Energy & Buildings 280 (2023) 112709

Contents lists available at ScienceDirect

Energy & Buildings

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

Evaluation of advanced control strategies for building energy systems

Phillip Stoffel *, Laura Maier, Alexander Kümpel, Thomas Schreiber, Dirk Müller

RWTH Aachen University, E.ON Energy Research Center, Institute for Energy Efficient Buildings and Indoor Climate, Mathieustrasse 10, 52074 Aachen, Germany

ARTICLE INFO

Article history:
Received 14 October 2022
Revised 18 November 2022
Accepted 1 December 2022
Available online 17 December 2022

Keywords:
Optimal building control
Artificial intelligence in buildings
Data-driven modeling
Adaptive control
Approximate MPC
Reinforcement learning

ABSTRACT

Advanced building control strategies like model predictive control and reinforcement learning can consider forecasts for weather, occupancy, and energy prices. Combined with system and domain knowledge, this makes them a promising approach to reduce buildings' energy consumption and CO_2 emissions. For this reason, model predictive control and reinforcement learning have recently gained more popularity in the scientific literature. Nevertheless, publications often lack comparability among different control algorithms. The studies in the literature mainly focus on the comparison of an advanced algorithm with a conventional alternative. At the same time, use cases and key performance indicators vary strongly. This paper extensively evaluates six advanced control algorithms based on quantitative and qualitative key performance indicators. The considered control algorithms are a state-of-the-art model-free reinforcement learning algorithm (Soft-Actor-Critic), three model predictive controllers based on white-box, gray-box, and black-box modeling, approximate model predictive control, and a well-designed rule-based controller for fair benchmarking. The controllers are applied to an exemplary multi–input–multi–output building energy system and evaluated using a one-year simulation to cover seasonal effects. The considered building energy system is an office room supplied with heat and cold by an air handling unit and a concrete core activation.

We consider the violation of air temperature constraints as thermal discomfort, the yearly energy consumption, and the computational effort as quantitative key performance indicators. Compared to the well-tuned rule-based controller, all advanced controllers decrease thermal discomfort. The black-box model predictive controller achieves the highest energy savings with 8.4%, followed by the white-box model predictive controller with 7.4% and the gray-box controller with 7.2%. The reinforcement learning algorithm reduces energy consumption by 7.1% and the approximate model predictive controller by 4.8%. Next to these quantitative key performance indicators, we introduce qualitative criteria like adaptability, interpretability, and required know-how. Furthermore, we discuss the shortcomings and potential improvements of each controller.

1. Introduction

With the publication of the Sixth Assessment Report's second part in February 2022, the international expert team *Intergovernmental Panel on Climate Change* emphasizes the need for immediate action to stop climate change and drastically reduce emissions [1]. In this context, establishing a green building stock is a crucial step toward climate neutrality.

Optimizing building energy systems' operation is a promising measure to reduce emissions from the current building stock quickly. In recent years, the scientific community has put tremendous effort into developing and comparing different optimal control approaches. Model predictive control (MPC) has proven to be a promising method for many application scenarios. Implementing MPC, different researchers achieve energy savings between 13% to 75% [2–7]. Even though many researchers have proven the potential, the transfer into practical applications is still lacking. According to [8], the small number of real-life applications is, among other things, caused by missing know-how, high hard- and software requirements, and the increased modeling effort compared to conventional control techniques. To mitigate the disadvantages and exploit the potential of MPC, the scientific community has introduced many modifications to the methodology.

The modeling effort of MPC applications mainly depends on the model development process [9,10]. Here, literature distinguishes between white-box (WB), gray-box (GB), and black-box (BB) mod-

* Corresponding author
E-mail addresses: phillip.stoffel@eonerc.rwth-aachen.de (P. Stoffel), laura.maier@eonerc.rwth-aachen.de (L. Maier), akuempel@eonerc.rwth-aachen.de (A. Kümpel), thomas.schreiber@eonerc.rwth-aachen.de (T. Schreiber), dmueller@eonerc.rwth-aachen.de (D. Müller).

https://doi.org/10.1016/j.enbuild.2022.112709
0378-7788/

Nomenclature

Symbol and Units

A	area, m^2
C	thermal capacity, $\frac{J}{K}$
c_p	specific thermal capacity, $\frac{J}{Kg \cdot K}$
d	disturbances, -
f	parameter, -
h	heat transfer coefficient, $\frac{W}{K}$
k	timestep, -
M	prediction horizon MHE, -
$\dot{m}$	mass flow, $\frac{kg}{s}$
N	prediction horizon MPC, -
Q	heat, W
$\dot{Q}$	heatflow, W
$\dot{q}$	specific heatflow, $\frac{W}{m^2}$
R	heat transfer resistance, $\frac{K \cdot m^2}{W}$
s	schedule, -
T	temperature, K
ΔT	Change in Temperature, $\frac{K}{s}$
u	control variable, -
x	state variable, -
y	controlled variable, -

Greek Symbols

α	learning rate, -
γ	discount factor, -
ϵ	slack variable, K
τ	Polyak update, -
ψ	split factor for radiation, -

Indices and Abbreviations

AHU	air-handling unit
AMPC	approximate MPC
ANN	artificial neural network
ARMAX	autoregressive-moving average with exogenous inputs
ARX	autoregressive model with exogenous inputs
BB	black-box
BBMPC	black-box MPC
BES	building energy system
CCA	concrete-core activation
CSC	constant setpoint control
DDPG	deep deterministic policy gradient
DeePC	data-enable predictive control
DQN	Deep Q-Networks
DT	decision trees
EKF	extended Kalman filter
FMU	functional mock-up unit
GB	gray-box
GBMPC	gray-box MPC
GPR	Gaussian process regression
HVAC	heating ventilation air conditioning
KPI	key performance indicator
MDP	Markov Decision Problem
MHE	moving horizon estimation
ML	machine learning
MPC	model predictive control
MSE	mean-squared error
OCP	optimal control problem
OL	online learning
PI	proportional-integral
PID	proportional-integral-derivative
RB	rule based
RBC	rule based controller
RC	resistance–capacitance
RF	random forests
RL	reinforcement learning
RSC	random setpoint control
SAC	Soft-Actor-Critic
SVM	support vector machines
UKF	unscented Kalman filter
WB	white-box
WBMPC	white-box MPC
ad	adapted
amb	ambient
conv	convection
dev	devices
floor	floor
hr	heat recovery
ig	internal gains
lb	lower boundary
meas	measured
occ	occupancy
rad	radiation
roof	roof
set	setpoint
sol	solar
ub	upper boundary
wall,int	internal walls
wall,ext	external walls
win	windows

els. While the former is based on full knowledge of the underlying physical behavior of the target system, the latter solely calculates statistical relations between in- and output using measurement data [11,12].

Gray-box models are a trade-off between both modeling approaches, for which the fundamental physical relations are implemented and enriched by data-based information. All three modeling approaches have been successfully applied in MPC applications [13].

According to [14], *white-box MPC (WBMPC)* show overall good performance due to their accurate representation of the target system. However, model development is costly, and detailed expert knowledge is necessary. In addition, the process model is not adaptive, which is why changes in boundary conditions and system periphery cannot be considered automatically. To face this challenge, *adaptive black-box MPC (BBMPC)* approaches are promising [15–17]. Instead of relying on a detailed, physics-based model, the process model is derived based on measurement data, usually by applying system identification or machine learning techniques. Consequently, the modeling effort is significantly lower than for WBMPC approaches. Furthermore, online learning can easily be integrated, addressing the challenge of changed boundary conditions over the system's lifetime [17,18]. However, high-quality data is crucial and robust extrapolation cannot be guaranteed. A compromise between the white- and the adaptive black-box approach lies in the *adaptive gray-box MPC (GBMPC)* method. Providing a basic model structure for the process model reduces the amount of measurement data and the necessary expert knowledge compared to the BB and WB approaches [19,20]. Thus, GB models are well-suited for optimization and control applications [21]. However, GBMPC can only capture effects that are considered in the basic model structure.

Common disadvantages of the MPC methods described above are the necessity of sophisticated data infrastructure and higher

hard- and software requirements compared to conventional control techniques [8,10]. These are the reasons why rule-based controllers are more popular and widespread in practice. In this context, *approximate MPC (AMPC)* applications are promising [22,23]. A machine-learning model is derived by learning the relation between the in- and output of an optimal controller. The resulting machine-learning model is deployable on local hardware [24]. Thus, the hardware and software requirements are significantly lower, and the data infrastructure is simpler. The training data results from the closed-loop operation of an MPC-controlled system, the so-called teacher MPC [24]. This teacher MPC can theoretically follow any MPC modeling paradigm (black-, white-, or gray-box), but white-box approaches are most common and adequate [22,24,23]. Another method that has attracted increased attention in the field of optimal control of building energy systems in recent years is *reinforcement learning (RL)* ([25]). It addresses the disadvantages of MPCs requiring high modeling effort, their lacking adaptability (at least in conventional MPC methods) as well as their limited maximum manageable complexity. In RL, an algorithm (or software agent) learns a control strategy by interacting with the system to be controlled. The standardized structure of an RL application consists of a closed loop where the algorithm is given a current state of the system, computes an action decision, writes it back to the system, and receives a reward signal. The algorithm's goal is then to maximize the cumulative reward over time ([26]). By doing so, RL can improve its control strategy over time in contrast to the traditional WBMPC and AMPC approach. We observe that scientific research has already proven that many methodologies and combinations of techniques exist that can contribute to optimized building energy system operation. Furthermore, we conclude that each method has its unique advantages and disadvantages and that all methods address different shortcomings of conventional or optimal control methods. However, to the author's best knowledge, the scientific literature is currently lacking a detailed comparative study that benchmarks all of the introduced advanced control methods using the same target system under the same boundary conditions. Our contribution to close these gaps is as follows:

- We present and compare the operation results of a white-box, a gray-box, a black-box, and an approximate MPC- as well as a reinforcement learning- and an advanced rule-based controller for a single office zone.
- We evaluate the controllers based on a comparable tool set using the same boundary conditions. We demonstrate it based on a supervisory control problem applied to a simulation model.
- Rather than solely focusing on quantitative results, we introduce soft criteria and discuss the involved engineering efforts to highlight each method's different advantages and disadvantages.

1.1. Methods, use-case, and structure of this work

Fig. 1 illustrates the five advanced control methods that show high research potential and are compared with each other in this study. As a fair benchmark system, we develop an advanced RB controller, which exploits the energy system's flexibility. The control methods are applied to a single office zone. The zone's energy system comprises an air-handling unit air-handling unit (AHU) and a concrete-core activation (CCA). The manipulated variables are the AHU's set temperature and the CCA's heat flow. The local control layer's actuators translate the set points into direct actuator control signals. We evaluate the control quality of the presented approaches using the thermal discomfort as integrated violation of comfort constraints and the overall energy consumption.

In this study, we first give a brief overview of each method's state-of-the-art and relevant work in Section 2. The use-case and control task is described in Section 3 followed by a detailed description of each implemented controller in Section 4. We apply the controllers to a single office zone for a full-year simulation in Section 5 and discuss them in Section 6.

2. Advanced control strategies for building energy systems in literature

2.1. Model predictive control for building energy systems

An MPC uses a mathematical process model to predict the system's reaction to the inputs. Then an optimizer calculates the optimal trajectory of inputs for a given cost function. Typically, in the context of building energy system (BES), the cost function covers energy costs as well as thermal comfort. The latter is either considered by applying desired temperature ranges or more complex indicators like the PMV index. The choice of comfort index can strongly influence the complexity of the control problem. Extensive work on (model predictive) comfort control for BES considering thermal and visual comfort as well as indoor air quality is provided by Castilla et al. [27,28].

Crucial for the performance of the MPC is the underlying process model. For BES, predominantly resistance–capacitance (RC) models are used as process models. The capacitances describe the various thermal masses linked to each other via heat transfer resistances, such as air volumes or components. The energy supply and distribution of the building are usually taken into account via power balances as an algebraic system of equations in the process model. [13,29].

To reduce the modeling effort of model-predictive building control, various publications use generic modeling libraries such as the Modelica libraries Buildings [30], and IDEAS [14], which are translated into optimization models via toolchains such as "TACO" [14]. Other examples are the Modelica library Aixlib [31] providing the process model for a nonlinear distributed MPC by Mork et al. [32] and the BRCM Matlab toolbox [33] which is used to implement an MPC for a swiss office building in [4].

However, the implementation effort is enormous, especially due to the modeling and the tuning effort of the controller, which is why the application of this control strategy is uneconomical in most cases [4,34].

2.2. Adaptive gray-box model predictive control

To overcome the aforementioned modeling and parameterization effort of WBMPC, adaptive control approaches are required. Adaptive control mechanisms for classical control approaches have been well-studied in recent decades. In contrast, adaptive model predictive control, especially for the building sector, requires further research ([35–37]). In adaptive GBMPC, a gray-box model includes basic equations describing the general behavior of the controlled system. At the same time, the parameters are continuously re-calibrated based on measurements. Since the model structure is known and the model behavior is adaptable, gray-box models are well suited for application in MPC. [21,20].

For instance, the authors of [38] develop a GBMPC for a thermal zone that is supplied by a floor heating system and an air-handling unit. An RC approach models the system, and the model parameters are updated weekly. Compared to a proportional-integral-derivative (PID) controller, the proposed adaptive GBMPC leads to 22.2% lower energy consumption and increased indoor comfort. Further, Zeng et al. [39] implement an adaptive GBMPC for an HVAC system. The model is linear, and parameter estimation is

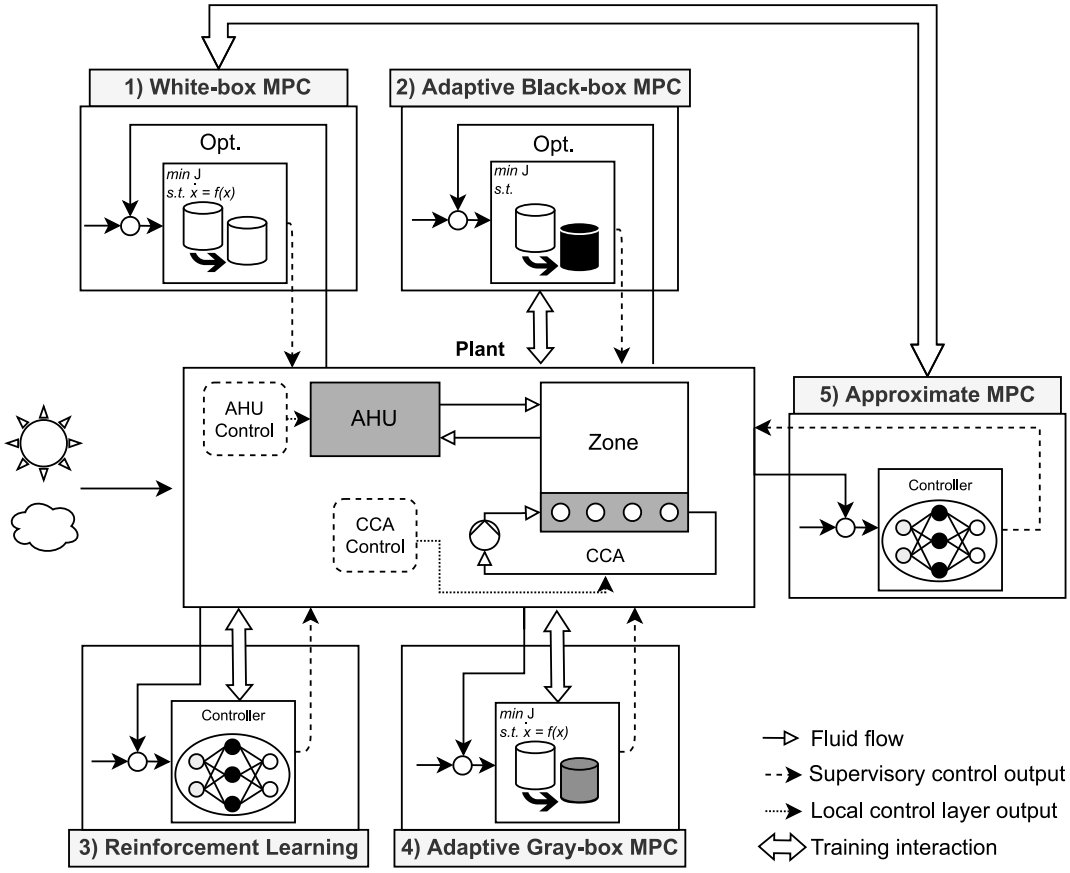

Fig. 1. This study's advanced control methods are applied to the simulation model of a single office zone. The control variables include an AHU's supply air temperature as well as a CCA's heat flow.

performed periodically. Compared to a baseline controller, the controller leads to 26.8% energy reduction. For online parameter estimation at each time step, Kalman-filter-based approaches such as the extended Kalman filter (EKF) ([40]) or the unscented Kalman filter (UKF) ([41]) are commonly used [42]. Fux et al. [43] present an adaptive GBMPC for a passive house and use an EKF for the online estimation of the model parameters. The parameters converge after three weeks and lead to a robust prediction. An UKFF filter is used in [44] for an adaptive GBMPC for a building. A disadvantage of the EKF and UKF is that parameter constraints cannot be explicitly considered. Here, the moving horizon estimation (MHE) is a promising approach and has been considered for parameter estimation in recent years. ([40,45]).

Kümpel et al. [46] present an adaptive GBMPC for heating and cooling coil subsystems of air-handling units based on a gray-box approach combined with an MHE. The adaptive GBMPC is applied to different heating and cooling coil subsystems in a simulation. The results show that the adaptive GBMPC leads to higher control quality and less energy consumption than a well-tuned PI controller. Additionally, the adaptive GBMPC needs no further tuning when applied to the different heating and cooling coil subsystems. Stoffel et al. [47] developed an adaptive GBMPC for a geothermal field using an RC approach for the ground. The geothermal field includes 41 probes, and a moving horizon estimator continuously determines the states and model parameters. Taken together, GBMPC with online parameter estimation is a promising approach. In this paper, a GBMPC is developed for a thermal zone that requires only a few input parameters and quickly adapts to the actual system behavior.

2.3. Data-driven black-box model predictive control

To mitigate the costly and time-consuming modeling effort, data-driven BBMPC approaches have increasingly become the focus of scientific literature [15]. In BBMPC, a data-driven black-box model represents the controlled system's dynamical behavior. Since physics-based models usually simplify and neglect certain aspects, well-trained black-box models can even outperform them [19]. Black-box modeling techniques that are often used in combination with model predictive control for buildings are, for example, linear systems identification methods such as autoregressive model with exogenous inputs (ARX), autoregressive-moving average with exogenous inputs (ARMAX), or 4SID [13,15,48], or machine learning methods like Gaussian process regression (GPR) [49,18,50], random forests (RF) [51,16,52], and artificial neural networks (ANNs) [53,54,17].

ANNs, in particular, show high accuracy in building modeling [55,19]. Several researchers demonstrate the application of BBMPC based on ANNs. Yang et al. [17] implement a BBMPC based on ANNs for a real-life office building and a lecture hall, resulting in energy savings of 58.5% and 36.7%, respectively. To solve the non-convex optimization problem, the authors first employ the exhaustive search method and then refine it with a gradient-based solver. Due to the long computation times, the authors also mimic the controller behavior with AMPC in another publication [56] (see chapter 2.4).

Bünning et al. [53] use input-convex ANNs, a special architecture of ANNs[57], to obtain a convex optimization problem when applying a BBMPC to control the temperature of a bedroom. The

controller maintains the room temperature within comfort boundaries while showing energy-saving behavior. The convex, nonlinear optimal control problem (OCP) is solved by the black-box algorithm COBYLA, which numerically approximates the gradients of the OCP.

To provide gradients more efficiently, Jain et al. [54] couple the automatic differentiation capabilities of Tensorflow with the interior-point optimizer Ipopt [58]. Using this methodology, the authors efficiently control the heating system of a two-story building resulting in energy savings of 5.7% compared to a baseline controller.

An advantage of ANNs is that they are categorized as a parametric machine learning method. Therefore, compared with Gaussian processes and random forests, the complexity of the OCP does not depend on the number of considered data points [50]. This facilitates the implementation of online learning by continuously improving the process model with new measurements [17].

The major disadvantage of common ANNs is the resulting nonlinear, nonconvex optimization problem, which reduces the scalability. It makes long-term simulations to test the controller time-consuming [53,59].

This publication presents a new methodology to efficiently integrate ANNs in OCPs using CasADi[60] specialized for nonlinear optimization. Additionally, we employ online learning to improve the controller quality continuously.

2.4. Approximate model predictive control

A potential method that addresses the MPC's drawbacks of high hard- and software as well as data infrastructure requirements is the concept of AMPC. For AMPCs, a simplified mathematical model, which is deployable on low-level hardware, is used to imitate the MPC's optimized control actions. This process is known as rule mining. In this context, statistical methods like logistic or linear regression and machine learning-based methods like decision trees (DT) and ANNs are promising for deriving mathematical models [61,24]. Over the last years, a few studies have applied the AMPC concept to BES. These studies can be categorized, e.g., regarding the chosen machine learning method, the control task, and the involved heating ventilation air conditioning (HVAC) components. Because of their comprehensive model structure, which follows the "if-then-else" concept commonly used for rule based controller (RBC) applications, DTs are a popular rule mining method due to their high interpretability [24,62–64]. Nonetheless, they tend to generalize poorly and to fail in imitating more complex control tasks [24]. Other more advanced ML methods comprise Random Forests [65], time-delayed ANNs [24], AdaBoost [66], recurrent ANNs [56] or support vector machines [67]. We focus on ANNs due to their high performance in the above-listed studies and the comparability with the other advanced control methods of this work. Apart from that, most of the analyzed studies applying AMPC have focused on decentral single zone control tasks with simplified generation systems. For example, Klaučo et al. [68] imitate an ideal heater and cooler. Furthermore, Žáčeková et al. [69] manage to mimic the AHU damper position of an office room while Drgoňa et al. [24] learn a gas boiler's optimal on–off schedule. To the authors' best knowledge, there is no study investigating the simultaneous imitation of two interacting control actions, which impedes robust control development. We close this gap by simultaneously imitating an AHU's set temperature as well as the set heat flow from a CCA.

2.5. Reinforcement learning for BES optimization

In recent years, more and more design principles and innovations from the field of RL are being published, and state-of-the-art algorithms, like the Deep Q-Networks (DQN) ([70]) for discrete control actions or the Soft-Actor-Critic (SAC) for continuous control actions, can quickly find a good and finally optimal control strategy if the RL specific design principles are well addressed. These principles concern in particular the proper parameterization of the algorithm according to the task and the clean formulation of the control problem in the form of a fully observable Markov Decision Problem (MDP). However, a particular challenge lies in the clean formulation of the MDP, which often stands in the way of its widespread application in practice ([25]). In the scientific literature, on the other hand, there is a growing number of successfully application examples for different use cases.

In ([71]), the use of deep RL for HVAC control under dynamic electricity prices was investigated. The authors show that the DQN algorithm used is able to effectively utilize the thermal inertia of a simulated building to save costs. Furthermore, a method for state-space description is proposed for problems that are not fully observable.

Another study on using RL for BES energy management applications was published by Brandi et al. ([72]). The authors investigate the use of a DQN algorithm for the supply water temperature control of a simulated office building and compare different training strategies and state-space variations. On the one hand, savings between 5 and 12 percent are achieved compared to a rule-based approach. On the other hand, the authors also show the importance of careful problem formulation when using RL.

Mathew et al. ([73]) used the latest design principles for the applied DQN algorithm. They tested an application for smart home energy management systems. In addition to the prioritized usage of stored training data, convergence was improved by an innovative exploration strategy and reward function. This allowed the algorithm to learn quickly and efficiently to avoid electrical peak loads.

A study with a stronger focus on indoor air quality was published in ([74]). The authors demonstrate the good adaptivity and applicability of the DQN algorithm. The algorithm achieves energy cost savings compared to a benchmark MPC under some climatic conditions.

Biemann et al. ([75]) apply a SAC algorithm to optimize the operation of a simulated data center HVAC system. They also compare four different model-free actor-critic algorithms. They show that the trade-off between thermal stability and energy savings is increasingly well handled during training by all four. While all algorithms were robust to changes in the boundary conditions, the SAC clearly showed the highest data efficiency and is therefore proposed for the studied problem.

Another study on the use of SAC was published in ([76]). The authors used the algorithm for cost optimization for heating an office building using the thermal mass of the building as flexibility. Applied to a simulated EnergyPlus model, the algorithm could save costs (against an RBC baseline) after three simulated months without violating thermal comfort. The algorithm was only provided inputs covered by standard sensors in conventional office buildings. Other applications for the successful use of RL for HVAC control and coordinating the power purchases of multiple buildings in a district can be found in ([77–79]). The following comprehensive review articles are recommended to the interested reader ([80–82]). In summary, the opportunities of RL for BES lie in the high adaptivity, the comparatively low engineering effort, and the performant processing of even very complex state spaces. On the other hand, challenges lie in the relatively long training times and the safe application and training on real systems.

In this paper, we use the state-of-the-art, model-free SAC algorithm ([83]), with hyper-parameters tuned via a Bayesian hyperparameter optimization. Beyond the comparison with the other methods, a significant contribution to the application of SAC lies

in the inclusion of problem-optimized hyper-parameters with a structured procedure. In most studies, this issue is addressed only by a sensitivity analysis or not at all.

2.6. Benchmarking and comparison of advanced BES control strategies in literature

The literature reviews above show that each of the considered control approaches offers great potential. This raises the question of which algorithm to choose for the individual control problem for a control engineer.

Hence, there is significant effort in the literature to benchmark control strategies among each other.

Picard et al. [84] derive linear state-space models from a building simulation model and apply model order reduction to obtain different process models for an MPC. The controllers based on the different process models are then compared based on quantitative key performance indicators (KPIs). It is concluded that energy consumption increases when the model mismatch is more considerable.

A comparison of WB-, GB-, and BBMPC for a real-life building is presented by Arroyo et al. [85]. In this publication, the black-box model has the same representation as the gray-box model but is not constrained by any physical insight. Therefore, all models have a similar structure derived from RC modeling while differing in size and parameter selection. The authors state that increasing the amount of physical detail in the model increases robustness in prediction and control performance but should be carefully handled.

A comparison of solely BBMPC algorithms is provided by Bünning et al. [59]. Here, the authors test BBMPC based on input-convex ANNs, random forests, and physics-informed ARMAX process models on a real-life building with one control input. In this publication, the ARMAX model outperforms the other approaches regarding sample efficiency and control quality.

In ([86]), the authors compared state-of-the-art continuous and discrete RL algorithms for maximum PV self-consumption in an optimal charging scheduling problem for electric vehicles. The authors underline that although RL is outperformed by different MPC (stochastic/deterministic) benchmarks, its performance is near-optimal and superior to rule-based control with much lower computational costs. They conclude that RL algorithms are a suitable technology for scalable and near-optimal electric vehicle charging.

A similar study is presented by Ceusters et al. [87]. They compare a mixed-integer MPC with perfect and non-perfect forecasts to two RL agents on two case studies of multi-energy systems. In this publication, the process model of the MPC is derived by linearization of the plant model. In the considered decision-making problem, the RL approach can outperform the MPC. This result is explained by the linearization error of the MPCs' process model. Furthermore, the RL agent learns an On/Off policy instead of a continuous policy.

Arroyo et al. [88,89] compare GBMPC, RL, and RL-MPC, a combined algorithm, to the Boptest benchmarking framework [90]. In these publications, the GBMPC outperforms the other algorithms, while the authors state that the GBMPC lacks *learning*. This means it cannot adapt to changing conditions. Furthermore, as stated by the authors, a limitation of the work is the lack of testing in multi-input building systems, where RL agents are expected to be more challenged. In our work, we contribute to these investigations by first providing adaptive GB- and BBMPC algorithms and second testing on a multi-input building energy system.

Another analysis of advanced control algorithms for BES is carried out by di Natale et al. [91]. Here, a gaussian process-based BBMPC, an RL approach, and a robust, bilevel, data-enable predictive control (DeePC) algorithm, each applied to a different real-life use case, are compared qualitatively. The authors conclude that none of the approaches can solve all BES control problems since each methodology has drawbacks. The gaussian process-based BBMPC is sample efficient but relies on more manual tuning. At the same time, DeePC is straightforward to deploy but is only well suited for linear problems. The RL algorithm reduces the needed expert knowledge and is flexible but requires large amounts of data, time-consuming offline training, and can cause online constraint violations.

2.7. Summary and Contributions

In summary, there is great interest in comparing different advanced control approaches for BES in scientific literature. Nevertheless, to the author's best knowledge, there is no extensive benchmark considering all MPC modeling paradigms, RL, and AMPC on the same use case using the same boundary conditions. For a reproducible study design, we choose a standardized use case. Furthermore, in the presented literature, the algorithms are either compared quantitatively or qualitatively, considering softer criteria. In this work, we pursue both a quantitative and a qualitative benchmark. Here, we also discuss the engineering effort involved in each method.

3. Use Case and control task

A detailed simulation model is used as a controlled system to test the different control strategies under repeatable boundary conditions. The controlled system consists of a parameterized thermal zone according to ASHRAE140 test case 900. [92]. The thermal zone is supplied with heat and cold using a CCA and an AHU, as depicted in Fig. 2. The system is modeled in Modelica using the AixLib-library[1] [31]. The hydronic models used for the air-handling unit and concrete core activation include all control-relevant sensors and actuators and cover the dynamic and static behavior at a high level of detail ([93]). The thermal zone model is based on a thermal RC approach as described and validated by Lauster et al. [94,95]. The AHU provides air at the desired temperature set point $T_{ahu,set} \in [18\,°C, 25\,°C]$ and includes a heater, a cooler, a heat recovery, and two fans. The heater and cooler are controlled by proportional-integral (PI) controllers, which adjust the mixing valves of the heat and cold supply to reach the desired temperature set point.

Furthermore, the air mass flow of the AHU can be adjusted by controlling the fans. The hydronic circuit of the CCA is coupled to the heat and cold supply via heat exchangers. The thermal power transmitted to the CCA $\dot{Q}_{cca,set} \in [-5\,kW, 5\,kW]$ is controlled by adjusting the mass flow of the heat and cold supply via regulating valves. Controlling the heat flow of a CCA is not state-of-the-art since, usually, the inlet temperature is controlled via a heating curve. Nevertheless, to assess the system's energy consumption, the mass flow of the CCA needs to be measured. This measurement also allows measurement and control of the heat flow. The AHU's and CCA's pumps are operated at a constant speed and deactivated if there is no demand for heat or cold. The system is influenced by weather and internal gains caused by humans, lights, and electrical devices. For the latter, the office load profiles provided by SIA [96] are used. The control task is to keep the air temperature of the thermal zone within the comfort constraints (Table 1) while minimizing the thermal energy consumption. Since an ideal heat and cold source is used, heat and cold consumption are assumed to be equally expensive. In a real setting, the AHU would typically require a higher temperature difference, making operation more

[1] The model is available at: https://github.com/RWTH-EBC/AixLib

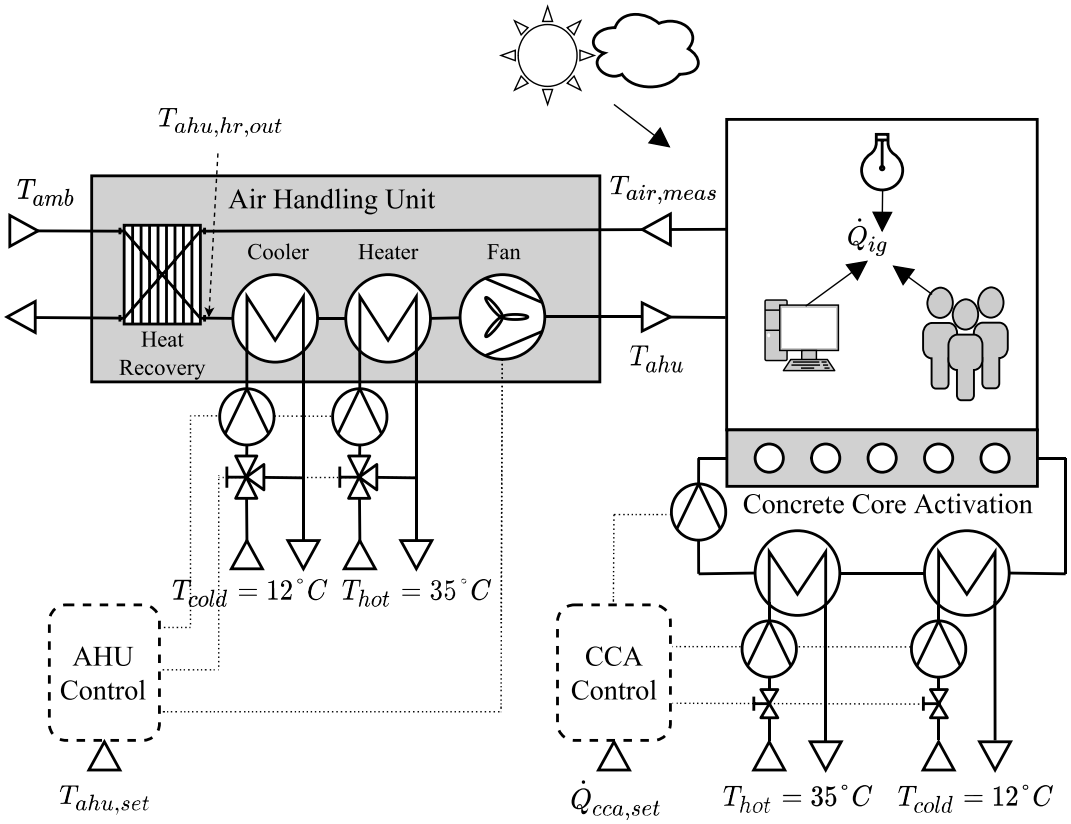

Fig. 2. Schematic representation of the thermal zone.

Table 1
Comfort constraints for the rooms' air temperature.

Time	$T_{air,min}$	$T_{air,max}$
Mon.-Fri. 7:00–19:00	21 °C	23 °C
Mon.-Fri. 19:00–7:00, Sat.-Sun.	17 °C	27 °C

expensive than operating the CCA. The task's challenge is integrating fast (AHU) and slow dynamics (CCA) and conflicting goals. Furthermore, the system controller relies on imperfect subsystem controllers to reach the set points $T_{ahu,set}$ and $\dot{Q}_{cca,set}$ while reacting to the various disturbances.

4. Implemented Controllers

4.1. Rule-based controller

For a fair benchmark with a baseline controller, we design a RB controller which exploits the comfort constraints similar to an MPC and deals with heating and cooling. To account for the thermal inertia of the system, the comfort constraints are adapted for the RB controller, as displayed in Fig. 3. The RB controller uses two PI controllers for the CCA, $\text{PI}_{\text{cca,cool}}$ and $\text{PI}_{\text{cca,heat}}$. $\text{PI}_{\text{cca,cool}}$ is set to follow the adapted upper boundary $T_{ub,ad}$ and computes the cooling power of the CCA $\dot{Q}_{cca,cool,set} \in [-5\,\text{kW}, 0\,\text{kW}]$. Analogously, $\text{PI}_{\text{cca,heat}}$ follows $T_{lb,ad}$ and computes the heating power $\dot{Q}_{cca,heat,set} \in [0\,\text{kW}, 5\,\text{kW}]$. Thus, the setpoint for the CCA calculates

$$\dot{Q}_{cca,cool,set} + \dot{Q}_{cca,heat,set} = \dot{Q}_{CCA,set} \tag{1}$$

The AHU is controlled by a single, limited PI controller, which provides the setpoint $T_{ahu,set} \in [18\,^{\circ}\text{C}, 25\,^{\circ}\text{C}]$ depending on the measured room temperature.

$$T_{ahu,set} = PI_{ahu}(T_{air,meas}, T_{air,set}) \tag{2}$$

With:

$$T_{air,set} = \begin{cases} T_{lb,ad}, & \text{if} \quad T_{air,meas} \leqslant T_{lb,ad} + 0,5 \\ T_{ub,ad}, & \text{if} \quad T_{air,meas} \geqslant T_{ub,ad} - 0,5 \\ T_{ahu,hr,out} & \text{else} \end{cases} \tag{3}$$

Here, $T_{ahu,hr,out}$ denotes the outlet temperature of the heat recovery system. Using this setpoint deactivates the operation of the heater and cooler. The latter is considered to maximize the free float operation of the AHU, saving energy. The tunable parameters of the controller are the P and I values of six individual PI controllers, the threshold for the AHU control, and the slope of the adapted comfort constraints. In conclusion, the RB controller exploits the comfort constraints by following the relaxed comfort constraints and enabling free float operation in between. Compared to the state-of-the-art, this controller is rather complex. Nevertheless, it represents a fairer benchmark to evaluate the performance of the advanced control strategies presented in this paper.

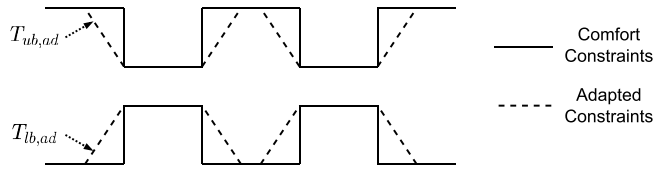

Fig. 3. Adapted comfort constraints for the rule based controller.

4.2. Detailed white-box MPC

The second investigated controller is a detailed WBMPC using an RC model with the same structure as the plant model. The model considers six thermal capacities; air volume (*air*), internal walls (*wall,int*), external walls (*wall,ext*), concrete core activation (*cca*), roof (*roof*), and windows (*win*). The capacities i and j are connected via the heat transfer coefficients h_{ij} (4). The heat transfer coefficients take radiation, convection, and conduction into account. In Eq. (4) the term $\dot{Q}_{u,d}$ sums up the influence of disturbances and inputs. This term is introduced in Eq. (5).

$$C_i \frac{dT_i}{dt} = \sum h_{ij} \cdot (T_j - T_i) + \dot{Q}_{u,d} \tag{4}$$

$$\dot{Q}_{u,d} = \begin{cases} \dot{Q}_{ig,conv} + \dot{m}_{air,ahu} \cdot c_{p,air}(T_{ahu,set} - T_{air}), & \text{for } i = air \\ \dot{Q}_{cca,set}, & \text{for } i = cca \\ h_{amb,i}(T_{amb,i} - T_i) + \psi_{i,sol} \cdot \dot{Q}_{sol} + \psi_{i,ig} \cdot \dot{Q}_{ig,rad} & else \end{cases} \tag{5}$$

The air volume is affected by the convective internal gains $\dot{Q}_{ig,conv}$ and the enthalpy flow from the AHU. For the process model of the WBMPC, perfect subsystem controllers are assumed. Therefore, the enthalpy is calculated with the temperature setpoint $T_{ahu,set}$. The same assumption is used for the CCA, which is affected by $\dot{Q}_{cca,set}$. The other capacities are affected by radiative internal gains, heat transfer to the environment, and solar radiation. The solar radiation $\dot{Q}_{sol}$ entering through the windows is allocated to the individual components based on view factors $\psi_{i,sol}$. The radiative internal gains $\dot{Q}_{ig,rad}$ are allocated in the same way, using the view factors $\psi_{i,ig}$. The internal gains $\dot{Q}_{ig,conv}$ and $\dot{Q}_{ig,conv}$ are functions of air temperature as well as schedules for light s_{light}, occupancy s_{occ} and devices s_{dev} [96]. To calculate the heat transfer to the environment, corrected ambient temperatures $T_{amb,i}$ for each component are used. These corrected temperatures are determined by considering radiative effects based on the component's orientation. These temperatures can be calculated in advance in the control loop based on weather forecasts.

To minimize the overall energy consumption in the controller, a model of the AHU's thermal power is needed. For this reason, an empirical, linear expression (6) considering the heat recovery is derived from simulation data to approximate the complex nonlinear behavior of the AHU.

$$\dot{Q}_{ahu} = \dot{m}_{air,ahu} \cdot c_{p,air}\left(T_{ahu,set} - \frac{0.95 \cdot T_{air} + 1.05 \cdot T_{amb}}{2}\right) \tag{6}$$

In total the process model has six states x, three algebraic variables y, two inputs u, and nine disturbances d (Eq. 7a–14d).

$$x = [T_{air}, T_{wall,int}, T_{wall,ext}, T_{roof}, T_{win}, T_{cca}] \tag{7a}$$
$$y = [\dot{Q}_{ahu}, \dot{Q}_{ig,rad}, \dot{Q}_{ig,conv}] \tag{7b}$$
$$u = [T_{ahu,set}, \dot{Q}_{cca,set}] \tag{7c}$$
$$d = [T_{amb}, T_{amb,roof}, T_{amb,win}, T_{amb,walls}, s_{occ}, s_{light}, s_{dev}, \dot{Q}_{sol}, \dot{m}_{air,ahu}] \tag{7d}$$

The detailed white-box controller uses the linear process model presented above in an economic MPC scheme (Eq. 8a–8h). Here, the violation of temperature constraints ϵ is penalized for maintaining comfort over the prediction horizon N. Furthermore, the thermal powers of the CCA and the AHU are considered in the cost function to minimize energy consumption. The last term in the cost function considers the change in decision variables to prevent the oscillations of the manipulated variables.

$$\min_{u,x,\epsilon} \sum_{k=0}^{N-1} \left(\epsilon_k^2 \cdot W + \dot{Q}_{ahu,k}^2 \cdot R + \dot{Q}_{cca,set,k}^2 \cdot R + \Delta u_k^2 \cdot dR\right) \tag{8a}$$
$$s.t. \quad x_{k+1} = f(x_k, u_k, d_k) \tag{8b}$$
$$y_k = g(x_k, u_k, d_k) \tag{8c}$$
$$u_{min} \leqslant u_k \leqslant u_{max} \tag{8d}$$
$$T_{air,min,k} - \epsilon_k \leqslant T_{air,k} \leqslant T_{air,max,k} + \epsilon_k \tag{8e}$$
$$x_0 = \hat{x}_0 \tag{8f}$$
$$0 \leqslant \epsilon_k \tag{8g}$$
$$\forall k \in [0, \ldots, N-1] \tag{8h}$$

The controller is implemented in PYOMO [97] using a collocation discretization scheme and Gurobi [98] as a optimizer. The tuning parameters of the WBMPC are the prediction horizon and the time step size, which are chosen to be 8 h and 15 min, respectively, and the weights of the cost function. The weights are $W = 250, R = 1$, and $dR = 2$. If the operating costs were different for the CCA and the AHU, the terms could be weighted differently.

4.3. Adaptive gray-box MPC

The third considered controller is an adaptive GBMPC based on a gray-box model. The general MPC formulation is equal to the WBMPC. However, the model equations, states, and considered disturbances of the GBMPC differ from the WBMPC and are described in the following. A gray-box model includes physics-based equations that reproduce the general behavior of the controlled system. However, the exact behavior of the model is described by parameters that have to be determined based on measured data. We implement an RC model to model the investigated thermal zone. Reynders et al. ([99]) showed that an RC model with two capacities and two resistances leads to acceptable model behavior. Based on this approach, we additionally consider the mass of the internal walls and the concrete core activation resulting in an RC model with four capacities and four resistances. Radiation and direct heat transfer between the walls is neglected. The developed gray-box model is given in Eq. 8–12. Here, the parameters $C_{\text{air}}, C_{\text{wall,int}}, C_{\text{wall,ext}}$ and C_{cca} denote the capacity of the air, the inner wall, the outer wall and the concrete core activation. The parameters $1/R_{\text{cca}}, 1/R_{\text{wall,int}}, 1/R_{\text{wall,ext}}$ and R_{cca} denote the resistances of the concrete core activation, the inner wall and the outer wall, respectively. Further, temperatures of the concrete core activation T_{cca}, the interior wall $T_{\text{wall,int}}$ and exterior wall $T_{\text{wall,ext}}$ as well as the solar radiation $\dot{q}_{\text{sol,dir}}$ and the internal gains $\dot{Q}_{\text{ig}}$ are considered. The areas of the floor, inner wall, exterior wall and the windows are constant and have to be set by the user. The parameter f_{rad} corresponds to the shading and transmissivity of the windows. The occupation times s_{occ} are based on a fixed profile.

$$\begin{aligned} C_{\text{air}} \cdot \frac{\mathrm{d}T_{\text{air}}}{\mathrm{d}t} = {} & c_{\text{p}} \cdot \dot{m}_{\text{air,ahu}} \cdot (T_{\text{ahu,set}} - T_{\text{air}}) + 1/R_{\text{cca}} \cdot A_{\text{floor}} \cdot (T_{\text{cca}} \\ & - T_{\text{air}}) + 1/R_{\text{wall,int}} \cdot A_{\text{wall,int}} \cdot (T_{\text{wall,int}} - T_{\text{air}}) \\ & + 1/R_{\text{wall,ext}} \cdot A_{\text{wall,ext}} \cdot (T_{\text{wall,ext}} - T_{\text{air}}) + \dot{q}_{\text{sol,dir}} \\ & \cdot A_{\text{win}} f_{\text{rad}} + \dot{Q}_{\text{ig}} \cdot s_{\text{occ}} \end{aligned} \tag{8}$$

$$C_{\text{wall,int}} \cdot \frac{\mathrm{d}T_{\text{wall,int}}}{\mathrm{d}t} = 1/R_{\text{wall,int}} \cdot A_{\text{wall,int}} \cdot (T_{\text{air}} - T_{\text{wall,int}}) \tag{9}$$

$$\begin{aligned} C_{\text{wall,ext}} \cdot \frac{\mathrm{d}T_{\text{wall,ext}}}{\mathrm{d}t} = {} & 1/R_{\text{wall,ext}} \cdot A_{\text{wall,ext}} \cdot (T_{\text{air}} - T_{\text{wall,ext}}) \\ & + 1/R_{\text{wall,ext}} \cdot A_{\text{wall,ext}} \cdot (T_{\text{amb}} - T_{\text{wall,ext}}) \end{aligned} \tag{10}$$

$$C_{\text{cca}} \cdot \frac{\mathrm{d}T_{\text{cca}}}{\mathrm{d}t} = \dot{Q}_{\text{cca,set}} + 1/R_{\text{cca}} \cdot A_{\text{floor}} \cdot (T_{\text{air}} - T_{\text{cca}}) \tag{11}$$

$$\dot{Q}_{\text{ahu}} = \dot{m}_{air,ahu} \cdot c_{p,air}\left(T_{ahu,set} - T_{air}\right) \tag{12}$$

The gray-box model has four states x, one algebraic variable y, two inputs u, and four disturbances d (Eq. 7a–14d).

$$x = [T_{air}, T_{wall,int}, T_{wall,ext}, T_{cca}] \tag{14a}$$
$$y = [\dot{Q}_{ahu}] \tag{14b}$$
$$u = [T_{ahu,set}, \dot{Q}_{cca,set}] \tag{14c}$$
$$d = [T_{amb}, \dot{q}_{\text{sol,dir}}, \dot{Q}_{\text{ig}}, \dot{m}_{air,ahu}] \tag{14d}$$

Similar to the WBMPC, the objective function includes the deviation from the comfort range ϵ_k ($a_1 = 1$), the change of the control variables $\Delta\dot{Q}_{\text{cca,set}}$ ($a_2 = 0.0355$) and $\Delta T_{\text{ahu,set}}$ ($a_3 = 26$) and the consumed energy of the CCA Q_{cca} and AHU Q_{ahu} ($a_4 = a_5 = 0.1$).

$$J_{\text{GB,MPC}} = \sum_{k=0}^{N-1}(a_1 \cdot \epsilon_k^2 + a_2 \cdot \Delta\dot{Q}_{\text{cca,set},k}^2 + a_3 \cdot \Delta T_{\text{ahu,set},k}^2 + a_4 \cdot \dot{Q}_{\text{cca,set},k} + a_5 \cdot \dot{Q}_{\text{ahu},k}) \tag{15}$$

To estimate the unknown capacities, resistances, the parameter f_{rad} and the internal gains as well as the current temperature of the walls and the CCA, an MHE based on the model presented above is used. The objective of the MHE is given in Eq. 16.

$$J_{\text{GB,MHE}} = \sum_{t=-M}^{0}(T_{\text{air},k} - T_{\text{air,meas},k})^2 \tag{16}$$
$$+ c_1 \cdot (R_{\text{cca}} - R_{\text{cca,act}})^2 \tag{17}$$
$$+ c_2 \cdot (R_{\text{wall,int}} - R_{\text{wall,int,act}})^2 \tag{18}$$
$$+ c_3 \cdot (R_{\text{wall,ext}} - R_{\text{wall,ext,act}})^2 \tag{19}$$
$$+ c_4 \cdot (C_{\text{air}} - C_{\text{air,act}})^2 \tag{20}$$
$$+ c_5 \cdot (C_{\text{cca}} - C_{\text{cca,act}})^2 \tag{21}$$
$$+ c_6 \cdot (C_{\text{wall,int}} - C_{\text{wall,int,act}})^2 \tag{22}$$
$$+ c_7 \cdot (C_{\text{wall,ext}} - C_{\text{wall,ext,act}})^2 \tag{23}$$
$$+ c_8 \cdot (f_{\text{rad}} - f_{\text{rad,act}})^2 \tag{24}$$
$$+ c_9 \cdot (\dot{Q}_{\text{ig}} - \dot{Q}_{\text{ig,act}})^2 \tag{25}$$

The objective includes the deviation between the estimated temperature $T_{\text{air},k}$ and the past measured temperature $T_{\text{air,meas},k}$. Further, the deviation of the estimated parameters from the previous estimate (subscript act) is penalized. The parameters c_1 to c_2 are chosen based on the initial guess $R, C_{i,ini}$ of the quantities: $c_i = 1/R, C_{i,ini}^2$. The initial guess is calculated based on the geometry and material properties. The MHE is performed with two different time horizons M: An MHE with a horizon of 6 h at each MPC step is executed to realize short computation times. Additionally, an estimation with a longer horizon of 72 h is performed every 48 h to consider the long-term behavior as well. In this way, the overall computational times can be kept small while considering a sufficient long estimation horizon.

As initial parameters of the GBMPC, the floor area and height are needed to calculate starting values for the capacities of the walls and the CCA. The air capacity is assumed to be constant at the starting value since the air capacity is small compared to the capacity of the walls and the CCA. The resistances are initialized with common default values. As in the case of the WBMPC, the GBMPC's tunable parameters are the prediction horizon (8 h), the step size (15 min), and the cost function's weights. Further, a perfect forecast of the occupation times $s_{\text{occ}}$and weather is assumed.

4.4. Data-Driven black-box MPC

For the BBMPC, we use ANNs to model the quantities of interest $\dot{Q}_{ahu}$ and T_{air}. To account for slow system dynamics caused by thermal inertia, the input features of the ANN consider an individual lag n. This means that a feature's last n values are used to calculate the neural network's output. Thus, the system dynamics are approximated by Eq. 29a and 29b.

$$\Delta T_{air,k} = f_{ann}(x_k, x_{k-1}, \ldots, x_{k-n}, u_k, u_{k-1}, \ldots, u_{k-n}, d_k, d_{k-1}, \ldots, d_{k-n}) \tag{26a}$$
$$\dot{Q}_{ahu,k} = g_{ann}(x_k, x_{k-1}, \ldots, x_{k-n}, u_k, u_{k-1}, \ldots, u_{k-n}, d_k, d_{k-1}, \ldots, d_{k-n}) \tag{26b}$$

The features and lags used to predict ΔT_{air} and $\dot{Q}_{ahu}$ are summarized in Table 2. In comparison to the WBMPC and the GBMPC, fewer variables are considered. The only considered state is the room temperature. The only algebraic variable is the power of the air handling unit. The internal gains are approximated by learning the influence of the time of the day and the day of the week. These are encoded as sinus and co-sinus functions, with frequencies of 24 h and 7 d, respectively. Furthermore, only the most significant weather quantities, the ambient temperature T_{amb}, and the direct solar radiation $\dot{q}_{dir}$ are considered. This results in the following state space:

$$x = [T_{air}] \tag{27a}$$
$$y = [\dot{Q}_{ahu}] \tag{27b}$$
$$u = [T_{ahu,set}, \dot{Q}_{cca,set}] \tag{27c}$$
$$d = [T_{amb}, \dot{q}_{sol,dir}, t_{day}, t_{week}] \tag{27d}$$

The optimization problem of the BBMPC is structured in the same way as in the case of the WBMPC. Only the dynamic constraints (8b) and (8c) are changed to (29a) and (29b). Furthermore, the integration step (28) is introduced.

$$T_{air,k+1} = \Delta T_{air,k} + T_{air,k} \tag{28}$$

The optimization problem is implemented in Casadi [60] and solved with the non-linear interior point solver Ipopt [58] using 'ma57' as a linear solver. Since the prediction of $\dot{Q}_{ahu}$ depends on the air temperature's prediction, a multiple shooting approach [100] is chosen as a transcription method for the optimization problem.

To integrate ANNs in Casadi, the authors developed an interface to Keras [101]. Thus, specialized training algorithms like Adam [102] from Keras can be used, while the optimization is performed in Casadi. This enables faster and more efficient optimization compared to gradient-free approaches. Thus, a prediction horizon of 8 h with a time step size of 15 min analogous to the WBMPC is chosen. The weights are also the same as in the case of the WBMPC. Therefore, the tunable parameters of the BBMPC are similar to the WBMPC. Additionally, we manually tuned the selected features, which should be automized by feature selection methods in the future.

4.4.1. Training and data generation

To successfully apply the BBMPC, the ANNs must accurately approximate the plant model. For this reason, the data generation process is crucial. We generate the training data with two strategies:

Table 2
Features and lags considered to predict the quantities of interest in the BBMPC.

	ΔT_{air}		$\dot{Q}_{ahu}$	
Feature	considered	lag	considered	lag
T_{air}	x	4	x	3
T_{amb}	x	4	x	3
$T_{ahu,set}$	x	3	x	3
$\dot{Q}_{cca,set}$	x	6	-	-
$\dot{q}_{sol,dir}$	x	3	-	-
time of the day	x	3	-	-
day of the week	x	3	-	-

- constant setpoint control (CSC)
- random setpoint control (RSC)

The first strategy controls the room temperature to a constant setpoint within the comfort boundaries. The second strategy sets a new random room temperature setpoint within the comfort boundaries every two hours. For both strategies, simple PI controllers are used to compute the AHU and the CCA setpoints.

Additionally, online learning can be used to improve the controller continuously. In this case, the model is retrained with operation data generated by the BBMPC. The retrained model is then passed to the BBMPC for use in the following period. Using online learning, the controller adapts to changing environmental conditions. In this work, we will investigate two different controllers; one trained for only two weeks in winter and then applying online learning. The other is trained in two winter and two summer weeks and doesn't employ online learning. The data sets are summarized in Table 3. The data sets are split into a training (70%), validation (15,%), and test set (15%). In a separate study, the influence of training data quantity and quality should be analyzed further. Nevertheless, the data are sufficient in this work to train a working controller. We perform a brute force hyperparameter optimization to determine the ANN architectures. Thus, we train several neural networks with a different number of hidden layers and neurons on each layer and choose the most accurate one on the test set. The mean-squared error (MSE) is used as loss function. An ANN with one hidden layer and 16 neurons is trained to predict the room temperature change. The second ANN approximating the AHU power has one hidden layer and eight neurons. Both ANNs use a sigmoid activation function and a batch-normalizing layer as the input layer.

4.5. Approximate white-box MPC

The AMPC approach uses the WBMPC as teacher controller (see Section 4.2).

According to the state of research, different ML algorithms are suitable as imitation methods. As discussed in Section 2.4, we focus on ANNs as machine learning (ML) method due to their high performance in mimicking the teacher MPCs. For both AMPCs, we apply the Python-based ML framework AddMo [103]. AddMo enables the automatic execution of necessary steps when working with ML models, ranging from data scaling, feature engineering, and selection to model selection and hyperparameter tuning. As this work solely focuses on ANNs, we skip the model selection step. We use a full-year simulation of the WBMPC's in- and outputs as input data. The WBMPC's outputs, namely $T_{\mathrm{ahu,set}}$ and $\dot{Q}_{cca,set}$, are the targets while its inputs are the features in the AMPC context. The following functional approximation can be deduced:

$$\Delta T_{ahu,set} = f_{ann}(x_k, x_{k-1}, \ldots, x_{k-n}, d_{k+n}, \ldots, d_{k+1}, d_k, d_{k-1}, \ldots, d_{k-n}) \tag{29a}$$

$$\dot{Q}_{cca,set,k} = g_{ann}(x_k, x_{k-1}, \ldots, x_{k-n}, d_{k+n}, \ldots, d_{k+1}, d_k, d_{k-1}, \ldots, d_{k-n}) \tag{29b}$$

Here, d_i also includes synthetically generated features like the time of day or weekday, which are helpful in the context of machine learning. In the case of $T_{\mathrm{ahu,set}}$, the ML-algorithms predict the setpoint difference compared to the previous time step's setpoint, i.e., $\Delta T_{\mathrm{ahu,set}}$, instead of predicting its absolute value. In this way, we prevent the tuning of purely autoregressive models, which could lead to poor closed-loop control performance as the setpoint tends to dominate the other features. For the manipulated variable $\dot{Q}_{cca,set}$, two types of models are compared in this study. The first model also predicts $\Delta\dot{Q}_{cca,set}$ instead of the absolute value $\dot{Q}_{cca,set}$ while the second predicts $\dot{Q}_{cca,set}$.

The features can be distinguished into states and disturbances. Among the former are solely values of the current or previous time steps. This corresponds to the idea of a simplified integration of local controllers based on measurement data without system modeling and its state predictions. Therefore, the states are integrated as conventional and lagged features. We consider previous, current, and future values for the disturbances corresponding to lagged, conventional, and lead features. Table 8 summarizes all of the considered and finally selected features for the two manipulated setpoints $\Delta\dot{Q}_{cca,set}$ and $T_{\mathrm{ahu,set}}$ of interest. We did not include feature selection for the former model to predict $\dot{Q}_{cca,set}$ as the results are not promising.

All in all, the AMPC approach comprises the following tuner parameters which would need adaptation when applying to another system:

- Number and type of features to be included (compare with Table 8)
- Potential consideration of synthetic features such as time of day, date of year, etc.

Table 3
Data sets used for the training of the DDMPC's process model.

Data	Initial data			Online
set	Total	Winter	Summer	Learning
1	28 d	7 d CSC/ 7 d RSC	7 d CSC/ 7 d RSC	-
2	14 d	7 d CSC/ 7 d RSC	-	x

- Amount of lags and leads for each feature (compare with Table 8)
- Training and testing period: in this study, we used identical test and training sets to make the controller more comparable
- Upper and lower thresholds for supplementary heuristic controller (see Table 4)

4.5.1. Supplementary heuristic control

To enable robust control, we complement the AMPC-based control by a supplementary, rule-based controller. This heuristic adjustment controller checks if the AMPC's output is between predefined ranges. More specifically, the controller limits the AMPC's output based on predefined upper and lower output thresholds and maximum changes between two time steps. The selection of these thresholds is based on the WBMPC's output range. Table 4 lists the resulting thresholds for the heuristic controller.

4.5.2. Application and open-loop training results

Overall, both models for the setpoint prediction is based on 35.040 training samples. The training and test period is the same to enable a fair comparison with the other methods. Consequently, an upper benchmark for the AMPC case is investigated. As described in the previous section, the selected open-source tool AddMo includes an automatic feature selection process. For imitating the controller output for $\Delta T_{\mathrm{ahu,set}}$, the tool selects 41 of a maximum of 59 features for the final model. It results in good open-loop accuracy, yielding an R^2 of 0.89 and a mean absolute error of 0.083. The best hyperparameters for the ANN are two layers with 65 and 33 neurons, respectively. The overall computation time of the training process is 149min. The open-loop training to predict $\Delta\dot{Q}_{\mathrm{cca,set}}$ results in similar high accuracies like the one for $\Delta T_{\mathrm{ahu,set}}$. For the whole year, an R^2 of 0.88 and a mean absolute error of 0.006 is realized. In contrast to the other target, of the maximum of 80 potential features, only nine are chosen by the algorithm. Among the selected features are the lag of $\Delta\dot{Q}_{\mathrm{cca,set}}$, its absolute value of the previous time step, and the room air temperature. Even though a reduced number of features is favorable from a practical point of view as the deployment is simpler and hardware interaction reduced, the closed-loop performance is not robust. The sole dependence on the manipulated variables yields a highly autocorrelated model which does not succeed in robust closed-loop control despite good open-loop performance.

In order to implement a robust closed-loop controller, we predict $\dot{Q}_{\mathrm{cca,set}}$ instead of $\Delta\dot{Q}_{\mathrm{cca,set}}$ and omit the feature selection process. We skip the feature selection for this target because even though good open-loop performance was realized by applying feature selection and consequently only using a limited amount of features, the closed-loop performance was not robust. This is why we integrate all of the possible features listed in Table 8. However, we highlight that feature selection should be an integral part of the training process as it generally simplifies the resulting model and is good practice in machine learning applications. For the prediction of $\dot{Q}_{\mathrm{cca,set}}$, however, including all potentially relevant features yielded a better closed-loop control performance than using the features chosen by automated features selection carried out with AddMo. This highlights that the conventional training process, that evaluates open-loop prediction accuracy, does not guarantee robust closed-loop performance. Overall, in the case of predicting $\dot{Q}_{\mathrm{cca,set}}$, we achieve an R^2 in open-loop prediction of 0.99 and a mean absolute error of 0.046. The number of features is 42, which is lower than the ones for predicting $\Delta T_{\mathrm{ahu,set}}$. We did not include the total number of 80 features in the training process for $\dot{Q}_{\mathrm{cca,set}}$ because the training results were poor and the computation time too high, in this case (see Table 8). The hyperparameter optimization with AddMo results in an ANN architecture of two layers with 55 and 54 neurons, respectively. The computation time for training is 107min. The shorter duration is caused by the omission of the feature selection process.

Table 4
Thresholds of supplementary heuristic controller for AMPC.

	$\dot{Q}_{\mathrm{cca,set}}$ in kW	$T_{\mathrm{ahu,set}}$ in K
Max. threshold	4.3	298.15
Min. threshold	−4.3	291.15
Max. difference	1.2	5.5

4.6. Reinforcement learning

We compare the presented MPC controllers to the model-free Reinforcement Learning Algorithm, SAC. SAC is a state-of-the-art algorithm first described in a publication in 2018 ([83]). SAC represents an evolution of the deep deterministic policy gradient (DDPG) ([104]) algorithm. Thus, SAC is also based on the actor-critic architecture and is suitable for learning continuous control problems with n-dimensional action spaces. For this purpose, the actor-critic architecture is based on several neural networks interacting. A so-called critic network learns to approximate the state-action-value function ([105]), while a so-called actor network computes the n-dimensional continuous action. The output of the critic network is used to calculate the gradients for the actor's stochastic-gradient-descent training process. Thus, the continuous actor is continuously trained by the progressively improving critic.

Fig. 4 shows a schematic representation of the interaction between the actor- and critic network and the environment. The DDPG was already equipped with some stability extensions presented in ([70]), namely target networks (which remain frozen over multiple interactions to avoid instabilities caused by oscillating policies and replay buffers serving as a static, growing training sample memory). While DDPG has long been the best-performing model-free continuous Reinforcement Learning approach, it has been outperformed by SAC in many application scenarios in recent years. SAC, unlike DDPG, optimizes a stochastic policy that allows taking uncertainties in the environment into account. The second key difference is the entropy-based exploration mechanism of SAC. While the DDPG used a defined reduction of the probability of random actions for exploration, SAC extends the objective function with an entropy term that rewards actions that explore previously unseen state-action-space regions. The increasing system

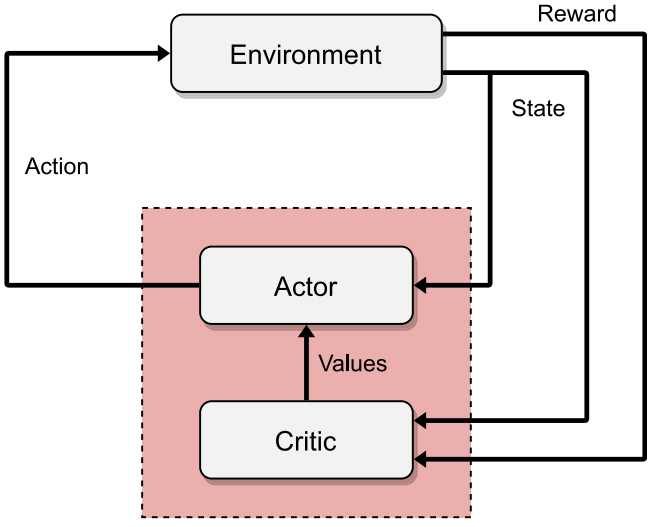

Fig. 4. Schematic structure of the actor-critic approach.

knowledge thereby is itself rewarded. This allows SAC to explore in a much more application-specific way and quickly achieve good results. Like the DDPG, the behavior of SAC is also determined by adjustable hyper-parameters, which we have determined for the use case with a Bayesian hyper-parameter optimization provided in the HyperOpt package ([106]). The hyper-parameters found are:

- Discount factor (γ) = 0.733. Determines the weighting of immediate against future rewards.
- Learning rate (α) = 0.0035. Determines the update of the ANN in each training cycle and therefore how much the policy is adjusted.
- Polyak update (τ) = 0.002. A SAC specific hyper-parameter which determines the degree of soft update of the target network.
- Batch size = 200. Determines the training samples used in each ANN training cycle.
- Buffer size = 758,600. Determines the memory capacity for past observation from the environment used training the ANN.

For stability reasons, we do not use a quadratic objective function. Instead, we use a tailored linear objective function for the SAC, formulated in Eq. 30. According to this equation, at each time step, the SAC algorithm receives a negative signal proportional to the violation of the comfort temperature range and a negative signal proportional to the thermal energy used. The algorithm's task is to find a policy that maximizes the cumulative reward signal over time by learning the relationship between the possible actions and the function.

$$reward = \begin{cases} -500 \cdot |T_{lb,rel} - T_{air,meas}| - 50 \cdot \dot{Q}_{total}, & \text{if } T_{air,meas} \leqslant T_{lb,rel} \\ -500 \cdot |T_{ub,rel} - T_{air,meas}| - 50 \cdot \dot{Q}_{total}, & \text{if } T_{air,meas} \geqslant T_{ub,rel} \\ -50 \cdot \dot{Q}_{total}, & else \end{cases} \tag{30}$$

Next, to the costs function, the tuning of the algorithm involves the feature selection for the state vector. The state vector (what the algorithm receives in each time step) is shown in Table 5 and is represented by a combination of current, historical, and predicted system variables. For the current and historical variables, the indoor air temperature and the AHU set temperature with lags of four are included. Furthermore, the AHU set temperature with it's last four values, the set temperature of the CCA with the last six values, the heating and cooling energy, each with the last three values, and the time of the day (sinoidal signal) with the last three values along with the day of the week as one integer, are included. For the predicted values, the solar radiation and the outdoor temperature are included along with the upper and lower comfort temperature bounds, with an eight-hour forecast horizon and a half-hour resolution. The state vector, therefore, has 94 entries and combines information regarding the current system dynamics and the constraints prediction.

Table 5
Features and lags considered to in the state-space of the SAC algorithm.

Feature	considered	lag	prediction
T_{air}	x	4	-
$T_{ahu,set}$	x	4	-
$\dot{Q}_{cca,set}$	x	6	-
$\dot{Q}_{heating}$	x	3	-
$\dot{Q}_{cooling}$	x	3	-
time of the day	x	3	-
day of the week	x	-	-
$T_{lb,rel}$	x	-	16
$T_{ub,rel}$	x	-	16
T_{amb}	x	-	16
$\dot{q}_{sol,dir}$	x	-	16

The possible actions are the AHU set temperature (between 18 and 25 °C) and the CCA heating or cooling energy (between −5 and 5 kW). Both the state vector and the action are normalized between −1 and 1 to be processable by the SAC algorithm.

The algorithm was trained on the system for two years before being tested for controller comparison in the third year. While severe comfort and energy efficiency limitations were observed in the first half of the first year, the interaction stabilized after that. In the second year, the RL controller almost had the same performance as in the third year.

5. Application of the implemented controllers

In the following, we evaluate the performance of the presented controllers in a one-year closed-loop simulation. The controlled system model is exported as a functional mock-up unit (FMU) and simulated in Python using the fmpy package to provide a standardized interface. For quantitatively benchmarking the performance, the KPIs energy consumption, thermal discomfort, and the computation time of the controllers are considered. Here, the thermal discomfort is expressed as an integrated violation of comfort constraints over the whole year in *Kh*.

Compared to the well-tuned RB controller, all investigated approaches perform better, leading to energy savings of 4.9% (AMPC) to 8.4% (BBMPC) and a reduction of thermal discomfort of 7.8% (AMPC) to 83.8% (BBMPC). The energy savings referred to the RB controller and thermal discomfort are displayed in Fig. 5. The black-box model predictive controller performs best in terms of energy consumption. At the same time, the online-learning BBMPC (BBMPC-OL) also outperforms the WBMPC concerning total discomfort.

These results reflect the model accuracies displayed in Table 6. For the air temperature prediction, we evaluate the one-step-ahead prediction error (k = 1) as well as the prediction error after five (k = 5), ten (k = 10), and 20 (k = 20) time steps. The accuracy is evaluated based on the controllers' forecasts over the year. Especially the nonlinear characteristics of the AHU are approximated more accurately by the BB models, leading to lower overall energy consumption. Furthermore, the GBMPC and WBMPC assume perfect subsystem controllers to reach the setpoints $T_{ahu,set}$ and $\dot{Q}_{cca,set}$. Looking at the one-step-ahead prediction error, the GBMPC profits from the initialization by the estimator. This information leads to a very low error.

Nevertheless, the prediction error increases with the prediction horizon due to the simplified model structure. The WBMPC shows

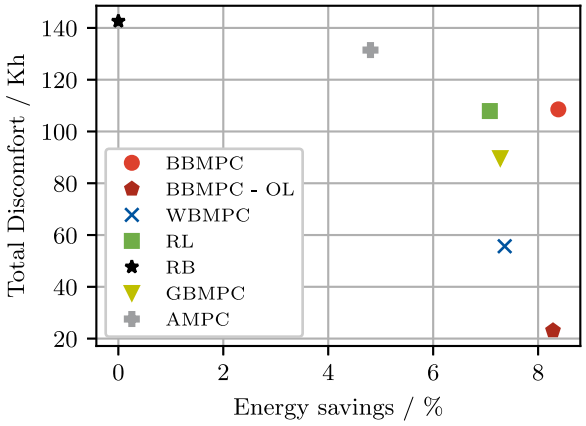

Fig. 5. Total energy consumption and discomfort of the controllers in a one-year closed loop simulation (BBMPC/ black-box MPC, BBMPC - OL/ online learning black box MPC, WBMPC/ White-box MPC, RL/ Reinforcement Learning, RB/ Rule-based controller, GBMPC/ Gray-box MPC, AMPC/ Approximate MPC).

Table 6
Accuracy of the WB-, GB-, and BB process models.

	WB	GB	BB	BB - OL
RMSE $\dot{Q}_{ahu}$	0.682 kW	0.724 kW	0.139 kW	0.125 kW
RMSE T_{air} (k = 1)	0.020 K	0.019 K	0.044 K	0.032 K
RMSE T_{air} (k = 5)	0.112 K	0.187 K	0.132 K	0.107 K
RMSE T_{air} (k = 10)	0.150 K	0.382 K	0.227 K	0.173 K
RMSE T_{air} (k = 20)	0.360 K	0.748 K	0.416 K	0.407 K

the highest accuracy with an increasing prediction horizon. In contrast, the online-learning BBMPC achieves the highest accuracy after five time steps, probably caused by the consideration of imperfect subsystem controllers. The latter allows the BBMPC-OL to react accurately in the short term, resulting in the lowest discomfort.

The model-free RL algorithm shows a comparable energy consumption to the GB- and WBMPC with a similar discomfort as the basic BBMPC. The AMPC does not approximate the WBMPC with perfect accuracy and therefore shows worse control quality but still outperforms the rule-based controller.

To evaluate the effects of different seasons, the weekly energy consumption and discomfort are plotted in Fig. 6. Compared to the RB controller, the advanced control strategies especially save energy in periods with low overall energy consumption. These are typically periods with mixed cooling and heating demand, where the advanced controllers profit from their anticipating behavior. The weekly discomfort differs strongly among the controllers. Through the adaptive behavior, the online learning BBMPC improves quickly in the first weeks and then converges to low thermal discomfort. The RL-controller also demonstrates a low discomfort, except when there is a high heating demand at the beginning and end of the year.

In contrast to the WB- and BBMPC, the RL controller penalizes comfort violations linearly. Therefore, comfort violations are more tolerated in periods with high energy demand due to the cost function. Thus, it is expected that further tuning of the RLs' cost function is likely to improve the results further. The AMPC shows few peaks in thermal discomfort, especially in cooling periods, indicating poor approximation accuracy in these periods. This poor accuracy likely results from the few cooling periods represented in the training data. In cooling periods, the adaptive GBMPC violates the comfort constraints more. In the case of the GBMPC, this behavior results from the underdetermined parameter estimation problem. During the observed discomfort peaks, the parameters computed by the MHE are close to their boundaries.

The observations based on Fig. 6 are confirmed considering the operation in the heating, cooling, and transitional period in Fig. 7. In the peak heating season, the RL algorithm violates the comfort constraints at the beginning and end of the occupancy period indicating a higher prioritization of energy savings than the other controllers. In contrast to the other controllers, the RL and the GBMPC use the AHU to cover the base load on the weekend. Notably, the WBMPC and the BBMPC show similar behavior in the CCA control but large differences in the AHU control. Therefore, the BBMPC probably approximates the nonlinear AHU behavior the most accurate.

In the presented cooling period, the GBMPC significantly undercools the system due to the inaccurate estimation of parameters. To correct this error, the controller spends additional heating energy. As seen before, the AMPC violates the comfort constraints in the cooling period, due to poor approximation accuracy.

Especially during the transition period, the advanced controllers profit from the anticipating behavior. Compared to the rule-based controller, this leads to a smoother temperature curve and energy savings.

Fig. 8 displays the computation times of the investigated control algorithms. Considering the step size of 15 min of the problem, all controllers are real-time capable with a neglectable computational burden. When looking at the computation time to calculate the control signals for one step, the simple RB controller outperforms the more complex approaches with an average time of less than 1 ms. Both AMPC and the RL-controller rely on evaluating one ANN, achieving computation times of 30 to 40 ms. Here, the most costly operation is the preparation of the inputs, which heavily relies on pandas evaluations. The SAC algorithm itself is close to the rule-based controller in processing the state signals and obtaining the actions. With a more efficient implementation, the computation time of the RL and the AMPC is expected to drop significantly. The WBMPC solves a linear optimization problem at each time step. Nevertheless, the computation time of 100 ms is comparable to the GBMPC, which solves a linear problem and a nonlinear estimation problem in 147 ms and the nonlinear BBMPC with 149 ms. This is attributable to the efficient implementation of the GB- and the BBMPC in CasADi[60], which provides are direct C++ interface to the solvers. In contrast, the WBMPC is implemented in Pyomo [97], which interfaces the solvers by writing text files. Nevertheless, for more time-critical processes, a more efficient implementation of a linear MPC can easily reach computation times below 10 ms.

6. Discussion

6.1. Performance of the algorithms

In summary, all controllers perform well on the control task, performing better than the rule-based controller. As stated in the literature (see Ref.,2.6), the different MPC controllers (WB, GB, BB) achieve better results than the RL controller on this continuous control problem. Due to its implicit learning of the underlying subsystem controllers and the more accurate approximation of the nonlinear AHU behavior, the online learning data-driven BBMPC based on neural networks shows the best comfort and energy consumption performance.

Nevertheless, there is still potential for improvement in the other MPC controllers. For example, the WBMPC and GBMPC could approximate the subsystem controllers as first-order lag elements to increase their model accuracy. Additionally, the parameter estimation of the GBMPC could be designed more robustly to avoid estimation inaccuracies. The GB-modeling accuracy could be further increased by considering the angle of the solar radiation to achieve accurate prediction in the summer.

The RL strategy could be improved further by tuning the underlying cost function and prolonged training. The latter is also expected to strengthen the AMPCs performance.

All algorithms require only a fraction of their control step size and therefore are real-time feasible. As expected, the optimization-based controllers show higher computation times. Despite the already little computational effort, all controllers could be implemented more efficiently to reduce the computational effort further. Here, the preprocessing routines of the input data could be improved. In general, the controllers should be implemented in a more lightweight programming language than python.

The algorithms need a different amount of initial data to execute the control task. While the RB controller doesn't need any

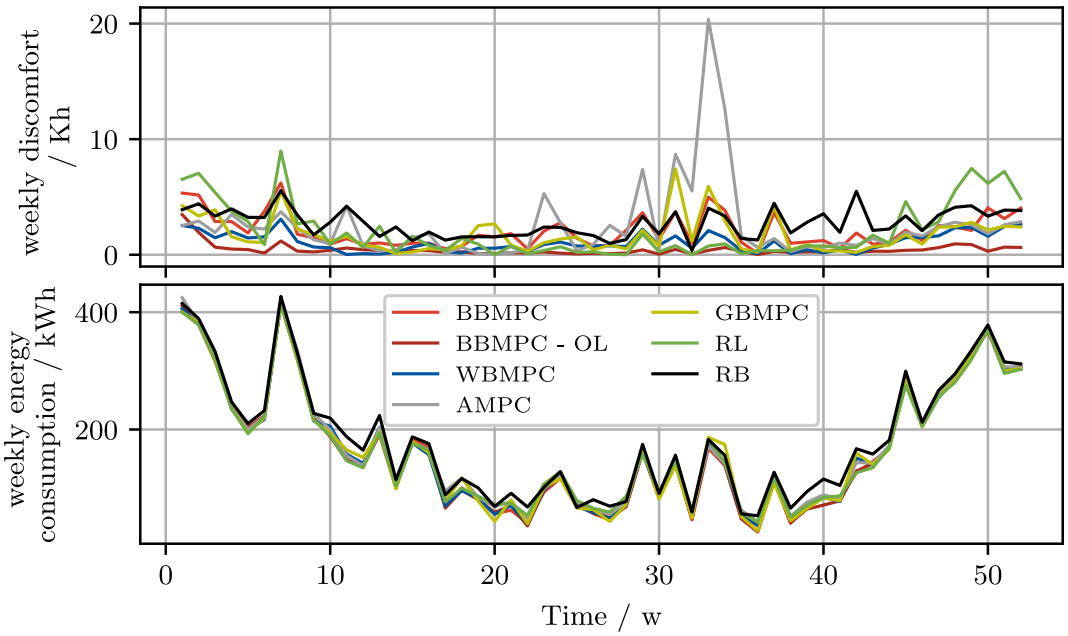

Fig. 6. Weekly discomfort and energy consumption during the simulation.

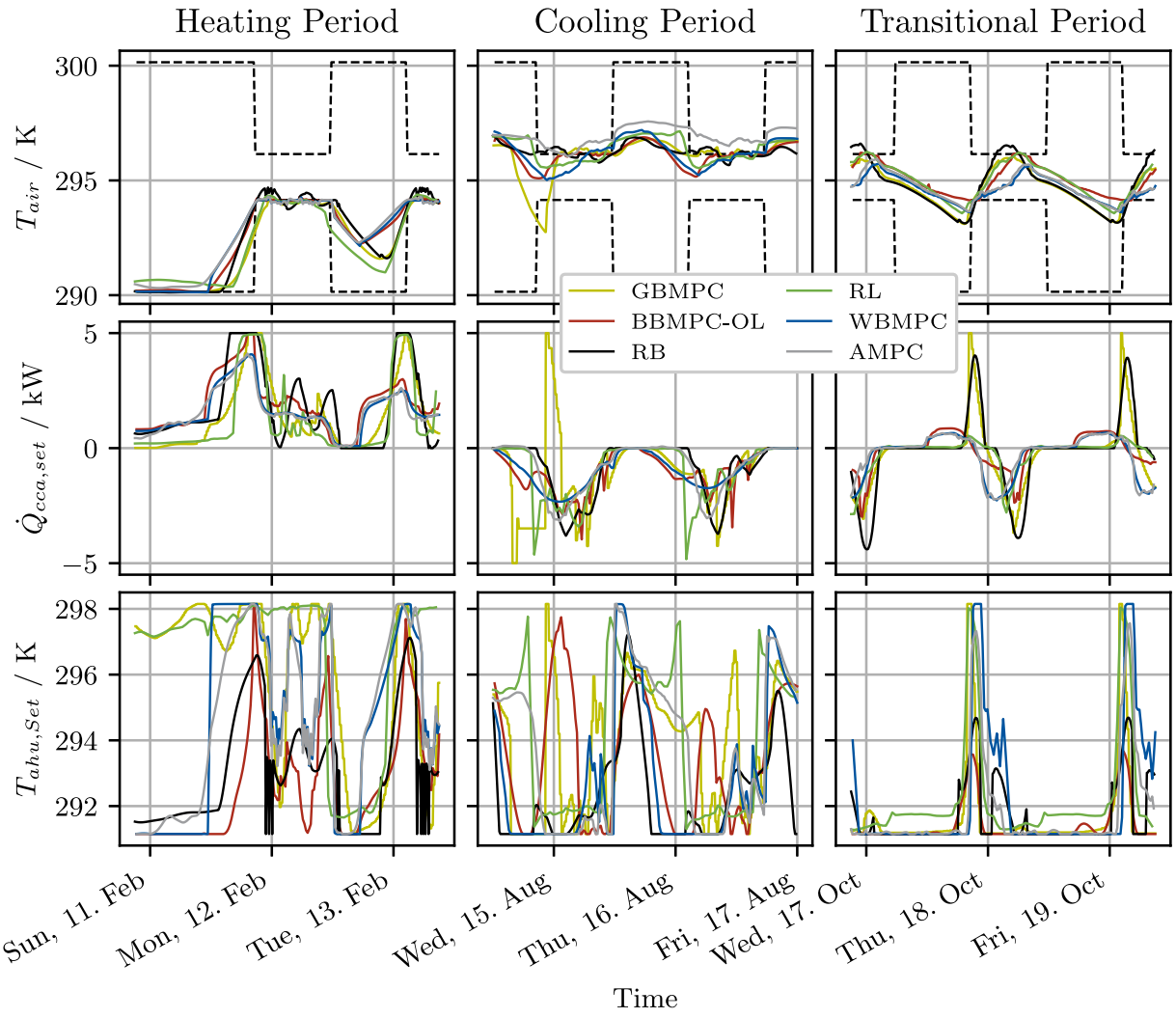

Fig. 7. Six representative days of operation in heating, cooling, and mixed operation.

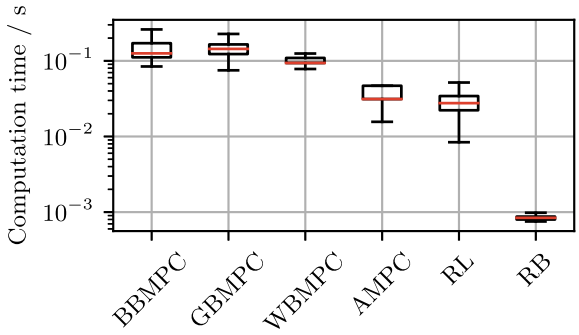

Fig. 8. Average computation time to calculate the control signals with each controller.

data, the GBMPC needs only 72 h of data for initial calibration. If all material properties are perfectly known (as is the case for our simulation), the WBMPC works without any data. In a real-life application, the WBMPC would also need data for calibration. The online learning BBMPC needs two weeks of training data. During the training, an excitation of the system is required to identify the system dynamics efficiently. Since comfort constraints are considered in the training process, only minor comfort violations occur. In a further study, the influence of training data quality and quantity should be investigated. The amount of required training data can likely be further reduced. In contrast, the approximate MPC uses one year of training data with regular MPC operation. Therefore no significant comfort violations are caused by the data generation for the approximate MPC. The RL controller is trained for two years to learn the presented control strategy. Here, serious comfort violations are observed during the first half of the first year of training.

The more extensive need of training data for the approximate MPC and the RL could be satisfied by both real measurement (e.g. approximate MPC [23], RL [107]) and simulation data (e.g. approximate MPC [24], RL [108]). Both training databases have been used in scientific literature [24,23]. Yet, we highlight at this point that one motivation to use approximate MPC and RL methods is to avoid implementing an MPC controller in practice which is why we recommend simulation-based approaches. Alternatively, the transfer to similar systems, including the choice of transferable system boundaries and the application of transfer learning (as proposed by Chen et al. [109]), are promising methodological approaches to apply simulation-based training efficiently.

These observations are based on a simulation with perfect forecasts and data availability. In the next section, we discuss aspects of the practical implementation of the controllers.

6.2. Practical applicability and implementation effort

All the advanced control methods discussed in this study show great potential to outperform conventional control strategies like RBCs (see Fig. 5). However, to become an alternative to existing methods, it is not only the control performance that needs to be considered [9]. To holistically compare the presented control methods and to consider practical applicability from model development to implementation, we introduce the following criteria and apply them to the methods implemented in this study:

1. **Pre-operation data need:** The dependence on synthetically generated (e.g., by simulations) or existing measurement data before actual deployment. The data is used for training or calibration purposes. The methods depend more or less significantly on their existence and quantity.
2. **Data quality requirements:** In addition to the needed data, the quality of the measured system states and disturbances affects the methods' performance. This category accounts for forecast uncertainty (e.g., non-perfect forecast), faulty sensors, and measurement inaccuracies and errors.
3. **Model developing effort:** This criterion accounts for the effort an engineer faces until the model performs well. This includes, e.g., domain analysis, mathematical formulation, training/calibration data generation, feature engineering and selection, hyperparameter tuning, etc. The more steps are necessary, the higher the developing effort.
4. **MIMO handling:** This study presents a MIMO problem. The methods differ in their capability to handle multiple in- and outputs and consider interdependencies.
5. **Adaptability:** Methods like the adaptive gray- and black-box MPC and the RL algorithm incorporate strategies to automatically adapt to the control domain, which can be beneficial for real-world applications and changing boundary conditions. I.e., the process model's or control model's parameters are adapted based on the model error or the system's control response.
6. **IT requirements:** Advanced control methods require different sets of sophisticated hard- and software. This criterion accounts for the necessity of, e.g., an online/cloud infrastructure, solver licenses, software, or high-performing controllers.
7. **Know-how dependence:** The applied methods require the know-how of different fields, namely machine learning, mathematical optimization, control engineering, domain knowledge, and system modeling. We define that the more fields are involved, the more demanding the need for expert know-how. This criterion is also recognized as a significant hurdle for modern control methods in practice by [24,9].
8. **Interpretability:** Another major hurdle for MPC applications recognized by [24,20,21] and RL is the missing interpretability of sophisticated control methods' underlying models, which leads to mistrust among commissioning engineers. According to Afroz et al. [21], white-box models are more interpretable, while black-box models are more easily transferable.
9. **Transferability:** High transferability facilitates the broad application of control methods as it decreases modeling development effort (see trade-off with interpretability of the previous criterion)[21].
10. **Scalability:** In this comparison, we evaluate scalability by taking a multi-zone system consisting of multiple versions of the assessed single office zone as an example. We explicitly differentiate scalability from transferability by assuming that upscaling does not involve a domain-specific change. I.e., an adaptation of model equations or input–output relations is not necessary.

Table 7 summarizes the qualitative comparison of the sophisticated and conventional control methods. We highlight at this point that the comparison is based on the way we implemented the methods. In scientific literature, various modifications for each of the presented methods exist. Consequently, the evaluation is not suitable for generalization. For example, for RL and approximate MPC, various training and tuning methods exist, which result in different model-developing efforts, higher or lower interpretability and transferability.

Regarding the need for pre-operation data, RBC and adaptive GBMPC outperform the other approaches. While RBC does not need any data before the operation, the adaptive GBMPC in this study only relies on 72h of training data for initial calibration. The BBMPC also needs a slightly higher amount of training data. In this study, we use two weeks of training. In contrast, the approximate MPC needs one year. At the same time, the RL method is based on three years of synthetically generated training data. Despite the amount

Table 7
Qualitative comparison of advanced control methods which are assessed in this study. The evaluation is based on the method implementation as realized in this study and not suitable for generalizations as different forms of method applications exists.

	RB	WB-MPC	Ad. GB-MPC	Ad. BB-MPC	Appr. WB-MPC	RL
Pre-operation data need	+	+	0 to +	0	- to 0	-
Data quality dependence	0	+	0	0 to -	+	-
Model dev. effort	+	-	0 to +	0 to +	-	0 to +
MIMO handling	-	+	+	+	0 to +	+
Adaptability	-	-	0 to +	+	-	+
IT requirements	+	-	-	-	0	0
Know-how dependence	+	-	0	0	-	0
Interpretability	+	0 to +	0 to +	0 to -	0	-
Transferability	+	-	0	+	+	+
Scalability	+	0 to +	0 to +	-	0 to +	0 to +

+ = positive, 0 = neutral, - = negative characteristic.

of synthetic or actual sensor data, data quality affects more or less significantly the methods' performance.

In general, strategies relying on an interaction with the system and not incorporating system knowledge, i.e., RL, RBC, and adaptive BBMPC, are more strongly influenced by faulty measurement data than approaches with system knowledge and/or no adaptability, i.e., WB and approximate WBMPC. However, they can adjust their parameters (of the process model (BBMPC, GBMPC) or the controller model (RL)) over time and account for prior modeling errors, changes in the target system, unseen boundary conditions or states, etc. For example, the RBC relies on a limited number of inputs. Consequently, measurement errors within these inputs have a stronger effect on the controller's output compared to, e.g., the WBMPC, using a detailed system or process model. In a real-life application, the latter requires a state estimator, like a Kalman Filter or an MHE, to estimate unmeasured states, disturbances, and modeling errors. Hence, one faulty sensor does not affect the control performance as significantly.

Apart from that, self-adaptation, or online learning, and the reliance on black-box instead of white-box modeling generally facilitate model development. RBC is only subject to tuning for actual operation and does not necessarily rely on prior model development, even though studies dealing with model-based RBC development have already been successfully presented. The adaptive BB and GBMPC, as well as the RL method, all include self-adaptation to the control domain. Thus, the model development effort is reduced to calibration and hyperparameter tuning. The same applies to the approximate MPC approach when considering the method itself. However, as it bases on the optimal operation of the WBMPC controller, which has a high model development effort, the overall effort is high.

Regarding handling MIMO problems like the one presented in this study, sophisticated control methods dominate the traditional RBC. Only the approximate WBMPC might have difficulties, including the interactions between manipulated variables. Still, they can be included in the training process. A significant advantage of statistical models is the ability to adapt automatically to the control domain. In this work, the adaptive GB and BB and the RL method exploit this advantage. This is especially important if boundary conditions or system characteristics change over time. Still, it comes at the expense of higher IT requirements.

Considering the IT requirements, the conventional RBC performs best, followed by the approximate WBMPC. While the adaptive GB and BBMPC both require a stable data infrastructure, advanced computer hardware as well as solvers with (often) commercial licenses, the approximate MPC and the RL can potentially be more easily deployed on low-level controllers using simplified software (also see Fig. 8). In our use case, both RL and AMPC also rely on forecasts of the disturbances, making an online data infrastructure for forecast incorporation crucial. This limits the theoretical advantage of a more simple infrastructure. Future studies should evaluate the trade-off between a simpler offline data infrastructure and the potential performance loss.

As described above, we evaluate the dependence on expert know-how based on the number of areas covered by the methodology. Here, RBC outperforms the sophisticated control methods. As methods like the adaptive GB-, BBMPC, and RL omit or tremendously reduce the effort of generating a process model by experts, they are categorized with a "0". If a modularized, software-framework for the controller generation is available, the effort could be reduced further. All areas mentioned above need to be considered for the WB-based approaches, which is why they are rated with "-". Furthermore, missing interpretability or comprehensibility has been detected as a major hurdle for MPC applications in practice [24,20,21].

However, we further differentiate the different approaches as the underlying model structure or methodological setup limits or supports interpretability. RL results in the biggest ANN (2 layers of 64 neurons) and is solely based on a BB model. Theoretically, the decision policy could be visualized, but it involves comparatively high effort [110]. The adaptive BBMPC is based on online learning and a BB model. Even though the underlying ANN is much smaller than the RL one, experts need to carefully analyze whether the decision is based on a faulty adapted process model or the optimization logic, resulting in a rather complex task. For the AMPC, feature selection which forms the basis of interpretability, is a challenge due to the missing link between open- and closed-loop performance. So even if the machine learning algorithm finds strong correlations and thus interpretability between the features and the control variables, it does not necessarily lead to good control performance. Even though the approximate WBMPC relies on a WB approach which is more comprehensive than a BB one, the control logic is learned by an ANN of a medium complex structure, limiting interpretability. Yet, we like to highlight at this point that research in AMPC partially focuses on the interpretability of the resulting BB model. In this case, however, tree-based BB models are used [24]. Apart from this, while the RBC is characterized by high interpretability, transferability to, e.g., a similar system is trickier due to a higher tuning effort. Here, the self-adapting BB-based methods are favorable. The WBMPC involves a unique detailed process model which limits the transferability to other systems. Finally, the conventional RBC is most suitable when it comes to scalability to a multi-zone approach with equal zones as it just involves re-tuning, if any. The remaining approaches are all evaluated similarly except for the adaptive BBMPC. The former all require, if any, more complex parameter adaptions (WBMPC, adaptive GBMPC) or retraining (RL, approximate WBMPC), but the methods are scalable. Regarding the BBMPC, Bünning et al. [59] highlight infeasibilities of ANN-based optimization for non-convex problems limiting the scalability.

7. Conclusion

This paper presents a holistic quantitative and qualitative comparison of popular advanced control methods. Based on the state of research (see Section 2), we included a white-box (Subsection 4.2), an adaptive gray-box (Subsection 4.3), an adaptive black-box (Subsection 4.4) and an approximate white-box MPC (Subsection 4.5) as well as a reinforcement learning-based controller (Subsection 4.6) in a comparative analysis. The advanced control methods are benchmarked with a rule-based controller representing a sophisticated state-of-the-art controller (Subsection 4.1). The rule-based controller is well-tuned and is designed to exploit the system's flexibilities, like the more advanced controllers.

Consequently, we provide a fair benchmark and do not degrade the conventional benchmark. We applied all methods to a Modelica-based simulation model of a single-zone office (Se ction 3) based on ASHRAE 140 and 900 and evaluate them based on the resulting annual discomfort and energy consumption. The office's energy system comprises an air handling unit and concrete core activation. As control outputs, the air handling unit's set temperature, and the concrete core activation's set heat flow are investigated, resulting in a multi–input–multi–output problem. In addition to the rather quantitative key performance indicators (see Subsection 6.1), we also compare the methods based on soft criteria (see Subsection 6.2). The results indicate that all advanced control methods outperform the conventional rule-based controller regarding discomfort and energy consumption. Among the advanced controllers, the adaptive black-box MPC approach performs best. Compared to the well-tuned, rule-based controller, energy savings between 4.9% (approximate white-box MPC) and 8.4% (adaptive black-box MPC) are realized (see Fig. 5).

Furthermore, we observe a reduction in thermal discomfort between a minimum of 7.8% (approximate white-box MPC) and 83.8% (adaptive black-box MPC). The performance of the non-adaptive black-box MPC, the reinforcement learning-based controller, and the adaptive gray-box MPC is similar. The white-box MPC yields the second lowest discomfort and energy consumption. At the same time, the approximate white-box MPC results in the highest discomfort and energy consumption among the advanced control methods. We highlight, at this point, that the performance of all controllers could be enhanced, which could potentially lead to a shift in results. While the white-box MPC could also incorporate the local controllers, the other methods (adaptive black- and gray-box and approximate white-box MPC as well as reinforcement learning) could benefit from an adaption in the training and tuning process. Considering the average computation time for a single control step, the conventional rule-based controller clearly outperforms the advanced methods (see Fig. 8). While the reinforcement-learning based and the approximate MPC controller yield medium computation times, the other MPC approaches all result in higher computation times. We highlight, at this point, that all controllers are real-time applicable as the longest control step computation time is below 0.3s. Considering a control sampling time of 15min, the real-time applicability is proven. To evaluate the methods' development to deployment cycle, we included additional characteristics upon which the methods are compared (Table 7). Here, we detect a trend that the methods that greatly depend on expert knowledge (especially white-box MPC and reinforcement learning) and have a high need for advanced hard- and software infrastructure (especially adaptive gray- and black-box MPC) result in lower energy consumption and discomfort. Therefore, the higher development effort is rewarded. Even though the rule-based controller results in the highest discomfort and energy consumption, it is favorable from a modeling development, IT infrastructure, interpretability, and expert knowledge dependence point of view. The approximate MPC approach is a compromise between a simple low-tech controller which can be deployed without extensive expert know-how but does not exploit a system's full performance potential and the more sophisticated white-box, adaptive gray-box, and (adaptive) black-box MPC-based as well as the reinforcement-learning-based controllers. However, the approximate MPC relies on prior model development to develop, as in our case, the white-box MPC. For future studies, we recommend applying the comparative analysis of the advanced control methods to other systems. As a use case, we used a single-zone office whose heating and cooling demand is covered by two systems: an air handling unit and a concrete core activation, respectively. As this represents a multi-output problem with systems of different inertia, we recommend future work to apply the presented methodology to a single-output system. Especially for the rule-based, reinforcement learning and approximate MPC controller, we expect a better performance due to simplified tuning. Apart from that, we suggest adopting the approach and applying it to a more standardized use case. Here, we see great potential in utilizing the BOPTEST framework as it provides standardized benchmarks and performance measures [90]. In addition, we propose to apply all methods to a real building based on actual measurement data and the inherent measurement errors. Plus, the effect of forecast uncertainty using real-life forecasts and actual disturbance variables on all methods should be investigated.

CRediT authorship contribution statement

Phillip Stoffel: Conceptualization, Methodology, Validation, Software, Writing - original draft, Investigation, Visualization, Formal analysis. **Laura Maier:** Conceptualization, Methodology, Validation, Software, Writing - original draft, Investigation. **Alexander Kümpel:** Conceptualization, Methodology, Validation, Software, Writing - original draft, Investigation. **Thomas Schreiber:** Conceptualization, Methodology, Validation, Software, Writing - original draft, Investigation. **Dirk Müller:** Conceptualization, Writing - review & editing, Supervision, Funding acquisition.

Data availability

Data will be made available on request.

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.

Acknowledgements

We would like to thank Steffen Eser for carefully proofreading the manuscript and valuable feedback.

We gratefully acknowledge the financial support by the Federal Ministry for Economic Affairs and Climate Action (BMWK), promotional references: 03EN3026C, 03ETW006A, and 03SBE0006A.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 101023666.

Appendix A. Feature selection results Approximate MPC

Table 8
Features included in training process and finally selected ones for approximate MPC model.

			$\Delta T_{AHU,set}$			$\dot{Q}_{CCA,set}$		
Features	provided		selec.	lag	lead	selec.	lag	lead
State variables	lag	lead						
$T_{air,in}$	1–4			2,4		x	1–4	
$T_{air,sup}$	1–4		x	1–4		x	1–4	
$T_{air,zone}$	1–4		x	1–4		x	1–4	
$\dot{Q}_{cold}$	1–4		x	1		x	1–4	
$\dot{Q}_{heat}$	1–4		x	1,4		x	1–4	
$\dot{Q}_{hydr}$	1–4		x	1–4		x	1–4	
Disturbance variables								
schedule light						x		
schedule devices			x			x		
schedule human			x			x		
$T_{ambient}$	1–4	1–4		4	1–4	x	1–4	
$\dot{Q}_{glo,hor}$			x			x		
rel. humidity			x			x		
sol. altitude angle	1–4	2–4			2–4	x	1–4	
sol. declination angle			x			x		
sol. hourly angle			x			x		
sol. time						x		
sol. zenith angle			x			x		
wind direction			x			x		
wind speed						x		
Synthetic features								
time of day			x			x		
weekday			x			x		
Controller specific								
$T_{air,ub}$			x			x		
$T_{air,lb}$						x		

References

[1] Working Group II, Climate change 2022: Impacts, adaptation and vulnerability: Summary for policymakers: Contribution to the sixth assessment report of the ipcc. URL: https://report.ipcc.ch/ar6wg2/pdf/IPCC_AR6_WGII_SummaryForPolicymakers.pdf.

[2] R. De Coninck, L. Helsen, Practical implementation and evaluation of model predictive control for an office building in Brussels, Energy and Buildings 111 (2016) 290–298, https://doi.org/10.1016/j.enbuild.2015.11.014, URL: https://linkinghub.elsevier.com/retrieve/pii/S0378778815303790.

[3] J. Drgoňa, D. Picard, L. Helsen, Cloud-based implementation of white-box model predictive control for a GEOTABS office building: A field test demonstration, Journal of Process Control 88 (2020) 63–77, https://doi.org/10.1016/j.jprocont.2020.02.007, URL: https://linkinghub.elsevier.com/retrieve/pii/S0959152419306857.

[4] D. Sturzenegger, D. Gyalistras, M. Morari, R.S. Smith, Model Predictive Climate Control of a Swiss Office Building: Implementation, Results, and Cost-Benefit Analysis, IEEE Transactions on Control Systems Technology 24 (1) (2016) 1–12, https://doi.org/10.1109/TCST.2015.2415411, URL: http://ieeexplore.ieee.org/document/7087366/.

[5] S. Freund, G. Schmitz, Implementation of model predictive control in a large-sized, low-energy office building, Building and Environment 197 (2021), https://doi.org/10.1016/j.buildenv.2021.107830, URL: https://linkinghub.elsevier.com/retrieve/pii/S0360132321002365.

[6] S. Sayadi, G. Tsatsaronis, T. Morozyuk, Reducing the Energy Consumption of HVAC Systems in Buildings by Using Model Predictive Control 13.

[7] D. Lindelöf, H. Afshari, M. Alisafaee, J. Biswas, M. Caban, X. Mocellin, J. Viaene, Field tests of an adaptive, model-predictive heating controller for residential buildings, Energy and Buildings 99 (2015) 292–302, https://doi.org/10.1016/j.enbuild.2015.04.029, URL: https://linkinghub.elsevier.com/retrieve/pii/S037877881500328X.

[8] G. Serale, M. Fiorentini, A. Capozzoli, D. Bernardini, A. Bemporad, Model Predictive Control (MPC) for Enhancing Building and HVAC System Energy Efficiency: Problem Formulation, Applications and Opportunities, Energies 11 (3) (2018) 631, https://doi.org/10.3390/en11030631, URL: http://www.mdpi.com/1996-1073/11/3/631.

[9] M. Killian, M. Kozek, Ten questions concerning model predictive control for energy efficient buildings, Building and Environment 105 (2016) 403–412, https://doi.org/10.1016/j.buildenv.2016.05.034.

[10] J. Cigler, D. Gyalistras, J. Široký, V. Tiet, L. Ferkl, Beyond theory: the challenge of implementing model predictive control in buildings, in: Proceedings of 11th Rehva World Congress, Clima. URL: https://opticontrol.ee.ethz.ch/Lit/Cigl_13_Proc-Clima2013.pdf.

[11] J. Woo, A.E. Fenner, A. Asutosh, D.-S. Kim, M. Razkenari, C. Kibert, A review of the state-of-the-art machine learning algorithms for building energy consumption prediction, 2018.

[12] A. Afram, F. Janabi-Sharifi, Review of modeling methods for HVAC systems, Applied Thermal Engineering 67 (1–2) (2014) 507–519, https://doi.org/10.1016/j. applthermaleng.2014.03.055.

[13] J. Drgoňa, J. Arroyo, I. Cupeiro Figueroa, D. Blum, K. Arendt, D. Kim, E.P. Ollé, J. Oravec, M. Wetter, D.L. Vrabie, L. Helsen, All you need to know about model predictive control for buildings, Annual Reviews in Control 50 (2020) 190–232, https://doi.org/10.1016/j.arcontrol.2020.09.001, URL: https://linkinghub.elsevier.com/retrieve/pii/S1367578820300584.

[14] F. Jorissen, W. Boydens, L. Helsen, TACO, an automated toolchain for model predictive control of building systems: implementation and verification, Journal of Building Performance Simulation 12 (2) (2019) 180–192, https://doi.org/10.1080/19401493.2018.1498537, URL: https://www.tandfonline.com/doi/full/10.1080/19401493.2018.1498537.

[15] A. Kathirgamanathan, M. De Rosa, E. Mangina, D.P. Finn, Data-driven Predictive Control for Unlocking Building Energy Flexibility: A Review, Renewable and Sustainable Energy Reviews 135 (2021), arXiv: 2007.14866. doi:10.1016/j. rser.2020.110120. URL: http://arxiv.org/abs/2007.14866.

[16] F. Bünning, B. Huber, P. Heer, A. Aboudonia, J. Lygeros, Experimental demonstration of data predictive control for energy optimization and thermal comfort in buildings, Energy and Buildings 211 (2020), https://doi.org/10.1016/j.enbuild.2020.109792, URL: https://linkinghub.elsevier.com/retrieve/pii/S0378778819320638.

[17] S. Yang, M.P. Wan, W. Chen, B.F. Ng, S. Dubey, Model predictive control with adaptive machine-learning-based model for building energy efficiency and comfort optimization, Applied Energy 271 (2020), https://doi.org/10.1016/j.apenergy.2020.115147, URL: https://linkinghub.elsevier.com/retrieve/pii/S0306261920306590.

[18] A. Jain, T. Nghiem, M. Morari, R. Mangharam, Learning and Control Using Gaussian Processes, in: 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS), IEEE, Porto, 2018, pp. 140–149, https://doi.org/10.1109/ICCPS.2018.00022.

[19] K. Arendt, M. Jradi, H.R. Shaker, C. Veje, Comparative Analysis of White-, Gray- and Black-box Models for Thermal Simulation of Indoor Environment: Teaching Building Case Study, in: Proceedings of the 2018 Building Performance Modeling Conference and SimBuild co-organized by ASHRAE and IBPSA-USA, ASHRAE, 2018, pp. 173–180.

[20] J. Drgoňa, J. Arroyo, I. Cupeiro Figueroa, D. Blum, K. Arendt, D. Kim, E.P. Ollé, J. Oravec, M. Wetter, D.L. Vrabie, L. Helsen, All you need to know about model predictive control for buildings, Annual Reviews in Control 50 (2020) 190–232, https://doi.org/10.1016/j.arcontrol.2020.09.001.

[21] Z. Afroz, G.M. Shafiullah, T. Urmee, G. Higgins, Modeling techniques used in building hvac control systems: A review, Renewable and Sustainable Energy Reviews 83 (2018) 64–84, https://doi.org/10.1016/j. rser.2017.10.044.

[22] P. May-Ostendorp, G.P. Henze, B. Rajagopalan, D. Kalz, Experimental investigation of model predictive control-based rules for a radiantly cooled office, in: HVAC and R Research, Vol. 19, pp. 602–615. doi:10.1080/10789669.2013.801303. URL: https://www.scopus.com/record/display.uri?eid=2-s2.0-84881269541&origin=inward.

[23] S. Yang, M.P. Wan, W. Chen, B.F. Ng, S. Dubey, Experiment study of machine-learning-based approximate model predictive control for energy-efficient building control, Applied Energy 288 (2021), https://doi.org/10.1016/j.apenergy.2021.116648, URL: https://linkinghub.elsevier.com/retrieve/pii/S0306261921001811.

[24] J. Drgoňa, D. Picard, M. Kvasnica, L. Helsen, Approximate model predictive building control via machine learning, Applied Energy 218 (2018) 199–216, https://doi.org/10.1016/j.apenergy.2018.02.156.

[25] Z. Wang, T. Hong, Reinforcement learning for building controls: The opportunities and challenges, Applied Energy 269 (2020), https://doi.org/10.1016/j.apenergy.2020.115036, URL: https://www.sciencedirect.com/science/article/pii/S0306261920305481.

[26] R.S. Sutton, A.G. Barto, Reinforcement learning: An introduction, MIT press, 2018.

[27] M.D.M. Castilla, J. Álvarez, F. Rodriguez, M. Berenguel, Comfort Control, Buildings (2014), https://doi.org/10.1007/978-1-4471-6347-3.

[28] M. Castilla, J.D. Álvarez, J.E. Normey-Rico, F. Rodríguez, Thermal comfort control using a non-linear MPC strategy: A real case of study in a bioclimatic building, Journal of Process Control 24 (6) (2014) 703–713, https://doi.org/10.1016/j.jprocont.2013.08.009.

[29] S. Zhan, A. Chong, Data requirements and performance evaluation of model predictive control in buildings: A modeling perspective, Renewable and Sustainable Energy Reviews 142 (2021), https://doi.org/10.1016/j.rser.2021.110835.

[30] M. Wetter, W. Zuo, T.S. Nouidui, X. Pang, Modelica Buildings library, Journal of Building Performance Simulation 7 (4) (2014) 253–270, publisher: Taylor & Francis _eprint: https://doi.org/10.1080/19401493.2013.765506. doi:10.1080/19401493.2013.765506. URL: https://doi.org/10.1080/19401493.2013.765506

[31] D. Müller, M.R. Lauster, A. Constantin, M. Fuchs, P. Remmen, AixLib - An Open-Source Modelica Library within the IEA-EBC Annex60 Framework, in: BauSim, Fraunhofer IRB Verlag, Stuttgart, 2016, pp. 3–9. URL: https://publications.rwth-aachen.de/record/681852.

[32] M. Mork, A. Xhonneux, D. Müller, Nonlinear Distributed Model Predictive Control for multi-zone building energy systems, Energy and Buildings 264 (2022), https://doi.org/10.1016/j.enbuild.2022.112066, URL: https://linkinghub.elsevier.com/retrieve/pii/S0378778822002377.

[33] D. Sturzenegger, D. Gyalistras, V. Semeraro, M. Morari, R.S. Smith, BRCM Matlab Toolbox: Model generation for model predictive building control, in: 2014 American Control Conference, 2014, pp. 1063–1069, iSSN: 2378–5861. doi:10.1109/ACC.2014.6858967.

[34] P. Rockett, E.A. Hathway, Model-predictive control for non-domestic buildings: a critical review and prospects, Building Research & Information 45 (5) (2017) 556–571, https://doi.org/10.1080/09613218.2016.1139885, publisher: Routledge _eprint:URL: https://doi.org/10.1080/09613218.2016.1139885.

[35] M. Gholamzadehmir, C. Del Pero, S. Buffa, R. Fedrizzi, N. Aste, Adaptive-predictive control strategy for hvac systems in smart buildings – a review, Sustainable Cities and Society 63 (2020), https://doi.org/10.1016/j.scs.2020.102480.

[36] S.J. Qin, T.A. Badgwell, A survey of industrial model predictive control technology, Control engineering practice 11 (7) (2003) 733–764.

[37] M. Benosman, Model-based vs data-driven adaptive control: An overview, International Journal of Adaptive Control and Signal Processing 32 (5) (2018) 753–776, https://doi.org/10.1002/acs.2862.

[38] R. Lv, Z. Yuan, B. Lei, J. Zheng, X. Luo, Model predictive control with adaptive building model for heating using the hybrid air-conditioning system in a railway station, Energies 14 (7). doi:10.3390/en14071996.

[39] T. Zeng, P. Barooah, An adaptive model predictive control scheme for energy-efficient control of building hvac systems, ASME Journal of Engineering for Sustainable Buildings and Cities 2 (3). doi:10.1115/1.4051482.

[40] J.B. Rawlings, D.Q. Mayne, Model predictive control: Theory and design, Nob Hill Pub, 2009.

[41] S.J. Julier, J.K. Uhlmann, Unscented filtering and nonlinear estimation, Proceedings of the IEEE 92 (3) (2004) 401–422, https://doi.org/10.1109/JPROC.2003.823141.

[42] R. Alexander, G. Campani, S. Dinh, F.V. Lima, Challenges and opportunities on nonlinear state estimation of chemical and biochemical processes, Processes 8 (11) (2020) 1462, https://doi.org/10.3390/pr8111462.

[43] S.F. Fux, A. Ashouri, M.J. Benz, L. Guzzella, Ekf based self-adaptive thermal model for a passive house, Energy and Buildings 68 (2014) 811–817, https://doi.org/10.1016/j.enbuild.2012.06.016.

[44] M. Maasoumy, M. Razmara, M. Shahbakhti, A.S. Vincentelli, Handling model uncertainty in model predictive control for energy efficient buildings, Energy and Buildings 77 (2014) 377–392, https://doi.org/10.1016/j.enbuild.2014.03.057.

[45] P. Kühl, M. Diehl, T. Kraus, J.P. Schlöder, H.G. Bock, A real-time algorithm for moving horizon state and parameter estimation, Computers & Chemical Engineering 35 (1) (2011) 71–83, https://doi.org/10.1016/j.compchemeng.2010.07.012.

[46] A. Kümpel, P. Stoffel, D. Müller, Self-adjusting model predictive control for modular subsystems in hvac systems, Journal of Physics: Conference Series 2042 (1) (2021), https://doi.org/10.1088/1742-6596/2042/1/012037.

[47] P. Stoffel, A. Kümpel, D. Müller, Cloud-based optimal control of individual borehole heat exchangers in a geothermal field, Journal of Thermal Science doi:10.1007/s11630-022-1639-0.

[48] M.D. Knudsen, L. Georges, K.S. Skeie, S. Petersen, Experimental test of a black-box economic model predictive control for residential space heating, Applied Energy 298 (2021), https://doi.org/10.1016/j.apenergy.2021.117227, URL: https://www.sciencedirect.com/science/article/pii/S0306261921006498.

[49] T.X. Nghiem, C.N. Jones, Data-driven demand response modeling and control of buildings with Gaussian Processes, in: 2017 American Control Conference (ACC), IEEE, Seattle, WA, USA, 2017, pp. 2919–2924. doi:10.23919/ACC.2017.7963394. URL: http://ieeexplore.ieee.org/document/7963394/

[50] E. Maddalena, S. Muller, R. Santos, C. Salzmann, C. Jones, Experimental Data-Driven Model Predictive Control of a Hospital HVAC System During Regular Use, 2021.

[51] F. Smarra, A. Jain, T. de Rubeis, D. Ambrosini, A. D'Innocenzo, R. Mangharam, Data-driven model predictive control using random forests for building energy optimization and climate control, Applied Energy 226 (2018) 1252–1272, https://doi.org/10.1016/j.apenergy.2018.02.126, URL: https://linkinghub.elsevier.com/retrieve/pii/S0306261918302575.

[52] A. Jain, M. Behl, R. Mangharam, Data Predictive Control for building energy management, in: Proceedings of the 2017 American Control Conference, IEEE, 2017.

[53] F. Bünning, A. Schalbetter, A. Aboudonia, M.H. de Badyn, P. Heer, J. Lygeros, Input Convex Neural Networks for Building MPC, arXiv:2011.13227 [cs, eess] ArXiv: 2011.13227. URL: http://arxiv.org/abs/2011.13227.

[54] A. Jain, F. Smarra, E. Reticcioli, A. D'Innocenzo, M. Morari, NeurOpt: Neural network based optimization for building energy management and climate control, arXiv:2001.07831 [cs, eess]ArXiv: 2001.07831. URL: http://arxiv.org/abs/2001.07831.

[55] A. Afram, F. Janabi-Sharifi, Black-box modeling of residential HVAC system and comparison of gray-box and black-box modeling methods, Energy and Buildings 94 (2015) 121–149, https://doi.org/10.1016/j.enbuild.2015.02.045, URL: https://linkinghub.elsevier.com/retrieve/pii/S0378778815001504.

[56] S. Yang, M.P. Wan, W. Chen, B.F. Ng, S. Dubey, Experiment study of machine-learning-based approximate model predictive control for energy-efficient building control, Applied Energy 288 (2021), https://doi.org/10.1016/j.apenergy.2021.116648.

[57] B. Amos, L. Xu, J.Z. Kolter, Input Convex Neural Networks, arXiv:1609.07152 [cs, math]ArXiv: 1609.07152. URL: http://arxiv.org/abs/1609.07152.

[58] A. Wächter, L.T. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Mathematical Programming 106 (1) (2006) 25–57, https://doi.org/10.1007/s10107-004-0559-y, URL: http://link.springer.com/10.1007/s10107-004-0559-y.

[59] F. Bünning, B. Huber, A. Schalbetter, A. Aboudonia, M. Hudoba de Badyn, P. Heer, R.S. Smith, J. Lygeros, Physics-informed linear regression is competitive with two Machine Learning methods in residential building MPC, Applied Energy 310 (2022), https://doi.org/10.1016/j.apenergy.2021.118491, URL: https://www.sciencedirect.com/science/article/pii/S0306261921017098.

[60] J.A.E. Andersson, J. Gillis, G. Horn, J.B. Rawlings, M. Diehl, CasADi: a software framework for nonlinear optimization and optimal control, Mathematical Programming Computation 11 (1) (2019) 1–36, https://doi.org/10.1007/s12532-018-0139-4, URL: http://link.springer.com/10.1007/s12532-018-0139-4.

[61] P. May-Ostendorp, G.P. Henze, C.D. Corbin, B. Rajagopalan, C. Felsmann, Model-predictive control of mixed-mode buildings with rule extraction, Building and Environment 46 (2) (2011) 428–437, https://doi.org/10.1016/j.buildenv.2010.08.004.

[62] P.T. May-Ostendorp, G.P. Henze, B. Rajagopalan, C.D. Corbin, Extraction of supervisory building control rules from model predictive control of windows in a mixed mode building, Journal of Building Performance Simulation 6 (3) (2013) 199–219, https://doi.org/10.1080/19401493.2012.665481.

[63] F.T. Bessler, D.A. Savic, G.A. Walters, Journal of Water Resources Planning and Management 129 (1) (2003) 26–34, https://doi.org/10.1061/(ASCE)0733-9496(2003)129:1(26), URL: https://ascelibrary.org/doi/pdf/10.1061/.

[64] C.-C. Wei, N.-S. Hsu, Optimal tree-based release rules for real-time flood control operations on a multipurpose multireservoir system, Journal of Hydrology 365 (3–4) (2009) 213–224, https://doi.org/10.1016/j.jhydrol.2008.11.038.

[65] L.M. Maier, S. Henn, P. Mehrfeld, D. Müller, Approximate optimal control for heat pumps in building energy systems. doi:10.18154/RWTH-2021-07442.

[66] A. Domahidi, F. Ullmann, M. Morari, C.N. Jones, Learning decision rules for energy efficient building control, Journal of Process Control 24 (6) (2014) 763–772, https://doi.org/10.1016/j.jprocont.2014.01.006, URL: https://www.sciencedirect.com/science/article/pii/S0959152414000304.

[67] K. Le, R. Bourdais, H. Guéguen, From hybrid model predictive control to logical control for shading system: A support vector machine approach, Energy and Buildings 84 (2014) 352–359, https://doi.org/10.1016/j.enbuild.2014.07.084.
[68] M. Klaučo, J. Drgoňa, M. Kvasnica, S. Di Cairano, Building temperature control by simple mpc-like feedback laws learned from closed-loop data, IFAC Proceedings Volumes 47 (3) (2014) 581–586, https://doi.org/10.3182/20140824-6-ZA-1003.01633.
[69] E. Žáčeková, M. Pčolka, J. Tabačk, J. Těžký, R. Robinett, S. Čelikovský, M. Šebek, Identification and energy efficient control for a building: Getting inspired by mpc, in: 2015 American Control Conference (ACC), IEEE, 01.07.2015 - 03.07.2015, pp. 1671–1676. doi:10.1109/ACC.2015.7170973.
[70] V. Mnih, K. Kavukcuoglu, D. Silver, et al., Human-level control through deep reinforcement learning, Nature 518 (7540) (2015) 529–533, https://doi.org/10.1038/nature14236.
[71] Z. Jiang, M.J. Risbeck, V. Ramamurti, S. Murugesan, J. Amores, C. Zhang, Y.M. Lee, K.H. Drees, Building hvac control with reinforcement learning for reduction of energy cost and demand charge, Energy and Buildings 239 (2021), https://doi.org/10.1016/j.enbuild.2021.110833, URL: https://www.sciencedirect.com/science/article/pii/S0378778821001171.
[72] S. Brandi, M.S. Piscitelli, M. Martellacci, A. Capozzoli, Deep reinforcement learning to optimise indoor temperature control and heating energy consumption in buildings, Energy and Buildings 224 (2020).
[73] A. Mathew, M.J. Jolly, J. Mathew, Improved residential energy management system using priority double deep q-learning, Sustainable Cities and Society 69 (2021), https://doi.org/10.1016/j.scs.2021.102812, URL: https://www.sciencedirect.com/science/article/pii/S2210670721001037.
[74] T. Yang, L. Zhao, W. Li, J. Wu, A.Y. Zomaya, Towards healthy and cost-effective indoor environment management in smart homes: A deep reinforcement learning approach, Applied Energy 300 (2021), https://doi.org/10.1016/j.apenergy.2021.117335, URL: https://www.sciencedirect.com/science/article/pii/S0306261921007431.
[75] M. Biemann, F. Scheller, X. Liu, L. Huang, Experimental evaluation of model-free reinforcement learning algorithms for continuous hvac control, Applied Energy 298 (2021).
[76] A. Kathirgamanathan, E. Mangina, D.P. Finn, Development of a soft actor critic deep reinforcement learning approach for harnessing energy flexibility in a large office building, Energy and AI 5 (2021), https://doi.org/10.1016/j.egyai.2021.100101, URL: https://www.sciencedirect.com/science/article/pii/S2666546821000537.
[77] Z. Zou, X. Yu, S. Ergan, Towards optimal control of air handling units using deep reinforcement learning and recurrent neural network, Building and Environment 168 (2020), https://doi.org/10.1016/j.buildenv.2019.106535, URL: https://www.sciencedirect.com/science/article/pii/S0360132319307474.
[78] G. Pinto, M.S. Piscitelli, J.R. Vázquez-Canteli, Z. Nagy, A. Capozzoli, Coordinated energy management for a cluster of buildings through deep reinforcement learning, Energy 229 (2021), https://doi.org/10.1016/j.energy.2021.120725, URL: https://www.sciencedirect.com/science/article/pii/S0360544221009737.
[79] G. Pinto, D. Deltetto, A. Capozzoli, Data-driven district energy management with surrogate models and deep reinforcement learning, Applied Energy 304 (2021).
[80] Z. Wang, T. Hong, Reinforcement learning for building controls: The opportunities and challenges, Applied Energy 269 (2020), https://doi.org/10.1016/j.apenergy.2020.115036.
[81] J.R. Vázquez-Canteli, Z. Nagy, Reinforcement learning for demand response: A review of algorithms and modeling techniques, Applied Energy 235 (2019) 1072–1089, https://doi.org/10.1016/j.apenergy.2018.11.002.
[82] A. Perera, P. Kamalaruban, Applications of reinforcement learning in energy systems, Renewable and Sustainable Energy Reviews 137 (2021), https://doi.org/10.1016/j.rser.2020.110618.
[83] T. Haarnoja, A. Zhou, P. Abbeel, S. Levine, Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, CoRR abs/1801.01290. arXiv:1801.01290. URL: http://arxiv.org/abs/1801.01290.
[84] D. Picard, J. Drgoňa, M. Kvasnica, L. Helsen, Impact of the controller model complexity on model predictive control performance for buildings, Energy and Buildings 152 (2017) 739–751, https://doi.org/10.1016/j.enbuild.2017.07.027, URL: http://www.sciencedirect.com/science/article/pii/S0378778817302190.
[85] J. Arroyo, F. Spiessens, L. Helsen, Comparison of Model Complexities in Optimal Control Tested in a Real Thermally Activated Building System, Buildings 12 (5) (2022) 539, https://doi.org/10.3390/buildings12050539.
[86] M. Dorokhova, Y. Martinson, C. Ballif, N. Wyrsch, Deep reinforcement learning control of electric vehicle charging in the presence of photovoltaic generation, Applied Energy 301 (2021), https://doi.org/10.1016/j.apenergy.2021.117504, URL: https://www.sciencedirect.com/science/article/pii/S0306261921008874.
[87] G. Ceusters, R. Rodríguez, A. García, R. Franke, G. Deconinck, L. Helsen, A. Nowe, M. Messagie, L. Ramirez Camargo, Model-predictive control and reinforcement learning in multi-energy system case studies, 2021.
[88] J. Arroyo, Comparison of Optimal Control Techniques for Building Energy Management, Frontiers in Built Environment 8. doi:10.3389/fbuil.2022.849754.
[89] J. Arroyo, C. Manna, F. Spiessens, L. Helsen, Reinforced model predictive control (rl-mpc) for building energy management, Applied Energy 309 (2022), https://doi.org/10.1016/j.apenergy.2021.118346, URL: https://www.sciencedirect.com/science/article/pii/S0306261921015932.
[90] D. Blum, J. Arroyo, S. Huang, J. Drgoňa, F. Jorissen, H.T. Walnum, Y. Chen, K. Benne, D. Vrabie, M. Wetter, L. Helsen, Building optimization testing framework (BOPTEST) for simulation-based benchmarking of control strategies in buildings, Journal of Building Performance Simulation 14 (5) (2021) 586–610, https://doi.org/10.1080/19401493.2021.1986574.
[91] L. Di Natale, Y. Lian, E.T. Maddalena, J. Shi, C.N. Jones, Lessons Learned from Data-Driven Building Control Experiments: Contrasting Gaussian Process-based MPC, Bilevel DeePC, and Deep Reinforcement Learning doi:10.48550/ARXIV.2205.15703.
[92] ASHRAE Standing Standard Project Committee, ASHRAE STANDARD - Standard Method of Test for the Evaluation of Building Energy Analysis Computer Programs (2011) 276.
[93] A. Kümpel, J. Teichmann, P. Mathis, D. Müller, Modular hydronic subsystem models for testing and improving control algorithms of air-handling units, Journal of Building Engineering 53 (2022), https://doi.org/10.1016/j.jobe.2022.104439.
[94] M. Lauster, J. Teichmann, M. Fuchs, R. Streblow, D. Mueller, Low order thermal network models for dynamic simulations of buildings on city district scale, Building and Environment 73 (2014) 223–231, https://doi.org/10.1016/j.buildenv.2013.12.016, URL: https://linkinghub.elsevier.com/retrieve/pii/S0360132313003727.
[95] M. Lauster, A. Constantin, P. Remmen, M. Fuchs, D. Muller, Verification of a Low Order Building Model for the Modelica Library AixLib using ASHRAE Standard 140, in: Proceedings of the 15th IBPSA Conference, San Francisco, CA, USA, 2017, p. 10. doi:https://doi.org/10.26868/25222708.2017.303.
[96] SIA, SIA 2024 - Raumnutzungsdaten für die Energie- und Gebäudetechnik, Tech. rep., Zurich, Switzerland (2015).
[97] W.E. Hart, C.D. Laird, J.-P. Watson, D.L. Woodruff, G.A. Hackebeil, B.L. Nicholson, J.D. Siirola, Pyomo – Optimization Modeling in Python, Vol. 67 of Springer Optimization and Its Applications, Springer International Publishing, Cham, 2017. doi:10.1007/978-3-319-58821-6. URL: http://link.springer.com/10.1007/978-3-319-58821-6.
[98] L. Gurobi Optimization, Gurobi Optimizer Reference Manual, 2021. URL: http://www.gurobi.com.
[99] G. Reynders, J. Diriken, D. Saelens, Quality of grey-box models and identified parameters as function of the accuracy of input and observation signals, Energy and Buildings 82 (2014) 263–274, https://doi.org/10.1016/j.enbuild.2014.07.025.
[100] H.G. Bock, K.J. Plitt, A Multiple Shooting Algorithm for Direct Solution of Optimal Control Problems*, IFAC Proceedings Volumes 17 (2) (1984) 1603–1608, https://doi.org/10.1016/S1474-6670(17)61205-9, URL: https://www.sciencedirect.com/science/article/pii/S1474667017612059.
[101] F. Chollet, others, Keras, 2015. URL: https://keras.io.
[102] D.P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, arXiv:1412.6980 [cs]ArXiv: 1412.6980. URL: http://arxiv.org/abs/1412.6980.
[103] M. Rätz, A.P. Javadi, M. Baranski, K. Finkbeiner, D. Müller, Automated data-driven modeling of building energy systems via machine learning algorithms, Energy and Buildings 202 (2019), https://doi.org/10.1016/j.enbuild.2019.109384, URL: https://linkinghub.elsevier.com/retrieve/pii/S0378778819316585.
[104] T.P. Lillicrap, J.J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, D. Wierstra, Continuous control with deep reinforcement learning. URL: http://arxiv.org/pdf/1509.02971v6.
[105] R.S. Sutton, A. Barto, Reinforcement learning: An introduction, second edition Edition, Adaptive computation and machine learning, The MIT Press, Cambridge, MA and London, 2018.
[106] J. Bergstra, B. Komer, C. Eliasmith, D. Yamins, D.D. Cox, Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures, Computational Science & Discovery 8(1 (2015), https://doi.org/10.5555/3042817.3042832.
[107] L. Yang, Z. Nagy, P. Goffin, A. Schlueter, Reinforcement learning for optimal control of low exergy buildings, Applied Energy 156 (2015) 577–586, https://doi.org/10.1016/j.apenergy.2015.07.050, URL: https://www.sciencedirect.com/science/article/pii/S030626191500879X.
[108] S. Touzani, A.K. Prakash, Z. Wang, S. Agarwal, M. Pritoni, M. Kiran, R. Brown, J. Granderson, Controlling distributed energy resources via deep reinforcement learning for load flexibility and energy efficiency, Applied Energy 304 (2021), https://doi.org/10.1016/j.apenergy.2021.117733, URL: https://www.sciencedirect.com/science/article/pii/S0306261921010801.
[109] Y. Chen, Z. Tong, Y. Zheng, H. Samuelson, L. Norford, Transfer learning with deep neural networks for model predictive control of HVAC and natural ventilation in smart buildings, Journal of Cleaner Production 254 (2020), https://doi.org/10.1016/j.jclepro.2019.119866, URL: https://linkinghub.elsevier.com/retrieve/pii/S0959652619347365.
[110] O. Kotevska, J. Munk, K. Kurte, Y. Du, K. Amasyali, R.W. Smith, H. Zandi, Methodology for interpretable reinforcement learning model for hvac energy control, in: 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 1555–1564. doi:10.1109/BigData50022.2020.9377735.