Universiteit
Leiden
The Netherlands

Deep Reinforcement Learning for Ventilation System Control in Pig Buildings

Zheng, Shuwen

Citation
Zheng, S. (2023). *Deep Reinforcement Learning for Ventilation System Control in Pig Buildings*.

Version:	Not Applicable (or Unknown)
License:	License to inclusion and publication of a Bachelor or Master Thesis, 2023
Downloaded from:	https://hdl.handle.net/1887/3642103

Note: To cite this publication please use the final published version (if applicable).

Universiteit
Leiden
The Netherlands

Deep Reinforcement Learning for Ventilation System Control in Pig Buildings

Shuwen Zheng

Thesis advisor: Dr. Xiaodong Cheng, Wageningen University & Research

Thesis advisor: Dr. Congcong Sun, Wageningen University & Research

Defended on September, 2023

MASTER THESIS

STATISTICS AND DATA SCIENCE

UNIVERSITEIT LEIDEN

Abstract

Indoor thermal environment plays an important role in pigs' health, welfare, production, and reproduction, and ventilation system focuses on ensuring the comfort of animals by using exhaust fans and air inlets commonly in livestock buildings. Model-free method Deep Reinforcement Learning(DRL), well known for the performance in game-playing and robotics control, recently has applied in buildings heating, ventilation, and air conditioning (HVAC) systems control. This study explored the effectiveness of a DRL algorithm, Deep Q-Network(DQN), in ventilation system control for pig buildings. The results showed that the DQN agent managed to maintain the room temperature within the comfortable range in 99.13%, 90.1%, 92.01% of test days in winter, spring and summer, respectively. The DQN agent outperformed the baseline method with the saving of power consumption by 17.66%, 30.04%, 6.89% in the test days of winter, spring and summer, respectively. The DQN algorithm applied the same neural network architecture and hyperparameter settings and was trained and tested in different periods of time, indicating the generalization capability of the DQN algorithm.

Keywords: Deep Reinforcement Learning (DRL), Pig Building, Ventilation system control, Thermal Discomfort, Energy Consumption

Contents

Chapter 1

Introduction

One of the most significant livestock industries worldwide is pig production. Pork made up around 32% of all meat production (FAO, 2022 [1]) and more over a quarter of the total protein consumed globally (Bruinsma, 2003 [2]). The expected growth in the world population and rising incomes in developing countries will likely increase the demand for meat and animal protein (United Nations, 2019 [3]). In China, about 53.8% of total meat products consumed were pork, and over the last decade, the market share of stocking of large-scale pig farms has risen from 20% to 60-80% (National Bureau of Statistics, PRC, 2021 [4]).

Indoor thermal environment plays an important role in pigs' health (Carroll et al., 2012 [5]), welfare (Huynh et al., 2005 [6]), production (Baxter et al., 2015 [7]), and reproduction (Zhao et al., 2015 [8]). When pigs are in thermal equilibrium, their body temperature remains constant, and when the environment is in pigs' comfort zone, their production performance and growth rate are at their peak (Renaudeau et al., 2012 [9]). Pigs would begin a thermal regulation system, such as limiting feed intake, which slows growth, causes heat stress that can harm their health or even result in mortality, if the environment got hot and humid (Gonçalves de Oliveira et al., 2021 [10]; Lucas et al., 2000 [11]).

Livestock ventilation focuses on ensuring the comfort of animals by considering their welfare, behavior, and health, and was related to conversion ratio, growth

rate, and mortality of animals (Clark, 1981 [12]). The primary goal of a ventilation system is to provide sufficient oxygen, eliminate moisture and odors, prevent heat accumulation, and reduce the concentration of air-borne disease-causing organisms. By regulating the exchange rate of air and the pattern of airflow, the optimal livestock indoor environment can be maintained, ensuring thermal comfort (based on temperature) and indoor air quality (based on contaminant gas concentration) within the ventilated structure (Tan and Zhang, 2004 [13]).

Traditionally, ventilation can be achieved through natural or mechanical methods. Natural ventilation relies on natural forces like thermal buoyancy and wind flow, but its effectiveness depends on the building design and it has limited use due to its passive nature and uncertain performance results. Mechanical ventilation controls air temperature and air movement using fans, thermostats, and air inlets. The most common method of mechanical ventilation is to use fans to exhaust air out of the livestock building, while fresh air is drawn in through inlets. Mechanical ventilation can be designed independently of the building and allows for flexibility in modification, but it is costly in terms of energy consumption.

Afram and Janabi-Sharifi (2014) [14] comprehensively reviewed the control techniques in heating, ventilation, and air conditioning (HVAC) systems, including classical control methods (i.e., on/off, P, PI and PID control), hard control methods (i.e., model predictive control (MPC), optimal control, robust control, nonlinear control and gain scheduling control), soft control methods (i.e., fuzzy logic (FL) control and neural network (NN) control), hybrid control methods (i.e., adaptive neuro, adaptive fuzzy, and fuzzy PID control), and other control methods (i.e., reinforcement learning (RL) control, two parameter switching control, preview control, pattern recognition adaptive controller, pulse modulation adaptive controller, direct feedback linear control).

Several control techniques, for example MPC, FL, and NN, have been applied on livestock buildings indoor climate control. Wu et al. (2006) [15] designed an MPC strategy for the hybrid ventilation systems and indoor climate of poultry barns, and applied thermal comfort parameters and a multi-zone method to develop a dynamic model which described the nonlinearity of ventilation and indoor climate. Yang et al. (2009) [16] applied a single-zone model to develop an opti-

mal control for the indoor climate of a large-sized livestock stable, and used the energy balance and the mass balance principle to simulate the thermal dynamic of the environment. Li et al. (2015) [17] used an MPC approach to control the CO2 concentration, temperature, and wind velocity in coops, and pointed out that the approach was advantageous for stabilizing the control of wind velocity-temperature-gas, and was able to forecast the trend of each variable. Mushtaq et al. (2016) [18] designed a FL controller by using FL based Mamdani model to produce the suitable temperature, humidity and air flow of a livestock shed. Gorczyca and Gebremedhin (2020) [19] highlighted that neural networks and random forests had the best accuracy among four machine learning algorithms in predicting the physiological responses of dairy cows, and revealed that the impact of air temperature on dairy cows' physiological responses ranked highest in environmental conditions. Lee et al. (2022) [20] developed recurrent neural network (RNN) models to predict the thermal and moisture environment in naturally and mechanically ventilated duck houses.

The use of Model-based Predictive Control (MPC) algorithm was gaining popularity in the agricultural building applications due to its efficient and flexible handling of system nonlinearities and constraints. However, MPC needs to use precise environmental models. While basic models have been created to estimate factors such as ammonia emissions in naturally ventilated livestock buildings, the intricacies and interrelationships between various environmental parameters make it challenging to fully understand the underlying mechanisms of these models. Reinforcement learning (RL) control have been investigated for the control of thermal energy storage in commercial buildings (Henze and Schoenmann, 2003 [21]; Liu and Henze, 2006 [22]), and deep reinforcement learning (DRL) have been studied for building HVAC control (Wei et al., 2017 [23]; Gao et al., 2019 [24]; Masburah et al., 2021 [25];Luo et al., 2022 [26]; Zheng, 2022 [27]). However, there has been little discussion on applications of (deep) reinforcement learning control for HVAC system in livestock buildings.

This thesis aimed to propose an efficient and effective ventilation system control using deep reinforcement learning, in order to improve livestock building indoor climatic conditions, increase animal welfare, and optimize energy consumption.

Chapter 2 introduced some theories of reinforcement learning and deep reinforcement learning. Chapter 3 described the mathematical model that can simulate the indoor thermal environment within a livestock building, the control problem formulation and the deep reinforcement learning algorithm. Chapter 4 elaborated the experiment setup and the experiment results. Chapter 5 summarized the conclusions, and discussed the limitations and future work, of this study. The application of reinforcement learning was the most innovative part of the work and had significant contribution, as it provided an alternative solution for the livestock building HVAC control problem.

Chapter 2

Background

This chapter included some theories of Reinforcement Learning and Deep Reinforcement Learning.

2.1 Reinforcement Learning

Reinforcement Learning (RL) is about an agent learning an optimal policy for decision making problems by interacting with the environment through trial and error and receiving negative or positive rewards as feedback for performing actions. [28]

2.1.1 Problem Setup

Figure 1 showed the RL process, which is also called a Markov Decison Process (MDP). At each time step t, the agent observes a state s_t in a state space S and takes an action a_t from an action space A, following a policy $\pi(a_t|s_t)$, which is the agent's brain, i.e., a mapping from state s_t to actions a_t telling the agent to select what action based on the given state. The agent gets a scalar reward r_t

from reward function $R(s, a)$, and the environment changes to the next state s_{t+1}, according to state transition probability $P(s_{t+1}|s_t, a_t)$. [28]

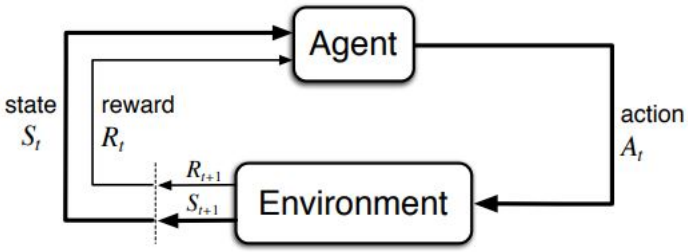


Figure 1: The RL process: a loop of state, action, reward and next state, reprinted from Sutton and Barto (2018) [29, Figure 3.1]

In an episodic problem, this process starts and ends when the agent reaches a terminal state in an episode. The return R_t defined as Equation 2.1 is the discounted cumulative reward, with the discount factor $\gamma \in (0, 1]$ indicating how much the agent values the long-term reward. The goal of the agent is to maximize the expected return from each state. [28]

$$R_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + ... = \sum_{k=0}^{\infty} \gamma^k r_{t+k} \tag{2.1}$$

2.1.2 Value Function

A value function is the expected, cumulative, discounted, future reward, implying how good a state s or a state-action pair (s, a) is. The state value function $V_\pi(s) = E[R_t|s_t = s]$ is the expected return for an agent starting at state s and following policy π for all time steps. According to the Bellman equation, $V_\pi(s)$ can be written as Equation 2.2. [28]

$$V_\pi(s) = E[r_t + \gamma V_\pi(s_{t+1} = s')|s_t = s] \tag{2.2}$$

The action value function $Q_\pi(s, a) = E[R_t|s_t = s, a_t = a]$ is the expected return for an agent starting at state s, choosing action a and then following policy π for all time steps. $Q_\pi(s, a)$ can be decomposed as Equation 2.3 according to the Bellman equation. [28]

$$Q_\pi(s, a) = E[r_t + \gamma Q_\pi(s_{t+1} = s', a_{t+1} = a')|s_t = s, a_t = a] \tag{2.3}$$

An optimal state value function $V^*(s) = max_\pi V_\pi(s) = max_a Q^*(s,a)$ is the maximum state value achieved for state s over all policies. An optimal action value function $Q^*(s,a) = max_\pi Q_\pi(s,a)$ is the maximum action value achieved for state s and action a over all policies. An optimal policy is denoted as π^*, and $\pi^*(s) = argmax_a Q^*(s,a)$ represents the link between the optimal policy and the optimal action value function. [28]

2.1.3 Temporal Difference Learning

Temporal difference (TD) learning is essential in RL, and it updates value function $V(s)$ at each step with TD target, which is an estimate of the expected return of an entire episode using bootstrapping. The update rule is shown in Equation 2.4,

$$V(s_t) \leftarrow V(s_t) + \alpha[r_t + \gamma V(s_{t+1}) - V(s_t)] \tag{2.4}$$

where α is a learning rate, $r_t + \gamma V(s_{t+1})$ is TD target, and $r_t + \gamma V(s_{t+1}) - V(s_t)$ is TD error. Precisely, this is TD(0) learning, i.e. one-step TD. [28]

TD learning is the learning strategy for value function update in Q-learning. Q-learning is an off-policy method which trains action value function $Q(s,a)$, i.e. Q function, to find the optimal policy. The Q function update rule in Q-learning is demonstrated in Equation 2.5,

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_t + \gamma max_{a'} Q(s_{t+1},a') - Q(s_t,a_t)] \tag{2.5}$$

where $max_{a'} Q(s_{t+1},a')$ means Q learning uses a greedy policy to select the highest state-action value for the next state. [28]

2.1.4 Value Function Approximation

Value function approximation is an approach to estimate the value function, when state and action spaces are very large or continuous, it's impractical to use the tabular method such as Q-learning which stores the state-action pair values in a table. The approximate of the action value function is parameterized with parameter vector θ as $Q(s,a;\theta)$.

Both linear functions or non-linear functions (for example neural networks) can be used as the function approximations with the parameters θ. Linear function approximates had been applied primarily in RL because they can converge. Deep neural networks, such as multilayer perceptrons (MLP), convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have recently been commonly served as function approximations for RL tasks since the convergence problems solved. [30]

2.2 Deep Reinforcement Learning (DRL)

Deep Reinforcement Learning (DRL) combines reinforcement learning and deep learning, which uses deep neural networks to approximate any components of reinforcement learning including value functions $V(s;\theta)$ or $Q(s,a;\theta)$, policy function $\pi(a|s;\theta)$, state transition function and reward function [28]. Recent work included model-free methods: Deep Q-Network (Mnih et al., 2015 [31]), Asynchronous Advantage Actor Critic (Mnih et al., 2016 [32]), Proximal Policy Optimization (Schulman et al., 2017 [33]), Deep Deterministic Policy Gradient (Lillicrap et al., 2015 [34]), Twin Delayed DDPG (Fujimoto et al., 2018 [35]) and Soft Actor-Critic (Haarnoja et al., 2018 [36]), and model-based methods: Imagination-Augmented Agents (Weber et al., 2017 [37]), Model-Based RL with Model-Free Fine-Tuning (Nagabandi et al., 2017 [38]), Model-Based Value Expansion (Feinberg et al., 2018 [39]) and AlphaZero (Silver et al., 2017 [40]).

Chapter 3

Methods

This chapter included the environment model, the Markov Decision Process (MDP) modeling of the ventilation system control problem in the pig buildings, and details of the Reinforcement Learning (RL) algorithms, the baseline method and evaluation metrics used in this thesis.

3.1 Environment Modeling

Xie et al. (2019) [41] [42] developed a dynamic thermal exchange model based on the energy balance equation (EBE) to simulate the heat transfer and the thermal environment in a pig building and tested it in three different seasons. This thesis used this model as the environment model, which defined the interactions between the output of the ventilation system and the thermal changes in the pig buildings, to train and test the Reinforcement Learning (RL) algorithms for swine buildings ventilation system control.

3.1.1 Pig Building Description

According to Xie et al. (2019) [41], "the swine building was located at the Animal Research and Education Center, Purdue University, West Lafayette, Indiana, USA. The pig building was a mixed steel and wood structure, and the ventilation system was closed mechanical ventilation. The dimensions of the building were 73.2 m × 24.4 m × 2.7 m (L × W × H) and the roof peak was 5.1 m high". Figure 2 was a photo of the pig building. According to Xie et al. (2019) [41], "the building had 12 pig rooms facing north or south, and each room had a capacity of housing 60 finishing pigs with a 11.0 m × 6.1 m × 2.7 m (L × W × H) pig living space (PLS). The building had two rows of 6 pens on each side and a center alley, and two 1.8 m deep manure pits below the PLS separated with a slatted concrete floor". Figure 3 showed the internal structure of the swine building.

Figure 2: Photo of the pig building, reprinted from Xie et al. (2017) [43, Figure 3a]

According to Xie et al. (2019) [41], "two air inlets that located on top of the east and west doors enabled fresh air to enter into the building, and air supplied to each room from ceiling and hallway inlets. Each room had two wall fans with single-speed to provide ventilation, one with 356-mm diameter (180 W) and another with 508-mm diameter (430 W), two pit fans with variable-speed and 250-mm diameters to provide room with minimum ventilation, and a heater to provide supplemental heating in winter".

Based on this swine building, we made some modifications for the pig building in this thesis. To simplify the building design, each room had one wall fan with different ventilation levels to provide minimum and maximum ventilation, there

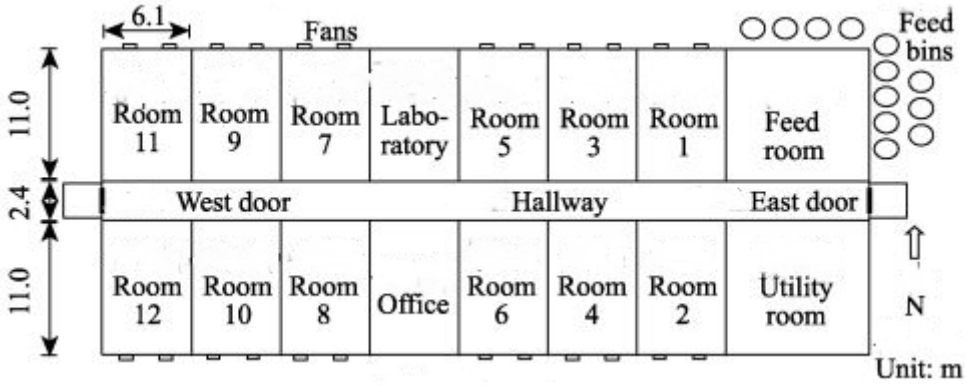


Figure 3: Stucture diagram of the swine building from top view, reprinted from Xie et al. (2019) [42, Figure 2a]

were no pit fans and manure pits, and floors were considered closed instead of slatted. The heater was turned off.

3.1.2 Model

According to Xie et al. (2019) [41], "heat exchange took place in the confined swine building through radiation, convection, conduction, and evaporation. Indoor temperature changes were significantly affected by solar radiation, heating system, ventilation, and conductive and radiative heat transfer between pigs and the buildings' interior structure". Figure 4 showed the heat transfers in the pig buildings.

Now we introduced the EBE model developed by Xie et al. (2019) [41] [42] to simulate the thermal environment in pig buildings. Firstly, according to the first law of thermodynamics, the difference of the heat gain and loss per unit of time was used to calculate the energy balance in the pig room as Eq.(3.1) [41].

$$\rho_a \cdot V \cdot c_p \frac{dT_i}{dt} = Q_h + Q_r + Q_p + Q_s + Q_f + Q_g \tag{3.1}$$

where ρ_a represented air density, kg/m^3; V represented air volume of pig building, m^3; c_p represented air specific heat capacity, $J/(kg \cdot ^\circ C)$; T_i represented indoor air temperature, $^\circ C$; $\frac{dT_i}{dt}$ represented temperature change rate, $^\circ C/s$; Q_h, Q_r, Q_p denoted heat gain in unit time from heating system, outside building envelope

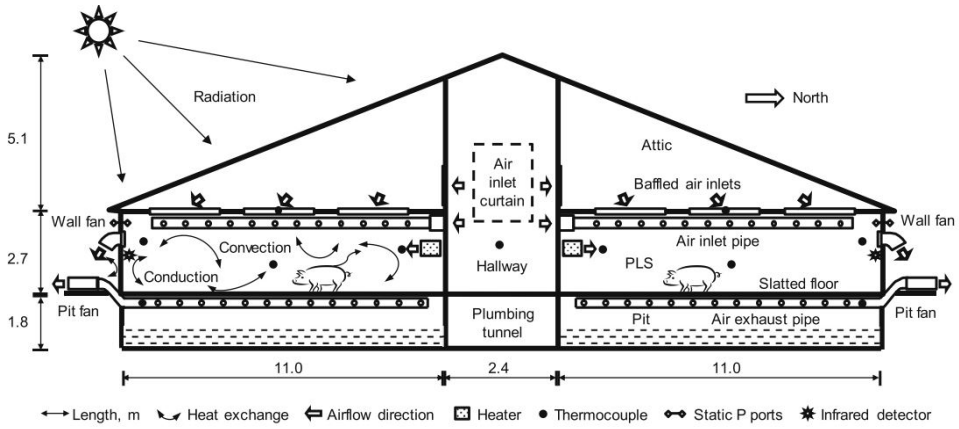


Figure 4: Diagram of heat exchange in swine building reprinted from Xie et al. (2019) [41, Figure 1]

received solar radiation, and pig body surface respectively, W; Q_s, Q_f, Q_g denoted heat loss in unit time from pig building envelop, ventilation system, and floor respectively, W [41].

Q_r was defined following the radiation law as Eq.(3.2), where ρ_r represented envelope material transmission coefficient, S_r represented envelope surface of pig building that received solar radiation, m^2; and I_D represented solar irradiance, Wm^{-2} [41].

$$Q_r = \rho_r \cdot S_r \cdot I_D \tag{3.2}$$

Eq.(3.3) expressed Q_h as the heating system used air convection to heat the pig room, where m_h represented mass of heated air, kg/s; and T_h represented temperature of heater surface, $°C$ [41]. $Q_h = 0$ in this thesis because the heater was switched off.

$$Q_h = m_h \cdot c_p \cdot (T_h - T_i) \tag{3.3}$$

Because heat is always transferred from the higher temperature side of an envelope to the lower temperature side, Q_s was associated with the temperature difference between the interior and exterior surfaces of the envelope and the heat transfer surfaces as Eq.(3.4), where k_s represented heat transfer coefficient of building envelope, $Wm^{-2}K^{-1}$; T_o represented outdoor air temperature, $°C$; and F_s represented area of building envelope, m^2 [41].

$$Q_s = k_s \cdot (T_i - T_o) \cdot F_s \tag{3.4}$$

Eq.(3.5) showed the primary heat exchange mechanism between a pig and the indoor air occurs through the pig's skin, where n represented pig number, n; Q_{pr} represented the radiative heat exchange of the pig's body surface, W; and Q_{pc} represented the convective heat exchange between the pig's body surface and the air, W [41]. In Eq.(3.6), A_p represented area of pig body surface, m^2, $A_p = 0.105 \cdot k \cdot \sqrt[3]{W_t^2}$; ε represented thermal emissivity of pig body surface; σ represented Stefan-Boltzmann constant, $Wm^{-2}K^{-4}$; and T_{pig} represented temperature of pig body surface, $°C$ [41] [42]. In Eq.(3.7), h_c represented convective heat transfer coefficient, $Wm^{-2°}C^{-1}$, $h_c = \sqrt[3]{270 \cdot v^2 + 23}$, v represented air speed, m/s [41] [42]. Therefore, Q_p was calculated as Eq.(3.8) [41].

$$Q_p = n(Q_{pr} + Q_{pc}) \tag{3.5}$$

$$Q_{pr} = A_p \cdot \varepsilon \cdot \sigma[(T_{pig} + 273)^4 - (T_i + 273)^4] \tag{3.6}$$

$$Q_{pc} = A_p \cdot h_c \cdot (T_{pig} - T_i) \tag{3.7}$$

$$Q_p = n \cdot A_p \left\{\varepsilon \cdot \sigma[(T_{pig} + 273)^4 - (T_i + 273)^4] + h_c(T_{pig} - T_i)\right\}; \tag{3.8}$$

Q_f was impacted by how efficiently the pit fans and wall fans operated as Eq.(3.9), where L_w and L_p represented ventilation rates of wall fans and pit fans, respectively, m^3/s; T_{hw} and T_p represented air temperatures of hallway and pit, respectively [41]. In this thesis, $T_{hw} = T_o$ to simplify the model, and $L_p = 0$ since we did not consider pit fans and pits.

$$Q_f = \rho_a \cdot c_p \cdot [L_w \cdot (T_i - T_{hw}) + L_p \cdot (T_i - T_p)] \tag{3.9}$$

Q_g was calculated as Eq.(3.10), where S_g represented floor area inside pig building, m^2; h_g represented heat exchange coefficient of floor, $Wm^{-2°}C^{-1}$; given the model assumption which considered the floor temperature T_g and the pit air temperature T_p the same [41]. $Q_g = 0$ in this thesis since we considered floors were closed instead of slatted, and no pits below floors.

$$Q_g = S_g \cdot h_g \cdot (T_i - T_p) \tag{3.10}$$

Figure 5 showed all the thermal exchanges in the pig building. Based on the energy balance Equations (3.1) to (3.10), this thesis used Python as the programming language to simulate the thermal exchanges in the pig room. The differential

Eq.(3.1) was numerically solved with a fourth-order using the classic Runge-Kutta method. The model was solved using parameter values described in the following at a fixed step of 5 minutes. The measured outdoor air temperatures and solar radiations were the model's input values. At t = 0, the initial value of the indoor air temperature was set to $20°C$. The calculated indoor temperature at the current time step served as the next time step's input value.

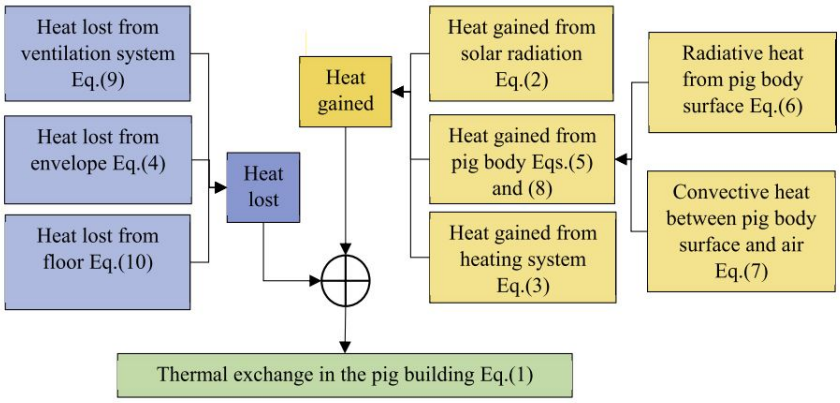


Figure 5: Thermal exchange in swine building, reprinted from Xie et al. (2019) [41, Figure 3]

3.1.3 Parameters

In this thesis, we selected Room 11 shown in Figure 3, because some detailed parameters of this room were provided by Xie et al. (2019) [42]. According to Xie et al. (2019) [42], Room 11 had 58 pigs with weight from 97.8 kg to 101.2 kg. The values of some parameters used in this thesis were collected from Xie et al. (2019) [41] [42] and shown in Table 1.

Parameter	Description	Value [unit]
c_p	Air specific heat capacity	1012 $[Jkg^{-1}{}^{\circ}C^{-1}]$
F_s	Area of the envelope	92.34 $[m^2]$
n	Pig number	58
S_g	Floor area inside the pig room	67.1 $[m^2]$
T_p	Pig body surface temperature	30 $[^{\circ}C]$
V	Volume of the pig room	181.17 $[m^3]$
W_t	Average weight of pigs	98 $[kg]$
ε	Thermal emissvity of pig body surface	0.95
σ	Stefan-Boltzman constant	5.67×10^{-8} $[Wm^{-2}K^{-4}]$
S_r	Envelope surface receiving solar radiation	29.7 $[m^2]$
h_g	Heat exchange coefficient between the floor and indoor air	6 $[Wm^{-2}]$
k_s	Heat transfer coefficient of the building envelope	0.405 $[Wm^{-2}K^{-1}]$
ρ_r	Envelope material transmission coefficient	0.48
k	Pig body surface correction factor	0.66
v	Air speed	0.15 $[ms^{-1}]$

Table 1: Parameters for the environment model

3.2 Markov Decision Process (MDP)

The building HVAC control problem can be seen as a Markov Decision Process (MDP) [23], and we formulated the details of the MDP process in the following.

3.2.1 State space

The state space was a vector of time, environmental conditions (i.e. solar irradiation and outdoor temperature) and indoor temperature, represented as

$$S = (t, I_D, T_o, T_i), s_t \in S \tag{3.11}$$

where s_t represented the state at time t, and the range of each variable were described in Table 2, and they were floating numbers. The measurements of solar irradiation and outdoor temperature in West Lafayette, Indiana, USA of 2021 were downloaded from National Solar Radiation Database (NSRDB) [1] and they were collected every 5 minutes, so the time interval of our MDP was 5 minutes. West Lafayette had a continental humid climate with four distinct seasons, with the lowest temperature -22.8 °C, and the highest temperature 34.7 °C, in 2021.

[1] https://nsrdb.nrel.gov

Variable [unit]	Description	Min	Max
t [h]	Time	0	24
I_D $[Wm^{-2}]$	Solar irradiation	0	1050
T_o $[°C]$	Outdoor temperature	-30	40
T_i $[°C]$	Indoor temperature	-30	40

Table 2: State space description

3.2.2 Action space

We considered the ventilation system of a pig room was a exhaust fan with four discrete levels of ventilation rates, i.e. $0.3m^3/s$, $0.6m^3/s$, $1.2m^3/s$, $1.8m^3/s$, to provide ventilation for the pig room. Therefore, the action space of the ventilation system control can be represented as

$$A = \{0, 1, 2, 3\}, a_t \in A \tag{3.12}$$

where a_t represented the action at time t. The minimum ventilation rate was $0.3m^3/s$, and the maximum ventilation rate was $1.8m^3/s$. Table 3 showed the ventilation rates and power consumption of the fan under each action.

Action	Ventilation rate, m^3/s	Power consumption, kW
0	0.3	0.1
1	0.6	0.18
2	1.2	0.36
3	1.8	0.54

Table 3: The ventilation rates and power consumption of the fan

3.2.3 Reward function

A well-designed reward function is essential to achieve a good performance in reinforcement learning. The reward function included two parts, the penalty of the power consumption of the fan and the penalty of the temperature deviation from the comfortable range, as shown in Equation (3.13).

$$r_t = -wP_t - \begin{cases} 0 & \text{if } \underline{\mathrm{T}} \leq T_i \leq \bar{T}, \\ T_i - \bar{T} & \text{if } T_i > \bar{T}, \\ \underline{\mathrm{T}} - T_i & \text{if } T_i < \underline{\mathrm{T}} \end{cases} \tag{3.13}$$

where r_t represented the reward at time t, w was the weight of the power consumption in the reward function, P_t was the power consumption at time t, $\bar{T}$ and $\underline{\mathrm{T}}$ were the upper bound and lower bound of the desired temperature range. More penalty was put on power consumption when w was a big value, less penalty was put on it if w was a small value, and w represented the trade-off between the importance to minimize energy consumption and to keep pig's thermal comfort. The goal of the reinforcement learning was to maximize the cumulative reward, so in this MDP problem was to keep the temperature inside the desired range as much time as possible while minimizing the power cost.

3.3 Deep Q Network (DQN)

In Deep Q Network (DQN), the artificial neural network was used to approximate the Q values as shown in Figure 6. The neural network can output the Q values for all possible actions at a given state.

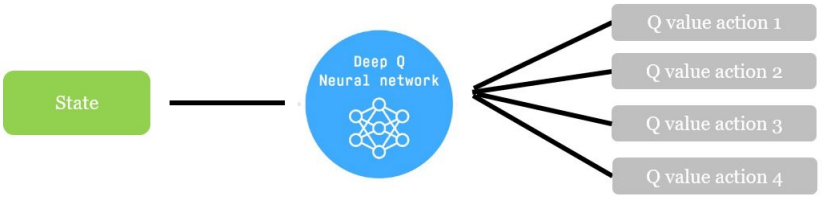


Figure 6: Deep Q Network

The DQN algorithm we used was based on the DQN algorithm proposed by Mnih et al. (2015) [31], and the pseudocode of the DQN algorithm was shown in Algorithm 1. The outer loop showed the number of training episodes, and the inner loop performed the training at each time step inside one episode.

Algorithm 1 Deep Q-Network (DQN), adapted from Mnih et al.(2015) [31]

1: Initialize memory D to capacity N
2: Initialize action-value function Q with random weights θ
3: Initialize target action-value function $\hat{Q}$ with random weights $\hat{\theta} = \theta$
4: **for** episode=1,M **do**
5: Reset environment to initial state s_1
6: **for** t = 1,T **do**
7: With probability ϵ select a random action a_t
8: otherwise select $a_t = argmax_a Q(s_t, a; \theta)$
9: Execute a_t in environment and observe reward r_t and next state s_{t+1}
10: Store transition (s_t, a_t, r_t, s_{t+1}) in D
11: Sample random minibatch of transitions (s_j, a_j, r_j, s_{j+1}) from D
12: Set $y_j = \begin{cases} r_j, & \text{if episode terminates at step } j+1 \\ r_j + \gamma max_{a'} \hat{Q}(s_{j+1}, a'; \hat{\theta}), & \text{otherwise.} \end{cases}$
13: Perform a gradient descent step on $(y_j - Q(s_j, a_j; \theta))^2$ with respect to network parameters θ
14: Every C steps reset $\hat{Q} = Q$
15: **end for**
16: **end for**

3.3.1 Initial setup

Before the training process, we first initialized an empty replay memory M, a neural network Q with random weights θ to approximate the action value function as Equation 2.3, and a neural network $\hat{Q}$ with weights $\hat{\theta}$ by copying the neural network Q and its weights to approximate the target action value function, as shown in Line 1-3 of Algorithm 1. At the beginning of every episode of training, as shown in Line 5 of Algorithm 1, the environment was set to the initial states $s_1 = (t, I_D, T_o, T_i)$, where t, I_D, T_o were time, solar radiation, and outdoor temperature from the external data, and $T_i = 20$ as we initialized the indoor temperature to 20°C.

3.3.2 Training process

During the training process, Line 7-8 of algorithm 1 indicated that the ϵ-greedy policy was applied to select the action, so the agent chose a random action with

probability ϵ to explore the action space, and chose the action with the maximum output value from the network Q, i.e. highest Q value, with probability $1-\epsilon$. The exploration rate ϵ gradually decreased during the training process until reaching the minimum value ϵ_{min}.

Then in Line 9 the action a_t was passed into the environment defined in Section 3.1 which calculated the temperature changes based on the given ventilation rate, and the environment changed from the current state s_t into a new state s_{t+1} and provided the reward r_t defined in Equation 3.13 to the agent. Line 10 showed the memory M stored the transition tuples $\langle s_t, a_t, r_t, s_{t+1} \rangle$, where s_t, a_t, r_t, s_{t+1} represented for current state, current action, current reward, and next state, respectively. Then random minibatch of transitions were sampled from memory M for training the network Q, as Line 11 of Algorithm 1.

Line 12 showed we used the target network $\hat{Q}$ to estimate the target Q value as

$$y_j = \begin{cases} r_j, & \text{if episode terminates at step } j+1 \\ r_j + \gamma max_{a'} \hat{Q}(s_{j+1}, a'; \hat{\theta}), & \text{otherwise.} \end{cases} \tag{3.14}$$

and the loss function is defined as the mean-squared error between the output value of the network Q and the target Q value as shown in Equation 3.15.

$$L(\theta) = E_\pi[(y_j - Q(s, a; \theta))^2] \tag{3.15}$$

Line 13 showed the weights θ of the network Q was trained by using a gradient descent method, which was used to minimize the loss function and updated the parameters θ following the rule $\theta \leftarrow \theta - \alpha \frac{\partial L(\theta)}{\partial \theta}$, where α is the learning rate, and the gradient was defined as Equation 3.16 with respect to the parameters θ. [30]

$$\frac{\partial L(\theta)}{\partial \theta} = E[(y_j - Q(s, a; \theta))] \frac{\partial Q(s, a; \theta)}{\partial \theta} \tag{3.16}$$

Line 14 suggested that the weights $\hat{\theta}$ of the target network $\hat{Q}$ were updated by copying the weights θ of the network Q every C steps, and C was a hyperparameter to be defined.

3.4 Baseline method

The baseline method was a rule-based method designed according to climate guidelines for pig buildings (2021) by Varkenshouderij Klimaatplatform [2]. In warm and hot days like spring and summer, the rules were shown in Table 4a, we selected the ventilation level to be 3 when the room temperature was higher or equal to $24°C$; and we selected the ventilation level to be 2 when the room temperature was higher or equal to $21.5°C$ but lower than $24°C$; we took the ventilation level to be 1 when the room temperature was higher or equal to $19°C$ but lower than $21.5°C$; we kept the fan on the minimum ventilation rate when the room temperature was lower than 19. In cold days like winter, rules were shown in Table 4b. Since the outdoor temperature was very low, high ventilation rate level 2 and 3 were not necessary, so we selected the ventilation level to be 1 when the room temperature was high or equal to $19°C$; and we kept the fan on the minimum ventilation rate if the room temperature was lower than 19 $°C$.

Temperature (°C)	Action
$T_i \geq 24$	3
$21.5 \leq T_i < 24$	2
$19 \leq T_i < 21.5$	1
$T_i < 19$	0

(a) In Spring and Summer

Temperature	Action
$T_i \geq 19$	1
$T_i < 19$	0

(b) In winter

Table 4: Baseline rule-based control

3.5 Evaluation Metrics

The evaluation metrics used in this thesis to indicate the performance of controllers are:

1. **Temperature violation rate**: it was defined by the proportion of the temperature out of the comfortable range during the test time period with the unit %.

[2] https://www.wur.nl/nl/show/Richtlijnen-klimaatinstellingen-varkenshouderij.htm

2. **Power consumption**: it represented the total power consumption of the HVAC system during the test time period with the unit kWh.

3. **Thermal discomfort**: it means the cumulative temperature deviation out of the comfortable rage during the test time period with the unit °C.

Chapter 4

Results

This chapter included the hyperparameter settings in the experiment setup and presented the experiment results.

4.1 Experiment Setup

The comfortable temperature range for pigs was between $15°C$ ($\underline{\mathrm{T}}$) and $23°C$ ($\bar{T}$) according to Chinese national standard of environmental management for intensive pig farms [44], and the weight of the power consumption in the reward function w was 10 by default. We used the same neural network architecture applied by Wei et al. (2017) [23] for building HVAC system control. The network Q and the target network $\hat{Q}$ had four fully-connected hidden layers, and each layer had 50, 100, 200, 400 neurons respectively. We used the rectified linear unit (ReLU) activation function in each hidden layer. The optimizer was Adam optimizer. We trained the DQN algorithm 400 episodes and the length of each episode was 48 hours. Table 5 shows the hyperparameter settings of the DQN algorithm. Buffer size was the size of the replay memory, batch size was the size of the minibatch for each gradient update, gamma was the discount factor, target update interval meant the target network get updated every that number of steps, exploration

max was the initial value of random action probability, exploration min was the final value of random action probability, max grad norm was the maximum value for the gradient clipping which helped stabilize the training process.

Hyperparameter	Value
Batch size	144
Buffer size	144*31
Learning rate	0.003
Gamma	0.99
Exploration max	1
Exploration min	0.1
Target update interval	144*5
Max grad norm	10
Net	[50,100,200,400]

Table 5: Parameters of DQN Algorithm

We trained three DQN algorithms using the same hyperparameter settings on three different months, i.e. January, April and July in 2021 to evaluate our DQN algorithm performance in different seasons such as spring, summer and winter. Table 6 showed the descriptive statistics for weather conditions such as solar radiation and outdoor temperature for these months. The heater was turned off all the time because the room had 58 finished pigs with the average weight of 98 kilograms so there were a lot of heat gain from those pigs' skin. We used the January, April and July in 2021 as the training data for the DQN algorithm, and we tested both the baseline method and the trained DQN agent on February 1 and 2, May 1 and 2, August 1 and 2 in 2021 to evaluate their performances.

	Solar radiation, Wm^{-2}	Outdoor temperature, $°C$
Period: 01/01/2021 - 31/01/2021		
Min	0	-12
Max	566	7.3
Mean	66.03	-2.10
SD	125.05	3.25
Period: 01/04/2021 - 30/04/2021		
Min	0	-5.4
Max	993	27.2
Mean	220.80	10.90
SD	301.92	6.90
Period: 01/07/2021-31/07/2021		
Min	0	11.9
Max	996	31.4
Mean	260.90	23.30
SD	316.61	4.13

Table 6: Summary statistics of weather data

4.2 Experiment Results

Outdoor temperatures in February 1 and 2, May 1 and 2, August 1 and 2, were shown in Figure 7a, 8a, and 9a, respectively. We can see from Figure 7a that outdoor temperatures in test winter days were below 0 °C mostly. Figure 8a showed large outdoor temperature differences between day and night on the fist spring test day. Figure 9a showed outdoor temperatures ranged between 14 °C and 28 °C on the summer test days.

Solar radiations in February 1 and 2, May 1 and 2, August 1 and 2, were shown in Figure 7b, 8b, and 9b, respectively. The figures showed that solar radiations were 0 during the night hours, increased during the day hours, and reached the peak values in a day around the noon hours. We can see that solar radiations were relatively low in winter test days with the maximum values close to 600 Wm^{-2} from Figure 7b, solar radiations were medium in spring test days with the maximum values around 800 Wm^{-2} from Figure 8b, and solar radiations were highest in summer test days with the maximum values close to 1000 Wm^{-2} from Figure 9b.

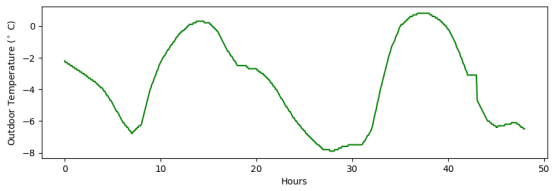
Figure 7c, 8c, and 9c showed the temperatures in the pig room in February 1 and 2, May 1 and 2, August 1 and 2, respectively, where the baseline rule-based

method performed control to operate the ventilation fan into different ventilation rate levels. We can see from 8c, and 9c that in spring and summer test days, the room temperatures had large deviations from the comfortable range when the solar radiation and outdoor temperature were relatively high in a day.

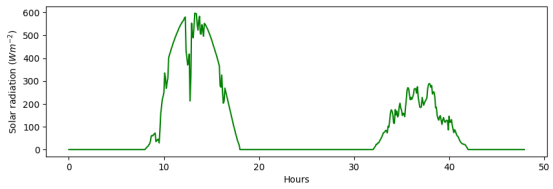
Figure 7d, 8d, and 9d showed the ventilation rate levels of the fan in the pig room in February 1 and 2, May 1 and 2, August 1 and 2, respectively, where the baseline rule-based method performed control to operate the ventilation fan. We can see from Figure 8d and 9d that during the spring and summer test days, the fan was mostly operated at the ventilation level 3 during the time when the solar radiations and outdoor temperatures were relatively high in a day.

Figure 7e, 8e, and 9e showed the temperature in the pig room in February 1 and 2, May 1 and 2, August 1 and 2, respectively, where the DQN algorithms performed control to operate the ventilation fan into three ventilation rate levels. We can see that the DQN algorithms were effective in keeping the room temperature within the comfortable range for pigs in test days of winter, spring and summer, suggesting that the DQN algorithm can also generalize very well since we used the same hyperparameter settings for all seasons.

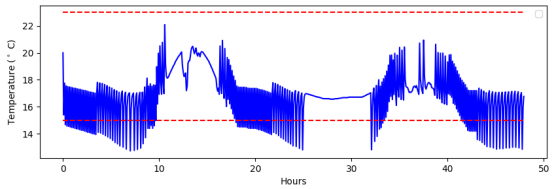
Figure 7f, 8f, and 9f showed the ventilation rate levels of the fan in the pig room in February 1 and 2, May 1 and 2, August 1 and 2, respectively, where the DQN algorithms performed control to operate the ventilation fan. We can see from Figure 7f that during the winter test days, the fan was at the minimum ventilation level for the most of the time, and at ventilation level 1 during the time when the solar radiations and outdoor temperatures were relatively high in a day. We can see from Figure 8f that during the spring test days, the fan was mostly operated at the ventilation level 2 during the time when the solar radiations and outdoor temperatures were relatively high in a day. We can see from Figure 9f that during the summer test days, the fan needed to operate at the maximum ventilation level during the time when the solar radiations and outdoor temperatures were relatively high in a day to maintain the temperature within the comfortable range. By comparing Figure 8d with Figure 8f, we can see that the fan operated more hours at the maximum ventilation level using the baseline rule-based control, compared to the DQN control, in spring test days.



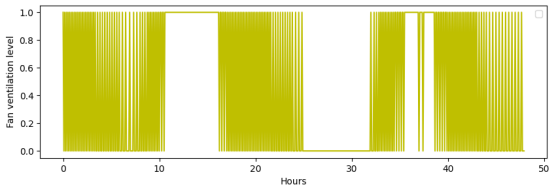
(a) Outdoor temperature



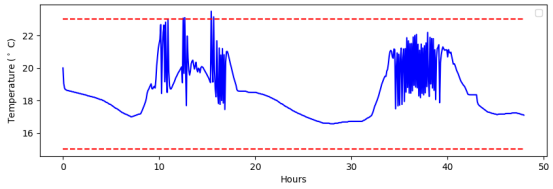
(b) Solar radiation



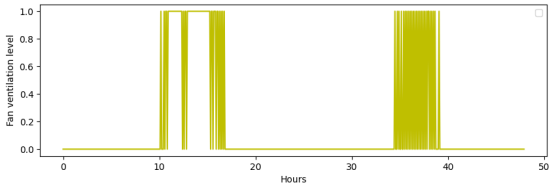
(c) Indoor temperature using baseline control



(d) Fan ventilation rate levels using baseline control

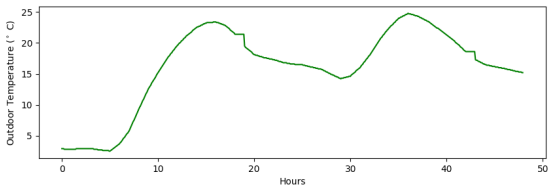


(e) Indoor temperature using DQN algorithm control

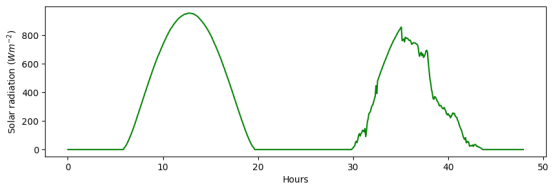


(f) Fan ventilation rate levels using DQN algorithm control

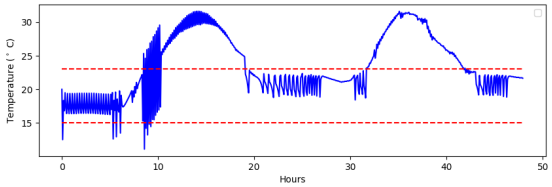
Figure 7: Results on February 1 and 2



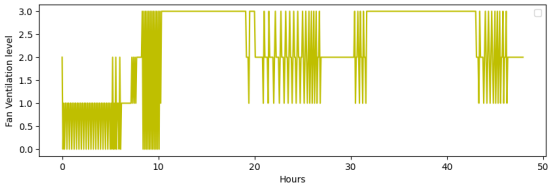
(a) Outdoor temperature



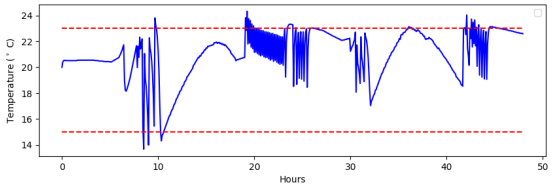
(b) Solar radiation



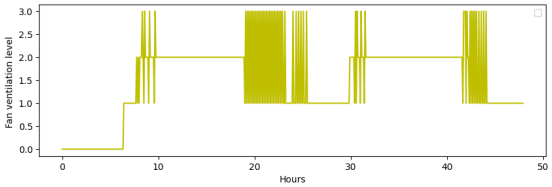
(c) Indoor temperature using baseline control



(d) Fan ventilation rate levels using baseline control

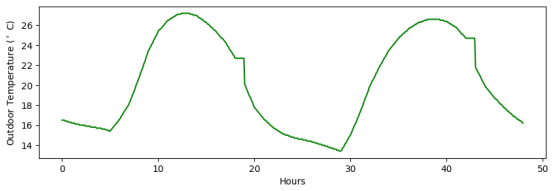


(e) Indoor temperature using DQN algorithm control

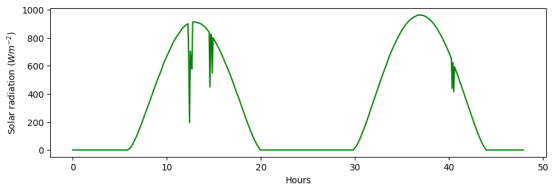


(f) Fan ventilation rate levels using DQN algorithm control

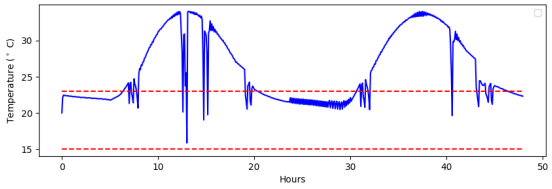
Figure 8: Results on May 1 and 2



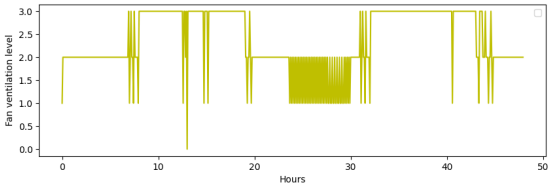
(a) Outdoor temperature



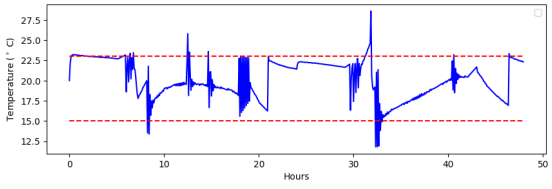
(b) Solar radiation



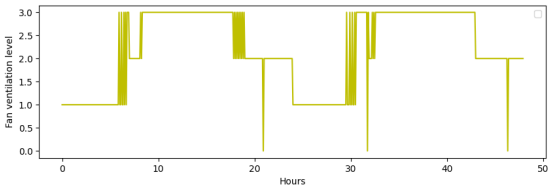
(c) Indoor temperature using baseline control



(d) Fan ventilation rate levels using baseline control



(e) Indoor temperature using DQN algorithm control



(f) Fan ventilation rate levels using DQN algorithm control

Figure 9: Results on August 1 and 2

Figure 10 compared the average frequency of uncomfortable temperature of the baseline strategy and the DQN algorithm. We can see that the DQN algorithms were able to keep the average frequency of uncomfortable temperature in the test days of all the three different seasons in a low level. To be more specific, the uncomfortable proportion of the room temperature under the baseline control method were 19.10%, 43.06%, and 56.60%, in the test days of winter, spring, and summer respectively. While the DQN algorithms had 0.87%, 9.90%, 7.99% average proportion of uncomfortable temperature in the test days of winter, spring, and summer respectively.

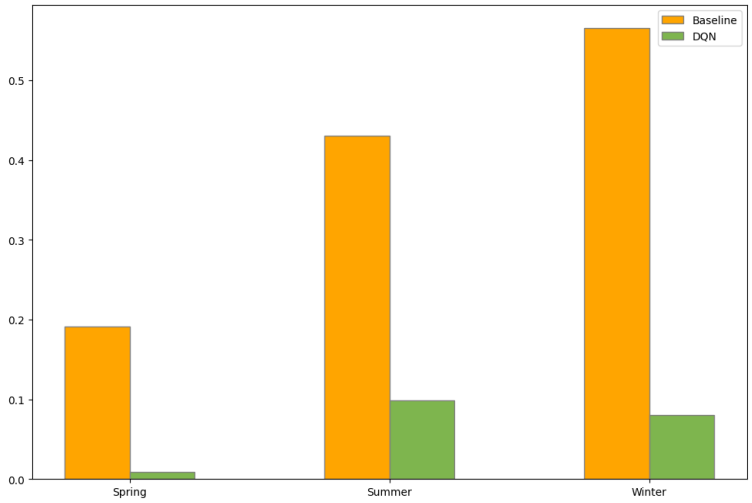


Figure 10: Comparison of the proportion of temperature out of the comfortable range between the baseline method and the DQN algorithm on the test days of three seasons

Figure 11 showed the total power consumption of the ventilation fan in the test days between the baseline method and the DQN algorithm. We can see that the DQN algorithms resulted in significant energy consumption reduction in the test days of spring, minor energy consumption reduction in the test days of winter and summer compared with the baseline method. In detail, the total power consumption of the baseline method was 6.57 kWh, 19.44 kWh, and 20.47 kWh for the test days in winter, spring, and summer, respectively. The total power consumption of the DQN algorithms were 5.41 kWh, 13.60 kWh, 19.06 kWh for the test days in winter, spring, and summer, respectively. Therefore, the

DQN algorithms achieved 17.66%, 30.04%, 6.89% power consumption reduction compared with the baseline approach in the test days of winter, spring and summer, respectively.

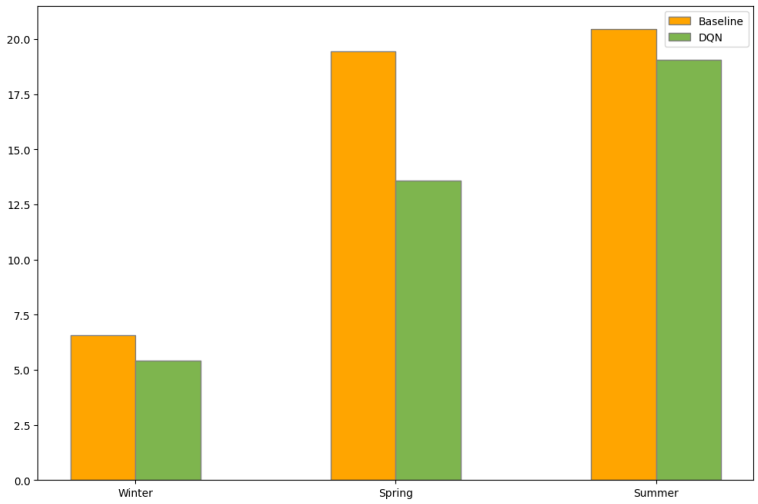


Figure 11: Comparison of the total power consumption of the ventilation fan between the baseline method and the DQN algorithm on the test days of three seasons

Table 7 showed the cumulative temperature deviation from the comfortable range during the test days, calculated by the sum of the absolute error between the indoor temperature and the upper or lower bound of the desired range as the second term in Equation 3.13, using the baseline method control and the DQN control. We can see that DQN algorithm managed to achieve low cumulative temperature deviation values in the test days of all three seasons, and had nearly perfect performance in the test days of winter. On the other hand, baseline rule-based control got quite large deviation sum values in the test days of all three seasons, especially in the test days of spring and summer.

4.3 Discussion

The performance of the baseline rule-based method was surprisingly poor on maintaining the temperature within the comfortable range, especially in the test days

	Baseline rule-based, °C	DQN, °C
Winter (Feberuary 1 and 2)	123.56	0.84
Spring (May 1 and 2)	1242.37	15.65
Summer (August 1 and 2)	2044.34	32.81

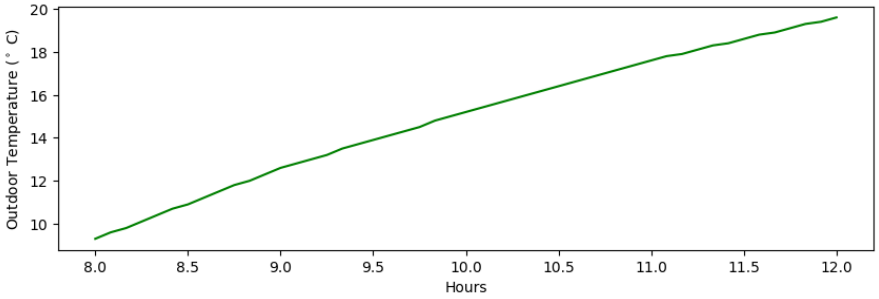
Table 7: Sum of temperature deviation from the comfortable range during the test days in three seasons using baseline rule-based control or DQN control

of spring and summer. We thought the possible reasons were:

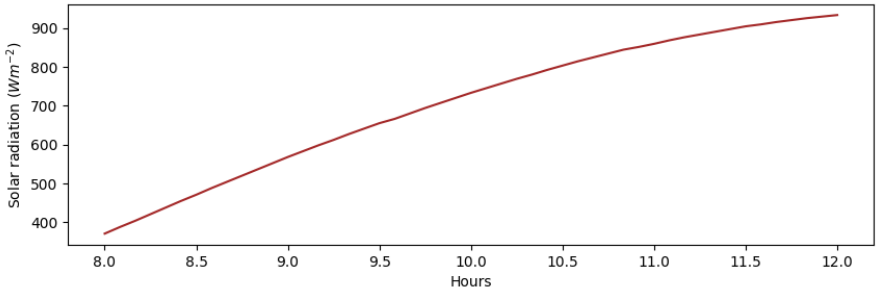
1. We assumed the ventilation rate was controlled on discrete levels to simplify the problem and save computation time, however, this could result in a underestimate of the performance of the rule-based method in reality when the ventilation rate can be controlled continuously.

2. The baseline rule-based method might have the problem of time lags since it decided the action only based on the indoor temperature at the current time step. On the contrary, the current time information incorporated in the states of the DQN algorithm enabled the agent to learn the time-varying weather conditions [23], so the agent might be able to decided the actions in a foreseeable way.

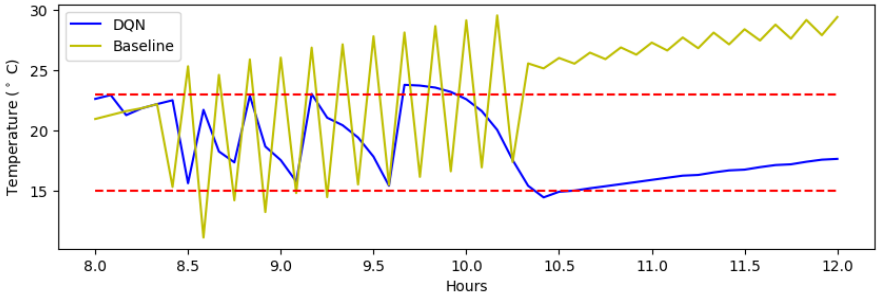
For example, we tested both the baseline method and the DQN approach on May 1, one of the test day in spring, from 8:00 to 12:00 as shown in Figure 12. We can see in Figure 12d that from 8:00 to 10:30 the baseline method and DQN algorithm chose different ventilation levels, the baseline method mainly chose level 0 and level 3, while the DQN method mainly chose level 1, 2, and 3. Both methods resulted in different indoor temperatures at 10:30 shown in Figure 12c, and the indoor temperature at 10:30 of DQN control was much lower than the baseline method which might imply the DQN method did not need to choose the maximum ventilation rate for the time after 10:30. From 10:30 to 12:00, the indoor temperatures under both the baseline method and DQN algorithm showed an increasing trend probably due to the increasing outdoor temperature and solar radiation.



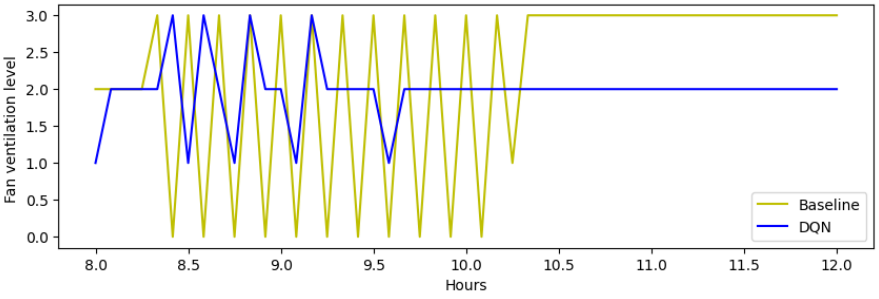
(a) Outdoor temperature



(b) Solar radiation



(c) Indoor temperature with baseline control and DQN control



(d) Fan ventilation rate levels with baseline control and DQN control

Figure 12: Results on May 1 8:00-12:00

Chapter 5

Conclusions and future work

In this chapter, we summarized the main research results of this thesis, discussed the limitations of the study and proposed some future work.

5.1 Conclusions

In this thesis, we presented the possibility to utilize Deep Reinforcement Learning method to optimally control the ventilation system in pig buildings, and the experiment results showed that:

1. The DQN agent managed to maintain the room temperature within the comfortable range after training. In detail, it achieved 99.13%, 90.1%, 92.01% average frequency of comfortable temperature, and 0.84°C, 15.65°C, 32.81°C cumulative temperature deviation from the comfortable range, in the test days of winter, spring and summer, respectively.

2. The DQN algorithm outperformed the baseline with the saving of power consumption by 17.66%, 30.04%, 6.89% in the test days of winter, spring and summer, respectively.

3. The DQN algorithm applied the same neural network architecture and hyperparameter settings and was trained and tested in different periods of time,

indicating the generalization capability of the DQN algorithm.

5.2 Limitations

Although this thesis has made certain contributions to the ventilation system control in pig buildings based on DRL, there are several limitations:

1. In this thesis we used discrete action space, however, it's more practical to use continuous action space as the ventilation rate usually can be controlled between the minimum and maximum in reality.

2. In this thesis we only worked on the control for ventilation system, the study will be more comprehensive if the control of heating system is added, since the temperature can be too cold for pigs to live in the room in some cities with very cold winter.

3. In this thesis we only used the DQN algorithm, we didn't explore other RL algorithms, such as policy-based method Proximal Policy Optimization (PPO), actor critic method Advantage Actor Critic (A2C).

5.3 Future work

Besides the limitations we discussed above, we suggested some future work that can be conducted on this topic further, as following:

1. Real price of electricity can be used to approximate the cost of power consumption, and it can be different on peak hours and off-peak hours.

2. Besides the indoor temperature, we can consider some other conditions such as humidity, carbon dioxide emission that can affect pigs' comfort.

3. In this thesis, the pig building was closed with mechanical ventilation. Furthermore, the ventilation system can be hybrid ventilation including both natural ventilation and mechanical ventilation.

Appendix A

This project used Google Colab notebook to train the algorithms with GPU computing, and the notebook can be found at url: https://colab.research.google.com/drive/1kLBKthdqRBj-2SAnkK2VQ83H9kw1tcjH?usp=sharing.

Bibliography

[1] FAO: Food and Agriculture Organization of the United Nations. *Food Outlook-Biannual Report on Global Food Markets*. FOOD & AGRICULTURE ORG, 2022.

[2] Jelle Bruinsma. *World agriculture: towards 2015/2030: an FAO perspective*. Earthscan, 2003.

[3] UN Desa. World population prospects 2019: Highlights. *New York (US): United Nations Department for Economic and Social Affairs*, 11(1):125, 2019.

[4] China Statistical Yearbook. National bureau of statistics of china, 2021 (in chinese). 2021.

[5] JA Carroll, NC Burdick, CC Chase Jr, SW Coleman, and DE Spiers. Influence of environmental temperature on the physiological, endocrine, and immune responses in livestock exposed to a provocative immune challenge. *Domestic Animal Endocrinology*, 43(2):146–153, 2012.

[6] TTT Huynh, AJA Aarnink, WJJ Gerrits, MJH Heetkamp, TT Canh, HAM Spoolder, B Kemp, and MWA Verstegen. Thermal behaviour of growing pigs in response to high temperature and humidity. *Applied animal behaviour science*, 91(1-2):1–16, 2005.

[7] Emma M Baxter, Oluwagbemiga O Adeleye, Mhairi C Jack, Marianne Farish, Sarah H Ison, and Sandra A Edwards. Achieving optimum performance in a loose-housed farrowing system for sows: the effects of space and temperature. *Applied Animal Behaviour Science*, 169:9–16, 2015.

[8] Yunxiang Zhao, Xiaohong Liu, Delin Mo, Qingsen Chen, and Yaosheng Chen. Analysis of reasons for sow culling and seasonal effects on reproductive disorders in southern china. *Animal reproduction science*, 159:191–197, 2015.

[9] D. Renaudeau, H. Gilbert, and J. Noblet. Effect of climatic environment on feed efficiency in swine. In *Feed efficiency in swine*, pages 183–210. Wageningen Academic Publishers, 2012.

[10] Débora Caroline Gonçalves de Oliveira, Melissa Selaysim Di Campos, Nady Passé-Coutrin, Cristel Onesippe Potiron, Ketty Bilba, Marie-Ange Arsène, and Holmer Savastano Junior. Modeling of the thermal performance of piglet house with non-conventional floor system. *Journal of Building Engineering*, 35:102071, 2021.

[11] EM Lucas, JM Randall, and JF Meneses. Potential for evaporative cooling during heat stress periods in pig production in portugal (alentejo). *Journal of agricultural engineering research*, 76(4):363–371, 2000.

[12] Jeremy Austin Clark. *Environmental aspects of housing for animal production.* Butterworths, 1981.

[13] Zhongchao Tan and Yuanhui Zhang. A review of effects and control methods of particulate matter in animal indoor environments. *Journal of the Air & Waste Management Association*, 54(7):845–854, 2004.

[14] Abdul Afram and Farrokh Janabi-Sharifi. Theory and applications of hvac control systems–a review of model predictive control (mpc). *Building and Environment*, 72:343–355, 2014.

[15] Zhuang Wu, Jakob Stoustrup, Klaus Trangbaek, Per Heiselberg, and Martin Riisgaard Jensen. Model predictive control of the hybrid ventilation for livestock. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 1460–1465. IEEE, 2006.

[16] Zhenyu Yang, Stefan K Greisen, Jette R Hansen, Niels A Pedersen, and Martin R Jensen. On the single-zone modeling for optimal climate control

of a real-sized livestock stable system. In *2009 International Conference on Mechatronics and Automation*, pages 3849–3854. IEEE, 2009.

[17] Ai-min Li, Rui Song, Ming-qu Fan, Meng-ran Sun, Sheng-lei Mao, and Xin-dan Wang. The research and design of intelligent wind velocity-temperature-co 2 control model of coop. In *2015 IEEE International Conference on Information and Automation*, pages 2715–2719. IEEE, 2015.

[18] Zohaib Mushtaq, Akbare Yaqub, Muhammad Jabbar, Adnan Khalid, Saania Iqbal, Kamran Zeb, and Abid A Naqvi. Environment control system for livestock sheds using fuzzy logic technique. In *2016 3rd International Conference on Information Science and Control Engineering (ICISCE)*, pages 963–967. IEEE, 2016.

[19] Michael T Gorczyca and Kifle G Gebremedhin. Ranking of environmental heat stressors for dairy cows using machine learning algorithms. *Computers and electronics in agriculture*, 168:105124, 2020.

[20] Sang-yeon Lee, In-bok Lee, Uk-hyeon Yeo, Jun-gyu Kim, and Rack-woo Kim. Machine learning approach to predict air temperature and relative humidity inside mechanically and naturally ventilated duck houses: application of recurrent neural network. *Agriculture*, 12(3):318, 2022.

[21] Gregor P Henze and Jobst Schoenmann. Evaluation of reinforcement learning control for thermal energy storage systems. *HVAC&R Research*, 9(3):259–275, 2003.

[22] Simeng Liu and Gregor P Henze. Experimental analysis of simulated reinforcement learning control for active and passive building thermal storage inventory: Part 2: Results and analysis. *Energy and buildings*, 38(2):148–161, 2006.

[23] Tianshu Wei, Yanzhi Wang, and Qi Zhu. Deep reinforcement learning for building hvac control. In *Proceedings of the 54th annual design automation conference 2017*, pages 1–6, 2017.

[24] Guanyu Gao, Jie Li, and Yonggang Wen. Energy-efficient thermal comfort control in smart buildings via deep reinforcement learning. *arXiv preprint arXiv:1901.04693*, 2019.

[25] Rumia Masburah, Sayan Sinha, Rajib Lochan Jana, Soumyajit Dey, and Qi Zhu. Co-designing intelligent control of building hvacs and microgrids. In *2021 24th Euromicro Conference on Digital System Design (DSD)*, pages 457–464. IEEE, 2021.

[26] Jerry Luo, Cosmin Paduraru, Octavian Voicu, Yuri Chervonyi, Scott Munns, Jerry Li, Crystal Qian, Praneet Dutta, Jared Quincy Davis, Ningjia Wu, et al. Controlling commercial cooling systems using reinforcement learning. *arXiv preprint arXiv:2211.07357*, 2022.

[27] Wanfu Zheng. Deep reinforcement learning for building control: A comparative study for applying deep reinforcement learning to building energy management, 2022.

[28] Yuxi Li. Deep reinforcement learning: An overview. *arXiv preprint arXiv:1701.07274*, 2017.

[29] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[30] Seyed Sajad Mousavi, Michael Schukat, and Enda Howley. Deep reinforcement learning: an overview. In *Proceedings of SAI Intelligent Systems Conference (IntelliSys) 2016: Volume 2*, pages 426–440. Springer, 2018.

[31] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[32] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PMLR, 2016.

[33] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[34] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[35] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR, 2018.

[36] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. PMLR, 2018.

[37] Théophane Weber, Sébastien Racaniere, David P Reichert, Lars Buesing, Arthur Guez, Danilo Jimenez Rezende, Adria Puigdomenech Badia, Oriol Vinyals, Nicolas Heess, Yujia Li, et al. Imagination-augmented agents for deep reinforcement learning. *arXiv preprint arXiv:1707.06203*, 2017.

[38] Anusha Nagabandi, Gregory Kahn, Ronald S Fearing, and Sergey Levine. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. *arXiv preprint arXiv:1708.02596*, 2017.

[39] Vladimir Feinberg, Alvin Wan, Ion Stoica, Michael I Jordan, Joseph E Gonzalez, and Sergey Levine. Model-based value estimation for efficient model-free reinforcement learning. *arXiv preprint arXiv:1803.00101*, 2018.

[40] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.

[41] Qiuju Xie, Ji-Qin Ni, Jun Bao, and Zhongbin Su. A thermal environmental model for indoor air temperature prediction and energy consumption in pig building. *Building and Environment*, 161:106238, 2019.

[42] Qiuju Xie, Ji-Qin Ni, Jun Bao, and Honggui Liu. Simulation and verification of microclimate environment in closed swine house based on energy and mass balance (in chinese with english abstract). *Transactions of the Chinese Society of Agricultural Engineering*, 35(10):148–156, 2019.

[43] Qiuju Xie, Zhongbin Su, Ji-Qin Ni, and Ping Zheng. Control system design and control strategy of multiple environmental factors in confined swine building (in chinese with english abstract). *Transactions of the Chinese Society of Agricultural Engineering*, 33(6):163–170, 2017.

[44] China Ministry of Agriculture and Rural Affairs. Environmental parameters and environmental management for intensive pig farms gb-t 17824.3-2008 (in chinese). 2008.