}$ = 289.9 MWh). Moreover, the energy consumption prediction model developed for the baseline RBC demonstrated robustness, with a difference of approximately 0.8% between the total predicted energy consumption and the hypothetical consumption if the baseline RBC had been implemented in the EnergyPlus environment.

Simultaneously, the developed training and deployment strategy for the DRL controller achieved better indoor temperature control. Despite being trained on a data-driven model of the building, the DRL controller was capable to reduce the cumulated sum of temperature violations during the 16 January–31 March period by more than 80% compared to the RBC ($T_{viol,DRL}$ = 34.8 °C vs. $T_{viol,RBC}$ = 200.0 °C).

To conclude, Fig. 10 shows the indoor temperature and supply water temperature profiles during the fifth deployment period (i.e., 1–14 February), with a focus on the occupancy period (i.e., 7:00–19:00) of the last day of the represented period, showing in detail the values of the indoor temperature setpoint as well as the lower and upper bound temperatures of the acceptability range.

Through the accurate management of the pre-heating phase, the DRL control agent successfully preheated the building prior to the arrival of occupants (i.e., 7:00). Moreover, thanks to the definition given for the temperature term of the reward function, considering the introduction of the temperature-term prize (θ) as shown in Eq. (2), the DRL controller kept the indoor temperature profile during the occupancy period near the lower limit of the temperature band, also exploiting higher heat gains during the day to decrease the supply water temperature and saving heating energy. During occupancy periods when DRL operates according to safety constraints, only for 3%

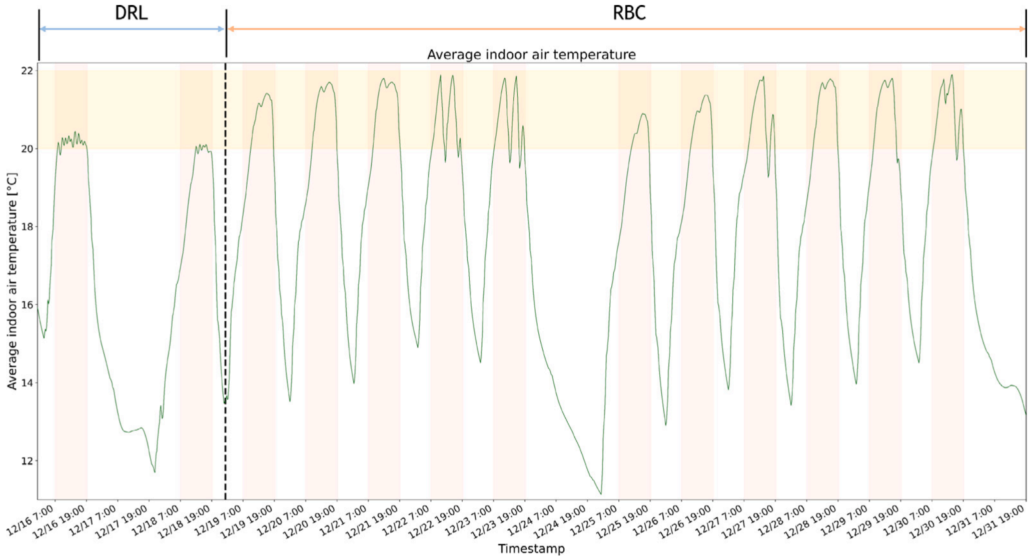


Fig. 9. Indoor temperature profile during the second deployment period of the DRL+RBC controller (after 1.5 training months).

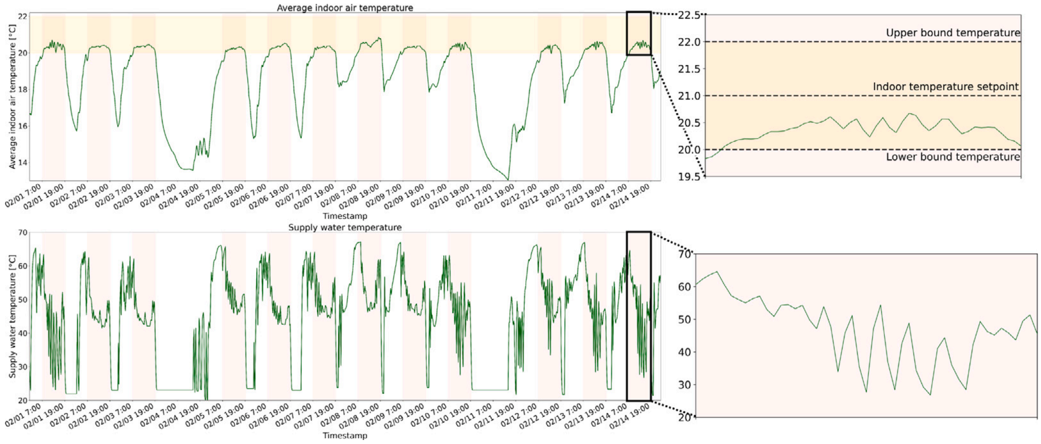


Fig. 10. Indoor temperature profile during the fifth deployment period of the DRL controller (after 3 training months).

of occurrences the indoor temperature was lower than the lower limit of the acceptability range (i.e., 20 °C) with an average value of deviation of 0.15 °C. This outcome can be considered more than acceptable considering both the size of the thermal zone and the type of terminal units (i.e., radiators). As a result, the learned control policy was able to effectively control the energy system implemented in the EnergyPlus environment under a range of weather conditions.

5. Discussion

The present paper focuses on developing an effective approach to offline pre-train a DRL control agent exploiting data-driven models of the building dynamics, aiming to address one of the main limitations to adopting DRL-based controllers in the building industry. Although the approach of exploiting engineering models is commonly employed to pre-train offline DRL agents before their effective deployment in a real-world context, it lacks of scalability considering that it is unfeasible to build a detailed model of each building before implementing an advanced control strategy. On the other hand, data-driven models are simpler to formalise compared to engineering models and require fewer input data. However, they have limited generalisation capabilities and their response from a physical perspective is heavily influenced by the quantity and quality of the data on which they are trained. Despite this limitation, data-driven models of building dynamics can represent an essential tool for increasing the scalability of DRL-based control strategies in buildings if correctly trained and tuned. In this context, the availability of historical data in terms of both volume and variety plays a key role for building robust data-driven models. As a wide availability of monitoring data in buildings is not always guaranteed, it is crucial to develop methodologies capable to handle a limited amount of data and being rapidly effective when implemented in a real building. In the present paper, differently from other applications reported in the literature, one of the key questions addressed is not to develop the most accurate data-driven model of the building dynamics but to propose an approach where data-driven models, despite their limitations, could provide a sufficiently robust representation of the control problem to enable a DRL agent to learn an effective control policy in an offline setting.

To this purpose, the proposed approach involves a recursive process based on frequent updates of both the data-driven model of building dynamics and the control policy learned by the agent. This training structure based on frequent updates allowed for the pre-training of the DRL controller in a data-efficient manner, making it possible to effectively map the building dynamics even when a limited amount of data are initially available. Moreover, daily and weekly safety checks are included to constantly verify that the performances of the DRL agent deployed on the systems are not diverging. If one of these rules is violated the controller switches to RBC mode until the next update.

The entire process is conceived to propose a scalable approach for the implementation of DRL based control strategies leveraging the potentialities of data-driven models of building dynamics.

The results show that the proposed approach can indeed provide a sufficiently robust representation of the control problem to enable a DRL agent to learn an effective control policy. By leveraging the potentialities of data-driven models of building dynamics, this approach offers a scalable solution for the implementation of DRL-based control strategies in buildings.

In this context, remarks for readers and future practitioners can be defined from the application of the proposed approach. Hyperparameters optimisation plays a fundamental role in the definition of deep learning architectures. In the proposed approach, automated hyperparameters optimisation routines are introduced due to offline training configuration. Moreover, data-driven model of building dynamics should not be only evaluated in terms of accuracy but also in terms of generalisation capabilities when employed to emulate building behaviour. Eventually, in the context of building energy management, the introduction of innovative approaches aims at increasing the scalability of existing DRL control algorithms considering real-world limitations can represent a promising research field.

In conclusion, this paper demonstrates that the proposed approach can effectively overcome the limitations of traditional approaches to pre-training DRL controllers in buildings. The results obtained validate the potential of data-driven models of building dynamics as a powerful tool for increasing the scalability of DRL-based control strategies, even when faced with an initial limited data availability.

6. Conclusion

The present paper proposes a pre-training strategy for enhancing the scalability of a DRL agent by means of a data-driven model of building dynamics. The proposed approach is tested for a controller managing supply water temperature setpoint of the heating system of an office building equipped with radiators. The goal of the controller is to minimise energy consumption while maintaining indoor air temperature values within an acceptability range.

In this application, an EnergyPlus environment is employed as a proxy of the real building and a RBC is chosen as a baseline. The experiment is carried out to emulate the deployment of a DRL control agent during a single heating season making the hypothesis of limited amount of monitored data to initially train the data-driven model.

A recursive approach alternating DRL agent training and deployment is conceived and tested. In this perspective, daily and weekly safety checks on temperature violations and energy consumption are implemented to determine in an automatic way the minimum training period required for the DRL controller to ensure acceptable performances. The results obtained shows that after 76 days the controller trained through the proposed approach is capable to converge to acceptable performance. In particular, the DRL agent is able to improve the indoor temperature control performances by 80% if compared to the baseline RBC strategy while consuming the same amount of energy. The obtained results suggest that if adequately trained, data-driven models of building dynamics can be effectively employed for the training of DRL control agents and considerably reduce the modelling effort with respect to detailed engineering models.

Future works will focus on extending the proposed approach in the following directions:

- Evaluation of the proposed approach on more complex case studies with different HVAC systems configurations considering renewable energy sources (i.e., PV system) and energy storage (i.e., batteries or TES).
- Development of a more detailed building simulation by exploiting Spawn of EnergyPlus [66], which enables the integration of the energy system modelled in Modelica with the building energy model developed in EnergyPlus. As a result, performances achieved by applying the proposed approach would be more similar to those obtained in real building operations.
- Implementation of Transfer Learning (TL) to share the data-driven model emulating the building dynamics and the DRL control policy between buildings with similar or different features (e.g., energy systems or building thermophysical properties) to enhance the scalability of the proposed approach.
- Real-world deployment of the DRL agent pre-trained on the LSTM model of the building dynamics, by developing an infrastructure to enable the correct implementation.

CRediT authorship contribution statement

Davide Coraci: Conceptualization, Methodology, Software, Investigation, Formal analysis, Data curation, Writing – original draft, Visualization. **Silvio Brandi:** Conceptualization, Methodology, Investigation, Writing – original draft, Writing – review & editing. **Alfonso Capozzoli:** Conceptualization, Methodology, Validation, Writing – review & editing, Supervision.

Declaration of competing interest

The 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.

Data availability

Data will be made available on request.

References

[1] Martinopoulos G, Papakostas KT, Papadopoulos AM. A comparative review of heating systems in EU countries, based on efficiency and fuel cost. Renew Sustain Energy Rev 2018;90:687–99. http://dx.doi.org/10.1016/j.rser.2018.03.060.

[2] Karasu S, Altan A. Recognition model for solar radiation time series based on random forest with feature selection approach. In: 2019 11th international conference on electrical and electronics engineering. ELECO, 2019, p. 8–11. http://dx.doi.org/10.23919/ELECO47770.2019.8990664.

[3] Altan A, Karasu S, Zio E. A new hybrid model for wind speed forecasting combining long short-term memory neural network, decomposition methods and grey wolf optimizer. Appl Soft Comput 2021;100:106996. http://dx.doi.org/10.1016/j.asoc.2020.106996.

[4] Piscitelli MS, Brandi S, Capozzoli A, Xiao F. A data analytics-based tool for the detection and diagnosis of anomalous daily energy patterns in buildings. Build Simul 2021;14(1):131–47. http://dx.doi.org/10.1007/s12273-020-0650-1.

[5] Wang Z, Hong T. Reinforcement learning for building controls: The opportunities and challenges. Appl Energy 2020;269:115036. http://dx.doi.org/10.1016/j.apenergy.2020.115036.

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

[7] Karasu S, Altan A, Bekiros S, Ahmad W. A new forecasting model with wrapper-based feature selection approach using multi-objective optimization technique for chaotic crude oil time series. Energy 2020;212:118750. http://dx.doi.org/10.1016/j.energy.2020.118750.

[8] Finck C, Beagon P, Clauß J, Péan T, Vogler-Finck P, Zhang K, Kazmi H. Review of applied and tested control possibilities for energy flexibility in buildings. In: Technical report from IEA EBC annex 67 - energy flexible buildings. 2017, p. 1–59. http://dx.doi.org/10.13140/RG.2.2.28740.73609.

[9] Capozzoli A, Piscitelli MS, Brandi S, Grassi D, Chicco G. Automated load pattern learning and anomaly detection for enhancing energy management in smart buildings. Energy 2018;157:336–52. http://dx.doi.org/10.1016/j.energy.2018.05.127.

[10] Molina-Solana M, Ros M, Ruiz MD, Gómez-Romero J, Martin-Bautista M. Data science for building energy management: A review. Renew Sustain Energy Rev 2017;70:598–609. http://dx.doi.org/10.1016/j.rser.2016.11.132.

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

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

[13] Naidu DS, Rieger CG. Advanced control strategies for heating, ventilation, air-conditioning, and refrigeration systems—An overview: Part I: Hard control. HVAC & R Res 2011;17(1):2–21. http://dx.doi.org/10.1080/10789669.2011.540942.

[14] Serale G, Fiorentini M, Capozzoli A, Cooper P, Perino M. Formulation of a model predictive control algorithm to enhance the performance of a latent heat solar thermal system. Energy Convers Manage 2018;173:438–49. http://dx.doi.org/10.1016/j.enconman.2018.07.099.

[15] Cho S, Zaheer-uddin M. Predictive control of intermittently operated radiant floor heating systems. Energy Convers Manage 2003;44(8):1333–42. http://dx.doi.org/10.1016/S0196-8904(02)00116-4.

[16] Ruusu R, Cao S, Manrique Delgado B, Hasan A. Direct quantification of multiple-source energy flexibility in a residential building using a new model predictive high-level controller. Energy Convers Manage 2019;180:1109–28. http://dx.doi.org/10.1016/j.enconman.2018.11.026.

[17] Seal S, Boulet B, Dehkordi VR. Centralized model predictive control strategy for thermal comfort and residential energy management. Energy 2020;212:118456. http://dx.doi.org/10.1016/j.energy.2020.118456.

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

[19] Vand B, Ruusu R, Hasan A, Manrique Delgado B. Optimal management of energy sharing in a community of buildings using a model predictive control. Energy Convers Manage 2021;239:114178. http://dx.doi.org/10.1016/j.enconman.2021.114178.

[20] Prívara S, Váňa Z, Gyalistras D, Cigler J, Sagerschnig C, Morari M, Ferkl L. Modeling and identification of a large multi-zone office building. In: 2011 IEEE international conference on control applications. CCA, 2011, p. 55–60. http://dx.doi.org/10.1109/CCA.2011.6044402.

[21] Prívara S, Cigler J, Váňa Z, Oldewurtel F, Sagerschnig C, Žáčeková E. Building modeling as a crucial part for building predictive control. Energy Build 2013;56:8–22. http://dx.doi.org/10.1016/j.enbuild.2012.10.024.

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

[23] Vázquez-Canteli JR, Nagy Z. Reinforcement learning for demand response: A review of algorithms and modeling techniques. Appl Energy 2019;235:1072–89. http://dx.doi.org/10.1016/j.apenergy.2018.11.002.

[24] Sutton RS, Barto AG. Reinforcement learning: an introduction. 2nd ed.. The MIT Press; 2018, URL http://incompleteideas.net/book/the-book-2nd.html.

[25] Watkins CJCH, Dayan P. Q-learning. Mach Learn 1992;8(3):279–92. http://dx.doi.org/10.1007/BF00992698.

[26] Capozzoli A, Mechri H, Corrado V. Impacts of architectural design choices on building energy performance applications of uncertainty and sensitivity techniques. In: Proceedings of the 11th international building performance simulation association conference, Glasgow, Scotland. 2009, URL https://publications.ibpsa.org/conference/?id=bs2009.

[27] Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, Graves A, Riedmiller M, Fidjeland AK, Ostrovski G, Petersen S, Beattie C, Sadik A, Antonoglou I, King H, Kumaran D, Wierstra D, Legg S, Hassabis D. Human-level control through deep reinforcement learning. Nature 2015;518(7540):529–33. http://dx.doi.org/10.1038/nature14236.

[28] Brandi S, Piscitelli MS, Martellacci M, Capozzoli A. Deep reinforcement learning to optimise indoor temperature control and heating energy consumption in buildings. Energy Build 2020;224:110225. http://dx.doi.org/10.1016/j.enbuild.2020.110225.

[29] Zhang Z, Chong A, Pan Y, Zhang C, Lam KP. Whole building energy model for HVAC optimal control: A practical framework based on deep reinforcement learning. Energy Build 2019;199:472–90. http://dx.doi.org/10.1016/j.enbuild.2019.07.029.

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

[31] Vázquez-Canteli J, Kämpf J, Nagy Z. Balancing comfort and energy consumption of a heat pump using batch reinforcement learning with fitted Q-iteration. Energy Procedia 2017;122:415–20. http://dx.doi.org/10.1016/j.egypro.2017.07.429, CISBAT 2017 International ConferenceFuture Buildings & Districts – Energy Efficiency from Nano to Urban Scale.

[32] Yu Z, Dexter A. Online tuning of a supervisory fuzzy controller for low-energy building system using reinforcement learning. Control Eng Pract 2010;18(5):532–9. http://dx.doi.org/10.1016/j.conengprac.2010.01.018.

[33] Brandi S, Gallo A, Capozzoli A. A predictive and adaptive control strategy to optimize the management of integrated energy systems in buildings. Energy Rep 2022;8:1550–67. http://dx.doi.org/10.1016/j.egyr.2021.12.058.

[34] Wang Y, Lin X, Pedram M. A near-optimal model-based control algorithm for households equipped with residential photovoltaic power generation and energy storage systems. IEEE Trans Sustain Energy 2016;7(1):77–86. http://dx.doi.org/10.1109/TSTE.2015.2467190.

[35] Abedi S, Yoon SW, Kwon S. Battery energy storage control using a reinforcement learning approach with cyclic time-dependent Markov process. Int J Electr Power Energy Syst 2022;134:107368. http://dx.doi.org/10.1016/j.ijepes.2021.107368.

[36] Wang Y, Velswamy K, Huang B. A long-short term memory recurrent neural network based reinforcement learning controller for office heating ventilation and air conditioning systems. Processes 2017;5(3). http://dx.doi.org/10.3390/pr5030046.

[37] Gao G, Li J, Wen Y. DeepComfort: Energy-efficient thermal comfort control in buildings via reinforcement learning. IEEE Internet Things J 2020;7(9):8472–84. http://dx.doi.org/10.1109/JIOT.2020.2992117.

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

[39] Park JY, Dougherty T, Fritz H, Nagy Z. LightLearn: An adaptive and occupant centered controller for lighting based on reinforcement learning. Build Environ 2019;147:397–414. http://dx.doi.org/10.1016/j.buildenv.2018.10.028.

[40] Du Y, Zandi H, Kotevska O, Kurte K, Munk J, Amasyali K, Mckee E, Li F. Intelligent multi-zone residential HVAC control strategy based on deep reinforcement learning. Appl Energy 2021;281:116117. http://dx.doi.org/10.1016/j.apenergy.2020.116117.

[41] Vázquez-Canteli JR, Ulyanin S, Kämpf J, Nagy Z. Fusing TensorFlow with building energy simulation for intelligent energy management in smart cities. Sustainable Cities Soc 2019;45:243–57. http://dx.doi.org/10.1016/j.scs.2018.11.021.

[42] Pinto G, Piscitelli MS, Vázquez-Canteli JR, Nagy Z, Capozzoli A. Coordinated energy management for a cluster of buildings through deep reinforcement learning. Energy 2021;229:120725. http://dx.doi.org/10.1016/j.energy.2021.120725.

[43] Deltetto D, Coraci D, Pinto G, Piscitelli MS, Capozzoli A. Exploring the potentialities of deep reinforcement learning for incentive-based demand response in a cluster of small commercial buildings. Energies 2021;14(10). http://dx.doi.org/10.3390/en14102933.

[44] Zhang B, Hu W, Li J, Cao D, Huang R, Huang Q, Chen Z, Blaabjerg F. Dynamic energy conversion and management strategy for an integrated electricity and natural gas system with renewable energy: Deep reinforcement learning approach. Energy Convers Manage 2020;220:113063. http://dx.doi.org/10.1016/j.enconman.2020.113063.

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

[46] Modelica Association. Modelica® - a unified object-oriented language for physical systems modeling. Tutorial. 2000, URL http://www.modelica.org/documents/ModelicaTutorial14.pdf.

[47] Crawley DB, Lawrie LK, Winkelmann FC, Buhl W, Huang Y, Pedersen CO, Strand RK, Liesen RJ, Fisher DE, Witte MJ, Glazer J. EnergyPlus: creating a new-generation building energy simulation program. Energy Build 2001;33(4):319–31. http://dx.doi.org/10.1016/S0378-7788(00)00114-6, Special Issue: BUILDING SIMULATION'99.

[48] Zhou X, Lin W, Kumar R, Cui P, Ma Z. A data-driven strategy using long short term memory models and reinforcement learning to predict building electricity consumption. Appl Energy 2022;306:118078. http://dx.doi.org/10.1016/j.apenergy.2021.118078.

[49] Li Q, Meng Q, Cai J, Yoshino H, Mochida A. Predicting hourly cooling load in the building: A comparison of support vector machine and different artificial neural networks. Energy Convers Manage 2009;50(1):90–6. http://dx.doi.org/10.1016/j.enconman.2008.08.033.

[50] Drgoňa J, Tuor AR, Chandan V, Vrabie DL. Physics-constrained deep learning of multi-zone building thermal dynamics. Energy Build 2021;243:110992. http://dx.doi.org/10.1016/j.enbuild.2021.110992.

[51] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9(8):1735–80. http://dx.doi.org/10.1162/neco.1997.9.8.1735, arXiv:https://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf.

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

[53] Bellman R. Dynamic programming. Science 1966;153(3731):34–7. http://dx.doi.org/10.1126/science.153.3731.34, arXiv:https://science.sciencemag.org/content/153/3731/34.full.pdf.

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

[55] Brockman G, Cheung V, Pettersson L, Schneider J, Schulman J, Tang J, Zaremba W. Openai gym. 2016, arXiv:1606.01540.

[56] Wetter M. Co-simulation of building energy and control systems with the building controls virtual test bed. J Build Perform Simul 2011;4(3):185–203. http://dx.doi.org/10.1080/19401493.2010.518631.

[57] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S. PyTorch: An imperative style, high-performance deep learning library. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché Buc F, Fox E, Garnett R, editors. Advances in neural information processing systems, Vol. 32. Curran Associates, Inc.; 2019, https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

[58] Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Kudlur M, Levenberg J, Mane D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shlens J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viegas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y, Zheng X. TensorFlow: Large-scale machine learning on heterogeneous distributed systems. 2016, arXiv:1603.04467.

[59] Hill A, Raffin A, Ernestus M, Gleave A, Kanervisto A, Traore R, Dhariwal P, Hesse C, Klimov O, Nichol A, Plappert M, Radford A, Schulman J, Sidor S, Wu Y. Stable baselines. 2018, https://github.com/hill-a/stable-baselines.

[60] Chen M-R, Zeng G-Q, Lu K-D, Weng J. A two-layer nonlinear combination method for short-term wind speed prediction based on ELM, ENN, and LSTM. IEEE Internet Things J 2019;6(4):6997–7010. http://dx.doi.org/10.1109/JIOT.2019.2913176.

[61] Lu K-D, Wu Z-G, Huang T. Differential evolution-based three stage dynamic cyber-attack of cyber-physical power systems. IEEE/ASME Trans Mechatronics 2022;1–12. http://dx.doi.org/10.1109/TMECH.2022.3214314.

[62] Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: A next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. KDD '19, New York, NY, USA: Association for Computing Machinery; 2019, p. 2623–31. http://dx.doi.org/10.1145/3292500.3330701.

[63] Bergstra J, Bardenet R, Bengio Y, Kégl B. Algorithms for hyper-parameter optimization. In: Proceedings of the 24th international conference on neural information processing systems. NIPS '11, Red Hook, NY, USA: Curran Associates Inc.; 2011, p. 2546–54, https://proceedings.neurips.cc/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf.

[64] Xin Q. 3 - optimization techniques in diesel engine system design. In: Xin Q, editor. Diesel engine system design. Woodhead Publishing; 2013, p. 203–96. http://dx.doi.org/10.1533/9780857090836.1.203.

[65] Zelany M. A concept of compromise solutions and the method of the displaced ideal. Comput Oper Res 1974;1(3):479–96. http://dx.doi.org/10.1016/0305-0548(74)90064-1.

[66] Wetter M, Benne KS, Gautier A, Nouidui TS, Ramle A, Roth A, Tummescheit H, Mentzer SG, Winther C. Lifting the garage door on spawn, an open-source bem- controls engine. In: 2020 building performance modeling conference and simbuild co-organized by ASHRAE and IBPSA-USA. 2020, https://simulationresearch.lbl.gov/wetter/download/2020-simBuild-spawn.pdf.