2018 International Conference on Intelligent Systems (IS)

Data-driven Optimization of Energy Efficiency and Comfort in an Apartment

Diego Nieves Avendano
IDLab, Ghent University - imec
Ghent, Belgium
diego.nievesavendano@ugent.be

Joeri Ruyssinck
IDLab, imec - Ghent University
Ghent, Belgium
joeri.ruyssinck@ugent.be

Steven Vandekerckhove
Renson Ventilation
Waregem, Belgium
steven.vandekerckhove@renson.be

Sofie Van Hoecke
IDLab, Ghent University - imec
Ghent, Belgium
sofie.vanhoecke@ugent.be

Dirk Deschrijver
IDLab, imec - Ghent University
Ghent, Belgium
dirk.deschrijver@ugent.be

***Abstract*—An important challenge in home automation is the energy efficient optimization of the indoor environment. This relies on the solution of a multi-objective optimization problem where energy efficiency and comfort parameters are maximized simultaneously. This paper presents three data-driven control algorithms based on machine learning techniques, which offer an alternative to traditional control methods. The results demonstrate that some data-driven methods can achieve similar results than rule-based systems. Moreover, they require no prior expert knowledge and have better scalability than standard approaches.**

***Keywords*—Model predictive control; Deep reinforcement learning; Multi-objective optimization; Home automation; Energy efficiency**

I. Introduction

Buildings consume over 40% of the total power in developed countries, 27% corresponding to household's consumption and 13% to other building types. More specifically, heating accounts for 55% to 67% of the final energy used in residential buildings [1]. New standards and directives in Europe aim towards energy efficient buildings, which brings attention to control algorithms that can maximize energy efficiency while maintaining adequate comfort conditions and air quality for the inhabitants.

Comfort is a subjective experience and depends on multiple parameters such as thermal comfort and air quality. Measuring related parameters, such as temperature or CO_2 levels, is nowadays possible due to the adoption of sensors for homes. The data collected by these sensors has potential applications in indoor environment control.

Thermal comfort is mainly dependent on room temperature and relative humidity (RH). Other factors include metabolic rate, clothing insulation, gender, user's expectations, air speed and mean radiant temperatures [2].

Indoor air quality is the degree in which the indoor air is pollutant free. It is indicated by the levels of indoor contaminants such as sulfur dioxide (SO_2), carbon dioxide (CO_2), ozone (O_3), volatile organic carbon (VOC), among others. Short and long-term exposure to such contaminants can have acute health effects. In household studies, CO_2 is not a concern as its levels are unlikely to exceed safe thresholds under normal building conditions. However, measuring CO_2 is often done as it provides information about air re-circulation and potential exposure to other pollutants. In addition, CO_2 levels are valuable indicators of room occupancy, which in turn is used as a proxy for ventilation, cooling and heating demands [3, 4].

In general, maximizing comfort comes with an increase in energy consumption. In order to keep a good indoor air quality, a high ventilation rate is needed, which in turn results in a large heat loss.

This paper focuses on developing three data-driven control algorithms for the optimization of comfort and energy-efficiency. The first method is a Moving Horizon Estimation (MHE) controller based on gradient boosted regression trees. The second and third methods use Reinforcement Learning (RL) techniques. The models are trained and validated using simulations of a two room apartment under different occupancy profiles. It is confirmed by numerical results that some data-driven methods can achieve similar results as rule-based systems.

The novel contributions of this paper are the following:

- Multi-parameter optimization: Most of the related work focuses on the optimization of only one or two parameters such as temperature, air quality, and energy efficiency (ventilation, heating or cooling). This approach optimizes for four parameters: Indoor temperature, CO_2 levels, heating energy and ventilation energy.

978-1-5386-7097-2/18/$31.00 ©2018 IEEE

- Data-driven and model-free: Compared to traditional rule-based control, data-driven models do not require expert knowledge, nor a description of the building's physical dynamics. Additionally, data-driven models can infer secondary information such as occupant's behavior and preferences, which can potentially improve control efficiency.

The remainder of the paper is organized as follows: Section II contains an overview of related work concerning model predictive control and reinforcement learning. Section III describes the simulation settings and defines the optimization problem. Section IV presents the methodology, elaborating on the different control strategies. Section V contains the results. Section VI gives the conclusions.

II. Related work

Traditionally, building automation problems are addressed by rule-based systems in which an expert uses best practices and mathematical models to create a set of rules that control different house components such as heating and ventilation. Some strategies include intelligent scheduling, set-point resets and demand-controlled ventilation [5]. Although rule-based systems are extensively used, they have limitations such as:

- Solutions rely on heuristics and are most likely sub-optimal.
- Their design is building specific and can hardly be transferred to other buildings.
- They offer poor scalability for larger buildings or multiple components [6].

Model predictive control (MPC) creates one or more models that approximate the physical properties of the building in order to predict the future states within a time horizon. The controller then takes the action(s) that maximize a target function within a given set of constraints.

In MPC, Artificial Neural Networks (ANN) have successfully been applied for optimizing the operation time of heating, ventilation and air conditioning units (HVAC) [7, 8], predicting temperature and relative humidity [9], and estimating future load demand [10, 11]. Other examples of soft-computing techniques include thermal comfort controllers based on neuro-fuzzy logic [12, 13], and Random Forests for occupancy estimation [14].

More recent papers have investigated the use of RL techniques, where the optimization problem is addressed by an agent that learns actions in a goal-oriented manner. Compared to MPC techniques, RL offers the possibility of optimizing long time horizons without causing time overheads during evaluation. Additionally, they can easily be extended to control multiple actuators and self adjust as user's preference change over time.

The multi-objective optimization of thermal comfort, energy efficiency and indoor air quality has been addressed with a radial basis approximator in Q-learning [15], Extreme Random Forests with Q-learning [16], ANNs with an Actor-Critic architecture [17], Deep Q-learning (DQN) [18, 19], and Convolutional Neural Networks (CNNs) [20].

For a more detailed review on building control we refer to the surveys concerning HVAC control based on ANN [21], time-series [22], and other soft-computing techniques [23].

III. Optimization problem

The aim is to develop an intelligent control system that optimizes comfort and energy efficiency by controlling multiple actuators in a simulated indoor environment. The building specifications of the environment are described in Section III-A. The simulation settings are provided in Section III-B. The optimization variables and objectives are outlined in Section III-C.

A. Building Specifications

The simulated building is a two zone apartment located in Brussels. Zone 1 (Z1) is composed of a living room and entrance and zone 2 (Z2) is a bathroom. Table I contains the area distribution and Fig. 1 its layout.

The roof and the facade with the window are in contact with the exterior and have a thermal transmittance (U-value) of 0.15 W/(m^{2}K). The other surfaces are considered to be adiabatic causing no energy loss. The window in zone 1 can be completely covered by an awning to reduce incident radiation, it has a solar transmittance of 0.05 and reflectance of 0.142. The ventilation system has a natural air supply and a mechanical air exhaust with an extractor in each zone. Zone 1 has an extraction rate of minimum ±4.5 m^3/h and maximum of ±70 m^3/h. Zone 2 has an extraction rate of minimum ±9 m^3/h and maximum ±50 m^3/h.

All the actuators are present in zone 1. Table II summarizes the actuators' effects and Table III shows their ten possible states. The numbers in parentheses represent ordinal quantities which are used to explain the control Algorithm 1 in Section

TABLE I
Simulated apartment's area distribution

Zone	Space	Dimensions	Surface area
Z1	Living room	5 x 6 x 2.7 m	30 m^2
	Entrance	1.2 x 2 x 2.7 m	2.4 m^2
	(Window)	(4.5 x 2.2 m)	(9.9 m^2)
Z2	Bathroom	3.8 x 2 x 2.7 m	7.6 m^2
Total		5 x 8 x 2.7 m	40 m^2

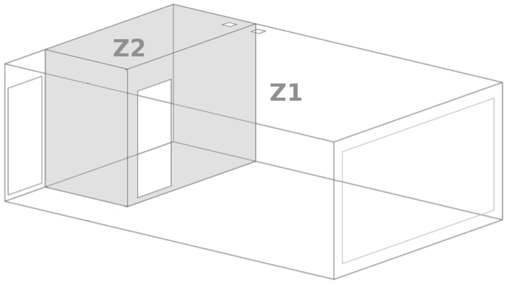

Fig. 1. Layout of the simulated two-room apartment

TABLE II
OVERVIEW OF ACTUATORS AND THEIR EFFECTS

Actuators	States	Effect	Z1	Z2
Ventilation		*Ventilation rate*	x	
	Low	4.5 m^3/h		
	Mid	17.6 m^3/h		
	High	62.8 m^3/h		
Window		*Ventilation rate*	x	
	Closed	dependent[a]		
	Tilted	106.2 m^3/h		
	Open	232.4 m^3/h		
Awning		*Incident radiation*	x	
	On	dampened[b]		
	Off	full		

[a] While the window is closed the rate is determined by the ventilation state.
[b] The incident radiation is reduced according to the awning's physical properties.

TABLE III
POSSIBLE ACTUATORS' STATES

Window	Ventilation	Awning
Closed (0)	Low (1)	On (1)
Closed (0)	Medium (2)	On (1)
Closed (0)	High (3)	On (1)
Closed (0)	Low (1)	Off (0)
Closed (0)	Medium (2)	Off (0)
Closed (0)	High (3)	Off (0)
Tilted (0.5)	Inactive (0)	On (1)
Tilted (0.5)	Inactive (0)	Off (0)
Open (1)	Inactive (0)	On (1)
Open (1)	Inactive (0)	Off (0)

* The numbers in parentheses represent ordinal quantities and are used to explain the algorithm in Section III-C.

III-C. It is important to notice two aspects of the actuators' functionality. First, the window can provide additional ventilation at no energy cost, with a possibly larger loss-heat due to the increased ventilation. Second, there is no direct control of the heating in zone 1, nor of the ventilation and heating in zone 2. These components are regulated by their own control systems that react according to CO_2 concentration and indoor temperature. Nevertheless their behavior is indirectly affected by the actions of the available actuators.

B. Simulation

The simulation is done using EnergyPlus [24] and interfaced with control algorithms through Building Controls Virtual Test Bed (BCVTB) [25].

The simulation goes from the beginning of February until the end of April. This period provides a varied range of conditions: in the beginning heating is essential for indoor comfort, and at the end additional ventilation is of importance. The outdoor temperature ranges from -9.1 °C to 22.7 °C, the incident radiation ranges from 0 W/m^2 to 837.7 W/m^2. The simulation is done in steps of ten minutes. Table IV shows the simulation variables.

Three realistic occupancy profiles are used in order to create different energy demand and air pollution scenarios. The profiles are generated with a technique that provides realistic behavior for Belgian households based on Aerts *et al.* [26]. The simulated occupancy profiles are:

A. One retired adult.
B. Two adults without children and both working full time.
C. Two adults without children, one working full time and the other part-time.

Fig. 2 shows the corresponding occupancy distributions.

TABLE IV
SIMULATION VARIABLES

	Outdoor	Z1	Z2
State variables	Temperature (°C) Time	Temperature (°C) CO_2 concentration (ppm) Relative humidity (%) Illuminance (lux) People count Heating power (W) Ventilation power (W) Incident radiation (W/m^2) Awning status Window status	Temperature (°C) CO_2 concentration (ppm) Relative humidity (%) Illuminance (lux) People count Heating power (W) Ventilation power (W)
Actuators	-	Awning Window Ventilation	-

C. Optimization Objective

Each control algorithm is evaluated in terms of comfort and energy efficiency profiles as follows. First each variable of interest is converted into a score according to the weights and formulas in Table V. Second, the variable scores are aggregated in a weighted sum which results in a comfort score (1) and an energy score (2).

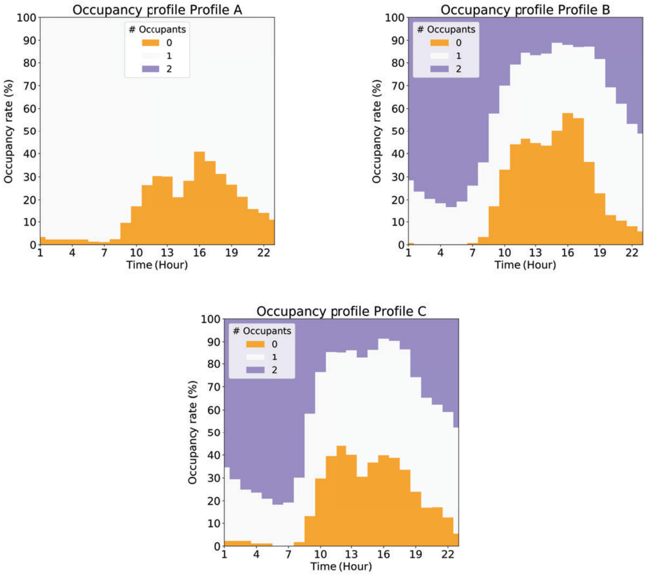

Fig. 2. Occupancy profiles. For each hour is shown the amount of people present in the apartment on average during the simulation.

TABLE V
SCORE FUNCTIONS

Temperature Day	$S_{\text{T day}} \begin{cases} 4, & \text{if} \quad 21.5 >= \text{Temp(Z1)} >= 20.5 \text{ or Occupancy} == 0 \\ \max(0, 5 - 2\vert\text{Temp(Z1)} - 21.5\vert), & \text{elsewise} \end{cases}$
Temperature Night	$S_{\text{T night}} \begin{cases} 4, & \text{if} \quad 22 >= \text{Temp(Z1)} >= 19 \text{ or Occupancy} == 0 \\ \max(0, 5.5 - \vert\text{Temp(Z1)} - 20.5\vert), & \text{elsewise} \end{cases}$
CO_2 concentration	$S_{\text{CO}_2\text{(Z1)}} \begin{cases} 4, & \text{if} \quad 800\text{ppm} >= \text{CO}_2\text{(Z1) or Occupancy} == 0 \\ \max(0, 4 - (\text{CO}_2\text{(Z1)} - 800)/133), & \text{elsewise} \end{cases}$
Ventilation Energy	$S_{\text{vent}} \begin{cases} 4, & \text{if} \quad 5000 > E_{\text{vent}} \\ \frac{5000 - E_{\text{vent}}}{11250} + 4, & \text{if} \quad 50000 > E_{\text{vent}} > 5000 \\ 0, & \text{elsewise} \end{cases}$
Heating Energy	$S_{\text{heat}} \begin{cases} 4, & \text{if} \quad 50000 > E_{\text{heat}} \\ \frac{50000 - E_{\text{heat}}}{75000} + 4, & \text{if} \quad 350000 > E_{\text{heat}} > 50000 \\ 0, & \text{elsewise} \end{cases}$
Weights	$w_{T\ day} = 2$, $w_{T\ night} = 1$, $w_{CO_2} = 2$, $w_{vent} = 1$, $w_{heat} = 3$

$$S_{\text{comfort}} = \frac{S_{\text{T day}} w_{\text{T day}} + S_{\text{T night}} w_{\text{T night}} + S_{\text{CO}_2} w_{\text{CO}_2}}{w_{\text{T day}} + w_{\text{T night}} + w_{\text{CO}_2}} \tag{1}$$

$$S_{\text{energy}} = \frac{S_{\text{vent}} w_{\text{vent}} + S_{\text{heat}} w_{\text{heat}}}{w_{\text{vent}} + w_{\text{heat}}} \tag{2}$$

Notice two important aspects: first, room temperature S_T is evaluated in two different ways in order to force a higher level of comfort during daytime and when people are present. The time between 8:10 and 23:00 corresponds to day-time and the rest to night-time. Second, the comfort score only considers the variables in Z1 as it is the zone where occupants spend most of their time. For the energy score, the energy consumption of both zones is considered.

In order to evaluate different user's preferences two profiles are proposed: Comfort priority (3) and energy efficiency priority (4).

$$\text{Comfort preference profile} = 0.9S_{\text{comfort}} + 0.1S_{\text{energy}} \tag{3}$$

$$\text{Energy preference profile} = 0.1S_{\text{comfort}} + 0.9S_{\text{energy}} \tag{4}$$

IV. Methodology

In this section, the rule-based system, the moving horizon estimation and the reinforcement learning approaches are briefly introduced.

A. Rule-based System

A rule-based system is a common approach for building automated control which can prioritize either for comfort or energy efficiency. The benchmark is a rule-based system designed for this specific case. The rules take into account room occupancy, room temperature, CO_2 levels, incident radiation and the previous actuator state. Algorithm 1 contains the pseudocode for the rule-based system that prioritizes comfort. The numeric values for the actuator states correspond to discrete actions as specified in Table III. The rules for the controller with energy efficiency priority follow a similar structure with different threshold levels. For the sake of brevity this pseudocode is not included.

Algorithm 1: Rule-based for comfort

```
Input  : P People count in both zones, T Z1 Temperature,
         R Z1 Incident radiation, C Z1 $CO_2$ level, Wp
         Previous window's status, Vp Previous ventilation
         status, Ap Previous awning's status,
Output: Ws Window's status, Vs Ventilation status, As
         Awning's status
$Ws = 0$ $As = 0$ $Vs = 1$
if P $> 0$ then
    if T $>= 22.2$ and Wp $== 0$ then Ws $= 0.5$
    else if T $>= 23$ and Wp $<= 0.5$ then Ws $= 1$
    else if T $<= 21.5$ and Wp $> 0$ then Ws $= 0$
    else Ws = Wp
else Ws $= 0$
if T $> 20.5$ then
    if T $> 21.5$ then
        if R $> 200$ then As $= 1$
        else As $= 0$
    else
        if R $> 240$ then As $= 1$
        else As $= 0$
else As $= 0$
if C $< 720$ then
    if T $> 21.8$ then
        Vs = Vp $+ 1$
        if Vs $> 3$ then Vs $= 3$
    else Vs $= 1$
else
    if Vp $== 1$ then
        Vs $= 2$
    else if Vp $== 2$ then
        if C $> 760$ or T $> 21.8$ then Vs $= 3$
        else Vs $= 2$
    else if Vp $== 3$ then
        if T $> 21.8$ then
            Ws $= 0.5$
        else if T $> 22.5$ then
            Ws $= 1$
        else Vs $= 3$
    else if Wp $> 0$ and Ws $== 0$ then
        Vs $= 3$
if Ws $== 0.5$ then Vs $= 0$
if Ws $== 1$ then Vs $= 0$
```

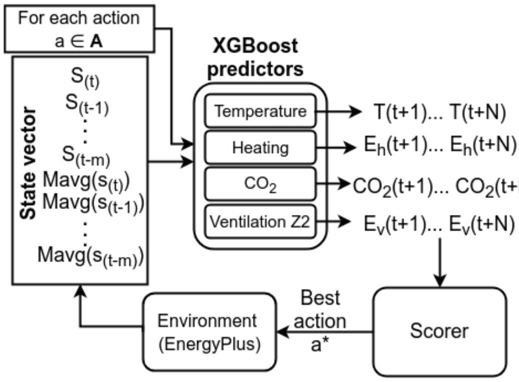

Fig. 3. Moving horizon estimation. *A* is the set of available actions (see Table III), *S* are the state variables (see Table IV) and *Mavg* the moving average of each state variable.

B. Moving Horizon Estimation

The MHE controller is composed of a predictive model and a scorer. The predictive model estimates the indoor temperature, CO_2, ventilation and heating demand for the next N time-steps based on the current state (see Table IV), moving averages of the state variables and the possible actions (see Table III). The scorer evaluates the predicted states and chooses the action that maximizes the score within the predicted horizon. Fig. 3 shows a schematic of the MHE controller.

The predictive models are based on extreme gradient boosted trees (XGBoost) [27]. To forecast multiple steps ahead, a rolling approach is applied, where only the information available up to current time step is used to predict the next N time-steps.

The number of possible outcomes increases exponentially with the number of time-steps N. In order to reduce the computation time, it is assumed that the controller repeats the same action for the next k time-steps, where $1 < k < N$, meaning that the controller may take an action for a shorter period than the prediction horizon. This is a sensible choice as prediction errors become larger as the horizon (N) grows.

C. Reinforcement Learning

In RL, an agent interacts with an environment in discrete steps and learns to perform the best actions over time. The agent in this case is equivalent to the controller. For a given state (s_t) the agent takes an action (a_t) by following a policy (π). Afterwards the system transitions to a new state s_{t+1} and receives a scalar reward (r). The objective then is to learn a policy that maximizes the rewards, in other words, that learns to take the best actions for all possible states. The objective can be expressed as maximizing the sum of discounted rewards over the episode $R = \sum_{t=t_0}^{\infty} \gamma^{t-t_0} r_t$. Where γ is a discount factor that makes the sum finite, and balances the importance of immediate rewards and future ones. Fig. 4 shows the general RL structure.

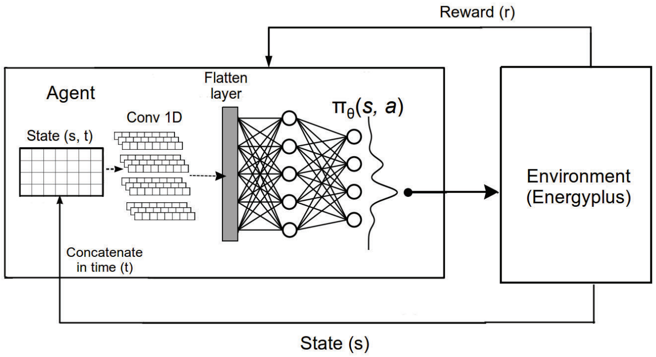

Fig. 4. Reinforcement Learning controller

Deep Q-learning (DQN) is a RL technique where ANNs are used as the approximator for the mapping function. In this paper two DQN extensions are implemented, namely Double Q-learning (DDQN) [28] and Rainbow (RBW) [29]. As far as we know, this is the first work to introduce RBW for building environment control.

Compared to the MHE controller, RL offers two advantages. First, the RL models learn to predict the reward of future states instead of the future state variables, which removes the need of having independent models for each variables of interest as seen in Fig. 3. Second, the MHE model predicts state variables for each subsequent time-step, whereas RL learns a sum of discounted future values, which removes the need of making the N time-step predictions. These advantages reduce the model complexity and the overhead of predicting multiple variables in subsequent time-steps, however, RL models require considerably more data in order to be trained.

V. Results

In this section the simulated building presented in Section III is evaluated following the control algorithms presented in Section IV. Section V-A contains the MHE specifications. Section V-B contains the RL specifications. Section V-C compares the models and discusses the results.

A. Moving Horizon Estimation Setup

For each occupation profile, a MHE controller is trained using offline recordings of the other occupancy profiles. This is done so the model is trained under different occupancy profiles, and then tested in an unseen profile. For each occupancy profile four datasets are recorded: two while using the rule-based system (see Algorithm 1). Similarly, two with a modified version of the rule-based system which takes random actions with a certain probability.

The controller consists of four independent XGBoost models and a scorer. The models forecast indoor temperature and CO_2 levels for zone 1, ventilation energy demand for zone 2, and heating energy demand for both zones. Note that ventilation of zone 1 is directly controlled. Each XGBoost model is tuned using 10-Fold cross-validation and grid search to find the best hyper-parameters. As subsequent time-series samples are temporally dependent, a split-block strategy is used, where the splits are done in full days, rather than individual samples [30].

During evaluation, the models predict the next *N* time-steps according to the current state (see Table IV) and possible actions (see Table III). Each prediction is then converted into a score following the preference profiles specified in (3) and (4). Finally, the controller selects the best action a*, which corresponds to the action that maximizes the score for the whole horizon *N*.

If the MHE algorithm is directly applied, the controller behaves as an on/off system that relies mostly on the window and seldom the ventilation. In practice this would be an undesired behavior, therefore, an additional constraint is added that restricts the controller from switching the window status more than 15 times within 12 hours. If the constraint is violated, the scorer takes the next action with highest score, and successively until the action is a valid one.

The *k* value is set to 1 and the prediction horizon *N* to 4 (40 minutes). Other k-values were evaluated but are not reported here as their performance was not significantly better than this configuration.

B. Reinforcement Learning Setup

In the case of RL, the controller is trained online while it interacts directly with the environment. A RL model is trained for each occupancy profile and each preference profile.

The input state corresponds to the concatenation in time of the last 18 time-steps (3 hours). The reward is the score for the new state according to the preference profiles in (3) and (4).

Table VI shows the agent specifications. The network architecture is slightly different between DDQN and RBW. In both cases the first two layer are convolutions in the time axis, which allow learning temporal features. The rest of the implementations is as follows:

- DDQN: Table VII shows the network architecture. The agent follows a ϵ-greedy policy with linear decay. After 28 episodes the ϵ value is kept to 0.05 and the simulation is run for another 12 episodes. This gives in total information of 120 months (~1.540.000 samples).

TABLE VII
DDQN NETWORK PARAMETERS

Network configuration			
#	**Type**	**Size**	**Specs**

1	Conv1D	32	window = 4
			stride = 3
			padding = Same

2	Conv1D	32	window = 3
			stride = 2
			padding = Same

4	Flatten	-	-
5	Dense	256	-
6	Dropout	-	p = 0.20
7	Dense	128	-
8	Dropout	-	p = 0.20
9	Softmax	10	-

- RBW: Table VIII shows the network architecture. The RBW agent uses noisy linear layers as mean of exploration, therefore no ϵ value is given. Convergence of RBW was faster than DDQN, therefore the simulation is run for only 30 episodes, equivalent to 90 months (~1.155.000 samples).

C. Evaluation

The results of the three models are summarized in Figs. 5 to 7 for each of the three models and the benchmark (REF). The results compare performance in terms of comfort and energy scores (1) and (2) for each occupancy profile. The dashed line corresponds to the Pareto front, and highlights all optimal solutions.

The RBW implementation was able to obtain similar results than the rule-based system under all of the occupancy profiles. In all cases the solutions found are non-dominated and part of the Pareto front. In some cases the trade-off for comfort and energy presents potential gains. For example, the comfort preference profile RBW under the occupancy profile B yields a saving of 17% of heating energy without a significant change in comfort (see Fig. 6). Table IX compares the energy consumption between RBW and the rule-based system for the three months period.

On the other hand, MHE and DDQN performances are below the Pareto front in most cases, except for occupancy-

TABLE VI
RL PARAMETERS

	DDQN	**RBW**
Discount	0.95	
Batch size	64	
Initial replay	12816	
Memory type	replay	
Memory size	25632	
Synch. frequency	6000	5000
ϵ **init** ϵ **decay step** ϵ **final**	1 2.5 e-6 0.05	-
Vmin **Vmax**	-	0 $\frac{1}{1-\gamma}$

TABLE VIII
RBW NETWORK PARAMETERS

Network configuration				
#	**Type**	**Size**	**Specs**	**Module**

1	Conv1D	32	window = [5]
			stride = 2
			padding = same
				Sequential

2	Conv1D	32	window = [3]
			stride = 1
			padding = same	

3	Flatten	-	-	
4	Dropout	-	p = 0.2	
5	Noisy Linear	512	σ = 0.5	Dueling
6	Noisy Linear	512	σ = 0.5	
7	Soft max	num. atoms	-	

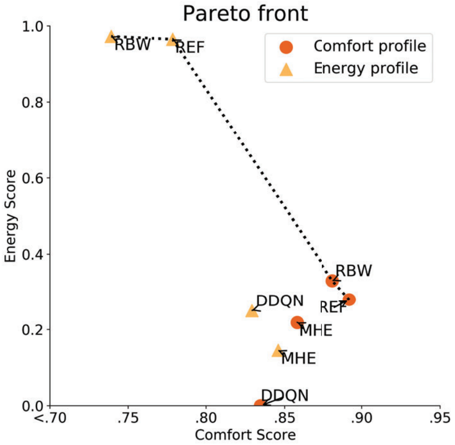

Fig. 5. Performance in occupation profile A

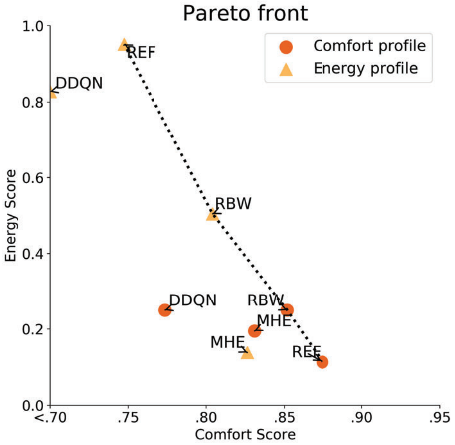

Fig. 6. Performance in occupation profile B

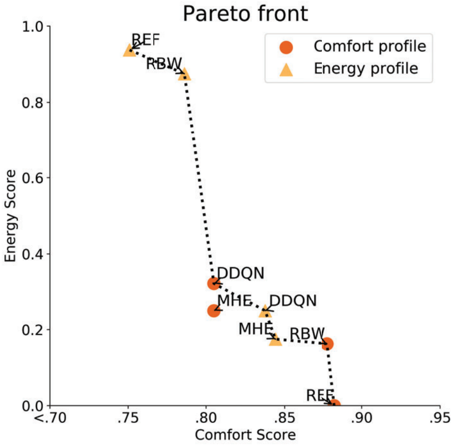

Fig. 7. Performance in occupation profile C

TABLE IX
ENERGY CONSUMPTION IN KWH

Profile		**Vent. Energy**		**Heat. Energy**	
		REF	**RBW**	**REF**	**RBW**
A	Comfort	25.82	51.87	291.73	217.92
	Energy	1.04	2.08	64.11	61.01
B	Comfort	54.82	5.02	304.63	357.33
	Energy	5.61	2.73	68.32	248.24
C	Comfort	50.59	51.61	351.19	288.65
	Energy	5.45	2.53	73.90	99.90

profile C (see Fig. 7) where also the DDQN and MHE models for energy optimization are part of the Pareto front.

MHE's poor performance can be attributed to two factors: first, the actuator's effect may occur in time spans longer than the optimization horizon of 40 minutes. Increasing the time horizon would in principle improve the performance with the drawback of requiring a more complex model and a considerable computational overhead. Second, the MHE was trained using data generated by the rule-based system which lacks information of undesirable states. In case the controller observes a previously unseen state, it may be taking suboptimal actions, which points to a poor generalization. In order to compensate for this, additional data was generated by allowing the controller to take random actions. However, based on the results, it seems that the amount of data generated was insufficient, or the random actions were not able to generate valuable information.

DDQN's poor performance can be attributed to a limited exploration due to following the ϵ-greedy policy. This problem is partially addressed by the noisy layers, which are included in the RBW implementation. This can also be related to a high similarity between the outcome of different actions. This can occur when external parameters have a considerable influence in the building's environment, such as wind speed, rain and clouds. The difference between taking different actions becomes less significant and may lead to under-performing policies. In such scenarios the dueling networks approach is able to find better policies [29].

VI. CONCLUSIONS

This work presented three multi-objective data-driven control approaches to maximize comfort and energy efficiency in a simulated building.

It was found that RBW is able to achieve a performance comparable to the traditional rule-based systems. It was also found that MHE and DDQN cannot achieve good performances under the chosen settings.

RBW offers an alternative to traditional building's environment control. As it is a driven-data algorithm, it can be trained using historical data. Thanks to its model free approach, it can be implemented in buildings with different physical properties, more zones and different user's behavior without requiring an extensive study of their interactions with the indoor environment.

Acknowledgment

The authors would like to thank An-Heleen Deconinck and Friedl Decock from Daidalos Peutz.

This work was partially funded by Flanders Innovation & Entrepreneurship (VLAIO) in the framework of the iCosy R&D project.

References

[1] J. Laustsen, P. Ruyssevelt, D. Staniaszek, D. Strong, S. Zinetti, C. Despret, M. Economidou, J. Maio, I. Nolte, and O. Rapf, "Europe buildings today," in *Europe buildings under the microscope: A country-by-country review of the energy performance of buildings* (B. Atanasiu, C. Despret, M. Economidou, J. Maio, I. Nolte, and O. Rapf, eds.), Buildings Performance Institute Europe (BPIE), 2011.

[2] P. O. Fanger, *Thermal comfort: analysis and applications in environmental engineering*. Danish Technical Press, 1970.

[3] X. Zhang, P. Wargocki, and Z. Lian, "Effects of exposure to carbon dioxide and human bioeffluents on cognitive performance," *Procedia Engineering*, vol. 121, pp. 138–142, jan 2015.

[4] J. A. Bernstein, N. Alexis, H. Bacchus, I. L. Bernstein, P. Fritz, E. Horner, N. Li, S. Mason, A. Nel, J. Oullette, K. Reijula, T. Reponen, J. Seltzer, A. Smith, and S. M. Tarlo, "The health effects of non-industrial indoor air pollution," *The Journal of allergy and clinical immunology*, vol. 121, pp. 585–91, mar 2008.

[5] M. Eftekhari and L. Marjanovic, "Application of fuzzy control in naturally ventilated buildings for summer conditions," *Energy and Buildings*, vol. 35, pp. 645–655, aug 2003.

[6] K. Maík, J. Rojíček, P. Stluka, and J. Vass, "Advanced HVAC Control: Theory vs. Reality," *IFAC Proceedings Volumes*, vol. 44, pp. 3108–3113, jan 2011.

[7] N. Nassif, "Modeling and optimization of HVAC systems using artificial neural network and genetic algorithm," *Building Simulation*, vol. 7, pp. 237–245, jun 2014.

[8] I.-H. Yang, M.-S. Yeo, and K.-W. Kim, "Application of artificial neural network to predict the optimal start time for heating system in building," *Energy Conversion and Management*, vol. 44, pp. 2791–2809, oct 2003.

[9] G. Mustafaraj, G. Lowry, and J. Chen, "Prediction of room temperature and relative humidity by autoregressive linear and nonlinear neural network models for an open office," *Energy and Buildings*, vol. 43, pp. 1452–1460, jun 2011.

[10] Y. Yao, Z. Lian, S. Liu, and Z. Hou, "Hourly cooling load prediction by a combined forecasting model based on Analytic Hierarchy Process," *International Journal of Thermal Sciences*, vol. 43, pp. 1107–1118, nov 2004.

[11] M. R. Biswas, M. D. Robinson, and N. Fumo, "Prediction of residential building energy consumption: A neural network approach," *Energy*, vol. 117, pp. 84–92, dec 2016.

[12] K. L. Ku, J. S. Liaw, M. Y. Tsai, and T. S. Liu, "Automatic control system for thermal comfort based on predicted mean vote and energy saving," *IEEE Transactions on Automation Science and Engineering*, vol. 12, pp. 378–383, jan 2015.

[13] Wu Jian and Cai Wenjian, "Development of an adaptive neuro-fuzzy method for supply air pressure control in HVAC system," in *2000 IEEE International Conference on Systems, Man and Cybernetics: Cybernetics Evolving to Systems, Humans, Organizations, and their Complex Interactions*, vol. 5, pp. 3806–3809, IEEE, 2000.

[14] L. M. Candanedo and V. Feldheim, "Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models," *Energy and Buildings*, vol. 112, pp. 28–39, jan 2016.

[15] K. Dalamagkidis, D. Kolokotsa, K. Kalaitzakis, and G. Stavrakakis, "Reinforcement learning for energy conservation and comfort in buildings," *Building and Environment*, vol. 42, pp. 2686–2698, jul 2007.

[16] G. T. Costanzo, S. Iacovella, F. Ruelens, T. Leurs, and B. J. Claessens, "Experimental analysis of data-driven control for a building heating system," *Sustainable Energy, Grids and Networks*, vol. 6, pp. 81–90, 2016.

[17] D. Du and M. Fei, "A two-layer networked learning control system using actorcritic neural network," *Applied Mathematics and Computation*, vol. 205, pp. 26–36, nov 2008.

[18] T. Wei, Y. Wang, and Q. Zhu, "Deep reinforcement learning for building HVAC control," in *Proceedings of the 54th Annual Design Automation Conference 2017 on - DAC '17*, (New York, New York, USA), pp. 1–6, ACM Press, 2017.

[19] Y. Chen, L. K. Norford, H. W. Samuelson, and A. Malkawi, "Optimal control of HVAC and window systems for natural ventilation through reinforcement learning," *Energy and Buildings*, vol. 169, pp. 195–205, jun 2018.

[20] B. J. Claessens, P. Vrancx, and F. Ruelens, "Convolutional Neural Networks for Automatic State-Time Feature Extraction in Reinforcement Learning Applied to Residential Load Control," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3259–3269, 2018.

[21] A. Afram, F. Janabi-Sharifi, A. S. Fung, and K. Raahemifar, "Artificial neural network (ANN) based model predictive control (MPC) and optimization of HVAC systems: A state of the art review and case study of a residential HVAC system," *Energy and Buildings*, vol. 141, pp. 96–113, apr 2017.

[22] C. Deb, F. Zhang, J. Yang, S. E. Lee, and K. W. Shah, "A review on time series forecasting techniques for building energy consumption," *Renewable and Sustainable Energy Reviews*, vol. 74, pp. 902–924, jul 2017.

[23] F. Belic, Z. Hocenski, and D. Sliskovic, "HVAC control methods - a review," in *2015 19th International Conference on System Theory, Control and Computing*

(ICSTCC), pp. 679–686, IEEE, oct 2015.

[24] D. B. Crawley, C. O. Pedersen, L. K. Lawrie, and F. C. Winkelmann, “EnergyPlus: energy simulation program,” *ASHRAE*, vol. 42, pp. 49—-56, 2000.

[25] M. Wetter and P. Haves, “A modular building controls virtual test bed for the integration of heterogeneous systems,” in *SimBuild 2008, July 30-August 1*, (Berkeley, CA), 08/2008 2008.

[26] D. Aerts, J. Minnen, I. Glorieux, I. Wouters, and F. Descamps, “A probabilistic activity model to include realistic occupant behaviour in building simulations,” in *International Building Performance Simulation Association IBPSA - eSIM*, (Ottawa), 2014.

[27] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” mar 2016.

[28] H. V. Hasselt, “Double Q-learning,” in *Advances in Neural Information Processing Systems* (J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, eds.), pp. 2613–2621, Curran Associates, Inc., 2010.

[29] M. Hessel, J. Modayil, H. van Hasselt, T. Schaul, G. Ostrovski, W. Dabney, D. Horgan, B. Piot, M. Azar, and D. Silver, “Rainbow: Combining improvements in deep reinforcement learning,” oct 2017.

[30] D. R. Roberts, V. Bahn, S. Ciuti, M. S. Boyce, J. Elith, G. Guillera-Arroita, S. Hauenstein, J. J. Lahoz-Monfort, B. Schröder, W. Thuiller, D. I. Warton, B. A. Wintle, F. Hartig, and C. F. Dormann, “Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure,” *Ecography*, vol. 40, pp. 913–929, aug 2017.