Published by
NIGERIAN SOCIETY OF PHYSICAL SCIENCES
Available online @ https://journal.nsps.org.ng/index.php/jnsps

J. Nig. Soc. Phys. Sci. 5 (2023) 1244

Journal of the Nigerian Society of Physical Sciences

A Data-Driven Approach Towards the Application of Reinforcement Learning Based HVAC Control

Constantin Falk*, Tarek El Ghayed, Ron van de Sand, Jörg Reiff-Stephan

ic3@Smart Production, University of Applied Sciences Wildau, Germany

Abstract

Refrigeration applications consume a significant share of total electricity demand, with a high indirect impact on global warming through greenhouse gas emissions. Modern technology can help reduce the high power consumption and optimize the cooling control. This paper presents a case study of machine-learning for controlling a commercial refrigeration system. In particular, an approach to reinforcement learning is implemented, trained and validated utilizing a model of a real chiller plant. The reinforcement-learning controller learns to operate the plant based on its interactions with the modeled environment. The validation demonstrates the functionality of the approach, saving around 7% of the energy demand of the reference control. Limitations of the approach were identified in the discretization of the real environment and further model-based simplifications and should be addressed in future research.

DOI:10.46481/jnsps.2023.1244

Keywords: Refrigeration system, Reinforcement learning, Optimized control. Q-learning

Article History :
Received: 09 November 2022
Received in revised form: 16 January 2023
Accepted for publication: 16 January 2023
Published: 24 February 2023

Communicated by: B. J. Falaye

1. Introduction

The United Nations (UN) adopted the Agenda for Sustainable Development for 2030, which entails 17 Sustainable Development Goals (SDG) [1]. An important topic regarding these goals is the energy consumption through cooling applications. Cooling processes in Germany for example, contribute for around 14% of the German electric power consumption [2]. Finding novel and innovative approaches is necessary in order to reduce and optimize the energy consumption regarding cooling processes like Heating Ventilation and Air Conditioning (HVAC). As a result, our society could benefit from lower energy costs, reduced emission of CO_2 and reduced peak loads for electricity providers [3]. A promising Industry 4.0 technology that could help reach the SDGs is decision-making processes like machine learning [1]. One such machine learning method, namely Reinforcement Learning (RL), most of all approaches based on Q-learning, showed promising results [4, 5]. Due to its simplicity, Q-learning is one of the most popular Reinforcement-Learning algorithms [6]. Yet there is hardly any research on the implementation of Q-learning in HVACs control concerning industrial settings. Thus, the objective of this paper is to provide a case study on a Q-learning based control strategy for an industrial chiller, which provides cooling capacity for two warehouses. A real HVAC serves as a model for the environment.

*Corresponding author tel. no: +49 3375508442
Email address: constantin.falk@th-wildau.de (Constantin Falk)

2. Related Work

For the development of a suitable RL based control strategy for industrial HVAC, this section aims at providing an overview of scientific contributions in recent years. Concerning the efficient control of cooling applications, many methods and models have been developed. Hovgaard et al. developed an economic-optimizing Model Predictive Control (MPC) scheme that reduces operating costs by utilizing thermal storage capabilities [7, 8]. Applying their proposed MPC controller on a simulation of a supermarket refrigeration system, it was shown that potentially savings up to 9-32 % could be achieved using thermal storage capacities in combination with prediction of varying loads and energy prices. In [9], a MPC with a least square tracking error criterion to solve and optimize a power balancing problem with flexible thermal storage units was presented. Their power balancing aggregator takes regulating power prices into account and they showed that the constraints and objectives for each unit are satisfied. In [10], a multiple nonlinear regression (MNR) model is used to predict the hourly cooling load in a library. They found that the MNR improved the accuracy of the predicted load, which is important for saving energy while operating the HVAC.

Several works have compared RL to conventional control strategies like Chiller-priority or Storage-priority control and MPC [11, 12]. Henze et al. stated that the RL controller does not rely on physical models of building energy systems and the environment [11]. It worked without the need for prediction and it was able to tune itself to the actual system, while complex learning tasks prolong the learning time. Beghi et al. addressed this issue partially in [13], in which a RL controller acted as a supervisory system and determined the setpoints for a local controller. They used domain knowledge to speed up the convergence of the learning algorithm. Schreiber et al. also addressed the issue of long training time, as well as in their following work [14, 15]. They trained a Deep Q-learning agent in advance for 10^5 interactions with a simulation of an admixing heater and then used it to control an injection heater and a throttle cooler. They found that after approximately 4200 interactions of online training the controller improved visibly and as a result the training time could be reduced. Liu et al. used three different techniques, namely Randomly Initialized Q-learning, Asynchronous Q-learning, Deep Q-learning in order to improve the learning speed and performance [4]. In addition, they stated that the Q-learning controller performs better than conventional control strategies (Chiller-priority) but is still outmatched by the MPC scheme. Overall, a large part of the literature deals with the implementation of different RL control schemes for commercial buildings. In regards to industrial applications Zhang et al. dealt with improving the efficiency of a refrigerating system by combining RL and a Coarse Model [16]. The coefficient of performance (COP) of the refrigeration system is used as the cost function. They found that the proposed algorithm exceeded the conventional conversion efficiency of the refrigeration system from the viewpoint of the average, although showing larger fluctuations. Li et al. used a Deep Reinforcement Learning (DRL) Framework to optimize a cooling application in an industrial setting [17]. Other approaches as in [18] are multi-agent DRL, which is shown to reduce thermal energy costs of Heating, Ventilation, and Air Conditioning (HVAC), compared to a Heuristic (HS) and Rule-Based Scheme (RS) with an ON/OFF policy. Chen et al. showed in a simulation, that a Q-learning based control strategy could achieve up to 23 % lower HVAC energy consumption, compared to a rule-based heuristic control [19]. Qiu et al. applied the mentioned RL algorithm for optimal chiller control for an office building [20]. They showed in their following work [21], that a Q-learning controller is also able to generate data with which data-driven chiller models can be trained and enhance the accuracy, generalization and robustness. Besides applying the algorithm in question on HVAC, in [5] Guo et al. proposes a Q-learning based RL controller for a district cooling energy plant which could achieve energy savings up to 8 %. The literature shows that the algorithm in question is more often applied for the air conditioning in office buildings. The authors suspect that because of the economic and safety risks that come with it, the literature concerning the investigation and use of Q-learning for the control of chiller in the industrial setting is scarce. Therefore, the objective of this work is presenting a case study with particular emphasizes on Q-learning for the control of a chiller for two industrial storage chambers.

3. Model

3.1. Principles

According to Kaelbling et al. Reinforcement Learning is based on an agent interacting with a given environment, which has a discrete set of states S [22]. The agent is able to perform an action a from a discrete set of actions A, in order to change the state s of the environment. As a result, the agent will receive a reward r, both potentially negative as well as positive, which is typically a boolean or real number. The agent's overall goal is to find a policy π, which maps states to actions in order to maximize the reward in the long run. RL is an iterative process, which can be distinguished by value iteration or policy iteration [6]. The investigated approach is characterized by value iteration with the objective of finding the optimal value function. According to [22] the optimal value function is defined as

$$V^*(s) = \max_a (R(s,a) + \gamma \sum_{s' \in S} T(s,a,s') V^*(s')), \forall s \in S, \quad (1)$$

where $\gamma \rightarrow [0,1]$ is the discount factor, which defines whether the agent should favor immediate rewards (γ=0) over long term rewards (γ=1). *T(s,a,s')* is the state transition function, which is the probability of making a transition from one state s to a next state s' using the action a. *R(s, a)* is the reinforcement function, which specifies the expected instantaneous reward as a function of the current state and action. Kaelbling et al. stated that $V^*(s)$ asserts that the value of a state s is the expected instantaneous reward plus the expected discounted value of the next state $V^*(s')$ using the best available action [22]. As a result given the optimal value function, the optimal policy can be

specified as

$$\pi^*(s) = \underset{a}{\operatorname{argmax}}(R(s,a) + \gamma \sum_{s' \in S} T(s,a,s')V^*(s')). \qquad (2)$$

According to Vazquez et al. RL can be further differentiated between the model-based and model-free approach [6]. They stated that Q-learning is the most used model-free RL algorithm worldwide, due to its simplicity. Kaelbling et al. stated that in the model-free approach a Markov Decision Process (MDP) model is not known [22]. According to White et al. additionally to the states and actions, a MDP Model consist of *T(s,a,s')* and *R(s, a)* [23]. White et al. indicated that a model is MDP if the transitions from one state to another depends only on the current state and are therefore independent of any previous states or agent actions [23]. Kaelbling et al. argued that the agent has to therefore obtain the optimal policy without knowledge of the mentioned MDP-Model [22]. This means that the agent has to directly interact with the environment in order to obtain and process information to produce the optimal policy.

Q-learning was first proposed by Watkins et al. in with the objective to determine an optimal policy π^* without knowing the MDP Model [24]. To achieve this task, a matrix of states by actions is built, in which the state-action values or Q-values are initialized. These Q-values are updated each iteration by using the Q-function. Vasquez et al. stated that the updating rule is defined as

$$Q(s,a) := Q(s,a) + \alpha(r + \gamma \max_{a'} Q(s'a') - Q(s,a)) \qquad (3)$$

where $\alpha \in [0,1]$ is the learning rate, which defines to what degree new knowledge overrides old knowledge [6]. For α=0, no learning occurs and for α=1 all prior knowledge is overriden. In order to update the Q-value, the difference between the previous Q-value *Q(s,a)* and the discounted next maximum Q-value *Q(s',a')* is formed and added to the reward *r*. This difference is then multiplied with the learning rate α and added to the previous Q-value. Updating the Q-value results in the enviroment entering a new state, in which the agent has to select the next action. For this selection a trade-off between exploration and exploitation, which is also called action-selection is made. Vazquez et al. state, that in the exploration phase the agent chooses actions at random, inversely in the exploitation phase the action with the highest Q-Value is chosen and that for managing this trade-off the ε-greedy policy can be applied [6]. It consits of taking the action with the greatest Q-value with the probability (1-ε), and selecting a random action with the probability ε. Overall the Q-values converge to the optimal Q-value $Q^*(s,a)$, which, as shown in [25], results in convergence to the optimal policy.

Reinforcement learning approaches are dependant on a learning environment. In some situations, the agent's intended use cases may fit as learning environment, for example when an agent learns playing a video game. However, in most use cases an agent should learn the consequences of its own possible actions in a simulated environment for mainly two reasons: firstly, to avoid harm or damage of entities in the real-word and, secondly, to accelerate the learning process itself. On the other hand, the biggest drawback of a simulation, however, is the need to create an environment that well represents real-world conditions, as this constitutes a non-negligable cost and time factor.

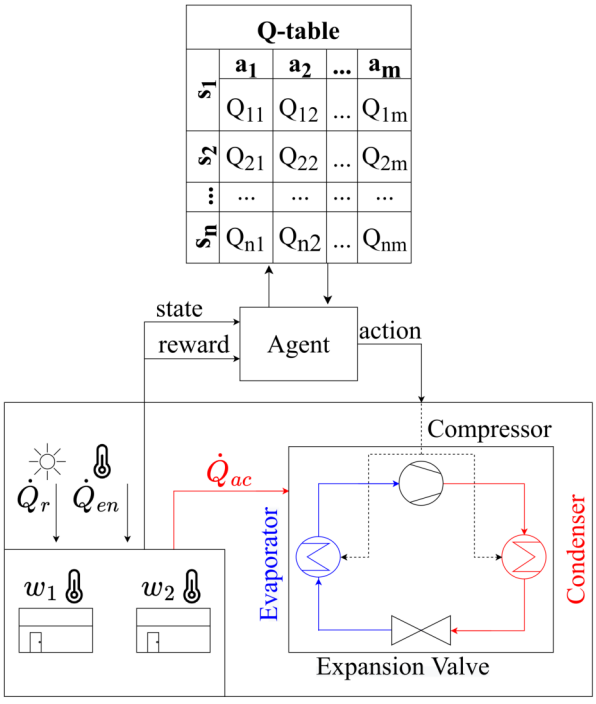

Figure 1. Case Study

3.2. Case Study

This work presents a case study for a Q-Learning based HVAC controlling strategy. Figure 1 shows a schematic representation on how such a controller could be realized. According to this representation the agent interacts with an environment, which entails the HVAC and both warehouses w_1 and w_2, through its actions and feedback in the form of states and rewards. Through its actions it is able to manipulate the screw compressor speed, cold-water feed-pump as well as the selection of which warehouse should be cooled. Through a feedback loop the agent is informed about the states of the environment and based on the resulting parameters a reward is calculated and fed back to the agent.

3.3. Environment

An ammonia-based vapor compression refrigeration system (VCRS) composed of three open screw compressors, refrigerant-, cold-water- and cooling-water circuit is adopted as case system. This system is used as a reference for tuning the model parameters based on real plant data. The cold water circuit supplies refrigeration energy to two separate warehouses, cooling them to different target temperatures. The refrigerant condensates whether in a water cooled condensator or in an air-water condensator, depending on the outdoor temperature and if heating of one of the warehouses is necessary. If heating of one of the warehouses is required, the water-circuit heat-exchanger is

used, while the heat pump can be switched on if not enough heat is available. The condensed refrigerant is fed from both condensers to the separator via throttle valves. If heat energy is required, the heat pump uses the cooled, liquid refrigerant to return it to the separator in a gaseous state. An additional deep freezing circuit powered by two reciprocating compressors utilizes the cold provided by the cold-water circuit. Both water circuits contain a water-glycol mixture, wherein the ciculation is realized by two pumps each. Two of the three screw compressors and all four water pumps are speed-regulated by use of frequency converters.

Matlab with the Simulink-package has been used, to build a computerized model of the given system, since Simulink is easy to use and well suited for creating physical simulation. Following [19], the system is simplified into the following Equation 4, while the structure of the model can be traced in a simplified version from Figure 2, where the input represents all external variables or constants which are not conditioned by the system itself, while not every block uses each of the available values.

$$\dot{Q}_{total} = \dot{Q}_{en} + \dot{Q}_r + \dot{Q}_{HX} \tag{4}$$

$\dot{Q}_{total}$ represents the total heat gain for the perspected room and is composed of the environmental heat gain $\dot{Q}_{en}$, the solar radiation $\dot{Q}_r$ and the cooling energy from the air conditioning $\dot{Q}_{ac}$. The environmental heat gain is the thermal energy gained through the building envelope and is the sum of the heat gain of all walls of the considered warehouse. It depends on the conductance of the wall h_i, surface area A_i of each external wall i and on the temperature difference between the inside and outside environments ($T_{out} - T_{in}$).

$$\dot{Q}_{en} = \sum_{i=0}^{n} h_i A_i (T_{out} - T_{in}) \tag{5}$$

The heat gain through solar radiation depends on the actual radiation per m² and the surface area of each external wall/ceiling of the warehouses that are exposed to the sun over a day. Depending on the position of the sun, the solar radiation per wall is determined approximately [26].

The HVAC itself gets simulated by a combination of regressions and simplifications. The cooling energy $\dot{Q}_{ac}$ gets neared with a second degree polynomial regression utilizing the compressors revolutions-per-minute (RPM), cooling-water and cold-water feed pumps RPM as well as the condenser inlet temperature and evaporator inlet temperature, based on data of the real plant. The model is allowed to cool only one of two warehouses simultaneously. However, this limitation is also present at the real plant, so that the warehouses must always be cooled one after the other. Utilizing the cooling energy $\dot{Q}_{ac}$ together with the evaporator inlet temperature T_{ei} and the mass-flow rate of the water-glycol mixture in the evaporator $\dot{m}_{fe}$ leads to the evaporator outlet temperature

$$T_{eo} = T_{ei} - \frac{\dot{Q}_{ac}}{\dot{m}_{fe} \cdot c_p} \tag{6}$$

where $\dot{m}_{fe}$ is approximated by use of a second-degree polynomial regression using the RPM of the feed-pump [27, p. 356f.]. For reasons of simplicity, the specific heat capacity of water c_p is used for the water-glycol mixture throughout this work and the heat loss to the environment is neglected. The resulting T_{eo} is used for the calculation of the heat gain in the heat exchangers in the warehouses, which also requires the corresponding warehouse temperatures T_{w1} and T_{w2} as well as m_{fe}. Furthermore, the chilled water either flows into w_1 or into w_2, depending on the actual set warehouse to cool. In this way, one of the warehouses always receives $m_{fe} = 0$ and the corresponding heat-flow is also $\dot{Q}_{HX} = 0$. However, the heat flow is given by

$$\dot{Q}_{HX} = k \cdot A \cdot \Delta t_m \tag{7}$$

where k represents the heat transfer coefficient of the heat exchanger, while A is the surface and Δt_m the logarithmic temperature difference in the heat exchanger. Another simplification was made, by using the water output temperature from the warehouse's heat exchanger as evaporator inlet temperature T_{ei}. In the same way, as $\dot{m}_{fe}$ only flows into one the warehouses, T_{ei} is used from the active warehouse to lose the loop. The heat-flow of the heat exchanger gets summed up with the other heat-flows for the corresponding warehouse to a total heat-flow. Using an estimated constant warehouse cooling capacity C_w and the total heat-flow leads to the warehouse temperature difference ΔT_w in a simulation step with a given duration.

For simplification purposes, only the heat-flows and temperatures are displayed in Figure 2. The estimation of the electrical power P_{total} utilizes the RPM of all compressor stages, feed-pumps, the outside temperature, T_{eo} and the condenser outlet temperature T_{co}. T_{co} is the only input of P_{total} which is not constant, an input variable of the model or already calculated. The calculation is similar to T_{eo} and is accordingly based on the condenser inlet temperature T_{ci}, the mass-flow rate of cooling-water $\dot{m}_{fc}$ and the condenser heat-flow $\dot{Q}_C$. While $\dot{m}_{fc}$ is also regression based on the actual RPM of the water feed-pumps, $\dot{Q}_C$ is simplified the sum of $\dot{Q}_{ac}$ and P_{total}. T_{ci} itself is also a regression-based value, which is approximated by P_{total} and the outside temperature.

Summing up the heat flows leads to the total heat flow of the respective warehouse as shown in Equation 4, with which the resulting temperature difference can be calculated for a considered time step. Therefore, both warehouses utilize an estimated thermal capacity C_W as constant input.

4. Validation

4.1. Simulation

The model presented in subsection 3.3 gets simulated in Matlab's Simulink environment. Thus, the agent, current state detection and the corresponding selection of the action based on the policy take place externally. To find a subset of states S_t which represents the most important system conditions, the most important influencing variables needs to be discretized. The actual state gets defined by the evaporator-inlet temperatur

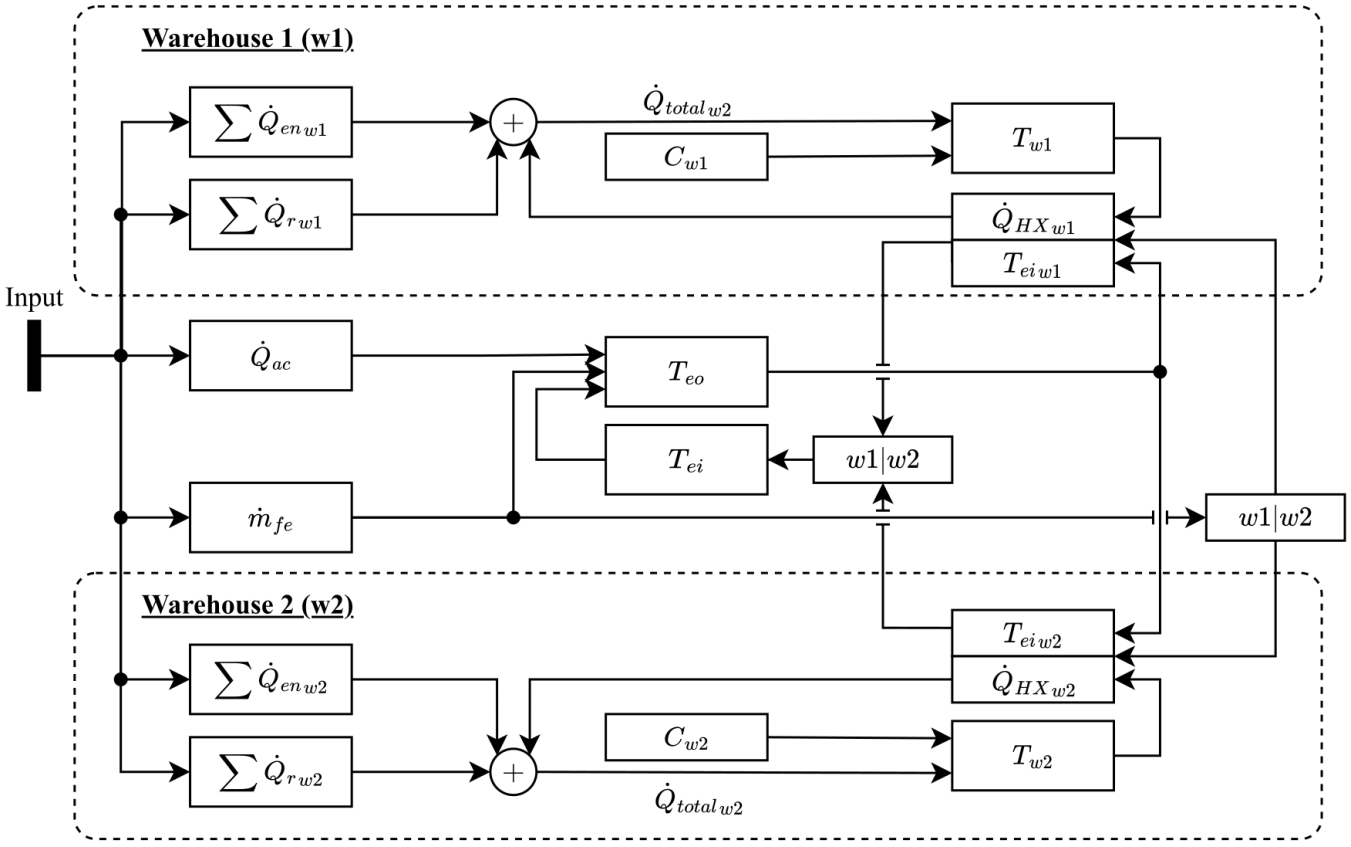

Figure 2. Simplified functionality of the model

T_{ei}, condenser inlet temperature T_{ci}, both warehouse temperatur T_{w1} and T_{w2}, outside temperature T_{out} and the actual sun intensity $p_{sun} \in [0, 1]$. The state space along

$$S_t = \begin{pmatrix} T_{ei} \\ T_{ci} \\ T_{w1} \\ T_{w2} \\ T_{out} \\ p_{sun} \end{pmatrix} \tag{8}$$

with three discretization steps for each variable can be seen in Equation 8, while the action space is

$$A_t = \begin{pmatrix} n_c \\ n_{cwp} \\ w \end{pmatrix} = \begin{cases} 0; 3000; 6000 \\ 0; 2944; 5888 \\ w_1; w_2 \end{cases} \tag{9}$$

The action space includes the screw compressor speed n_c, cold-water feed-pump speed n_{cwp} and the selection of the warehouse w to be cooled. The Q-table represents the complete State-Action-Space following $Q = S \times A$.

It is worth noting, that the actual cardinality of action-state-pairs directly depends the time of convergence to an optimal policy. Increasing the number of states or actions to increase the achievable policy precision, simultaneously increases the time required to achieve convergence [13]. Thus, the presented cardinality is a tradeoff between precision and processing time, with the available computing power having a corresponding effect on speed.

Because the agent initially has to observe the environment, with the look-up table $Q(s, a)$ initialized to zero, it's interactions are divided into exploration- and exploitation-mode. From earlier experiments it can be deduced that the learning rate α should be initialized with 0.7 in exploration-mode and decreases to 0.125 in exploitation. The discount factor γ is set to 0.7 to provide the agents with a solid foresight. Following the ε-greedy policy introduced in subsection 3.1, ε= 0.9 in exploration-mode, such that a random action is chosen 9 times out of 10 in the current state. In exploitation-mode, on the other hand, ε= 0.1. Both, α and ϵ, are not set to 0 to ensure that it is possible to react to possible changes in the environment even during operation.

The reward r_t represents the system performance and depends on the system's power consumption P_{total} as well as on the actual warehouse temperatures T_w. Different states use different formulas to calculate rewards, with the respective conditions, derived from Beghi et al., defined in Equation 10 and the reward calculation in Equation 11 [13].

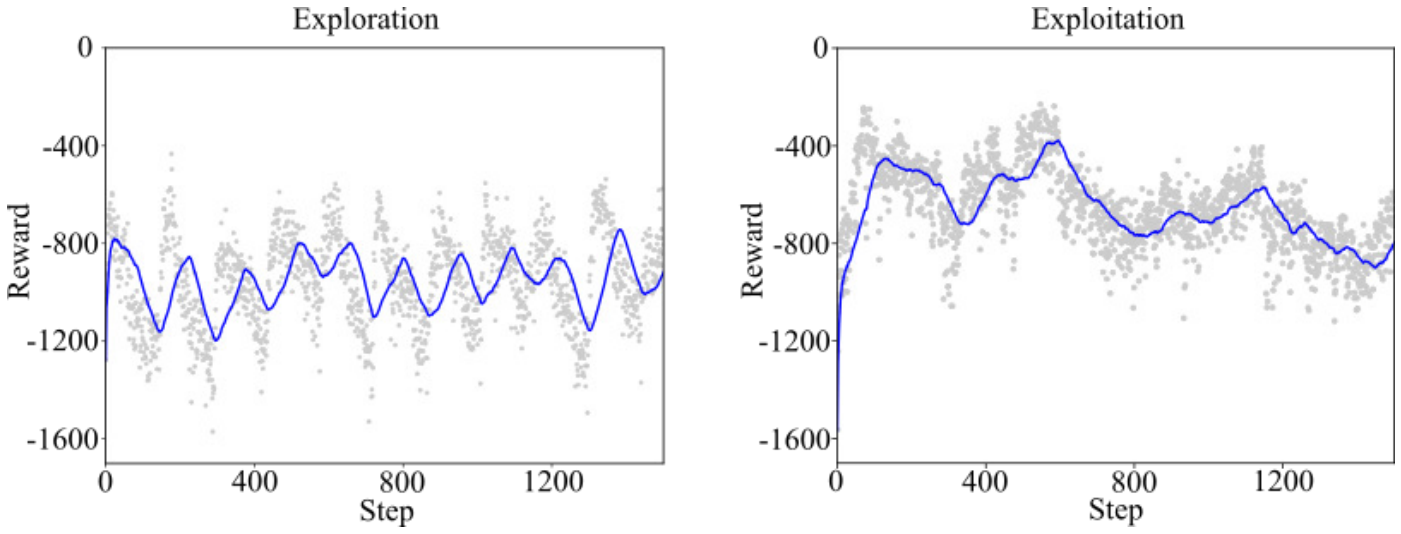

Figure 3. Agent's rewards in exploration and exploitation mode

$$
\begin{aligned}
c1 &= \{T_w | T_w < T_{wmin} \wedge T_w > T_{wmax}\} \\
c2 &= \{T_w | T_w \leq T_{wmin} + \Delta T_{set} \wedge T_w \geq T_{wmax} - \Delta T_{set}\} \\
c3 &= \{T_w | T_{wmin} + \Delta T_{set} < T_w < T_{wmax} - \Delta T_{set}\}
\end{aligned} \tag{10}
$$

$$
r_{tw} = \begin{cases} -1000 - 50 \cdot \Delta T_{min} & ; \text{if } c1 \\ 100 - 50 \cdot \frac{\Delta T_{set}}{\Delta T_{min}} & ; \text{if } c2 \\ 100 & ; \text{if } c3 \end{cases} \tag{11}
$$

$$
r_{tP} = -j \cdot P_{total}
$$

$$
r_t = r_{tw1} + r_{tw2} + r_{tP}
$$

A separate reward r_{tw} is calculated for both warehouse temperatures and depends if the actual T_w whether is outside the specified warehouse temperature, within the safety margin of limiting temperatures or within all limits. Furthermore, ΔT_{min} specifies the minimal deviation from the warehouse limits T_{wmin} and T_{wmax}, while ΔT_{set} is an user-defined safety margin that shall not be exceeded as this might cause unintended operating conditions or even damage to the system. Temperatures outside the given limits will be always rewarded negative, while temperatures in the safety margin are tolerated and rewarded depending on the distance to the limits. Temperatures inside all limits are always rewarded constantly good to avoid influencing the agent's temperature set-point. Rewards of the actual power consumption r_{tP} are always negative, while it's influence on the total reward can be adjusted by the weight factor j.

In general, one simulation step represents 10 minutes. To speed up the exploration of state-actions, the simulation gets parallelized, such that multiple environments are simulated simultaneously step-by-step, while the Q-table is updated after every step with updates from all environments. The parameters of the environments are instantiated randomly with orientation to the discretization steps. As a consequence, it is possible for the system to start outside the specified limits in some environments.

4.2. Results

To ensure comparability of the results, a reference 2-step-controller has been implemented, similar to the real plant, and simulated on the presented model. The 2-step-controller works like a real plant, by cooling one room after the other using the cold-water feed-pumps if necessary and only turning on the compressors, if the water temperature T_{ei} is too high.

After 1500 steps of exploration, the agent switches into exploitation mode and uses the filled Q-table to control the plant properly. Figure 3 displays the averaged reward earned by the parallel working agents per simulation step. It can be seen, that in exploration mode the rewards are very unsteady with a conspicuousness in the recurring reward peaks. This is due to the fact that all environments have been re-instantiated after a simulated day, so they are mostly within the given system limits (see Equation 11) and accordingly receive rewards for staying within the boundaries at the beginning. While exploring the environment, most agents maneuver the plant out of the target range, earning negative rewards.

It can be seen, that even in exploitation-mode averaged rewards from all environments are always negative. This can be attributed to the big impact of the actual power consumption on the rewards, since the weight factor was instantiated as $j = 4$. It is also noticeable that the average rewards achieved decreased over the further steps. This indicates that some agents utilized the margin of the temperature limits, which in turn can be attributed to agents acting too short-sightedly.

Figure 4 displays the average electrical energy consumption of the refrigeration system in different simulation environments. Therefore, a set of 30 simulation environments were instantiated and controlled on the one hand by the Q-learning agent and on the other hand by the reference 2-step controller. The set of environments includes all seasons with different weather conditions and outdoor temperatures. In about 10 simulated days, the 2-point controls consumed on average about 450 kWh more than the Q-learning based controls. The course of the difference over the simulated period can be seen in the line chart.

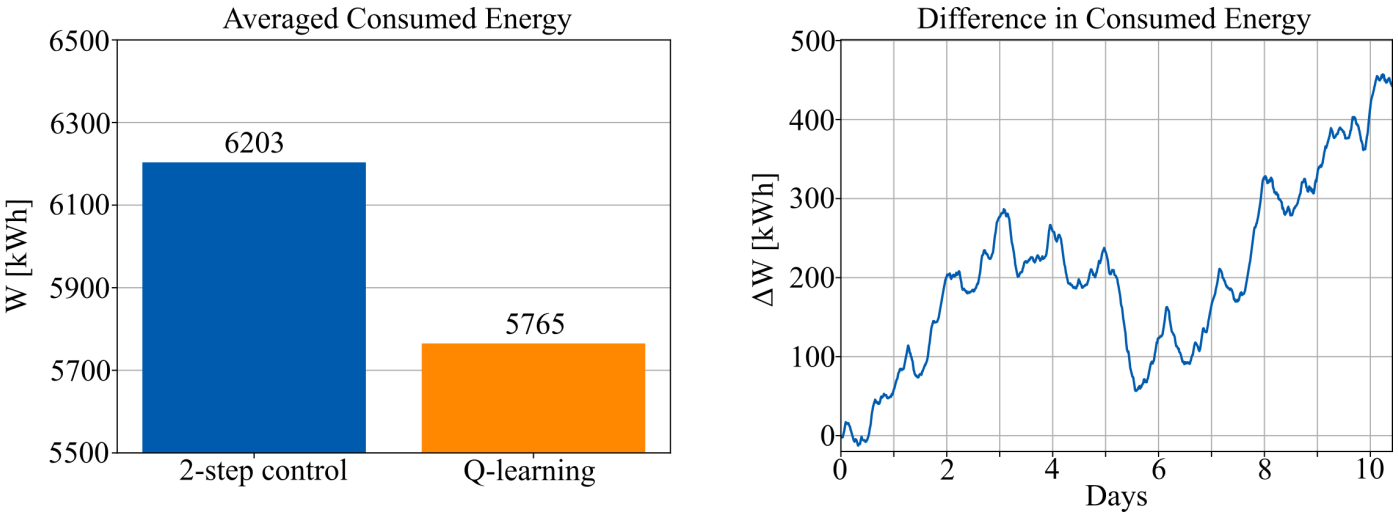

Figure 4. Comparison of averaged energy consumption with (left) the average energy consumption and (right) the difference between the 2-step-control and the Q-learning approach

In summary, the refrigeration plant controlled by the Q-learning approach consumes an average of 5.765 kWh in the 250 hours (1500 steps), while that controlled by the 2-step controller requires 6.203 kWh. This results in energy savings of about 7 % using the Q-learning approach compared to the conventional control.

The results demonstrate that the chosen approach is functioning, although performance can still be improved. Especially the consistent compliance with the temperature limits should be focused by more far-sighted control in a follow-up version. However, the approach also shows a lot of potential, as the environmental model created so far does neither simulate load behaviour in warehouses nor weather forecasts. A promising approach to predict selected meteorological parameters was proposed in [28] and could be considered in future research. These are recurring characteristics or dependencies, which a RL-approach can learn very well and derive optimal control behaviors accordingly. Furthermore, the approach could be improved by extending the action space to increase the agent's scope, while also requiring greater computational power. Another variant could be to apply a Deep-Q-learning algorithm, as already successfully implemented by Yu. et al. in a commercial building, where no states or actions would have to be discretized [29].

5. Conclusion

In this paper, we conducted a case study to examine potentials in energy efficiency of Q-learning in cooling applications. For this purpose, a model based on a real refrigeration plant including warehouses and environmental influences was implemented and used as training environment for the Q-learning algorithm. The simulations cover different environmental influences typical for German weather conditions including different seasons. The controller is able to perform actions with predetermined parameters and receives a reward as feedback as well as the new state of the system, which allows the controller learning the behavior of the system directly through the performed actions. The results confirm that Q-learning is a suitable approach to derive a control policy to optimize energy consumption. However, it was also shown that constraints such as discretized state and action spaces or simplified models limit the possibilities of Q-learning. Future works should investigate the benefits of reinforcement learning on recurrent load behaviors compared to conventional controls, as well as identify potential benefits of deep Q-learning application on refrigeration systems.

Acknowledgements

This research was supported by the Ministry for Economic Affairs, Labour and Energy, State of Brandenburg (No: 80237615). We would also like to thank our partner Potsdamer Anlagenbau und Kältetechnik GmbH for their great support in setting up the test facility as well as during the commissioning phase.

References

[1] M. M. Mabkhot et al., "Mapping industry 4.0 enabling technologies into united nations sustainability development goals", Sustainability **13** (2021) 2560.

[2] VDMA e.V. Allgemeine Lufttechnik, "Energiebedarf für Kältetechnik in Deutschland: Eine Abschätzung des Energiebedarfs von Kältetechnik in Deutschland nach Einsatzgebieten", (2017).

[3] K. Mason & S. Grijalva, "A review of reinforcement learning for autonomous building energy management", Computers Electrical Engineering **78** (2019) 300.

[4] S. Liu & G. P. Henze, "Evaluation of Reinforcement Learning for Optimal Control of Building Active and Passive Thermal Storage Inventory", Journal of Solar Energy Engineering **129** (2006) 215.

[5] Z. Guo, A. R. Coffman & P. Barooah, "Reinforcement Learning for Optimal Control of a District Cooling Energy Plant", 2022 American Control Conference (ACC) (2022) 3329.

[6] J. R. Vázquez-Canteli & Z. Nagy, "Reinforcement learning for demand response: A review of algorithms and modeling techniques", Applied Energy **235** (2019) 1072.

[7] T. G. Hovgaard, L. F. S. Larsen & J. B. Jorgensen, "Flexible and cost efficient power consumption using economic MPC a supermarket refrigeration benchmark", 2011 50th IEEE Conference on Decision and Control and European Control Conference (2011) 848.
[8] T. G. Hovgaard, L. F. Larsen, K. Edlund & J. B. Jørgensen, "Model predictive control technologies for efficient and flexible power consumption in refrigeration systems", Energy **44**((2012) 105.
[9] R. Halvgaard, L. Vandenberghe, N. K. Poulsen, H. Madsen & J. B. Jorgensen, "Distributed Model Predictive Control for Smart Energy Systems", IEEE Transactions on Smart Grid **7** (2016) 1675.
[10] C. Fan & Y. Ding, "Cooling load prediction and optimal operation of HVAC systems using a multiple nonlinear regression model", Energy and Buildings **197** (2019) 7.
[11] G. Henze & J. Schoenmann, "Evaluation of Reinforcement Learning Control for Thermal Energy Storage Systems", HVAC&R Research **9** (2003) 259.
[12] D. Ernst, M. Glavic, F. Capitanescu & L. Wehenkel, "Reinforcement learning versus model predictive control: a comparison on a power system problem", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) **39** (2009) 517.
[13] A. Beghi, M. Rampazzo & S. Zorzi, "Reinforcement Learning Control of Transcritical Carbon Dioxide Supermarket Refrigeration Systems", IFAC-PapersOnLine **50** (2017) 13754.
[14] T. Schreiber, S. Eschweiler, M. Baranski & D. Müller, "Application of two promising Reinforcement Learning algorithms for load shifting in a cooling supply system", Energy and Buildings **229** (2020) 110490.
[15] T. Schreiber, A. Schwartz & D. Muller, "Towards an intelligent HVAC system automation using Reinforcement Learning", Journal of Physics: Conference Series **2042** (2021) 012028.
[16] D. Zhang & Z. Gao, "Improvement of Refrigeration Efficiency by Combining Reinforcement Learning with a Coarse Model", Processes **7** (2019) 967.
[17] Y. Li, Y. Wen, D. Tao & K. Guan, "Transforming Cooling Optimization for Green Data Center via Deep Reinforcement Learning", IEEE transactions on cybernetics **50** (2020) 2002.
[18] L. Yu, Y. Sun, Z. Xu, C. Shen, D. Yue, T. Jiang & X. Guan, "Multi-Agent Deep Reinforcement Learning for HVAC Control in Commercial Buildings", IEEE Transactions on Smart Grid **12** (2021) 407.
[19] Y. Chen, L. K. Norford, H. W. Samuelson & A. Malkawi, "Optimal control of HVAC and window systems for natural ventilation through reinforcement learning", Energy and Buildings **169** (2018) 195.
[20] S. Qiu, Z. Li, Z. Li & X. Zhang, "Model-free optimal chiller loading method based on Q-learning", Science and Technology for the Built Environment **26** (2020) 1100.
[21] S. Qiu, Z. Li, R. He, J. Li, and Z. Li, "How does the control logic influence the establishment of a data-driven chiller model?", Journal of Physics: Conference Series **2006** (2021) 012002.
[22] L. P. Kaelbling, M. L. Littman & A. W. Moore, "Reinforcement learning: A survey", Journal of artificial intelligence research **4** (1996) 237.
[23] C. C. White & D. J. White, "Markov decision processes", European Journal of Operational Research **39** (1989) 1.
[24] C. J. C. H. Watkins, *Learning from delayed rewards* (1989).
[25] C. J. C. H. Watkins & P. Dayan, "Q-learning", Machine Learning **8** (1992) 279.
[26] S. A. Khalil, "Performance Evaluation and Statistical Analysis of Solar Energy Modeling: A Review and Case Study", Journal of the Nigerian Society of Physical Sciences **4** (2022) 911.
[27] G. Cerbe & G. Wilhelms, "Technische Thermodynamik: Theoretische Grundlagen und praktische Anwendungen", Carl Hanser Verlag GmbH Co KG (2021).
[28] F. O. Aweda, J. A. Akinpelu, T. K. Samson, M. Sanni & B. S. Olatinwo, "Modeling and Forecasting Selected Meteorological Parameters for the Environmental Awareness in Sub-Sahel West Africa Stations", Journal of the Nigerian Society of Physical Sciences **4** (2022) 820.
[29] K. H. Yu, Y. A. Chen, E. Jaimes, W. C. Wu, K. K. Liao, J. C. Liao, K. C. Lu, W. J. Sheu & C. C. Wang, "Optimization of thermal comfort, indoor quality, and energy-saving in campus classroom through deep Q learning", Case Studies in Thermal Engineering **24** (2021) 100842.

Appendix

A, A_t	set of actions
A_i	surface area of the external wall
a	action
a'	next action
C_W	warehouse cooling capacity
c_p	capacity of water
h_i	conductance of the wall
j	weight factor
k	heat transfer coefficient of the heat exchanger
$\dot{m}_{fe}$	mass-flow rate of the water-glycol mixture in the evaporator
$\dot{m}_{fc}$	mass-flow rate of cooling-water
n_c	screw compressor speed
n_{cwp}	cold-water feed-pump speed
P_{total}	electrical power
p_{sun}	sun intensity
$Q(s,a)$	Q-value
$Q^*(s,a)$	optimal Q-value
$\dot{Q}_{ac}$	cooling energy from air conditioning
$\dot{Q}_C$	conderser heat-flow
$\dot{Q}_{en}$	enviromental heat gain
$\dot{Q}_{HX}$	heat flow
$\dot{Q}_r$	solar radiation
$\dot{Q}_{total}$	total heat gain
$R(s, a)$	reinforcement function
r, r_t	reward
r_{tP}	Rewards of the actual power consumption
r_{tw}	separate reward
S, S_t	Set of states
s	state
s'	next state
$T(s,a,s')$	transition function
T_{ci}	condenser inlet temperature
T_{co}	condenser outlet temperature
T_{ei}	evaporator inlet temperature
T_{eo}	evaporator outlet temperature
T_{in}	inside temperature
T_{out}	ouside temperature
T_w	warehouse temperature
T_{wmin}, T_{wmax}	warehouse temperature limits
$V^*(s')$	value function
w	warehouse
α	learning rate
γ	discount factor
ΔT_{set}	safety margin
Δt_m	logarithmic temperature difference in the heat exchanger
π	policy
π^*	optimal policy