공학박사학위논문

연료전지 스택의 열화를 고려한 연료전지 하이브리드 차량 대상의 Actor-critic 알고리즘을 활용한 진보한 실시간 동력분배전략의 개발

Advanced Real Time Power Management Strategy using Actor-Critic Algorithm Considering Degradation of Fuel Cell Stack in Electric Vehicles

2021년 2월

서울대학교 대학원

기계항공공학부

송 창 희

Advanced Real Time Power Management Strategy using Actor-Critic Algorithm Considering Degradation of Fuel Cell Stack in Electric Vehicles

By

SONG Changhee

A Dissertation Submitted in Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

Department of Mechanical and Aerospace Engineering

Seoul National University

February 2021

연료전지 스택의 열화를 고려한 연료전지 차량 대상의 Actor-critic 알고리즘을 활용한 진보한 실시간 동력분배전략의 개발

Advanced Real Time Power Management Strategy using Actor-Critic Algorithm Considering Degradation of Fuel Cell Stack in Electric Vehicles

지도교수 차 석 원

이 논문을 공학박사 학위논문으로 제출함

2020 년 10 월

서울대학교 대학원

기계항공공학부

송 창 희

송창희의 공학박사 학위논문을 인준함

2020 년 12 월

위 원 장 _____ 윤 병 동 _____ (인)

부위원장 _____ 차 석 원 _____ (인)

위 원 _____ 송 한 호 _____ (인)

위 원 _____ 김 남 욱 _____ (인)

위 원 _____ 조 구 영 _____ (인)

Abstract

Advanced Real Time Power Management Strategy using Actor-Critic Algorithm Considering Degradation of Fuel Cell Stack in Electric Vehicles

As vehicle emission regulations are becoming more and more stringent, vehicle manufacturers are making efforts to develop hybrid electric vehicle (HEV) as an alternative to increase fuel efficiency. The HEV is defined as vehicle with two or more power sources. Due to the advantage that individual power source can be operated at an efficient operating point, HEVs exhibit higher efficiency compared to internal combustion engine vehicles. However, the high efficiency of the HEV can only be guaranteed only if a valid power distribution strategy is in place.

Due to the importance of the power distribution strategy on the efficiency of the HEVs, many studies have been conducted on the development of the power management strategy. The related studies have been developing the energy management strategies for the HEVs based on rule-based control, optimal control theory, and reinforcement learning theory. The power distribution strategy based on the optimal control theory has the advantage of achieving high fuel efficiency, but the power distribution strategy has the disadvantage of low applicability and generalization performance in that future driving information must be considered. On the other hand, since rule-based control and reinforcement learning do not require future driving information, the vehicle applicability and the generalization performance are high, but the fuel efficiency is relatively low. Currently, the related research is focusing on developing the energy management strategy that is excellent in both the generalization performance and the efficiency.

Most power distribution strategies for the hybrid electric vehicles have been developed for general HEVs whose powers sources consist of an internal combination engine and a battery. However, along with the popularization of fuel cell hybrid electric vehicles (FCHEV), research on the development of the power management strategies for the fuel cell hybrid electric vehicles is increasing in recent years. The power source of the FCHEV is usually composed of a combination of a fuel cell stack and a battery, and compared with a general hybrid vehicle, it does not emit exhaust gas at all, have simplified power train configuration, and achieves high efficiency. However, since the fuel cell stack for of the FCHEV is vulnerable to durability problems, it is necessary to develop the energy management strategy in consideration of the deterioration of the fuel cell stack. The power distribution problem for the FCHEV belongs to a multi-objective problem that needs to be solved by considering both the degradation of the fuel cell stack and the fuel consumption. Due to these characteristics, the power distribution strategy for the FCHEV is more complicated than the power distribution strategy for the HEV that optimizes only in terms of the fuel consumption. Nevertheless, the development of the power management strategy for the FCHEV has not been much researched compared to the development of the energy management strategy for the general HEV that have already been popularized.

In this study, the power distribution strategy for the FCHEV was developed using the reinforcement learning. The reinforcement learning has recently made great progress through convergence with deep neural networks and deep reinforcement learning which combines the reinforcement learning and the deep neural networks has been proven through many studies. We developed the power distribution strategy that optimizes the degradation of the fuel cell stack and fuel consumption by utilizing the deep reinforcement learning (DRL) based on the Actor-Critic algorithm. Since the DRL derives the control strategy based on the current

state, it has the advantage of high generalization performance and is also very excellent in terms of scalability. By utilizing the high scalability of the deep reinforcement learning, the power distribution strategies for various hybrid systems can be developed through the same learning framework. In addition, the DRL has the advantage of being able to respond to the degradation of the fuel cell stack that occurs in real time through self-learning and online-learning. In this study, we developed the energy management strategy for the FCHEV that secure both the generalization performance and the scalability by utilizing all the advantages of the deep reinforcement learning. In addition, we developed a methodology that efficiently updates the existing DRL model based on the online learning.

Keyword : fuel cell hybrid electric vehicle (FCHEV), power distribution strategy, reinforcement learning, Markov decision process, equivalent consumption minimization strategy (ECMS), dynamic programming (DP), deep neural network

Student Number : 2014-22488

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

1.1. Hybrid Electric Vehicles

The electrification and hybridization of vehicles are accelerating due to increasingly strengthened regulations for exhaust gas. Figure 1 shows the emission regulation trend for the vehicles in Europe [1]. The emission regulations have demanded a sharp reduction in vehicle emissions whenever the regulation is changed from Euro I to Euro VI. Hybrid electric vehicles (HEVs) are one of the most effective alternatives to tightening emissions regulations in the current situation, where this trend towards emissions regulation is expected to continue.

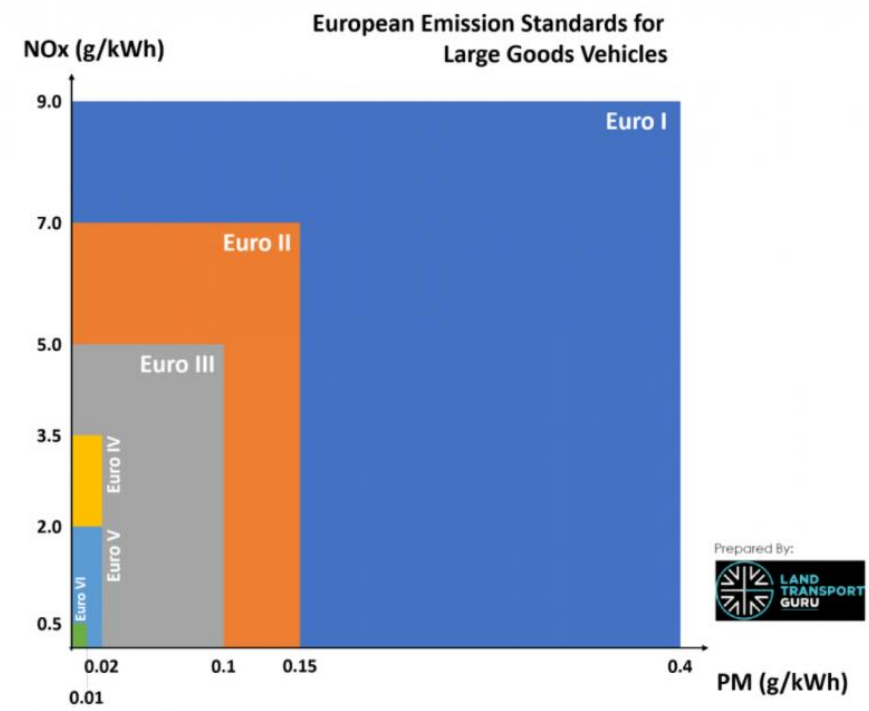


Figure 1. European Emission standards trend [1]

In a general hybrid electric vehicle, the power source is composed of an internal combustion engine and a battery, and the power by the two power sources is controlled by the vehicle's power management controller. As such, the HEV is a system having two or more power sources, so there is a large difference in efficiency according to the method of the distributing the driver's required power to the power sources. For this reason, many studies on the development of the power distribution strategies for HEVs have been conducted.

The development of power distribution strategies is largely being studied based on three theories including rule-based control, optimal control theory, and reinforcement learning. In a study based on rule-based control theory, a driving mode according to driving conditions is designed based on human experience and knowledge [2-5]. Rule-based theory has a low computational amount and does not consider future driving conditions for control, so its applicability to actual vehicles is high, but the effect of improving fuel economy is small. In addition, there is a disadvantage in that the scalability is small in that the control strategy is constructed based on human experience and knowledge. In addition, it has the disadvantage of small scalability in that the control strategy is constructed based on human experience and knowledge. The power distribution strategy based on the optimal theory can be divided into the control strategy based on dynamic programming (DP) that guarantees a global optimum and the control strategy based on real-time optimization theory. The DP-based power distribution strategy can guarantee global optimality because it derives optimal control by considering all the control cases based on the bellman equation [6-10]. However, the applicability of the actual vehicle is low, since the DP-based control strategy requires a large amount of computation and the future driving information must be reflected in the control. And the power distribution strategy based on real-time optimization theory is being studied mainly using Pontryagin's minimum principle (PMP) and equivalent

consumption minimization strategy (ECMS) [11-16]. The essential pursuit of both studies is to derive a control value which minimizes the cost function that integrates the electrical energy and fuel consumption. Therefore, it is important to derive an appropriate value for co-state, a kind of equivalent factor that converts electrical energy into fuel consumption in the cost function. Kim et al. has been proven through previous studies that when the co-state is properly set, results corresponding to the global optimum can be obtained [17]. The control strategy based on real-time optimization theory has a small amount of computation, but the strategy has a disadvantage of poor generalization performance because the co-state is a variable dependent on the driving cycle. The strategy based on reinforcement learning is formulated in the Markov decision process (MDP) as shown in Figure [2]. The action corresponding to the control value is derived only through the current state and does not require a future state. Therefore, the reinforcement learning-based control strategy can secure high generalization performance and real vehicle applicability in that it derives the control value only through the current state, but it is generally less efficient than the strategy based on the optimal control theory [18-24].

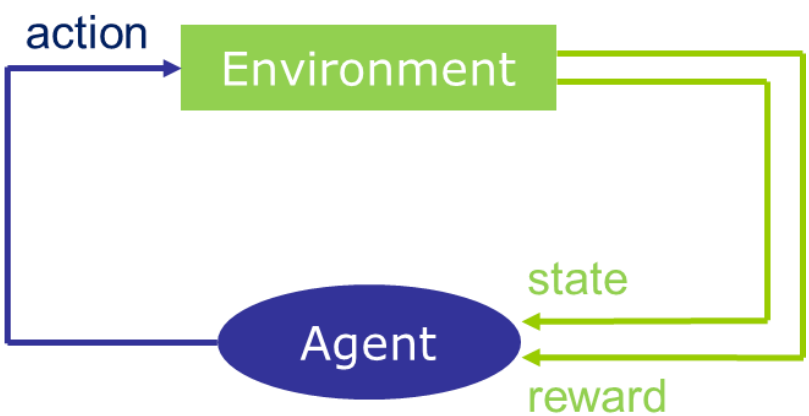


Figure 2. Schematic of the Markov decision process

Recently, reinforcement learning has made great progress through the fusion of deep artificial neural networks [25]. Deep reinforcement learning (DRL), which combines the theory of reinforcement learning with a deep artificial neural network, has proven that it can effectively solve complex problems through many studies [26, 27]. Recently, some studies to develop control strategies for the HEVs using the DRL algorithm have been conducted and the DRL algorithm has been proven to be effective in developing the energy management strategy for the HEV [28-36].

1.2. Fuel Cell Hybrid Electric Vehicles

As the fuel cell hybrid electric vehicles (FCHEV) are beginning to be released on the market, research on the power distribution problem for the FCHEV is also actively progressing. The power source of the FCHEV is usually composed of a combination of fuel cell stack and battery. Like the HEV, the power management controller performs energy management between the two power sources. The FCHEV is attracting attention as an eco-friendly vehicle of the future because it can eliminate or simplify the transmission system and does not emit exhaust gas.

However, FCHEV's fuel cell stack has the disadvantage of being vulnerable to durability. The deterioration of the fuel cell stack is caused by a combination of various causes, such as reduction of the surface area of catalysts, mechanical stress, and contamination [37]. The degradation of the fuel cell stack is considered one of the biggest obstacles to the popularization of the FCHEV. US department of energy (DOE) estimates that the operation time required for the popularization of the FCHEV is 5000 hours, but most of the FCHEVs currently fail to achieve this goal.

Since the degradation of the fuel cell stack has a large effect on the FCHEV, research to investigate the deterioration of the fuel cell stack is also ongoing. One of them is a study to analyze the degradation of the fuel cell through a physical deterioration model [38]. The physical model does not require experimental data and has the advantage of high generalization performance, but the physical model that can sufficiently represent the degradation has not been developed due to the complexity of the degradation phenomenon. To overcome the limitations of this physical model, some studies was conducted to construct a degradation model for the fuel cell stack of the FCHEV based on the actual driving data [39, 40]. In the representative study (H. Chen, 2015), the degradation model according to the operating mode of the fuel cell stack was developed and verified through actual vehicle driving data [39].

As such, the FCHEVs is more sensitive to the durability than general HEVs, so it is necessary to develop the power distribution strategy in consideration of both the fuel consumption and the degradation. Therefore, the problem of the energy management for the FCHEV should be extended to the multi-objective problem as shown in Eq (1). $\dot{m}_{fc}$ and L represent the fuel consumption and the voltage drop of the fuel cell stack due to the deterioration, respectively, and x and u represent state and control, respectively. The boundary condition for Eq (1) is represented in Eq (2), where $SOC(t_f)$ and $SOC(t_o)$ mean the final state of charge and the initial state of charge, respectively. And the secondary condition for the problem is as Eq (3). P, ω, and T represent power, rotational speed, and torque, respectively, and the notation $(\cdot)_{fcs}$, $(\cdot)_{bat}$, and $(\cdot)_{mot}$ represent fuel cell system, battery, and motor, respectively.

$$\min \sum \dot{m}_{fc}\big(x(k), u(k)\big) + L\big(x(k), u(k)\big) \tag{1}$$

$$SOC(t_f) = SOC(t_0) \tag{2}$$

$$\begin{aligned} P_{fcs,min} \le P_{fcs}(t) \le P_{fcs,max} \\ P_{bat,min} \le P_{bat}(t) \le P_{bat,max} \\ \omega_{mot,min} \le \omega_{mot}(t) \le \omega_{mot,max} \\ T_{mot,min} \le T_{mot}(t) \le T_{mot,max} \end{aligned} \tag{3}$$

In other words, FCHEV's power distribution strategy should ensure SOC sustain-ability in the system's operable area, while minimizing fuel consumption and the deterioration of the fuel cell stack. Studies on the development of strategies to minimize the degradation of the fuel cell stack and the fuel consumption are continuously being carried out, and among them, many studies are being conducted based on the optimal control theory [41-45].

Chapter 2. Research Background

2.1. Deep Reinforcement Learning

Q-learning, a type of reinforcement learning, is a representative model-free algorithm. The purpose of Q-learning is to learn the optimal policy in which the agent derives the optimal action corresponding to an arbitrary state in the Markov decision process as shown in Figure 2. "Q" in Q-learning symbolizes the quality of the action taken in the current state, and the quality is quantified through Q-value. Q-value is defined by Eq (4) as follows, where π, s, a, and R represent policy, state, action, and reward, respectively, and ρ represents a discount factor. The discount factor, which has a value between 0 and 1, is a value designed in terms of mathematical convenience and the present value is greater than the future value. The Q-value is optimized through recursive execution of the bellman optimality equation such as Eq (5), and an optimal policy such as Eq (6) is derived.

$$Q_\pi(s,a) = E_\pi[R_{t+1} + \rho R_{t+2} + \rho^2 R_{t+3} + \cdots \mid S_t = s,\ A_t = a] \tag{4}$$

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha \cdot \left(r_t + \rho \max_a Q(s_{t+1},a) - Q(s_t,a_t) \right) \tag{5}$$

$$\pi^*(s) = \underset{a}{\operatorname{argmax}}\, Q^*(s,a) \tag{6}$$

The biggest weakness of Q-learning optimized by value-evaluation is that it is difficult to apply to complex problems with large size of the states and actions. However, through the deep Q-network (DQN) algorithm (V. Mnih, 2015), a groundbreaking study that applied a deep neural network to the framework of Q-learning, the field of reinforcement learning was able to achieve rapid development [25]. In this study, the state was defined as four 84×84 gray scale stack images, so

the state has a very large size of $256^{84 \cdot 84 \cdot 4} (\approx 10^{67970})$. However, in this study, the problem was effectively solved by using an agent based on a convolutional neural network that acts as a function approximator on the framework of the Q-learning. In the DQN algorithm, the weights θ^Q of the deep neural network are optimized through a backpropagation algorithm to minimize loss such as Eq (7).

$$Loss(\theta^Q) = [y_t - Q(s, a|\theta^Q)]^2$$
$$\because y_t = r_{t+1} + \rho \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}|\theta^{Q'}) \tag{7}$$

The DQN introduces two methods to stabilize the learning algorithm. One was to solve the temporal dependency problem of mini-batch by developing a replay memory that stores experiences used for the training of the agent. And the other one was to stabilize learning by creating a separate target Q-network with weights $\theta^{Q'}$ to derive the target Q-value. Through this, it was proved that the DQN algorithm can achieve successful results in many game environments of Atari emulator.

However, DQN has the disadvantage that it cannot derive continuous action values. To compensate for the shortcomings of the DQN algorithm, a DRL model using a policy gradient algorithm was developed. A representative study among them is the deep deterministic policy gradient method based on the actor-critic (A2C) architecture (T. P. Lilliccrap, 2016). The architecture of the actor-critic based DRL is shown in Figure [3]. Both actor and critic are composed of the deep neural network. The actor derives action for the state, and the critic performs value approximation of the actor's action for the state. The training for the actor is progressed based on the policy gradient method as in Eq (8), and θ^A, θ^C means the weights of the actor network and the critical network. And the learning for the critic proceeds in the same way as the training of the DQN like Eq (9), where $\theta^{A'}, \theta^{C'}$ represents the weights of the target actor network and the target critic network.

It has been demonstrated in several papers that the DRL can secure scalability, the ability to solve various kinds of problems through the similar training framework [25, 27]. In addition, the DRL is easy to apply online-learning algorithm because it has the characteristic that the training of the agent proceeds based on one's own experience. This study develops the effective power management strategy for the FCHEV by utilizing the advantages of the DRL algorithm.

$$\begin{aligned} \nabla_{\theta^A} J &= \frac{1}{N}\sum \nabla_{\theta^\mu} Q(s, \mu(s|\theta^A)|\theta^C) \\ &= \frac{1}{N}\sum \nabla_a Q(s, a|\theta^C) \nabla_{\theta^A} \mu(s|\theta^A) \end{aligned} \tag{8}$$

$$\begin{aligned} Loss(\theta^C) &= [y_t - Q(s, a|\theta^C)]^2 \\ \because y_t &= r_{t+1} + \rho Q(s_{t+1}, \mu(s_{t+1}|\theta^{A'})|\theta^{C'}) \end{aligned} \tag{9}$$

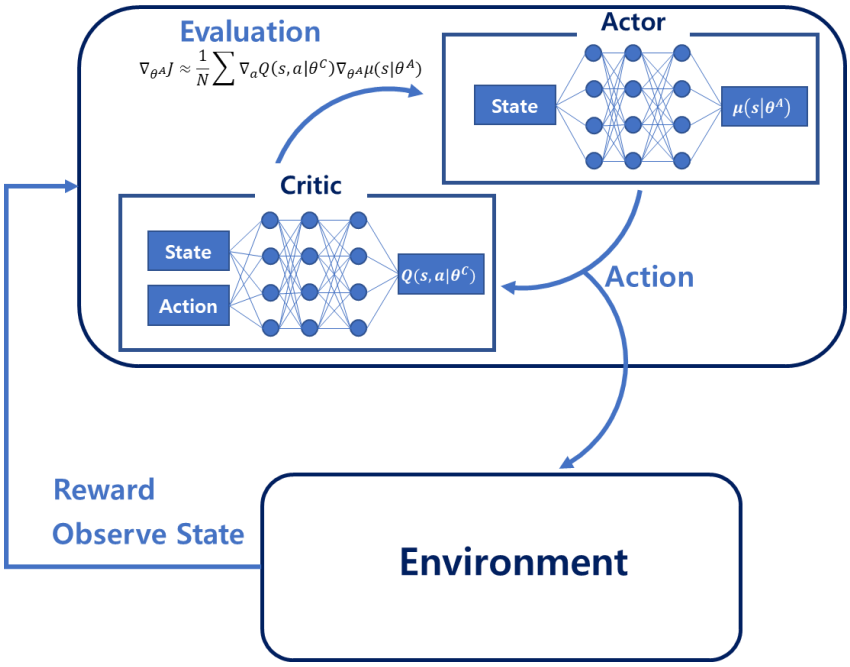


Figure 3. Schematic of the actor-critic model architecture

2.2. Existing studies

We investigated research related to the power management strategy of the hybrid electric vehicle. Related studies were investigated in four aspects including "stack degradation", "online-learning", "generalization performance" and "scalability". Since the power system in the FCHEV is highly affected by the stack degradation, it is important to consider the stack degradation in developing the power distribution strategies. Since the power system of the FCHEV continuously changes due to the stack degradation, it is important to develop a methodology that can adapt to system changes. Therefore, it is necessary to apply the online-learning methodology in developing the power management strategy for the FCHEV. In addition, it is very important that the control strategy guarantees the generalization performance, as the control strategy must be effective for all driving profiles. Now that the structure and types of the HEVs are diversifying, the development of the power management strategy that guarantees the scalability has great significance in the industrial aspect. Therefore, the recent related studies were analyzed focusing on the four factors that become issues in the development of the control strategies.

The power distribution strategies for the fuel cell hybrid electric vehicles have been mainly studied based on the optimal control theory. W. Zou et. al. conducted a study to optimize the durability and fuel efficiency of the FCHEV through the dynamic programming, but the aspects of the generalization, the scalability, and the online-learning were not considered in the development of the power distribution strategy [7]. H. Li et. al. developed a power distribution strategy considering the degradation of power sources using the ECMS [12]. In addition, an online learning methodology was developed that updates the equivalent factor based on the deterioration of the power sources in the study. However, the research did not develop the control strategy that considers the scalability, and since the update of the equivalent factor is based on the degradation model of the power sources, the

effectiveness of the online-learning algorithm is decreased if the accuracy of the degradation model for the power sources is not guaranteed. C. Geng et. al. developed a rule-base control strategy for the FCHEV using fuzzy logic algorithm [3]. The study did not consider the degradation of the fuel cell stack and did not consider the scalability and the online-learning algorithm in developing the power distribution strategy.

There has been little research on the development of the DRL-based control strategies for the FCHEVs, and the power distribution strategy using the DRL algorithm has been applied to general HEVs composed of the internal combustion engine and the battery. X. Han et. al. developed a power distribution strategy based on double deep Q-network (DDQN) that derives discrete actions for the hybrid electric tracked vehicles [33]. In the study, the power management strategy was developed focusing on securing the generalization performance. H. Tan et. al. developed a power distribution strategy for the HEV that can derive continuous action through the deep deterministic policy gradient (DDPG) algorithm [34]. In the study, the power distribution strategy was developed focusing on the generalization aspect, and the remaining three items were not considered. Y. Hu developed a control strategy based on DDQN for the HEV, and they considered the generalization aspect and application of the online-learning algorithm in developing the control strategy [36]. However, the control strategy was developed without considering the degradation and the scalability.

2.3. Research Motivation

The general power distribution strategy for the actual HEV is produced as a map in a simple look-up table format due to the vehicle's limited computing power. This map-based strategy is made based on the results of the optimal control theory and the experience and knowledge of experts. Since the power distribution strategy is developed based on human experience and subjectivity, it has the disadvantage of low generalization performance and scalability.

However, considering the reality that the computation power of vehicles is rapidly increasing due to the advent of autonomous vehicles, electrification of vehicles, and development of cloud computing technology, it is necessary to develop advanced power distribution strategies for the hybrid electric vehicles. In this study, DRL model was used to develop the advanced energy management strategies for the FCHEVs.

DRL has proven its high scalability through several studies, and the DRL has the characteristic of self-learning in which the training is performed based on one's own experience, so the performance of existing models can be improved through the online-learning. In particular, the DRL-based control strategy can cope with to system changes caused by the degradation of the fuel cell stack through the characteristics that online-learning for the DRL is easy. As such, DRL is a very effective theory in relation to the development of the control strategies for FCHEVs, but the DRL-based energy management for FCHEVs has not yet been developed. Also, there is hardly any research that has developed the power management controller in consideration of the scalability aspect.

In this study, a comprehensive and systematic study was conducted in developing the power management strategy based on the DRL algorithm. Through this research, we developed the power distribution strategy that not only excels in generalization performance, but also secures high scalability and online-learning.

Chapter 3. Research and Results

3.1. Overview of Research Framework

Figure [4] shows the overall research framework of this study. In this study, the power distribution strategy was developed using the actor-critic structure DRL model that can derive continuous action. The environment of MDP is composed by vehicle model and driver's driving pattern. Agent is composed of deep neural network of actor-critic architecture. The environment of the MDP is composed by vehicle model and driver model (driving cycles). And the agent is composed of the deep neural network with the actor-critic architecture. The actor of the agent derives the action for the state, and the environment derives the reward and the next state corresponding to the action. In the DRL, tuple composed of (state [s], action [a], reward [r], next state [s']) is defined as experience. These experiences are stored in the replay memory and used for the training process of the agent. The specific DRL algorithm is described in Algorithm [1].

Regarding the method of selecting an action during the training, we used the ε-greedy method, which is a representative exploration strategy. The agent conducts the exploration that randomly selects an action without depending on the policy according to the exploration probability ε in the ε-greedy method. Through this method of the agent exploration, it is possible to prevent the policy having the local maximum performance from being trained. The training algorithm is designed such that the exploration probability decreases as the learning progresses. Therefore, the agent actively performs exploration at the beginning of the training but hardly conducts exploration at the end of the training. This chapter consists of three parts. In 3.1.1, we introduce the definition of the state, the action, and the reward. In 3.1.2, the structure of the neural network used in the actor-critic model is explained, and in 3.1.3, the target FCHEV model is introduced.

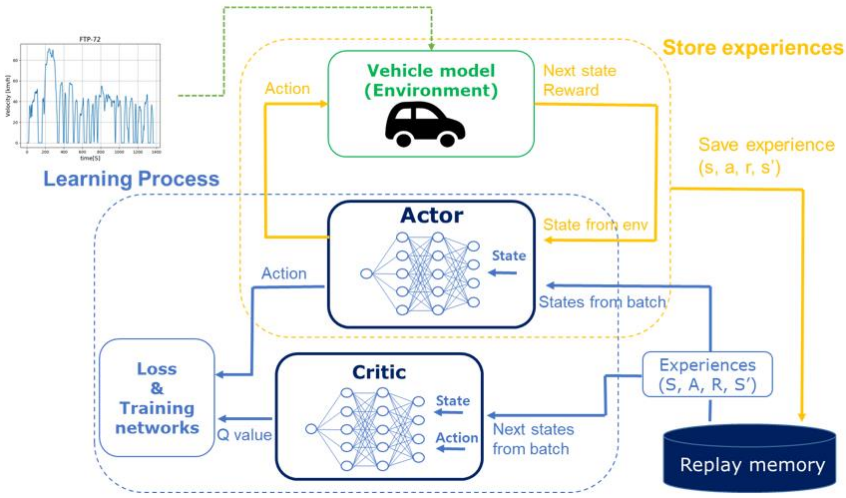


Figure 4. Schematic diagram of the research framework

Algorithm 1.

Randomly Initialize critic network $Q(s,a|\theta^C)$ and actor $\mu(s|\theta^A)$ with θ^C and θ^A

Initialize target network Q' and μ' with $\theta^{C'} \leftarrow \theta^C$ and $\theta^{A'} \leftarrow \theta^A$

For episode =1 to M do:

Receive initial observation state s_0

For t=1 to T do:

Select action $a_t = \mu(s_t|\theta^A)$ according to $\varepsilon - greedy$ algorithm

Execute action a_t and observe reward r_t and new state s_{t+1}

Store experience (s_t, a_t, r_t, s_{t+1}) in the replay memory

Calculate target Q-value, $y = r_t + \rho Q'(s_{t+1}, \mu'(s_{t+1}|\theta^{A'})|\theta^{C'})$

Update critic by minimizing the loss, $L = \frac{1}{N}\sum_i \left(y_i - Q(s_i, a_i|\theta^Q)\right)^2$

Update the actor using the policy gradient:

$$\nabla_{\theta^A} J = \frac{1}{N}\sum \nabla_a Q(s,a|\theta^C)\nabla_{\theta^A}\mu(s|\theta^A)$$

Update the target networks:

$$\theta^{A'} \leftarrow \tau\theta^A + (1-\tau)\theta^{A'}$$

$$\theta^{C'} \leftarrow \tau\theta^C + (1-\tau)\theta^{C'}$$

End for

End for

3.1.1. Definition of the state, action, and reward

The state, the action, and the reward are the main elements of the MDP and need to be defined appropriately for the problem situation. This part explains how the state, the action, and the reward are defined.

The power of the fuel cell stack according to the current density is expressed in Eq (10), where $j, A, n, V(\cdot)$ denote the current density, area of cells, number of cells, and cell voltage, respectively. And like Eq (11), the battery power P_{bat} is calculated as the difference between the demanding power P_{dmd} and the fuel cell stack power when the power of the fuel cell stack is determined. Therefore, in this study, the current density of the fuel cell stack is defined as the agent's action.

$$P_{fc} = (jA) \cdot (nV(j)) \tag{10}$$

$$P_{bat} = P_{dmd} - P_{fc} \tag{11}$$

State is information representing the current situation and is an input necessary for the agent to derive the action. In this study, the state was constructed based on four factors such as Eq (12). The P_{dmd} means the required power of the vehicle, and ΔSOC represents the SOC deviation between the initial SOC, SOC_{init} and the current SOC as in Eq (13). j_{min} and j_{max} are the minimum current density and the maximum current density of the fuel cell stack according to the demanding power. The j_{min} and the j_{max} are expressed as Eq (14).

$$\mathrm{s} = [P_{dmd}, \Delta SOC,\ j_{min,}\ j_{max}] \tag{12}$$

$$\Delta SOC = SOC - SOC_{init} \tag{13}$$

$$j_{min} = \min_{j} f(P_{dmd}, j) \tag{14}$$

$$j_{max} = \max_{j} f(P_{dmd}, j)$$

The reward is a mathematical expression that is transmitted to the agent so that the training of the agent can be made appropriate to the defined problem. Therefore, it is important to define the appropriate type of reward in order to derive the appropriate agent. In this study, as Eq (15), the reward was designed by considering the fuel consumption rate and the battery SOC. The γ in Eq (15) serves as a kind of equivalent factor that equalizes the absolute value of the battery SOC deviation and the fuel consumption rate. The reward is divided into a part related to fuel consumption and a part related to SOC deviation. We tried to create the agent that secures the SOC sustain-ability while minimizing the fuel consumption through this reward composition.

$$\mathrm{R} = -(\dot{m}_{\mathrm{fc}} + \gamma|\Delta SOC|) \tag{15}$$

3.1.2. Neural network structure in the actor-critic model

The neural network structure for the actor and the critic is shown in Figure [5]. The actor network is responsible for mapping the action from the state. The actor network has two hidden layers, both of which are made up of 512 neurons and the activation function is defined as a rectified linear unit (ReLU) such as Eq (16). In Eq (16), z denotes a value derived by a linear combination of layer weights and input values. The activation function of the actor's output layer is designed as the sigmoid function, and the mathematical expression for the sigmoid function is as Eq (17).

$$h(z) = max\,(0, z) \tag{16}$$

$$\mathrm{h(z)} = \frac{1}{1 + e^{-z}} \tag{17}$$

The output layer of the actor has a value between 0 and 1 by the sigmoid activation function, and we designed the agent's action as Eq (18) by utilizing the characteristics of this activation function. That is, the physical meaning of the output of the actor network, $h_{out}(z)$ can be viewed as a ratio of the current density of the action and the maximum current density value.

$$a = j_{max} \cdot h_{out}(z) \tag{18}$$

Critic network receives the state and the action information and calculates the value for the action corresponding to the state. In the critic network, the representation feature for the state and for the action are individually derived and then integrated into one layer through a concatenate layer. And the activation function of the output layer is composed of a linear function. Therefore, the output value is derived through the linear combination of the weights of the layer and the input values.

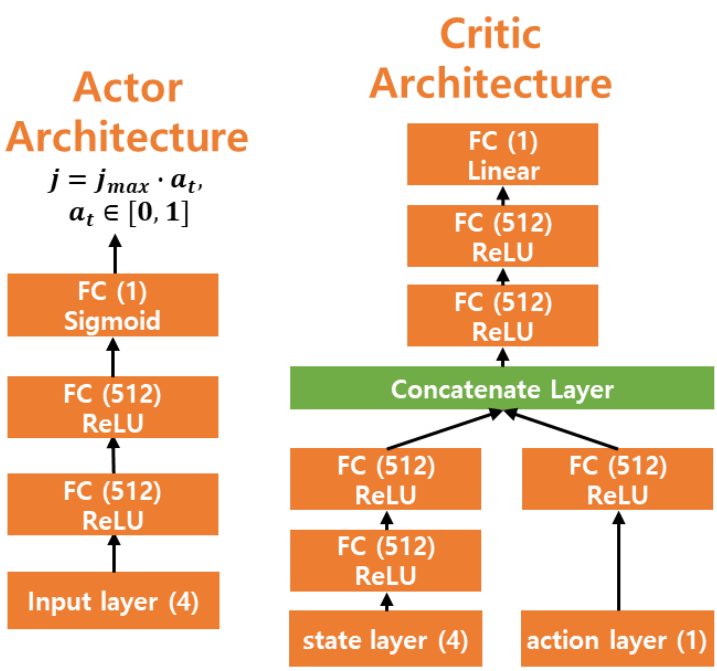


Figure 5. Architecture for the actor network and the critic network

3.1.3. FCHEV model

In this study, the FCHEV consisting of a combination of the fuel cell stack and the battery as the power source was set as the target vehicle. The structure of the vehicle to be studied is shown in Figure [6], and the main specifications of the target vehicle are specified in Table 1. The maximum power of the fuel cell stack and the maximum power of the battery are 67kW and 39kW, respectively, and the mass of the vehicle is set to 1200 kg.

The fuel cell model was developed by referring to 1-D proton exchange membrane fuel cell (PEMFC) flux balance modeling (F. Prinz, Cha, 2014) [46]. The fuel cell stack system includes components such as stack and compressor, and the area and number of cells in the stack are 200cm^2, 400, respectively. And in this study, considering the stability of the stack, the maximum value of the current density was set to 1.0A/cm^2 and the idling current density was set to 0.001 A/cm^2. The detailed PEMFC and fuel cell stack system model are described in Appendix 1.

The battery model was developed based on the equivalent open-circuit voltage model and internal resistance model expressed as a function of SOC. The required power of the battery is the same as Eq (19), where η_{mot} represents the motor efficiency. Given the battery power, the time derivative of SOC, $\dot{SOC}$ is expressed as Eq (20), where Q_{bat}, R_{bat}, and V_{OC} represent battery capacity, internal resistance of the battery, and battery open-circuit voltage, respectively.

$$\begin{aligned} P_{bat} &= P_{dmd} - P_{fc} \\ &= \eta_{mot}^{-sgn(T_{mot})} \cdot T_{mot} \cdot \omega_{mot} \end{aligned} \tag{19}$$

$$\dot{SOC} = -\frac{1}{Q_{bat}} \frac{V_{OC}(SOC) - \sqrt{V_{OC}(SOC) - 4P_{bat}R_{bat}(SOC)}}{2R_{bat}(SOC)} \tag{20}$$

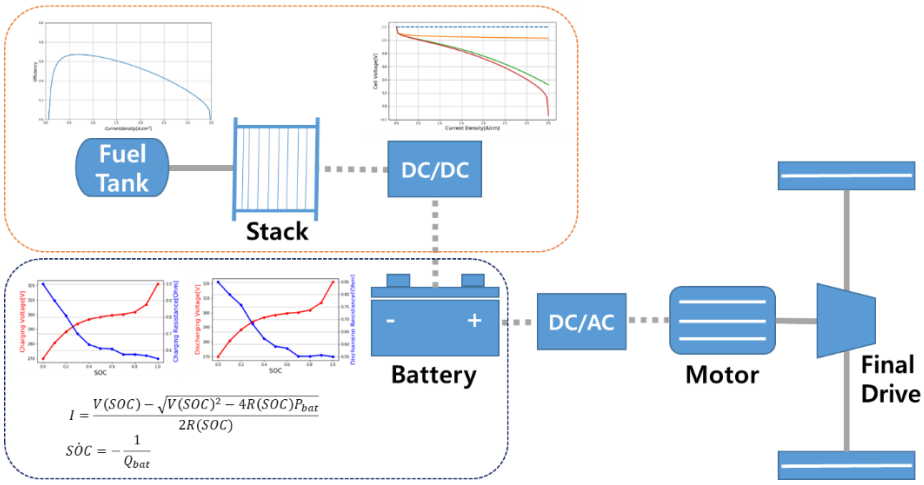


Figure 6. Configuration for the research target vehicle

Table I. Target FCHEV model specification

Properties	Values
Maximum stack power [kW]	67
Maximum battery power [kW]	39
Vehicle mass [kg]	1200
Tire radius [m]	0.337
Final gear ratio	3.648
Initial SOC	0.6
Efficiency of the final drive [%]	96
Efficiency of the converter [%]	97

Figure [7] shows the stack system modeled in this study. On the anode side where hydrogen is supplied, pressure, temperature, and humidity are controlled to a constant state through valve, humidifier, and heater. And on the cathode side, the pressure, temperature and humidity of the air are controlled in a certain state through the compressor, the value, the humidifier, and the heater.

The power generated from the fuel cell stack system is expressed as Eq (21), P_{FC} represents the output power from the stack, and P_{AUX} represents the power consumed by auxiliary equipment belonging to the mechanical balance of plant (MBOP).

$$P_{sys} = P_{FC} - P_{AUX} \tag{21}$$

The power generated from the stack is expressed as Eq (22), and the stack voltage, V_{stack} and stack current, and i_{stack} can be expressed as NV_{cell} and jA, respectively. N, V_{cell}, and A denote the number of cells, cell voltage, and effective cell area in the stack, respectively.

$$P_{FC} = V_{stack} \cdot i_{stack} = (NV_{cell}) \cdot (jA) \tag{22}$$

The compressor that compresses the air supplied to the cathode side belongs to the main auxiliary equipment in the stack system. The power supplied to the compressor is represented as Eq (23), where η_{comp} means compressor efficiency, C_p means heat capacity of air, γ means ratio of the specific heat of air, and $\dot{m}_{comp}$ means air flow rate. In addition, the air flow rate is expressed as a function of the current density as in Eq (24), where M_{air} is molecular weights of the air.

$$P_{comp} = \frac{C_p T_{air}}{\eta_{comp}} \left(\left(\frac{P_{out}}{P_{in}} \right)^{\frac{\gamma-1}{\gamma}} - 1 \right) \cdot \dot{m}_{comp} \tag{23}$$

$$\dot{m}_{comp} = M_{air} \frac{N \cdot i_{stack}}{4 x_{o_2} F} \tag{24}$$

Also, we constructed the stack system by assuming that the power consumed by the auxiliary equipment except the compressor is constant at 3kW.

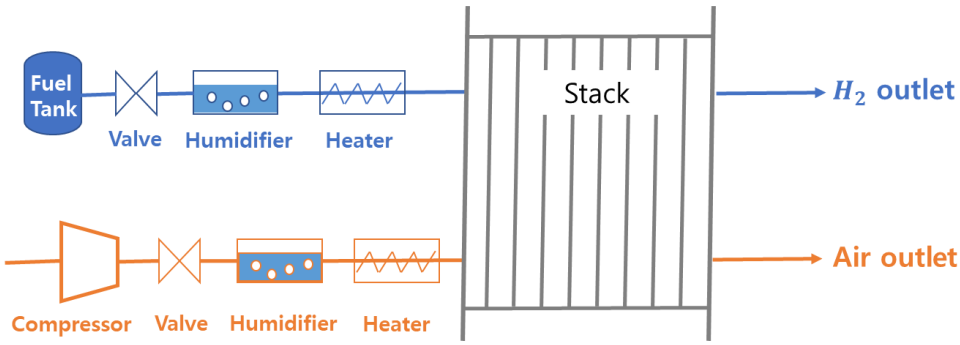


Figure 7. Schematic diagram of the stack system

3.2. Study for the Stability and the Scalability

The reinforcement learning differs from supervised learning in which a target value is given in advance in that training of model is performed based on one's own experience. Due to these characteristics, DRL is classified as semi-supervised learning, and since the training of the DRL is very difficult compared to general supervised learning, it is important to derive training conditions for stable learning.

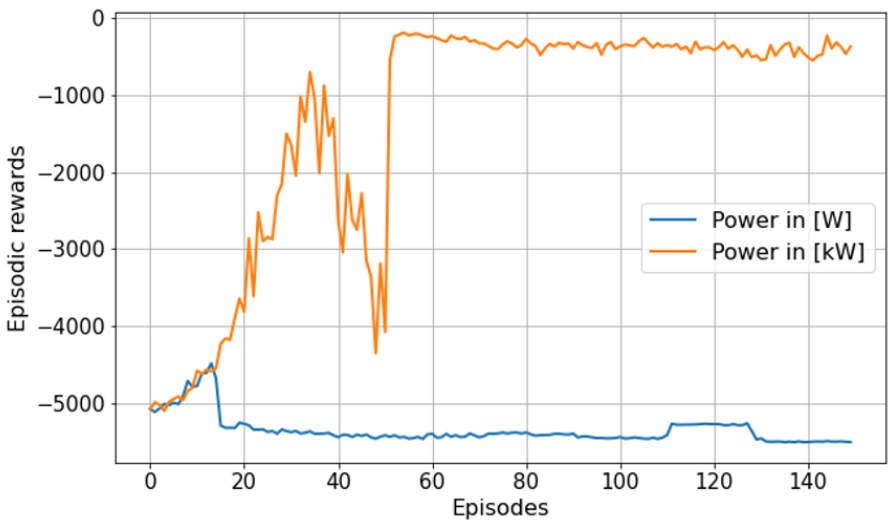


Figure 8. Trend of episodic reward according to the difference in units of demanding power

Figure [8] shows episodic rewards according to the episode. The episodic reward refers to the total reward which the agent has earned in one episode. In this study, the start and end of the driving for the FCHEV model are defined as one episode, and episodic reward refers to the sum of the rewards the agent gets in one episode. Figure [8] clearly shows the difficulty of learning process of the DRL. In the defined state, SOC and current density constraints are normalized values for battery capacity and cell area, respectively, and mainly have values between 0 and 1. On the other hand, demanding power is not a normalized value, and it has a value

within about five order of magnitude in the case of the required power represented in unit of W. And if the required power is expressed in units of kW, it has a value within about 2 order of magnitude. In Figure [8], when the demanding power of the state is expressed in units of W, it can be confirmed that the learning of the agent is unstable, which is caused by a large deviation between the state features.

As such, it is important to properly process the state because the learning stability of the DRL is determined according to the processing method for the state. The method of converting the unit to reduce the deviation between the features of the state has low scalability because the size of the demanding power is different for each problem situation. In order to ensure scalability and increase the stability of the learning, it is necessary to normalize the required power. We approached the above problem through two methodologies.

The first normalization method is shown in Figure [9]. Whenever a new experience is stored in the replay memory, the running mean and standard deviation of the demanding power are updated as Eq (25) and Eq (26), where $\mu, \sigma,$ and N represent running mean, running standard deviation, and the number of experiences respectively. And the mini-batch used for the network training is normalized based on the running mean and the standard deviation like Eq (27).

$$\mu \leftarrow \mu + \frac{P_{dmd,i}}{N} \tag{25}$$

$$\sigma \leftarrow \sigma + \sqrt{\frac{\left(P_{dmd,i} - \mu\right)^2}{N-1}} \tag{26}$$

$$P_{norm} = \frac{P_{dmd} - \mu}{\sigma} \tag{27}$$

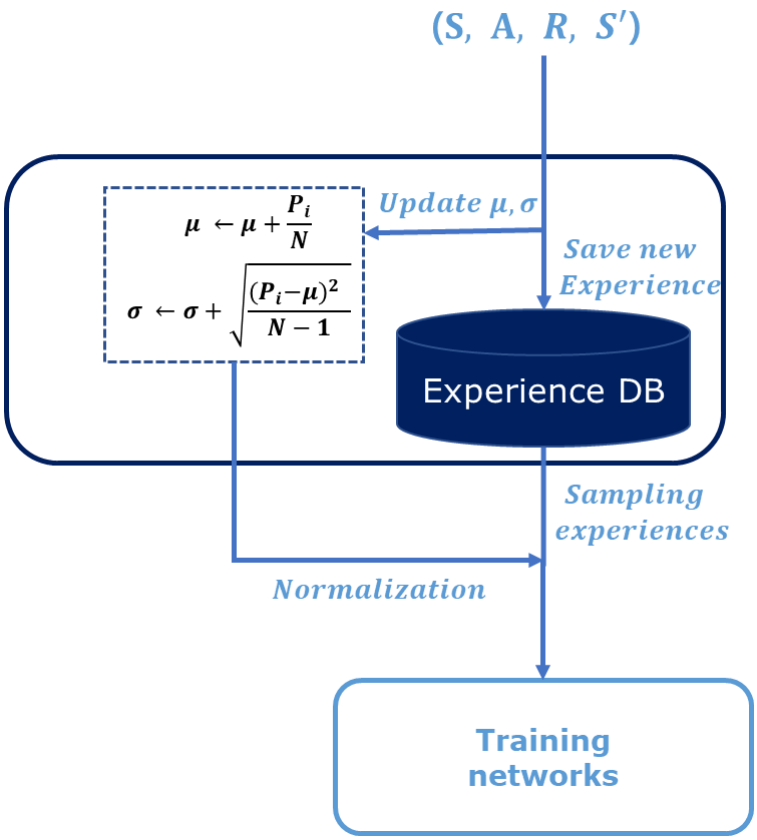


Figure 9. Normalization process by calculating the running mean and the running standard deviation for the demanding power in the state

The other normalization method is to use a batch normalization layer. The batch normalization layer normalizes the mini-batch by deriving the mean and standard deviation of the mini-batch and has weights for the scale and shift of the mini-batch [47]. Batch normalization layer has the advantage of being able to speed up the learning speed of the deep neural networks as well as normalization between features. In this study, the batch normalization layer was added to the existing actor-critic network as shown in figure [10]. We place the batch normalization layer right after the state input layer of the actor network and the critical network for state normalization.

We compared the learning results according to the four state representations, and Figure [11] shows the learning results according to the state representation. In terms of learning speed and convergence of episodic rewards, it can be seen from Figure [11] that the state representation that has normalized required power shows superior results compared to the other cases. In this study, state normalization was performed by using the batch normalization layer with the best reward convergence among the two normalization techniques.

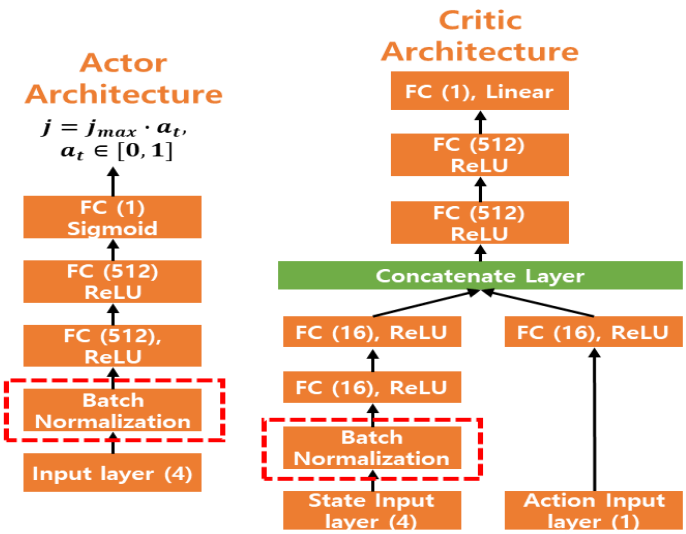


Figure 10. Actor-critic network architecture adding the batch normalization layer

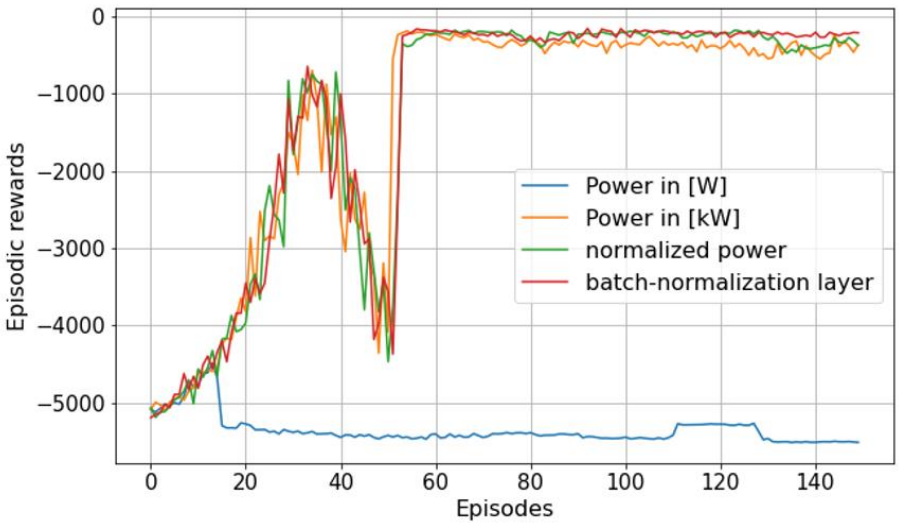


Figure 11. Simulation results with the state representation

In order to ensure the stability of the agent's learning, it is important to derive the hyper parameters. We determined the appropriate replay memory size and network size based on the simulation result according to the replay memory size and the simulation result according to the network size.

The number of experiences stored is determined depending on the replay memory, and the old experience is deleted as new experience is input. We compared the simulation results by setting the replay memory size to 500, 1000, 10000, and 100000. Figure [12] shows the simulation results according to the replay memory size. Figure [12] represents that when the replay memory size is low, such as 500 and 1000, stability of the training is not secured. In this study, the replay memory size was set to 100000 for stable training for the agent.

We derived the appropriate network size by adjusting the size of the last two hidden layers of the actor-critic network. The sizes of the last two hidden layers for the actor-critic agent were set to 16, 128, and 512, and the simulation results were compared. Figure [13] shows the simulation results according to the hidden layer size of the last two layers for the actor-critic agent. As shown in Figure [13], when the size of the hidden layer is set to 16, it can be confirmed that there is a part in which training of the agent becomes unstable. On the other hand, when the hidden layer size is set to 128 and 512, it can be confirmed that the training of the agent secures the learning stability. We set the hidden layer size to 128 so that the agent secures the learning stability while the agent does not require a lot of the computation power during the training and the inference process.

Therefore, the actor-critic agent used in this study has the structure as shown in Figure [14].

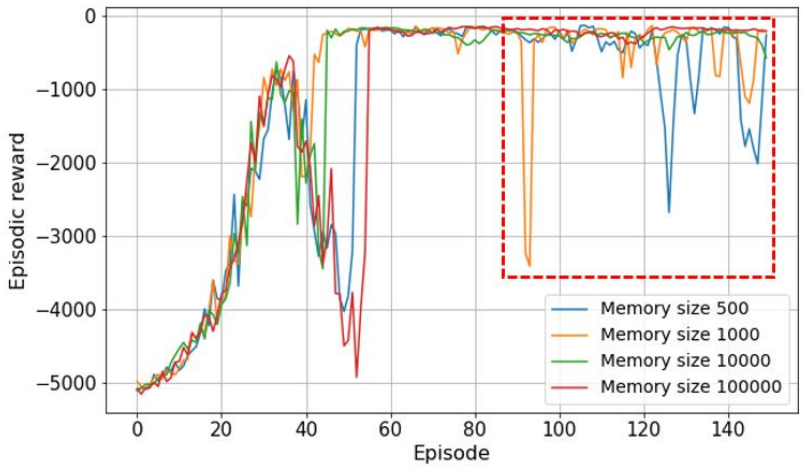


Figure 12. Simulation results with replay memory size

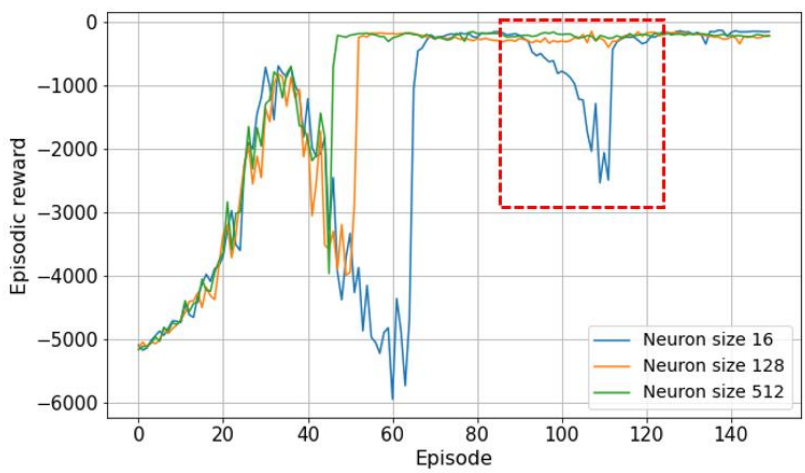


Figure 13. Simulation results with the last two hidden layer size

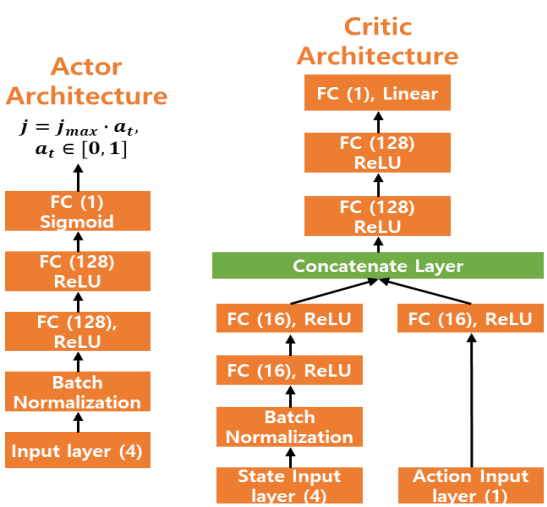


Figure 14. Actor-critic network architecture after the comparison experiment with hidden layer size

3.3. Learning Process for the DRL Agent

In order to derive a suitable agent, it is necessary to understand the training process of the agent. This chapter examines the training process of the agent and explains the development of a methodology to make the scalable power distribution strategy through such consideration.

3.3.1. Understanding of the learning process

The reward of this study is divided into two parts as shown in Figure [15] below. One is related to the fuel consumption and the other is related to the SOC deviation. We tried to analyze how the influence of the two terms changes with the learning process. Figure [16] shows the ratio of terms related to fuel consumption in the reward and the ratio of terms related to SOC deviation in the rewards according to episodes. The result shown in Figure [16] is the result derived by setting the reward factor, γ to 10. In Figure [16], the blue solid line indicates the proportion of the term related to fuel consumption to reward, and the orange solid line indicates the proportion of the SOC deviation related term to the reward. At the beginning of the training, the SOC deviation related terms occupy a dominant proportion of the total reward, but after about 40 episodes, it can be seen that the shares of the two terms in the reward become similar. In this study, the section in which the SOC deviation related terms exert the dominant influence was named “SOC dominant region”, and the section in which the two terms of the reward had similar weights was named “Training equilibrium region”.

$$R = \boxed{-\dot{m}_{fc}} \boxed{-\gamma\left|SOC - SOC_{ref}\right|}$$

Fuel consumption SOC Deviation

Figure 15. Configuration of the reward with two terms

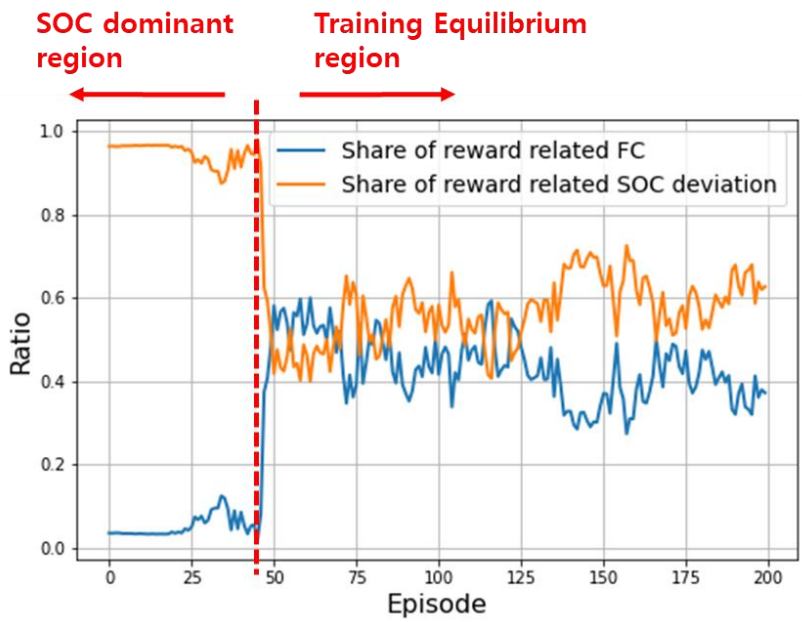


Figure 16. Changes in the share of the rewards of two terms according to the training

The SOC trajectories according to the episode are visualized in Figure [17] and the fuel consumption and the shares of the two terms to the reward are shown as Figure [18]. In Figure [17], (a), (b), (c), and (d) show SOC trajectory according to episode 1, episode 40, episode 80, and episode 120. And in Figure [18], (a) represents the fuel consumption according to the episode, and (b) shows the ratio of two terms of the reward according to the episode. From Figure [17]-(b), it can be seen that the fuel consumption is decreasing as shown in Figure [18]-(a) because the agent learns in a way that actively utilizes the battery at the beginning of learning. Figure [17]-(b) shows that the agent is trained in a way that actively utilizes the battery at the

beginning of the training. Due to this, it can be confirmed that the fuel consumption on the SOC dominant region is decreasing as shown in Figure [18]-(a). Therefore, the decrease in the fuel consumption on the SOC dominant region is a phenomenon that occurs in the process of the agent regulating the SOC trajectory. After about 50 episodes, as shown in Figure [17]-(c) and Figure [17]-(d), the agent reaches the training equilibrium region by limiting the SOC trajectory. Figure [18]-(a) shows that the fuel consumption in the training equilibrium region continuously decreases with the episodes. This is because the agent is trained in a way that maximizes the reward by regulating the SOC-trajectory and reducing the fuel consumption at the same time.

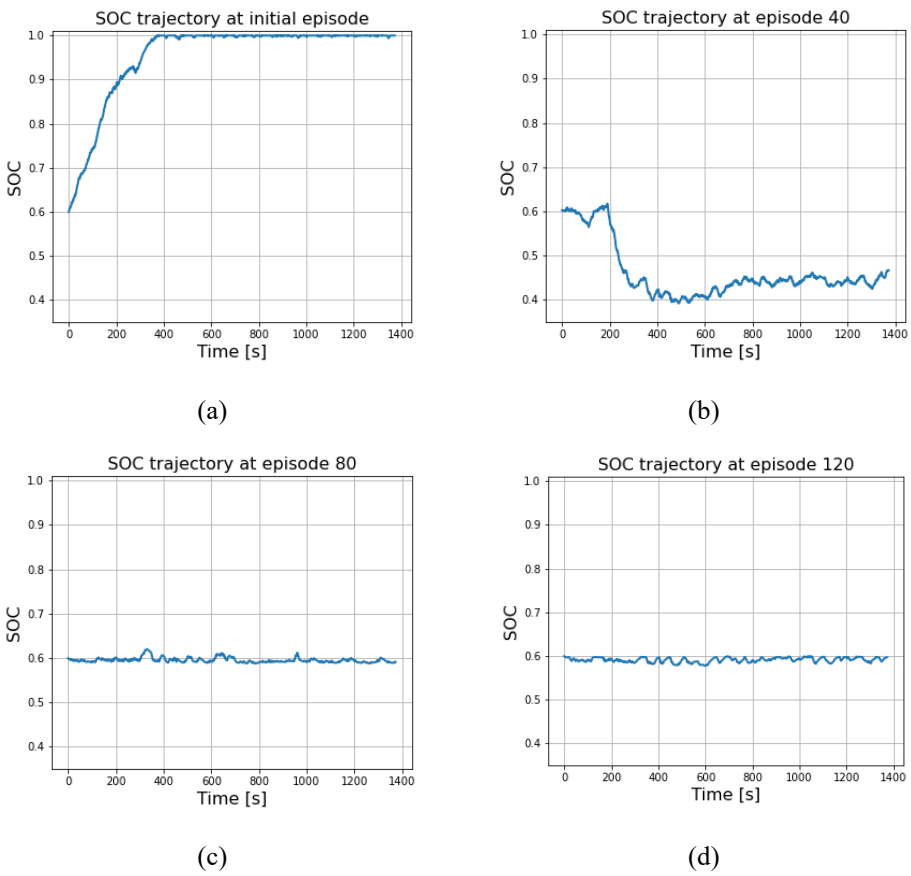


Figure 17. SOC trajectory according to the episode: (a) SOC trajectory at initial episode, (b) SOC trajectory at episode 40 (c) SOC trajectory at episode 80 (d) SOC trajectory at episode 120

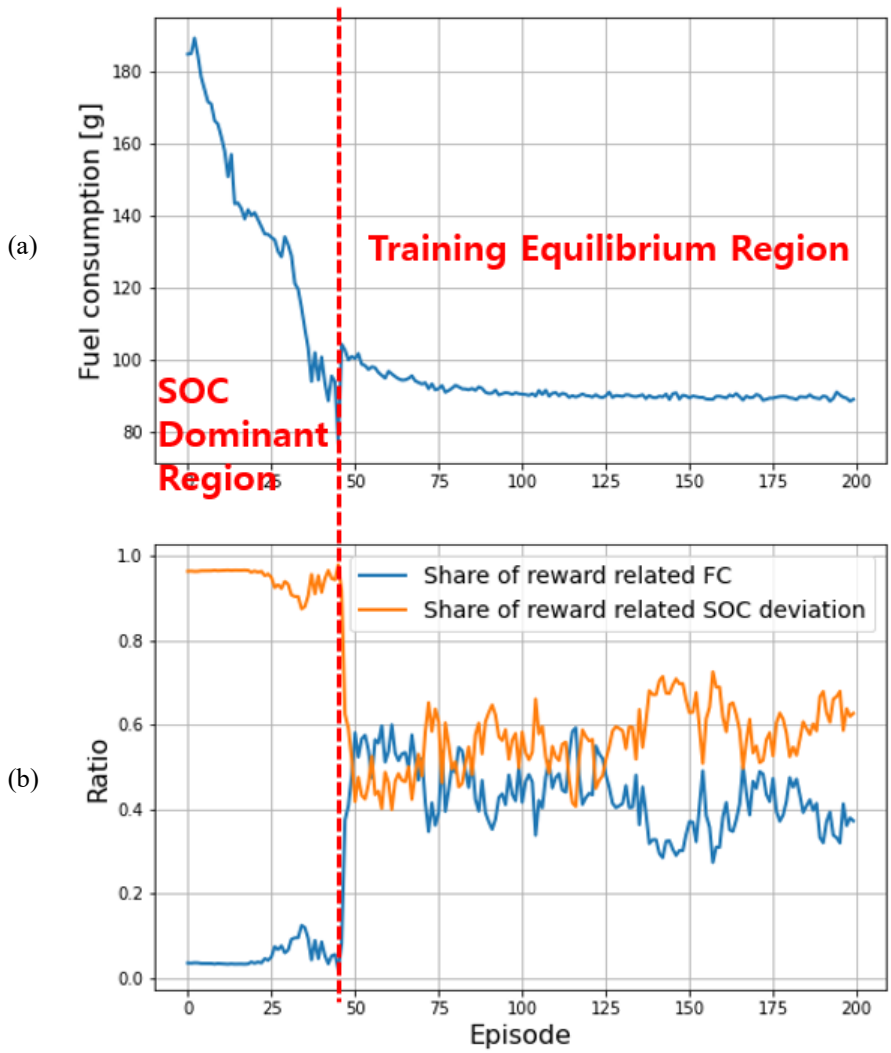


Figure 18. Simulation results with training process: (a) fuel consumption with episode, (b) shares of the reward with episode

3.3.2. Methodology development for the scalability

We confirmed that the agent is trained in a way that restricts the SOC trajectory in the early training of the agent. In addition, we can see that in the training equilibrium region, the agent is trained to minimize fuel consumption while limiting SOC trajectory. In this study, we analyzed whether this learning process proceeds in the same way according to the reward factor.

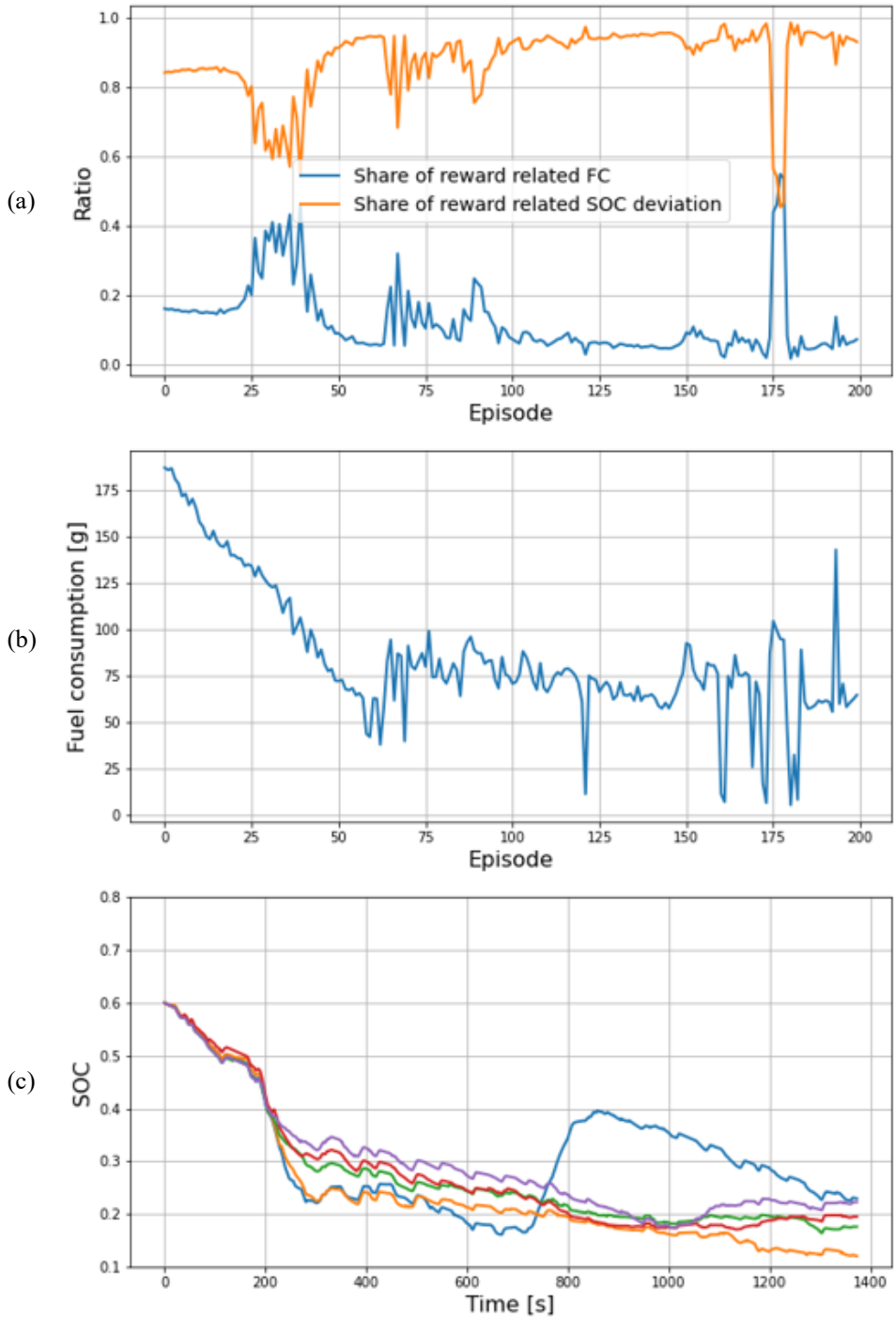


Figure 19. Training characteristics when the reward factor is set to 2: (a) The shares of two reward components to reward with the episode, (b) The fuel consumption with the episode, (c) SOC trajectories derived from the last 5 episodes

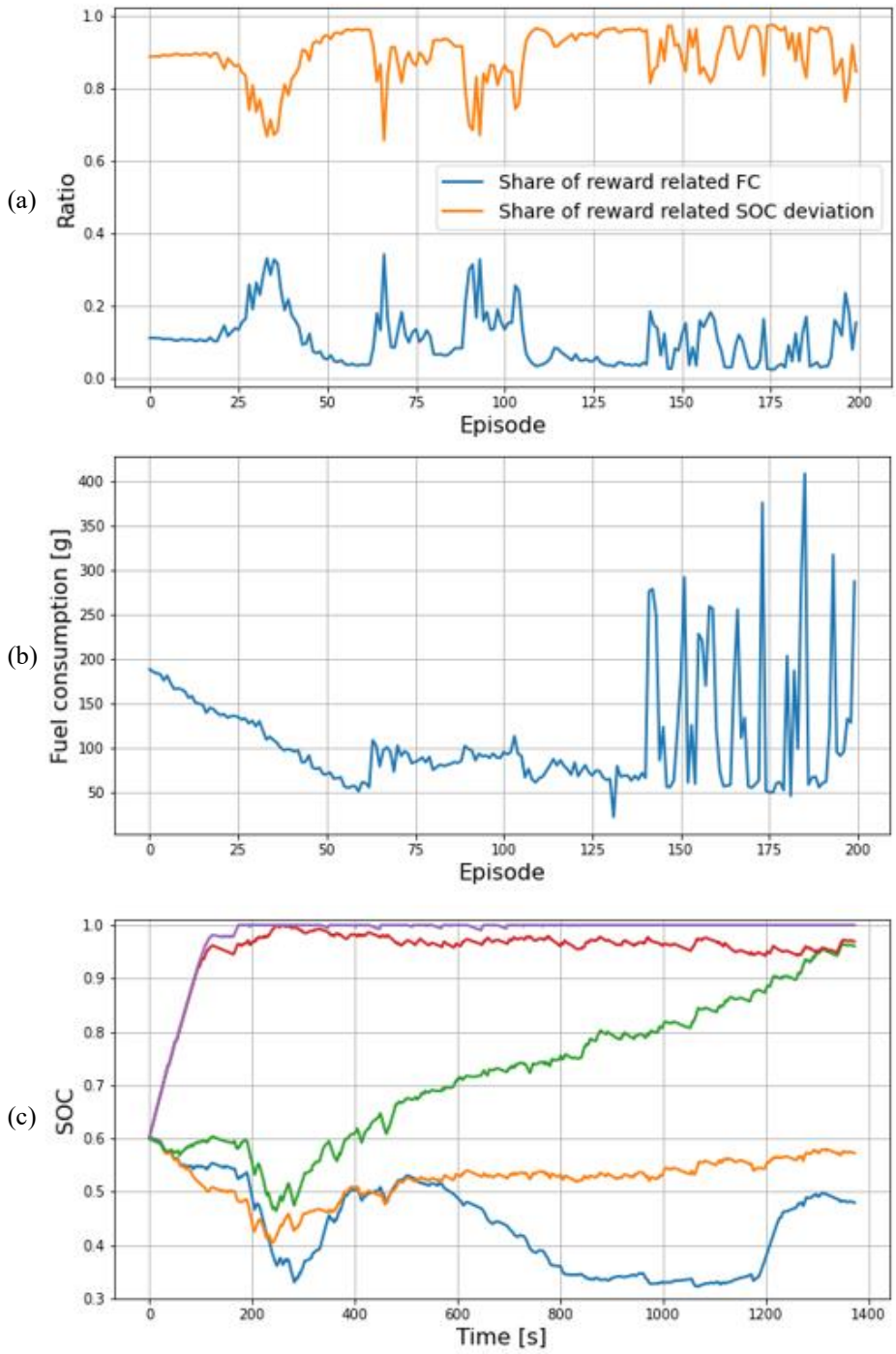


Figure 20. Training characteristics when the reward factor is set to 3: (a) The shares of two reward components to reward with the episode, (b) The fuel consumption with the episode, (c) SOC trajectories derived from the last 5 episodes

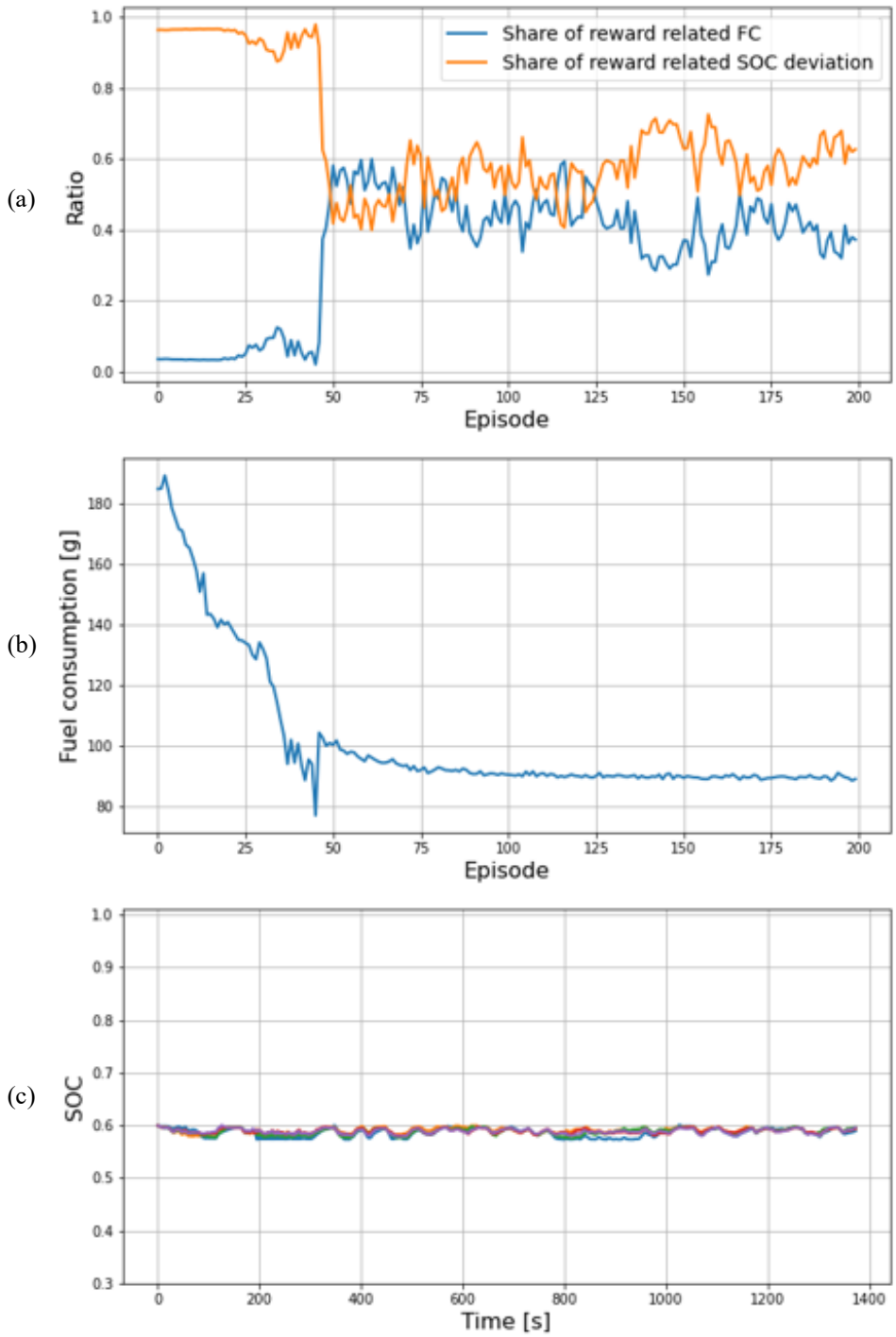


Figure 21. Training characteristics when the reward factor is set to 10: (a) The shares of two reward components to reward with the episode, (b) The fuel consumption with the episode, (c) SOC trajectories derived from the last 5 episodes

We analyzed the change in characteristics of the training process while setting the reward factor to 2, 3, 10. Figures [19], [20], and [21] show the results when the reward factor is set to 2, 3, and 10 respectively. In the figures, the (a) shows the shares of the two components to the total reward according to the episode, and the (b) shows the agent's fuel consumption according to the episode, the (c) represents the SOC trajectory derived from the last 5 training episodes.

From the results in Figures [19] to [21], it can be seen that a reward factor that is too low causes instability in the training process. Figure [19] and Figure [20] show the results when the reward factor is set to 2 and 3. It can be seen that the training equilibrium region is not formed at the reward factor low like 2 and 3, and as a result, the learning of the agent is not conducted efficiently. Since the reward factor is an equivalent factor that means the importance of SOC deviation to the fuel consumption rate, we can see that the training equilibrium region is not formed due to insufficient regulation of the SOC trajectory at the low reward factor, resulting in instability of the training. On the other hand, the simulation results with the reward factor designed as 10 show that the training equilibrium region exists and the learning of the DRL is done effectively. From these results, we can see that it is necessary to select an appropriate reward factor for the successful learning of the agent.

In order to examine the effect of the reward factor on the training more closely, we conducted an analysis of the training process of the agent. The DRL agent collects various types of experiences by taking random actions with high probability in the initial episode. And as the training progresses, the DRL agent takes an action according to the trained policy while reducing the probability of taking a random action. Figure [22] shows the change in the episodic reward and the exploration probability, ε according to the training episodes when the reward factor is set to 10.

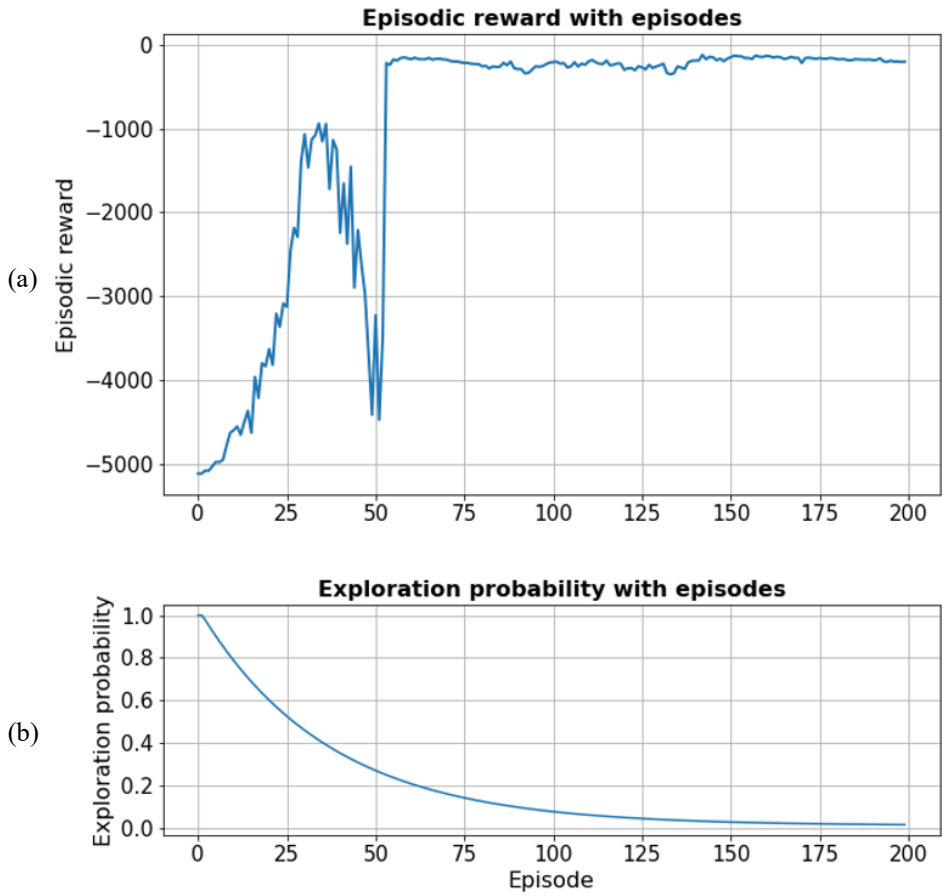


Figure 22. The episodic reward and the exploration probability with the training episodes when the reward factor is set as 10: (a) The episodic reward, (b) exploration probability

The typical episodic reward according to the episodes is expressed in Figure [23]. The training process for the agent can be divided into four phases, as shown in Figure [23]. In the "phase 1", the episodic reward increases with the training, and the episodic reward decreases as the training progresses in the "phase 2". Also, the episodic reward increases rapidly in the "phase 3", and the episodic reward converge in the "phase 4". We analyzed each phase and tried to understand the learning process of the DRL agent.

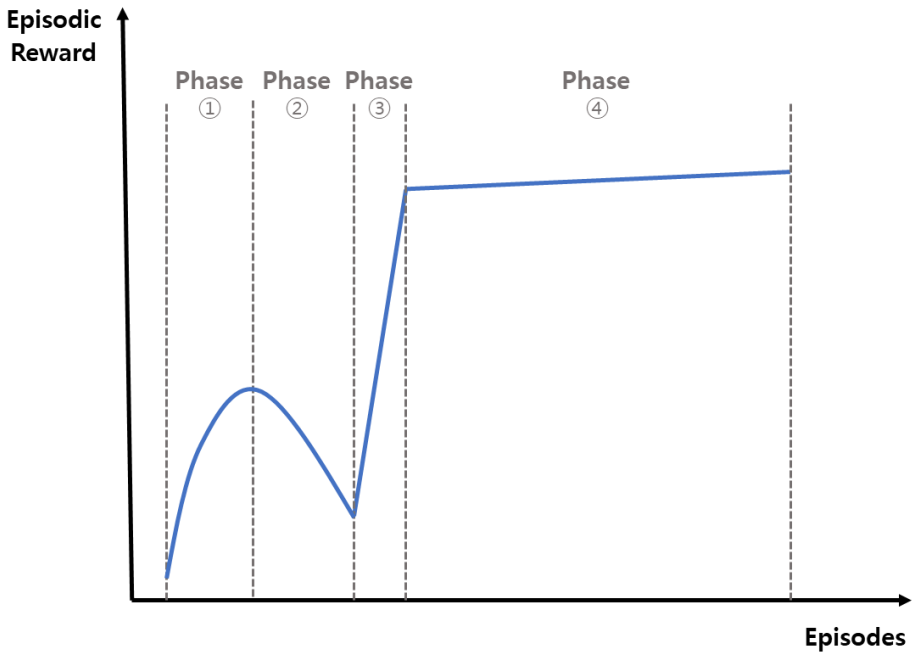


Figure 23. Illustration of the typical episodic reward according to the training episode

In the phase 1, the action of the agent is mainly selected in a random manner. The action profile and SOC trajectory in the first episode where the action is selected randomly are illustrated in Figure [24]. When the actions are selected randomly, the SOC is maintained close to 1.0 because the agent uses excessive power of the fuel cell stack. This operation results in high fuel consumption and large SOC deviations.

Therefore, the agent is trained in the direction of minimizing the use of the power for the fuel cell stack in order to increase the episodic reward. That is, the agent mainly selects the idling current density as an action. Figure [25] shows how the trained agent takes an action in the phase 1. We classified the action selected in a random manner and the action selected by the policy of the agent, and the results is visualized through Figure [25]. Figure [25] shows that most of the actions derived by the policy are the same as the idling current density.

(a)

(b)

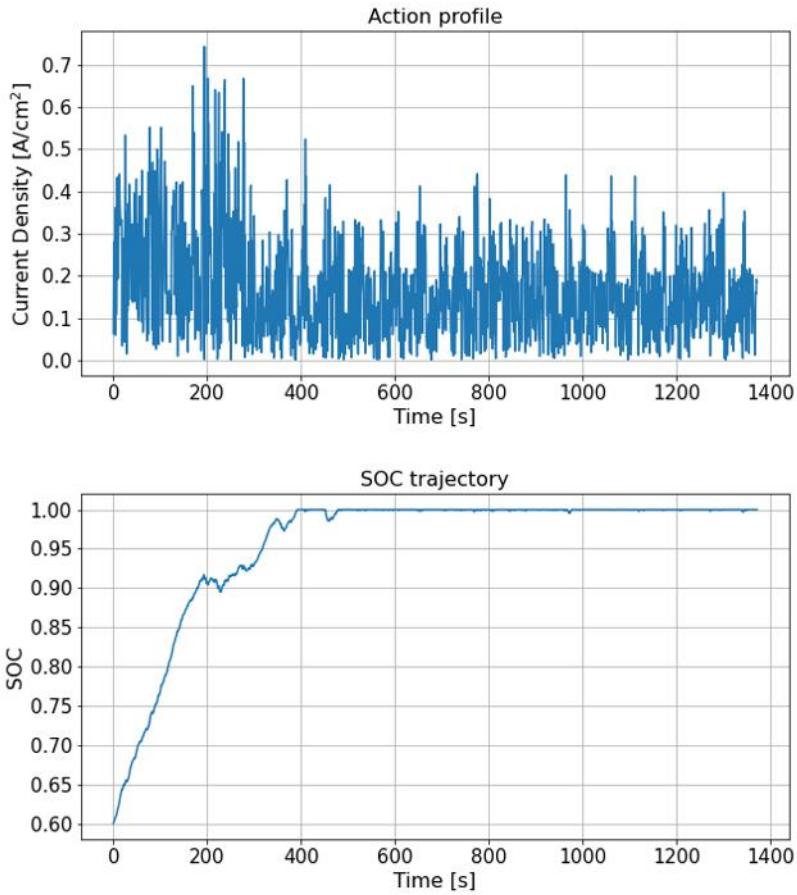


Figure 24. (a) The action profile and (b) the SOC trajectory in the initial episode

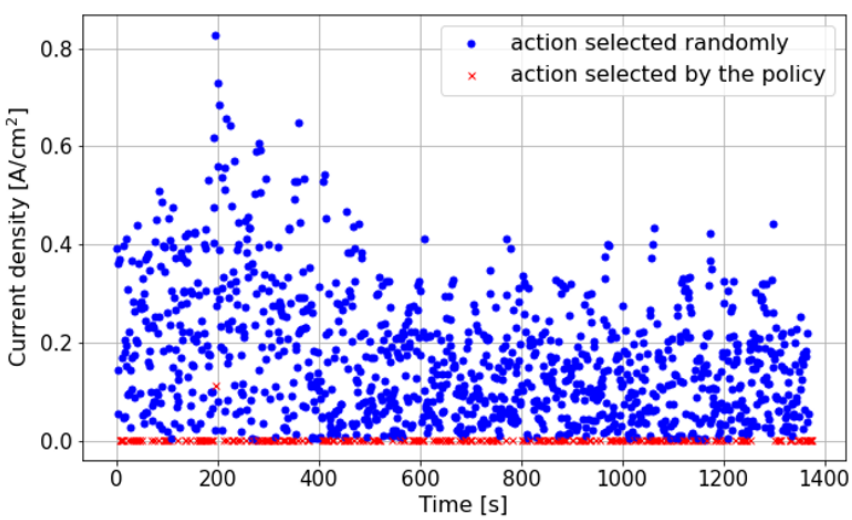


Figure 25. The action selected by the policy and the action selected randomly in the phase 1

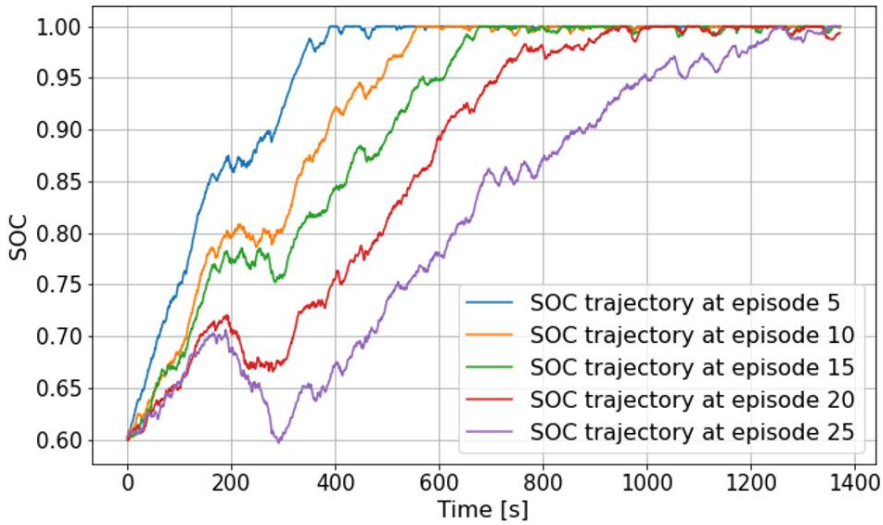


Figure 26. The SOC trajectories according to the episode in the phase 1

In the phase 1, if the agent selects the idling current density continuously, the SOC trajectory moves near the initial SOC of 0.6. Figure [26] shows the SOC trajectory change according to the episode in the phase 1. As a result, the agent can reduce the SOC deviation and reduce the amount of fuel consumed of the fuel cell stack by selecting the idling current density. Therefore, the agent can increase in episodic reward by choosing the idling current density.

It can be seen from Figure [27] that the agent still selects the idling current density as an action even in the phase 2. However, in the phase 2, when only the idling current density is selected, the SOC-deviation no longer decreases, but rather increases, so the episodic reward increases as a result.

Figure [28] shows SOC trajectories according to episodes in the phase 2. As the episode progresses, the agent selects more idling current density as an action, the SOC is distributed more frequently in a region which is lower than the initial SOC. Therefore, the episodic reward for the agent decreases as the episode progresses in the phase 2.

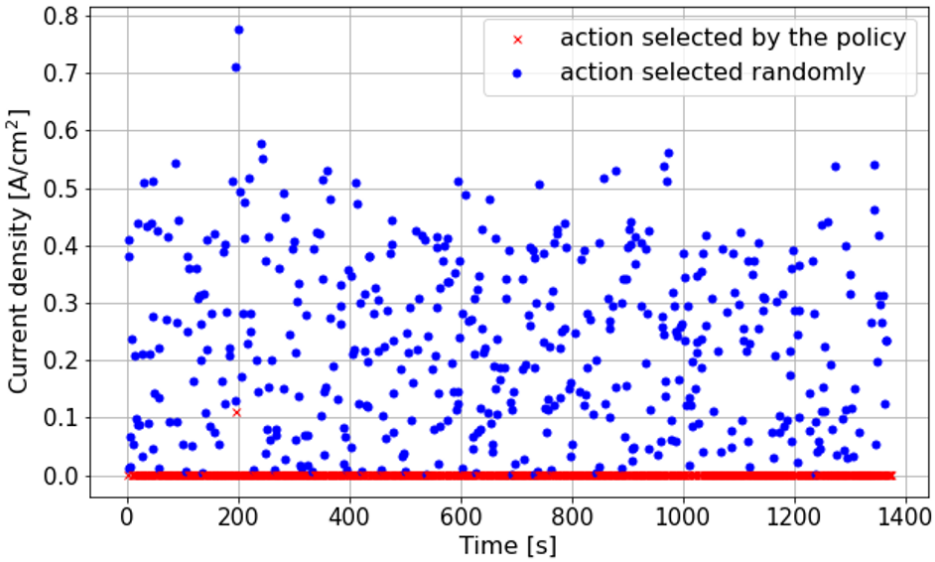


Figure 27. The action selected by the policy and the action selected randomly in the phase 2

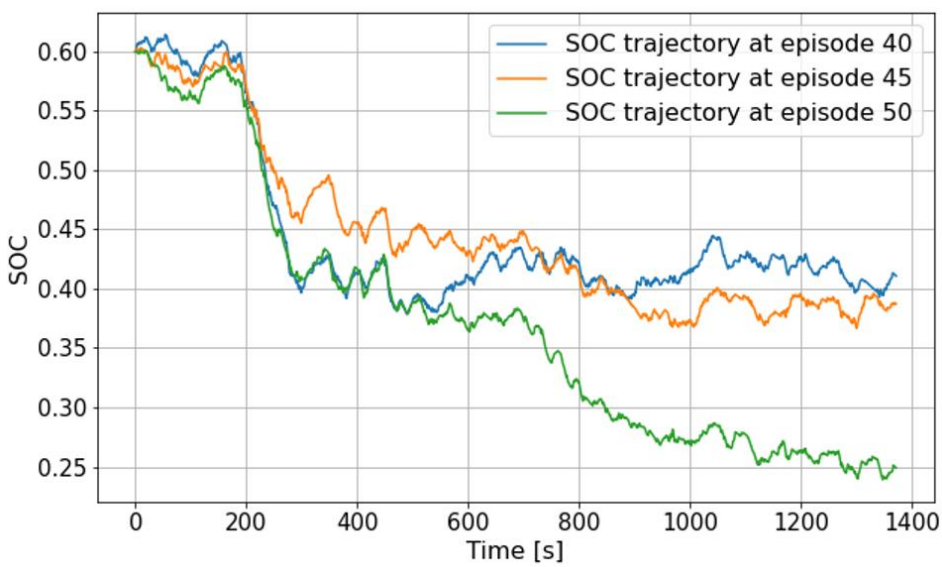


Figure 28. The SOC trajectories according to the episode in the phase 2

In Phase 3, the episodic reward increases dramatically. This rapid increase in the episodic reward is because the agent is trained in a way that can regulate the SOC-trajectory sufficiently. Figure [29] shows the SOC trajectory and distribution

of action in one episode belonging to the phase 3. Through the figure, it can be confirmed that the SOC is distributed around the initial SOC.

And we can see from Figure [30] that the SOC regulation occurring in the phase 3 is closely related to the "training equilibrium region". That is, the training equilibrium region is developed through the SOC regulation.

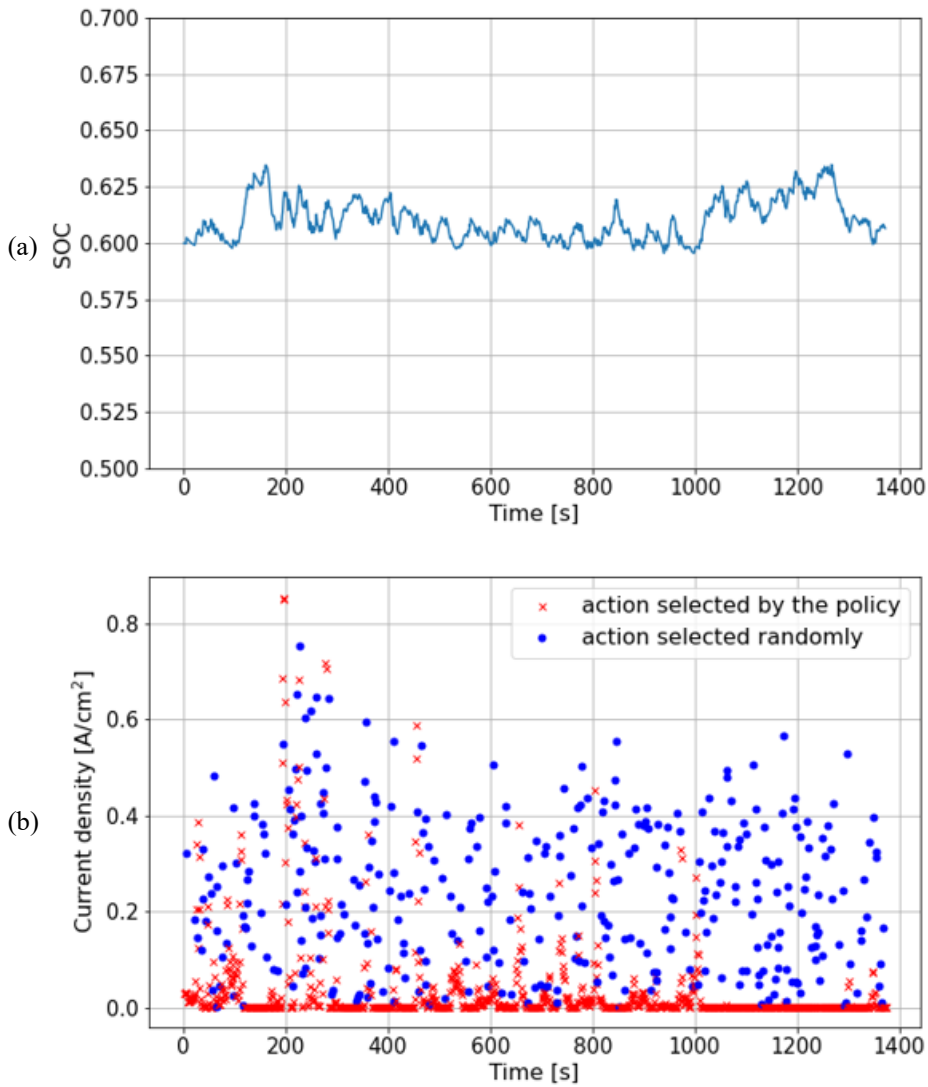


Figure 29. (a) The SOC trajectory and (b) the action distributions in the phase 3

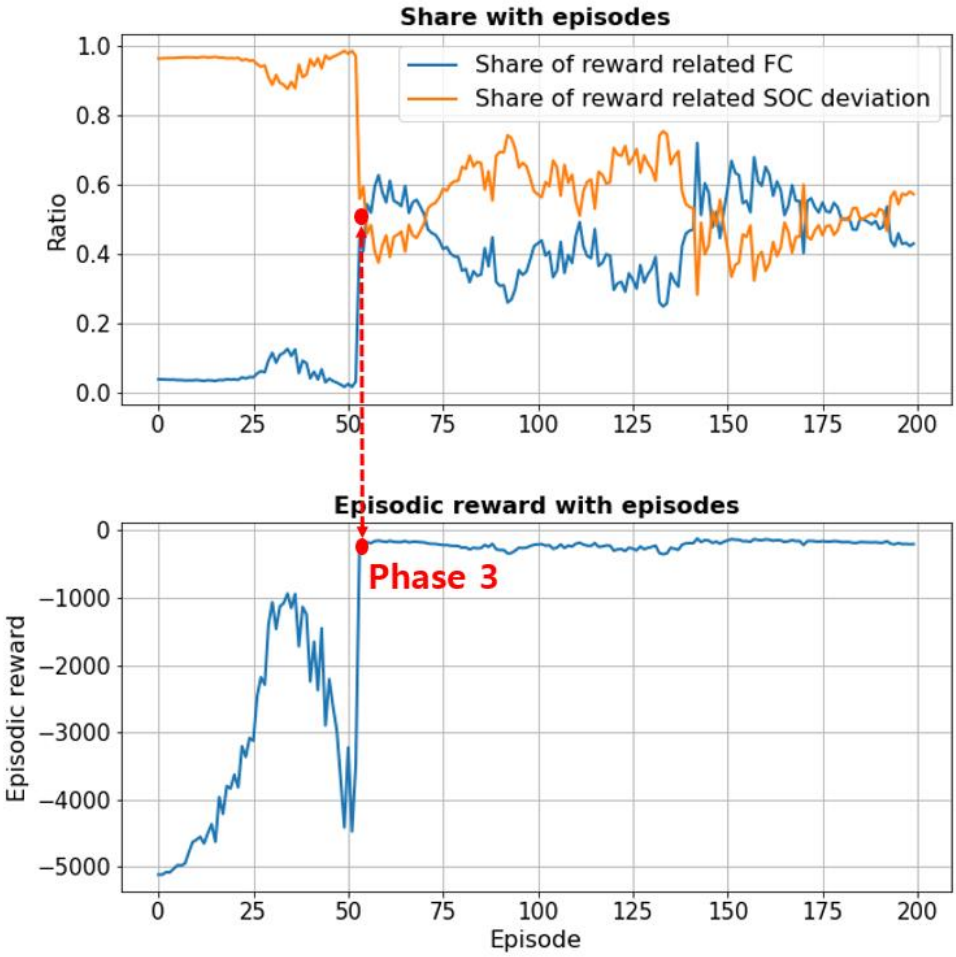


Figure 30. Relationship between the start of the training equilibrium region and the phase 3

And the in the phase 4, the episodic reward either converges or rises gently. The agent is trained to minimize the fuel consumption while regulating the SOC in the phase 4. We classified the fuel consumption in the phase 4 and the fuel consumption in the rest of areas and visualized the fuel consumption in both areas. Figure [31] shows the result for the visualization. In the figure, the blue solid line represents the fuel consumption according to the episode in the phase 4. We can see from the figure that the fuel consumption decreases as the learning of the agent progresses in the phase 4.

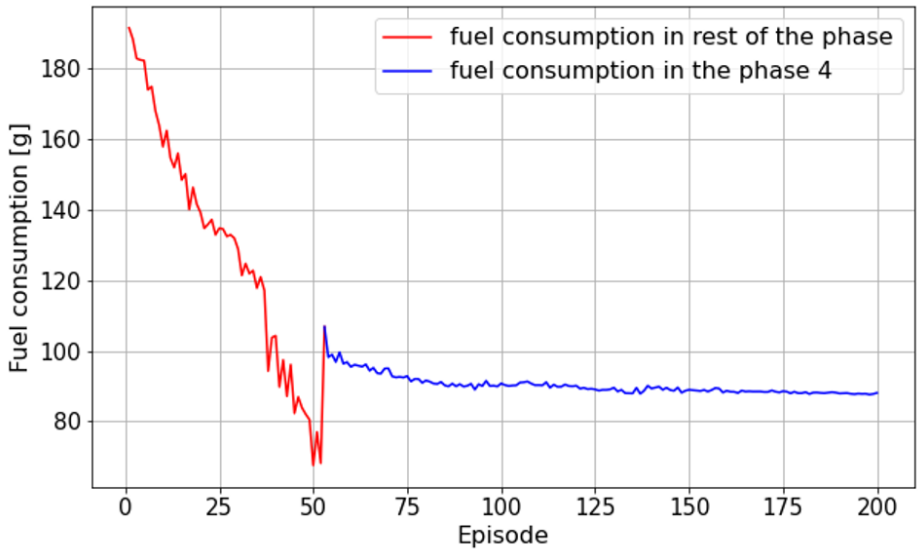


Figure 31. The fuel consumption according to the training episodes

However, the SOC regulation is not sufficiently implemented at a relatively low reward factor. As a result, the phase 3 in the episodic reward and the training equilibrium region are not developed. Figure [32] shows the training results of the DRL agent when the reward factor is set to 2. Figure [32]–(a) shows the episodic rewards according to the episodes, and Figure [32]–(b) shows the shares of two reward components according to the episodes. When the reward factor is set to 2, it can be seen from the figure that the phase 3 in the episodic reward and the training equilibrium region are not formed, and the training of the agent is also unstable because the agent does not regulate the SOC sufficiently.

Reward factor has the meaning of the weight about the SOC deviation to fuel consumption. Therefore, if the reward factor is set low for the training of the DRL agent, the agent does not regulate the SOC sufficiently, and as a result, the agent's learning does not proceed effectively. Therefore, it is important to derive an effective reward factor according to the problem definition and the FCHEV system.

(a)

(b)

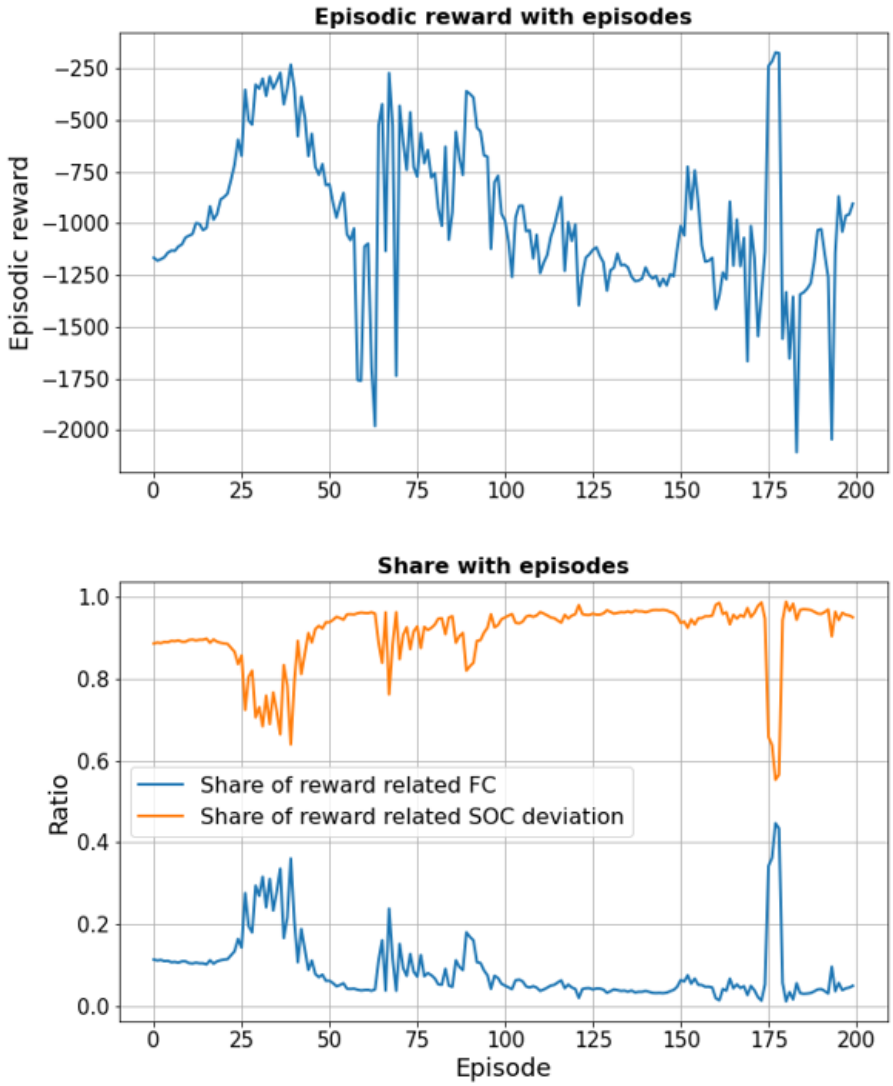


Figure 32. The training results when the reward factor is set as 2: (a) The episodic reward and (b) the shares of two reward components according to the episode

Since the effective reward factor is different depending on the FCHEV system and the problem definition, it is necessary to develop a methodology that can derive the reward factor automatically without the experience and the knowledge of the experts for the scalability of the power distribution strategy. In this study, we developed a methodology that can derive the reward factor through the process shown in Figure [33].

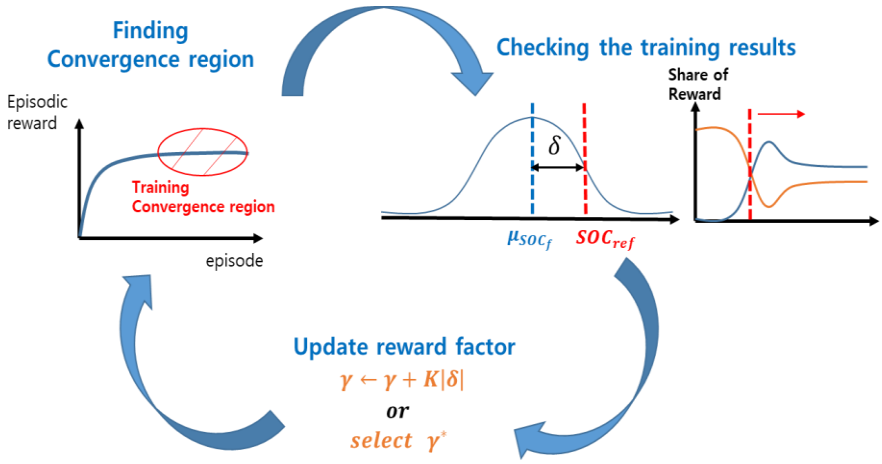


Figure 33. Schematic diagram of the process of finding the reward factor

The process to find the reward factor is as follows. We checked whether the valid training was performed or not through the final SOC distribution of the SOC trajectories in the episodes at the end of the training and the existence of the training equilibrium region. The deviation between the final SOC of the SOC trajectories at the end of the training and the initial SOC, δ is equal to Eq (28), and the δ acts as a variable that determines whether the reward factor is updated or not. In Eq (28), N is the number of episodes corresponding to the convergence region, and $SOC_{f,i}$ is the final SOC of the SOC trajectory for each episode.

$$\delta = \mu_{SOC_f} - SOC_{init}$$
$$\because \mu_{SOC_f} = \frac{1}{N}\sum_i SOC_{f,i} \tag{28}$$

In this study, when the absolute value of the δ is within 0.015 and the training equilibrium region is formed, the corresponding reward factor is determined as a reward factor suitable for the training. And we designed the algorithm to execute the training process again by adjusting the reward factor as shown in Equation (29) if an arbitrary reward factor does not satisfy the above two conditions. That is, as the

difference between the initial SOC and the final SOC increases, the range of adjustment of the reward factor is increased so that the effective SOC regulation is made.

$$\gamma \leftarrow \gamma + K|\delta| \tag{29}$$

We derive the reward factor suitable for this study through the search method of the reward factor. The initial reward factor was designed as 3, after that, the reward factor was 4.45, and finally 5.80 was selected as the suitable reward factor. The results are expressed in Figures [34]-[36]. (a) and (b) in figures show the weight of the two components of the reward for each episode and the SOC paths for episodes at the end of the learning. When the reward factors are 3, we can see that the training equilibrium region does not exist and the constraint for the SOC paths are not properly regulated. Figure [35]-(a) shows that when the reward factor is 4.45, the training equilibrium region does not exists during the training process of the agent. As shown in Figure-(b), when the reward factor is 4.45, the regulation on SOC trajectory is not implemented, and the power distribution strategy does not guarantee SOC-sustainability. Figure [36] shows the result when the reward factor is set to 5.80. Through Figure [36], when the reward factor is set to 5.80, it can be seen that the training equilibrium region is formed and the SOC trajectories are limited around the initial SOC.

(a)

(b)

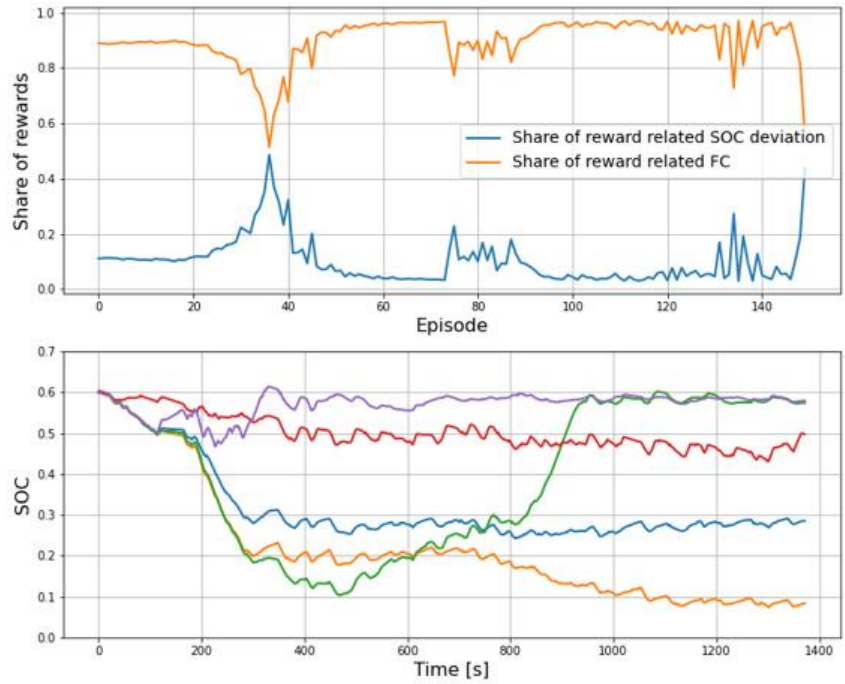


Figure 34. Simulation results when the reward factor is 3.00: (a) The shares of two terms to reward with the episode, (b) SOC trajectories at the end of the training episode

(a)

(b)

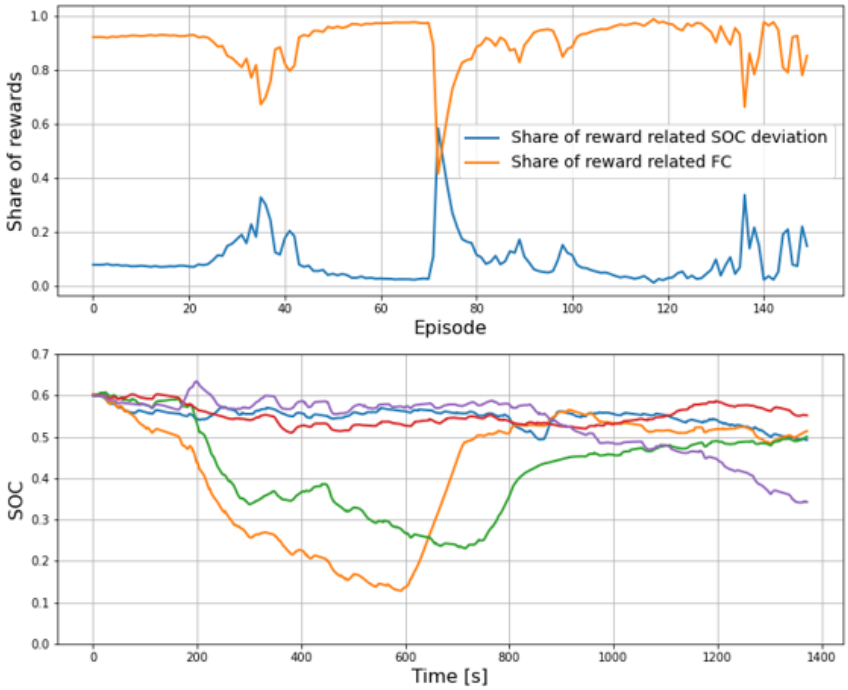


Figure 35. Simulation results when the reward factor is 4.45: (a) The shares of two terms to reward with the episode, (b) SOC trajectories at the end of the episode

(a)

(b)

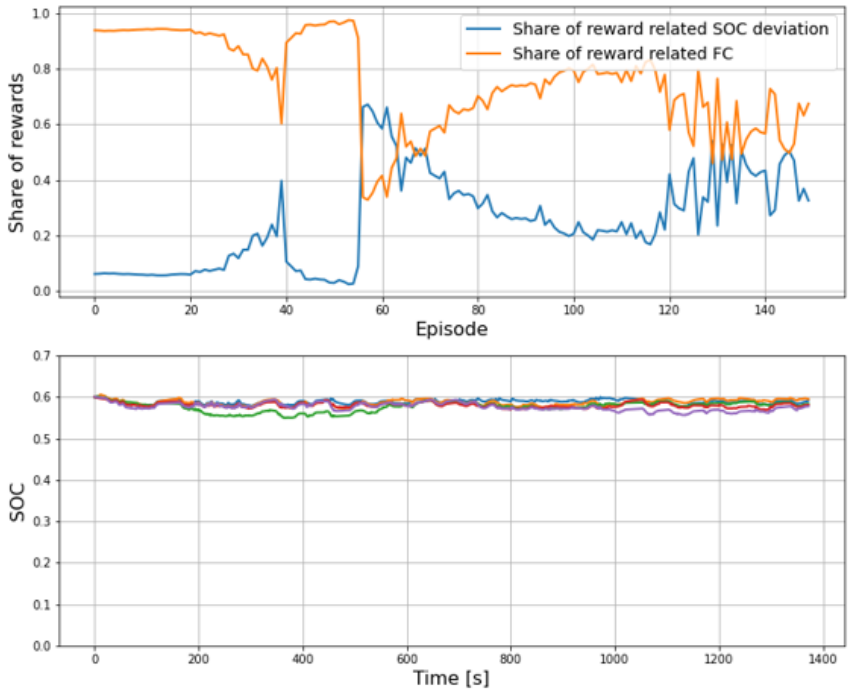


Figure 36. Simulation results when the reward factor is 5.80: (a) The shares of two terms to reward with the episode, (b) SOC trajectories at the end of the episode

3.4. Generalization Performance of the Trained Agent

One of the most important goals in the field of the machine learning and the artificial intelligence (AI) is to construct a model with generalization performance. The generalization performance is defined as the ability of a model trained with arbitrary training data to derive valid results even for the data not used for the training. The DRL-based power distribution strategy is trained by specific driving patterns or driving cycles. Therefore, the DRL-based control strategy that secures generalization power must perform effective energy management even for the driving cycles that are not used for the learning of the agent.

This chapter consists of three parts. In 3.4.1, the training framework in which the agent is trained to secure the generalization power of the DRL model is explained. And the driver model based on the Markov decision process developed to verify the DRL model is described in 3.4.2. Finally, in 3.4.3, the process and results of the validation test for the DRL model are described.

3.4.1. Training framework for the generalization performance

Since the generalization performance increases in proportion to the quantity of the data used to develop the algorithm, using many driving cycles in the development of the DRL model for the FCHEV helps to increase the generalization performance of the agent. In this study, the DRL model was trained based on more than 20 reference driving cycles representing general driving situations, such as FTP-75 and UDDS cycle for the training of the DRL model.

Figure [37] shows the process of developing the DRL-based power distribution strategy. First, we find the effective reward factor based on one driving cycle through the methodology that derives the reward factor described above. In this study, the

effective reward factor was derived based on the FTP-72 reference driving cycle shown in Figure [38]. After that, we improved the generalization performance of the DRL model by re-training the DRL agent through various standard cycles and the derived reward factor. In order to improve the learning speed, when training the DRL model through multiple reference cycles, we used a transfer learning method that initializes the weights of the model to the weights of the model previously trained in one cycle.

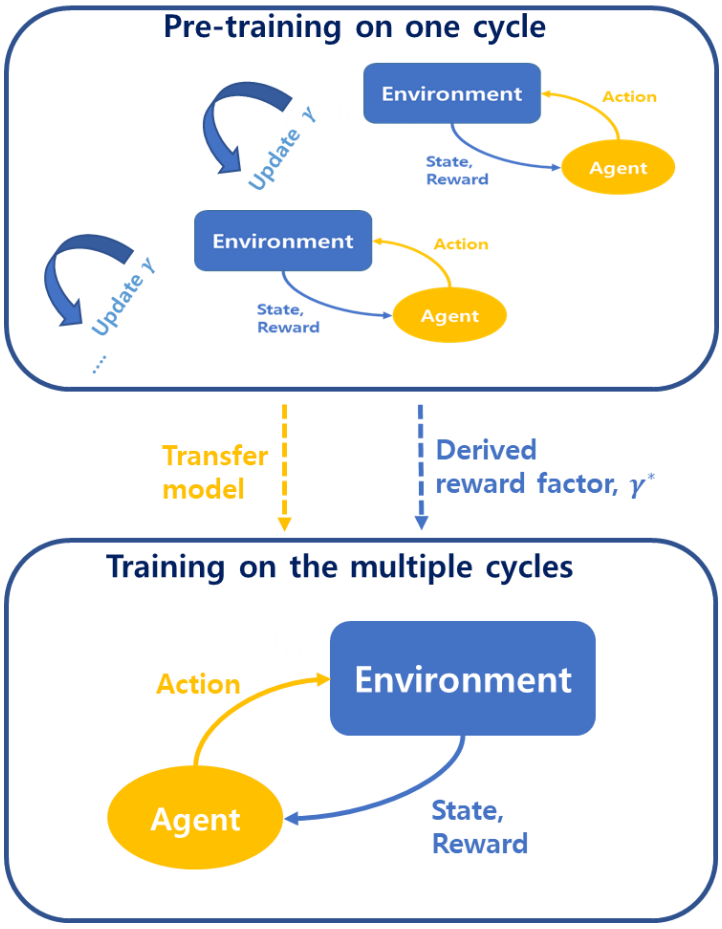


Figure 37. Training framework to secure the generalization power of the agent

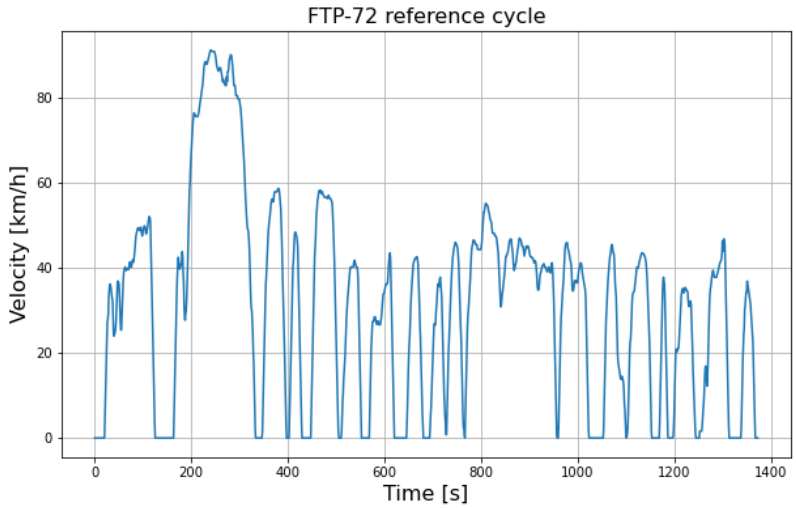


Figure 38. FTP-72 reference driving cycle

3.4.2. Development of driver model based on the Markov decision process

Since the DRL model was trained through most of the reference driving cycles, it was necessary to generate additional driving patterns and driving cycles to verify the DRL model. We developed the driver model based on the Markov decision process to verify the DRL model for various driving patterns.

The MDP-driver model was developed based on the statistical characteristics of the reference driving cycles. As in Eq (30), the MDP-driver model derives the vehicle speed of the next step based on the vehicle speed at the current step. In Eq (30), $v, a, and\ P$ mean velocity, acceleration, and transition probability, respectively. The transition probability represents the probability that an arbitrary acceleration will be derived under a specific velocity condition.

$$v_{t+1} = v_t + a_t \Delta t, \quad a_t \sim P(a_t | V = v_t) \tag{30}$$

And the transition probability, P_v^a is expressed as Eq (31) below, where N_v^a means the number of transitions to acceleration, a under the condition of a specific velocity, v. The number of transitions, N_v^a is expressed by Eq (32). In Eq (32), ε is an arbitrary constant value and plays a role of suppressing the occurrence of a dead zone where the transition probability becomes 0. Also, it makes the transition probability different from the average transition distribution of the reference cycles.

$$P_v^a = \frac{N_v^a}{\sum_a N_v^a} \tag{31}$$

$$N_v^a = \sum \mathbf{1}(\cdot \mid V = v, A = a) + \varepsilon \tag{32}$$

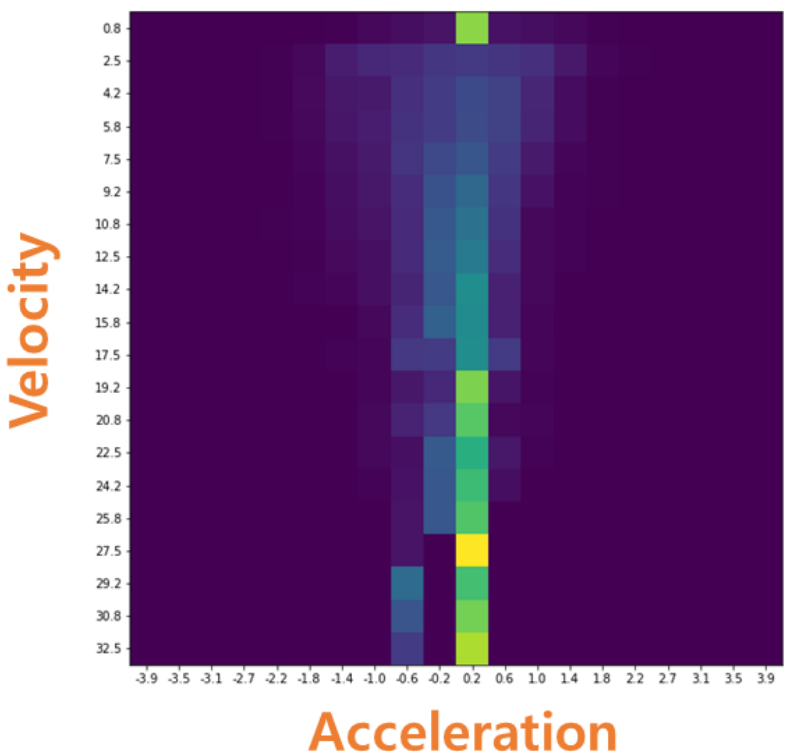


Figure 39. Visualization of the transition probability matrix

Figure [39] shows the visualization result for the transition probability matrix. Y-axis stands for the velocity and X-axis stands for the acceleration. And the color of the grid corresponding to each velocity and acceleration represents the transition probability. The closer to yellow, the closer to 1, and the closer to purple, the closer to 0.

When the verification experiments are conducted with only the reference cycles, the number of developed reference driving cycles is limited, so only limited verification experiments are possible. In order to overcome these limitations, we developed the MDP-based driver model and carried out extensive DRL model verification experiments through a number of the validation driving cycles.

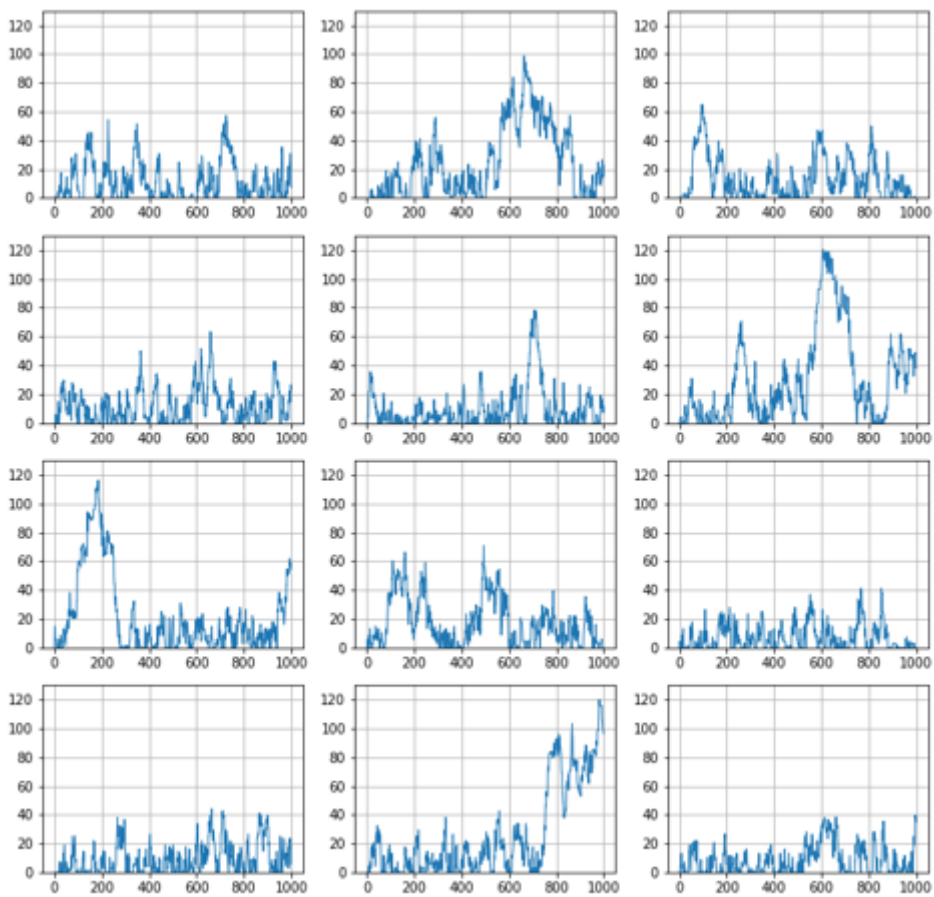


Figure 40. Driving cycles generated from the MDP driver model

3.4.3. Experiments for the validity and the results

Figure [40] shows examples of the driving cycles for verification created through the MDP-driver model described above. It can be seen from the figure that the generated driving cycles have different driving characteristics from each other. In this study, the generalization power of the trained DRL model was evaluated with a number of driving cycles generated through the MDP-driver model.

(a)

(b)

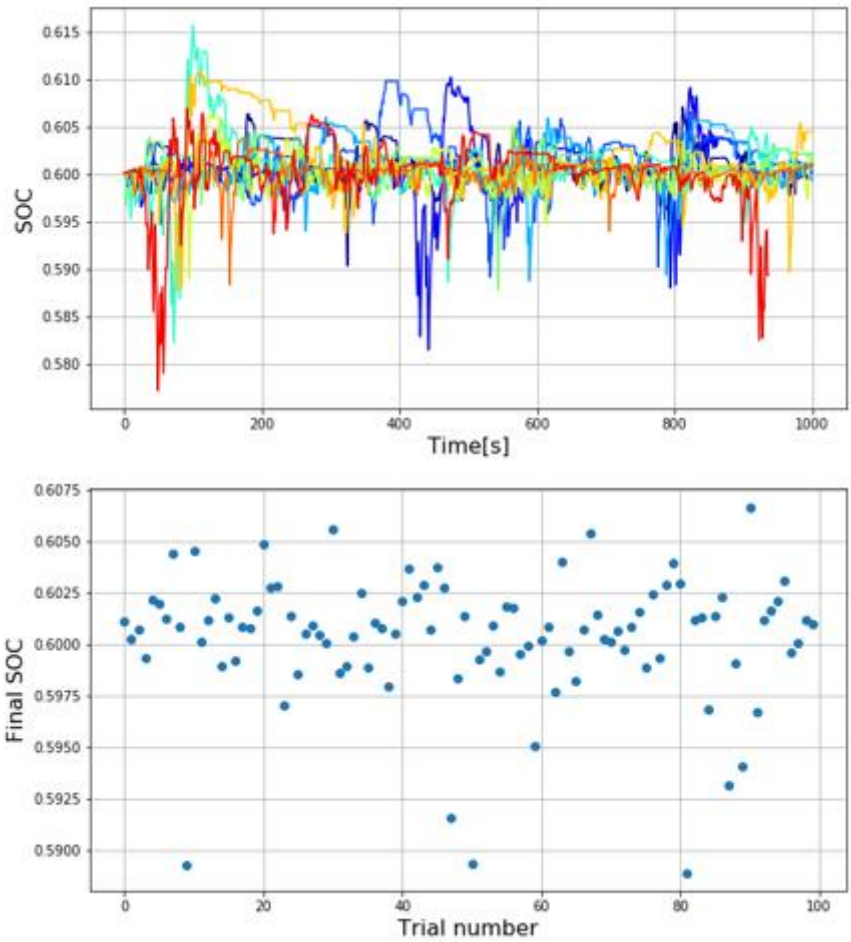


Figure 41. Simulation results for the generalization power of the DRL model in terms of the SOC sustainability: (a) SOC trajectories of the trained DRL model on the test driving cycles, (b) final SOCs of the trained DRL model on the test driving cycles

We first generated 100 driving-cycles through the MDP-driver model in order to evaluate the generalization power of the trained DRL model in terms of the SOC-sustainability. In these 100 driving-cycles, we evaluated the SOC-sustainability of the agent, and the evaluation results are shown in Figure [41]. (A) of Figure [41] is the result showing some of the SOC trajectories derived while the trained agent runs the 100 test cycles. And (b) of Figure [41] refers to the final SOC, which is the SOC at the end of the 100 test cycles.

Figure [41] – (b) shows that the final SOCs for the trained DRL model are very similar to the initial SOC value set to 0.6 even when the agent is tested on the 100 validation driving cycles generated by the MDP-driver model. The difference between the final SOC and the initial SOC shows a difference of less than about 0.01, and through this, it can be confirmed that the trained agent guarantees the generalization power in terms of SOC-sustainability.

We confirmed that the derived DRL model secures the generalization performance in terms of the SOC-sustainability. However, the DRL agent must be able to guarantee the generalization performance in terms of the fuel efficiency. In order to verify the fuel efficiency of the DRL agent, we conducted a study using Equivalent consumption minimization strategy (ECMS) as a reference model. ECMS is a theory derived based on the PMP (Pontryagin's minimum principle) theory, and is controlled by deriving a control value, u, which minimizes an instantaneous cost function such as Eq (33). λ is defined as a Lagrange multiplier or co-state, and physically has the meaning of an equivalent factor that equalizes the amount of SOC change to instantaneous fuel consumption in Eq (33).

$$H = \dot{m}_{fc} + \lambda \cdot \dot{SOC} \tag{33}$$

$$u^* = arg\min_{u} H \tag{34}$$

ECMS is a simple theory, but a solution derived by the ECMS close to the optimum if the co-state is derived appropriately. N. Kim, S. W. Cha, et. al proved that if the co-state that satisfies the SOC-constraint can be derived, it can show a result comparable to the dynamic programming (DP) guaranteeing a global optimal solution.

DP provides a methodology that can derive an optimal solution without searching all paths by repeatedly storing the optimal solution of the divided problem. The process of deriving the optimal cost, $J^*_{k,N}$, which is consumed from the Nth step to the kth step, can be expressed through Eq (35), where L is the cost consumed from the k+1th step to the kth step. Figure [42] is illustration showing the mathematical expression of DP.

$$J^*_{k,N}(x(k)) = \min_{u}(L\big(x(k), u(k)\big) + J^*_{k+1,N}(x(k+1))) \tag{35}$$

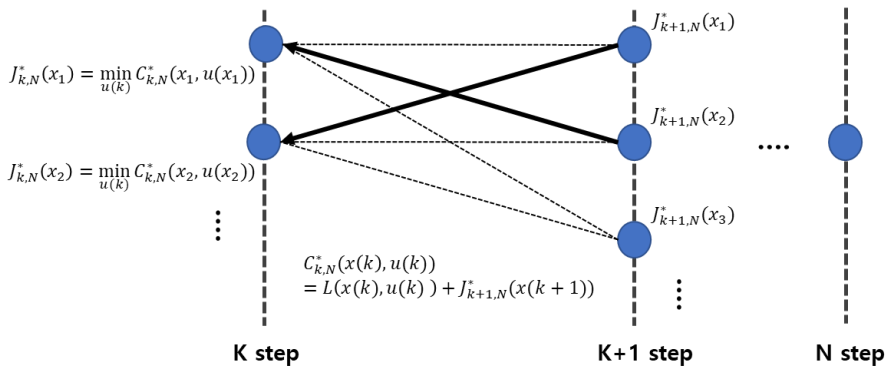


Figure 42. Schematic diagram of the calculation process of the dynamic programming

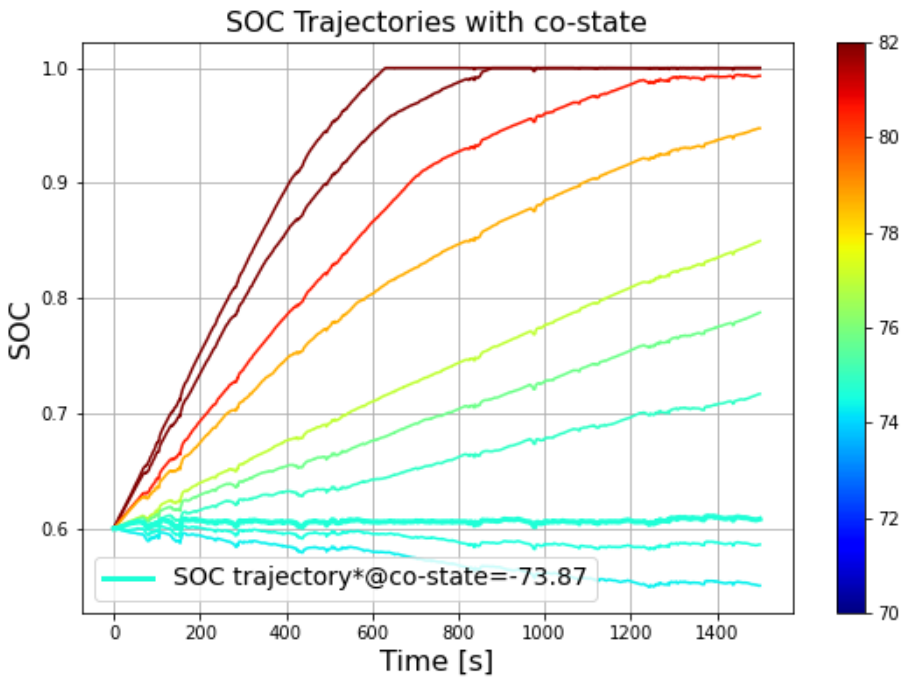


Figure 43. SOC trajectory created by shooting method

We used the shooting method to derive the optimal co-state for the ECMS. That is, we derive the optimal co-state that can satisfy the SOC constraint by repeatedly updating the co-state value for an arbitrary driving cycle. Figure [43] shows the SOC-trajectories generated by updating the co-state until the optimal co-state value for the ECMS is derived on an arbitrary driving cycle. When the co-state is about -71.3, the SOC-constraint can be satisfied.

In order to confirm the effectiveness of the ECMS used as the reference model, we conducted the comparison experiment between the ECMS and the DP which can guarantee an optimal solution. 50 driving cycles corresponding to 1500 seconds were created from the MDP-driver model, and the fuel economy comparison experiment between the two optimal control algorithms was conducted. Figure [44] shows the scatter plot of the equivalent fuel consumption between DP and ECMS. The equivalent fuel consumption refers to the fuel consumption consumed when the final

SOC value is replaced with the initial SOC when the final SOC is different from the initial SOC. The x-axis of the scatter plot shows the fuel consumption for the ECMS and the y-axis shows the fuel consumption for the DP. The blue dotted line represents the decision boundary in Figure [44]. If there is a point at the bottom of the decision boundary, it means that the DP recorded less fuel consumption than the ECMS, and if the point is above based on the decision boundary, it means that the ECMS recorded less fuel consumption than the DP. All points exist below the decision boundary, but points are distributed at points very close to the decision boundary, which means that the two optimization algorithms show similar efficiencies.

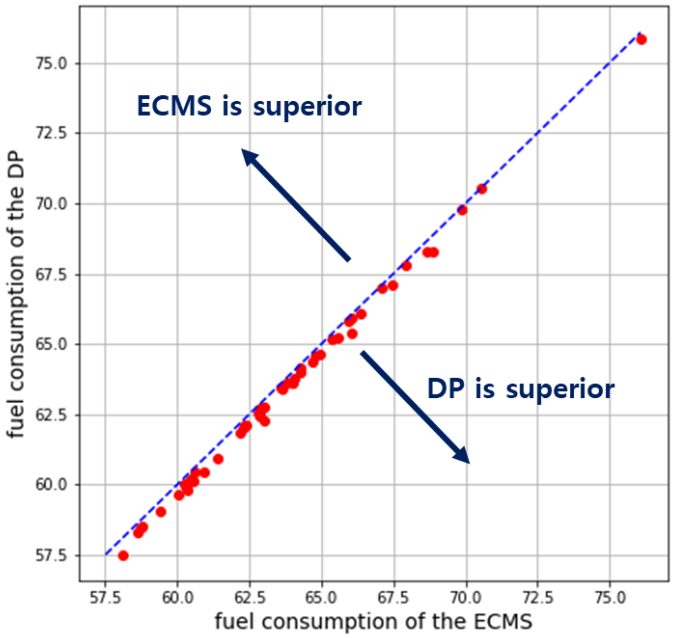


Figure 44. Scatter plot for the fuel consumption of the ECMS and the fuel consumption of the DP

Table II. Comparison of simulation results between DP and ECMS

Maximum ΔFC [%]	Minimum ΔFC [%]	Average ΔFC [%]
1.19	0.0014	0.47

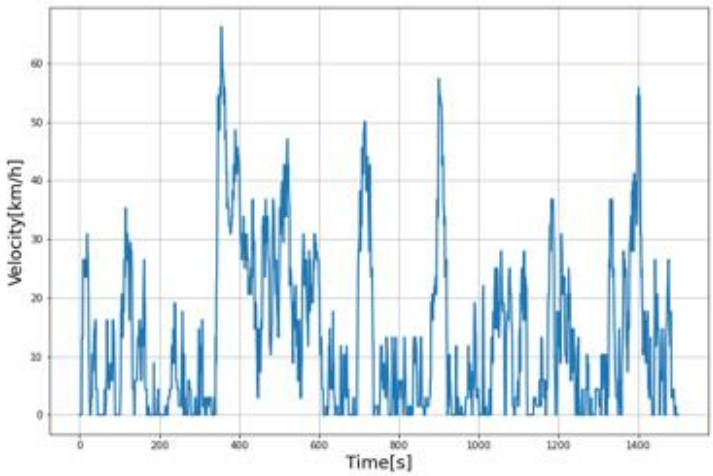
Table 2 shows information on the difference in the equivalent fuel consumption between the ECMS and the DP. The difference about the equivalent fuel consumption between the optimal control theory algorithms for the 50 driving cycles is about 1.19% at the maximum, and about 0.001% at the minimum. On average, the two optimal control algorithms show a difference of about 0.47%, which means that if the co-state value of the ECMS is well set based on the future driving information, there is no significant difference from the DP result.

As such, we confirmed the validity of the AC model developed in this study by comparing the ECMS algorithm belonging to the real-time optimal control which can derive control close to the optimal.

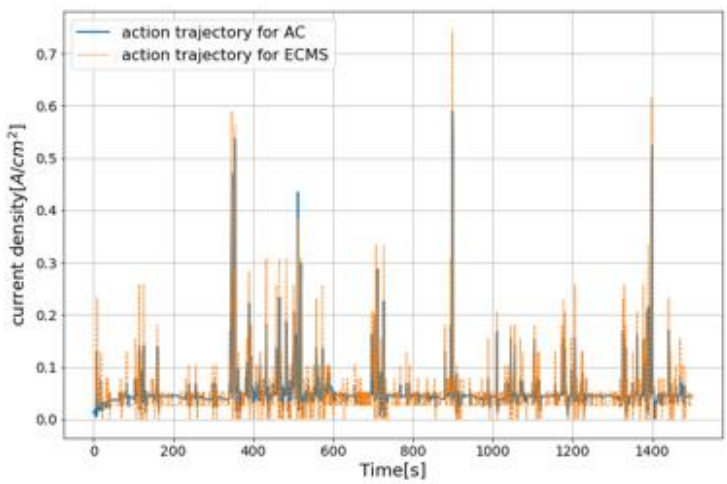
Figure [45] shows the simulation results of the ECMS algorithm and AC agent for the cycle corresponding to 1500 seconds generated through the MDP-driver model. (a), (b), and (c) of Figure [45] show the driving cycle, the action profiles of the two algorithms, and the SOC trajectories for the two algorithms, respectively. In Figures [45]-(b) and Figure [45]-(c), the solid blue line shows the action profile and SOC trajectory for the DRL-based AC model, and the orange dotted line shows the action profile and SOC trajectory of the ECMS. The figure shows that the action profiles for the ECMS and the AC models are similar.

And Table 3 shows the comparison simulation results between the ECMS and the AC agent. In Table 3, FC stands for the fuel consumption and $\mathrm{FC}_{@\mathrm{ref}}$ stands for the equivalent fuel consumption. It can be seen that the difference between the equivalent fuel consumption for the AC model and the equivalent fuel consumption for the ECMS is 0.3%, which is very similar.

(a)



(b)



(c)

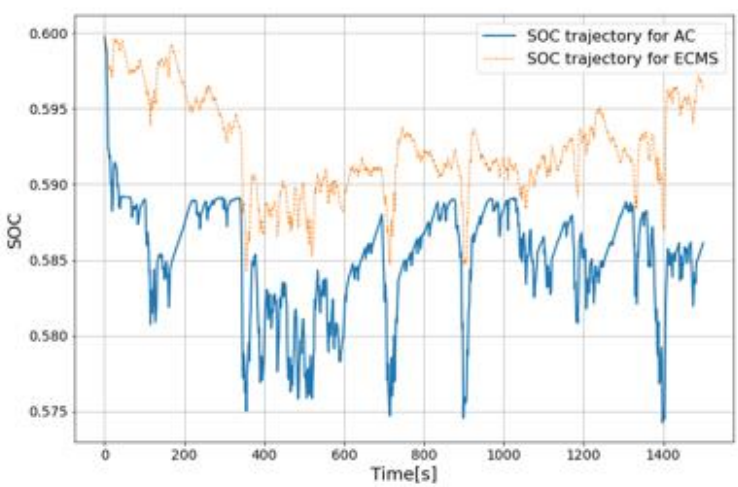


Figure 45. Generalization performance test for the DRL model: (a) driving cycle from the MDP-driver model, (b) action profiles from the two models, (c) SOC trajectories for the two models

Table III. Comparison simulation results for the two models on a driving cycle generated from MDP-driver model

	ECMS	AC
Final SOC	0.596	0.586
FC [g]	66.3	65.7
$\mathbf{FC_{@ref}}$ **[g]**	66.5	66.7
$\Delta \boldsymbol{FC}$ **[%]**	-	0.3%

And Figure [46] is the scatter plot showing the equivalent fuel consumption of the ECMS and the AC model in 100 simulations based on the driving cycles generated from the MDP-driver model. The x-axis represents the equivalent fuel consumption for the ECMS algorithm, and the y-axis represents the equivalent fuel consumption for the AC model. The blue dotted line is the decision boundary that classifies the efficiency advantages of the two models. If the points are distributed above the decision boundary, it means that the efficiency of the ECMS is better than the DRL-based AC agent, and if the points are distributed below the decision boundary, it means that the efficiency of the AC model is better than the ECMS. Although most of the points are distributed above the decision boundary, it can be seen that they exist near the decision boundary, which means that the difference for the efficiency between the two algorithms is small.

Then, the experimental results for the 100 validation simulations between the ECMS and the AC agent are shown in Table 4. In simulations, the ECMS showed better efficiency than the DRL model in the general point of the view, but the difference in fuel efficiency between the two models is less than 0.84%. Considering the fact that the AC agent performs control by reflecting only the current state, it can be said that the AC agent guarantees a high level of the generalization performance.

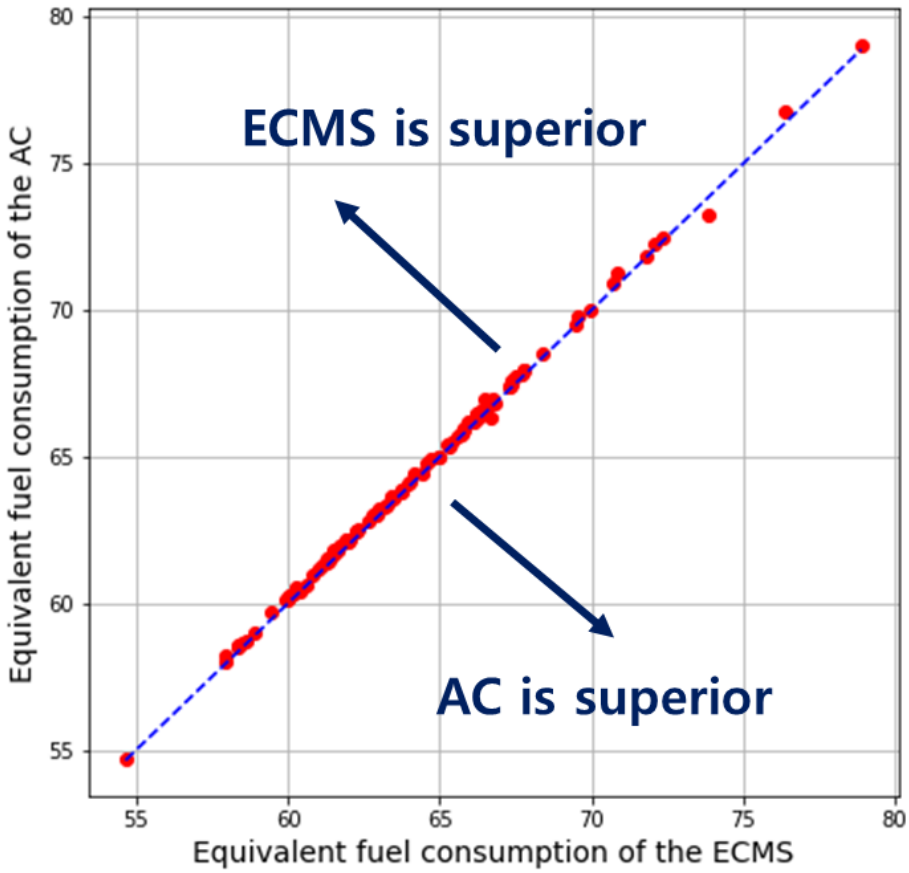


Figure 46. Scatter plot between the AC agent and the ECMS for equivalent fuel consumption

Table IV. Comparison results between the AC model and the ECMS on the 100 driving cycles generated from the MDP-driver model

	ECMS	**AC**
Number of wins	98 / 100	2 / 100
Biggest difference with ECMS (%)	-	+ 0.84
Lowest difference with ECMS (%)	-	- 0.79
Average difference with ECMS (%)	-	+ 0.25

3.5. Development of the Agent considering the Degradation of the Fuel Cell Stack

Research so far has focused on deriving power distribution strategies that minimize the fuel consumption and ensure the SOC sustainability. Through the previous research, we developed the methodology to train the DRL model that can guarantee the scalability and the generalization. However, since the fuel cell stack is vulnerable to the deterioration, it is necessary to develop the power distribution strategy that considers the degradation of the fuel cell.

The degradation of the fuel cell stack is caused by a wide variety of causes such as mechanical stress, reduction of the surface area of the catalyst, contamination, etc. [37]. The deterioration of the fuel cell stack for the FCHEV is caused by mechanical shock as well as by electrochemical reaction. In addition, since one degradation factor affects other degradation factors, the deterioration of the fuel cell stack is very complex. As such, since the degradation process of the fuel cell stack of the FCHEV involves a lot of complexity, many studies have been conducted to model the deterioration of the fuel cell through a data-driven approach [39].

In a representative study to diagnose the FCHEV's fuel cell degradation based on using a data driven approach, the stack operation mode was divided into four and the deterioration of the fuel cell according to each operation mode was quantified (H. Chen, 2015). “Idling operation”, “high power condition”, “load change operation”, and “start & stop operation” were defined as four operation modes that cause degradation in the fuel cell stack in the study. Table 5 shows the degree of deterioration of the fuel cell stack according to each operating mode. In this study, the fuel cell degradation model was developed with reference to the relevant research, and the energy management strategy of the FCHEV that that reflects the deterioration of the fuel cell stack was developed based on this degradation model.

Table V. Fuel cell stack degradation model according to the operation mode

Operation conditions	Voltage degradation
Load change operation	0.4185 $\mu V/cyc$
Idling operation	8.662 $\mu V/h$
High power condition	10.00 $\mu V/h$
Start & Stop operation	13.4 $\mu V/cyc$

3.5.1. Reformulation of the reward considering the degradation

Since there is no degradation-related term in the reward designed in the previous study, it is necessary to modify the existing reward to develop the DRL-based control strategy that reflects the degradation of the fuel cell stack. We designed a new reward as Eq (36) to develop the DRL model that considers the degradation of the fuel cell stack. In Eq (36), $\Delta\dot{V}$ and σ mean the instantaneous voltage drop of the fuel cell due to the degradation and the equivalent factor that equalize the instantaneous voltage drop to fuel consumption rate.

$$R = -\left(\dot{m}_{fc} + \sigma\Delta\dot{V}\right) - \gamma\left|SOC - SOC_{ref}\right| \tag{36}$$

We wanted the reward considering the degradation of Eq (36) to maintain the same format as the reward that does not reflect the degradation of Eq (19), which consists of two components: the fuel consumption and the SOC deviation. Therefore, we introduced the equivalent factor of the stack degradation for the fuel consumption rate of σ in Eq (36), and it has a unit of $[g/\mu V]$. We carried out an economic evaluation such as Eq (37)-(39) to derive an appropriate value for σ. In Eq (37),

$Cost_{stack}$ and $Cost_{kw}$ represent stack price and stack price per kW power. In Eq (37), $Cost_{drd}$ means the economic cost due to the stack degradation, ΔV_{limit} represents limiting voltage-drop, and the lifespan of the stack is considered until the deterioration exceeding the limiting voltage-drop occurs. And $Cost_{H_2}$ of Eq (39) represents hydrogen price. In this study, $Cost_{kw}$, ΔV_{limit}, and $Cost_{H_2}$ were defined as 40[$/kW], 0.12V, and 8[$/kg], respectively, based on related research and current market conditions [48, 49]. As a result, we derive the value of σ as 2.79 $[g/\mu V]$. In other words, the voltage drop corresponding to 1 μV is equivalent to additional consumption of 2.79 g of hydrogen.

$$Cost_{stack} = P_{fc,max} \cdot Cost_{kw} \tag{37}$$

$$Cost_{drd} = \frac{Cost_{stack}}{\Delta V_{limit}} \tag{38}$$

$$\sigma = \frac{Cost_{drd}}{Cost_{H_2}} \tag{39}$$

And we added a term related to the additional fuel consumption due to the voltage-drop to the reward. As shown in Eq (40), the power that the fuel cell should be responsible for an arbitrary demanding power is the same when deterioration occurs and when no deterioration occurs. In Eq (40), δV refers to the voltage-drop due to the stack degradation, and δj refers to the current density to be increased due to the stack degradation.

$$V_{cell} \cdot j = (V_{cell} - \delta V)(j + \delta j) \tag{40}$$

If we ignore the term for $\delta V \cdot \delta j$, Eq (40) can be expressed as Eq (41).

$$\delta j = \frac{j}{V_{cell}} \delta V \tag{41}$$

Then, the fuel consumption rate, $\dot{m}_{fc,add}$ additionally consumed by δj is equal to Eq (42). In Eq (42), M, N, A, and F denote the molecular mass for hydrogen, the number of cells, the cell area, and Faraday number, respectively.

$$\dot{m}_{fc,add} = M \frac{NA\delta j}{2F} = M \frac{NAj}{2FV_{cell}} \delta V \tag{42}$$

Finally, the designed reward is represented to Eq (43). It should be noted that there is a difference in that $\Delta\dot{V}$ represents the instantaneous voltage-drop caused by the stack degradation, and δV represents the voltage-drop accumulated by the stack degradation during the simulation in Eq (43).

$$R = -\left(\dot{m}_{fc} + \sigma\Delta\dot{V} + M \frac{NAj}{2FV_{cell}} \delta V\right) - \gamma\left|SOC - SOC_{ref}\right| \tag{43}$$

Therefore, the structure of the reward considering the stack degradation can be considered to be divided into a term related to the fuel consumption rate and a term related to the SOC deviation, similar to the structure of the reward which is not reflect the degradation. Figure [47] clearly shows the reward structure. In Figure [47], $(\dot{m}_{fc} + \sigma\Delta\dot{V} + M \frac{NAj}{2FV_{cell}} \delta V)$ is a term related to the fuel consumption which contains actual fuel consumption and converted fuel consumption due to the deterioration. Therefore, the term related to the fuel consumption is called the effective fuel consumption rate and is expressed as Eq (44) in this study.

$$\dot{m}_{eff} = \dot{m}_{fc} + \sigma\Delta\dot{V} + M\frac{NAj}{2FV_{cell}}\delta V \quad (44)$$

$$R = -(\dot{m}_{fc} + \sigma\Delta\dot{V} + M\frac{NAj}{2FV_{cell}}\delta V)$$

Effective Fuel Consumption

$$-\gamma|SOC - SOC_{ref}|$$

SOC deviation

Figure 47. Configuration of the reward with two terms

3.5.2. Development of the power distribution strategy considering the stack degradation

Based on the reward reconstructed in the previous study, the DRL agent that considers stack degradation was trained. When training the agent through the reward that reflects the stack deterioration, the reward factor was selected as 28.25.

We compared the simulation results of the agent trained based on the new reward and the agent in the previous study derived through the reward that does not include the degradation factor. Table 6 shows the main characteristics of the agent without considering the stack deterioration and the main characteristics of the agent considering the stack deterioration. In the table, “Agent 1” refers to the agent trained without considering the degradation factor, and “Agent 2” refers to the agent trained by considering the degradation factor. Compared to the agent 1, the agent 2 not only

has the different reward, but also has the different state. The action of the previous step, a_{prev} is added to the existing state in the state for the agent 2.

Since one of the main deterioration factors of the stack is the load change, if the current density by action is different from the current density in the previous step, the stack degradation occurs. Therefore, the state of size 5, such as Eq (45), was constructed so that the agent2, which make policy in consideration of the deterioration, can cope with the degradation of the stack due to load change.

$$s = [P_{dmd}, \Delta SOC, j_{min}, j_{max}, a_{prev}] \tag{45}$$

Table VI. Main features of the trained agents

	Agent 1	**Agent 2**
Algorithm	Actor-Critic	Actor-Critic
State configuration	$[P_{dmd}, \Delta SOC, j_{min}, j_{max}]$ Size of the state = 4	$[P_{dmd}, \Delta SOC, j_{min}, j_{max}, a_{prev}]$ Size of the state =5
Action configuration	$a = \hbar_{out}(z) \cdot j_{max}$	$a = \hbar_{out}(z) \cdot j_{max}$
Reward configuration	$R = -(\dot{m}_{fc} + \gamma\lvert\Delta SOC\rvert)$	$R = -(\dot{m}_{eff} + \gamma\lvert\Delta SOC\rvert)$
Reward factor	5.80	28.25

Figure [48] shows the test cycle derived through the MDP-driver model, and Figure [49] shows the results of the comparative experiment of the two agents. The (a) of Figure [49] refers to the cumulative fuel consumption according to the time of the two agents, and the (b) refers to the cumulative voltage-drop according to the time of the two agents. In Figure [49], the blue solid line represents the simulation result for the agent 1, and the orange solid line represents the simulation result for the agent 2. Figure [49] shows that agent 1 is somewhat more efficient than agent 2 in terms of the fuel consumption, but agent 2 shows a great advantage over agent 1

in terms of the degradation of fuel cell stack. Figure [49] shows that the agent 1 is somewhat more efficient than the agent 2 in terms of the fuel consumption, but the agent 2 shows a great advantage over the agent 1 in terms of the degradation of the fuel cell stacks. The results of the comparative experiment are specified in Table 7. In Table 7, ΔFC and ΔDegradation are the relative difference of the agent 2 to the agent 1 in terms of the equivalent fuel consumption and the relative difference of the agent 2 to agent 1 in terms of the degradation. In terms of the equivalent fuel consumption, the agent 2 consumes about 19% more fuel than the agent 1, but in terms of the fuel cell degradation, the voltage-drop from the agent 2 is reduced by about 80% compared to the voltage-drop from the agent 1.

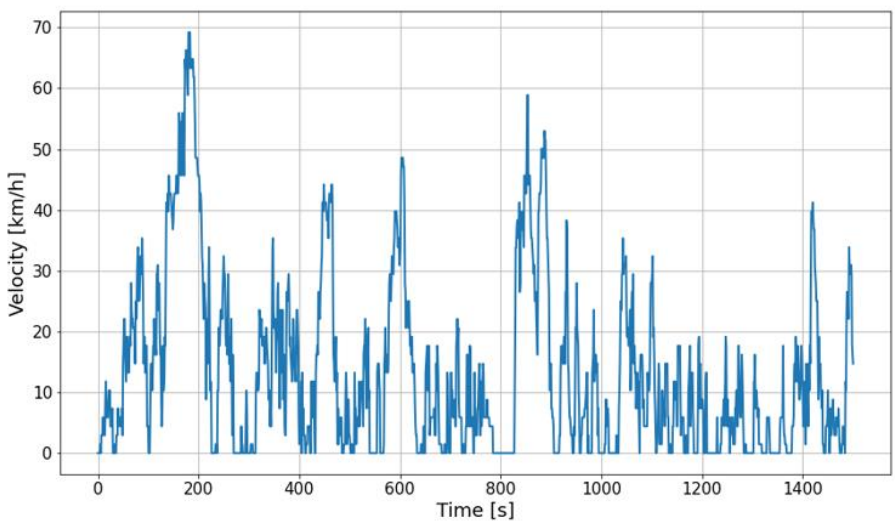


Figure 48. Test cycle from MDP-driver model

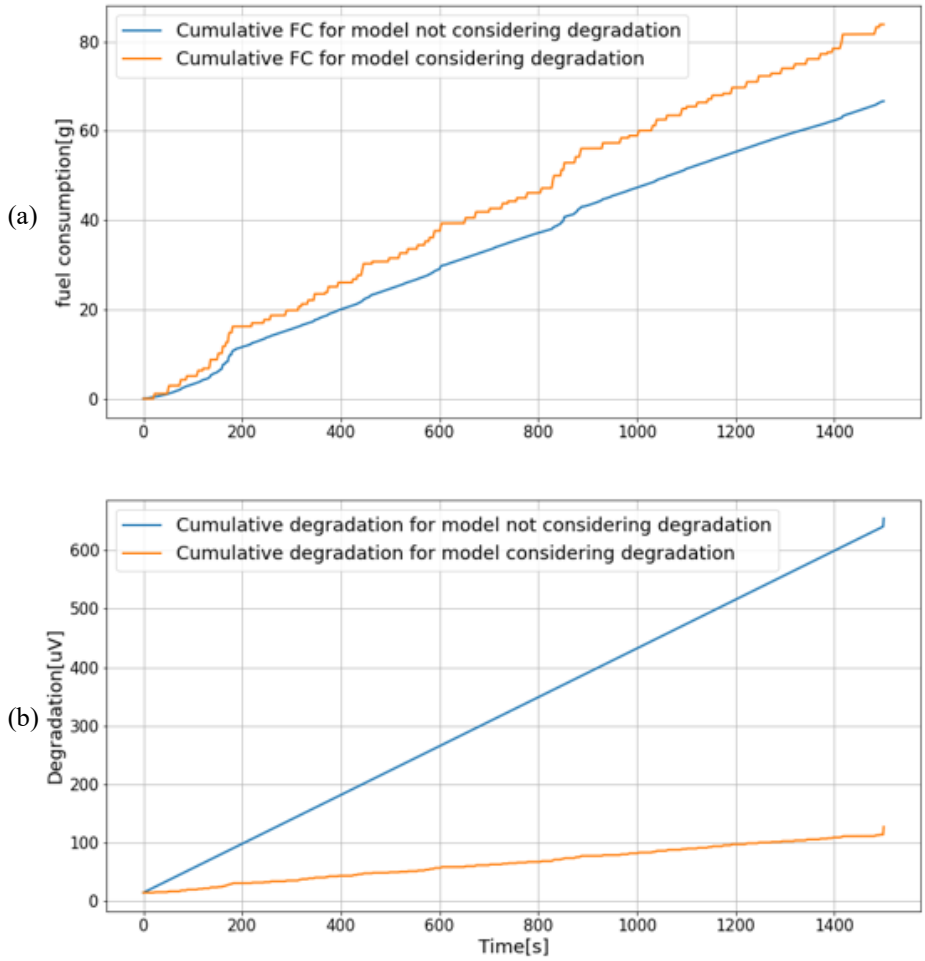


Figure 49. Comparative experiment between the agent1 and the agent2: (a) comparison results regarding the fuel consumption (b) comparison results regarding the voltage-drop due to the degradation

Table VII. Simulation results with the agent 1 and the agent 2

	Agent 1	**Agent 2**
$\boldsymbol{FC_{@SOC_{ref}}}$ $[\boldsymbol{g}]$	67.7 $(SOC_f = 0.586)$	84.0 $(SOC_f = 0.597)$
Degradation $[\boldsymbol{\mu V}]$	655.0	126.7
ΔFC [%]	-	+19.4%
ΔDegradation [%]	-	-80.7%

We carried out the comparison experiment on 100 test-cycles generated through the MDP-driver model for a robust comparison experiment. Figure [50] and Table 8 show the comparison experiment results in terms of the fuel consumption. Figure [50] shows the scatter plot of the fuel consumption consumed by the agent 1 and the agent 2 through the 100 driving cycles. The blue dotted line is the decision boundary. If the red dot is on the left based on the decision boundary, it means that the agent 1 is more efficient than the agent 2, and if the red dot is on the right based on the decision boundary, the agent 2 is more efficient than the agent 1. It can be seen that all points of the scatter plot are formed on the left based on the decision boundary, and the related results are clearly seen in Table 8. The agent 1 shows superior performance in terms of the fuel consumption compared to the agent 2 in all 100 test drives, and the agent 1 shows excellent fuel efficiency performance of about 24% on average compared to the agent 2.

On the other hand, Figure [51] and Table 9 show the comparison experiment results in terms of the stack degradation. Figure [51] shows the scatter plot of the voltage-drop caused by the stack degradation for the agent 1 and the agent 2 according to the 100 test cycles. Based on the decision boundary of the scatter plot, all red dots are on the right, which means that the control strategy of the agent 2 is superior to the control strategy of the agent 1 in terms of the stack degradation. And the dots of the scatter plot are concentrated in the lower right region, which means that the deterioration of the stack has low dependency on the driving-cycles. Table 9 shows the comparison results for the control strategy of the agent 2 and the control strategy of the agent 1 regarding the voltage-drop. It can be seen that the agent 2 shows superior performance in terms of the degradation compared to the agent 1 for all comparison experiments. On average, the agent 2 can reduce the voltage-drop caused by the degradation by about -78% compared to the agent 1.

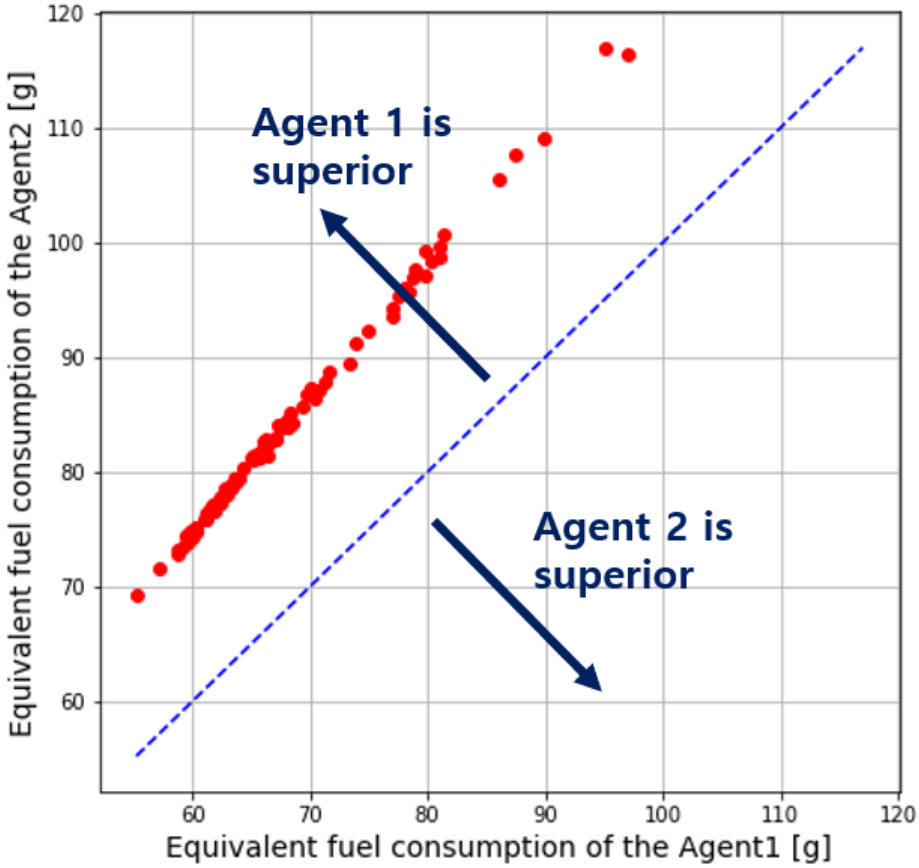


Figure 50. Scatter plot of agent1's fuel consumption and agent2's fuel consumption on 100 driving cycles

Table VIII. Comparison experiment results in terms of the fuel consumption with 100 driving cycles

Fuel Consumption	**Agent 1**	**Agent 2**
Number of wins	100 / 100	0 / 100
Maximum Δ (%)	-	+25.5
Minimum Δ (%)	-	+20.0
Average Δ (%)	-	+23.9

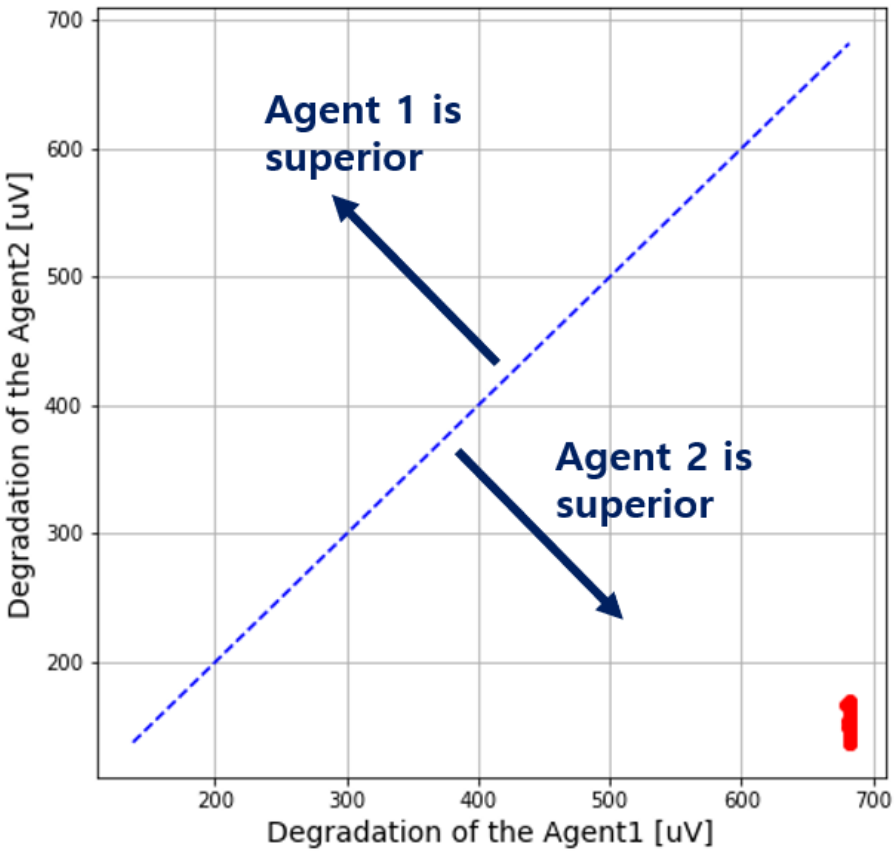


Figure 51. Scatter plot of agent1's stack degradation and agent2's stack degradation on 100 driving cycles

Table IX. Comparison experiment results in terms of the voltage-drop with 100 driving cycles

Degradation	**Agent 1**	**Agent 2**
Number of wins	0 / 100	100 / 100
Maximum Δ (%)	-	-75.0
Minimum Δ (%)	-	-79.8
Average Δ (%)	-	-78.1

As such, it can be seen that the agent 2 shows very superior performance with regard to the stack degradation compared to the agent 1. In order to find out the cause of the large difference in the voltage-drop between the agent 2 and the agent 1, we created a validation cycle through the MDP-driver model and derived the action profiles of the two agents. Figure [52] shows the action profiles of two agents for the validation cycle. The (a) and (b) of Figure [52] show the action profiles of the agent 1 and the agent 2, respectively. We can confirm that agent 2 relatively reduces the load change operation by actively using the idling operation through (b) of Figure [52].

Table 10 represents the voltage-drop for each operation mode that occurred while the agent 1 and the agent 2 drive the same validation cycle. It is noteworthy that the load change operation exerts a dominant influence on the overall stack degradation for both the agent 1 and the agent 2. The phenomenon that the load change operation has a great influence on the fuel cell degradation has been proven through past studies [42].

From the results of Figure [52] and Table 10, we can see that the agent 2 suppresses the load change operation that dominates the stack deterioration by actively utilizing the idling operation, which has relatively little influence on the degradation. The control strategy of the agent 2 shows about 80% superior performance than the control strategy derived by the agent 1 in terms of the total degradation.

And Figure [53] visualizes the total amount of effective fuel consumption generated by driving of two agents on the validation cycle. The agent 1 and agent 2 consume 1920g and 464g of the effective fuel consumption, respectively. As a result, the agent 2 shows improved performance of 75.8% compared to the agent 1 with regard to the effective fuel consumption through the reduction of the fuel cell degradation.

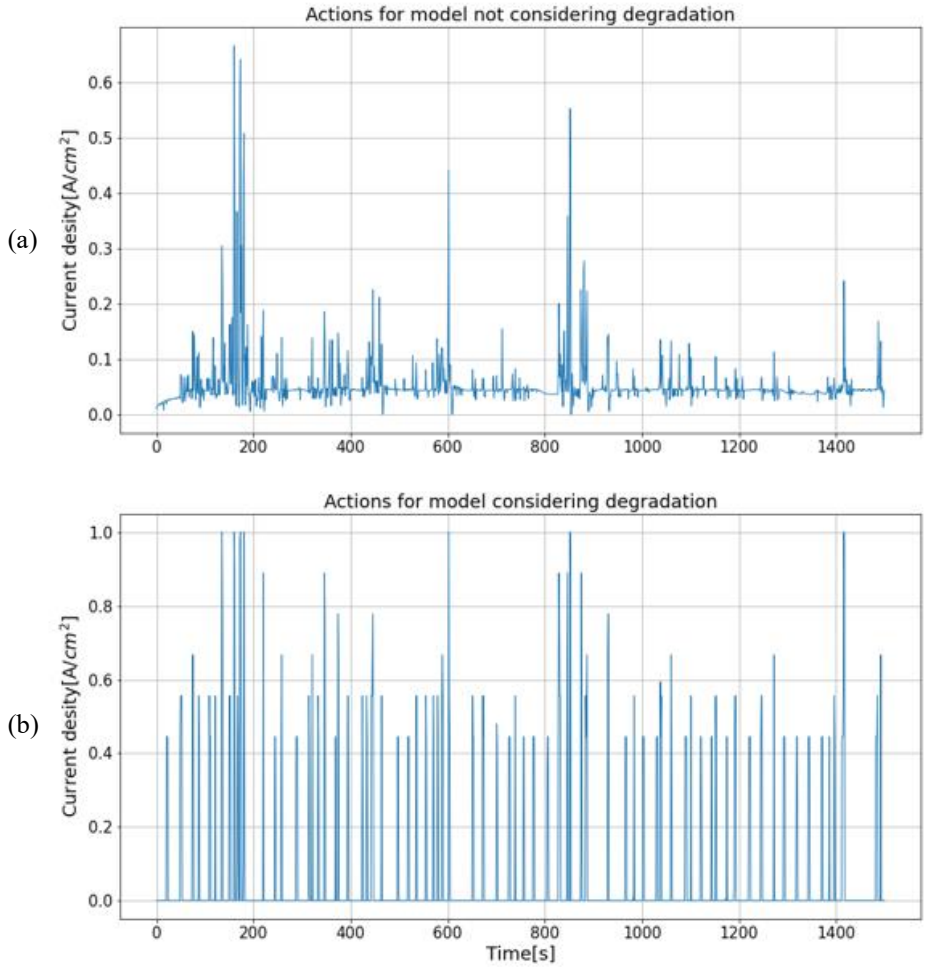


Figure 52. Action profiles of two agents on the validation cycle generated from MDP-driver model: (a) action profile of the agent 1, (b) action profile of the agent 2

Table X. Voltage-drop by the operation conditions for two agents

Degradation Factors	**Agent 1**	**Agent 2**
Idling (μV)	0.007	3.15
High load (μV)	0.0	0.06
Load change (μV)	628.2	96.7
Start & Stop (μV)	26.8	26.8
Total (μV)	655.0	126.7

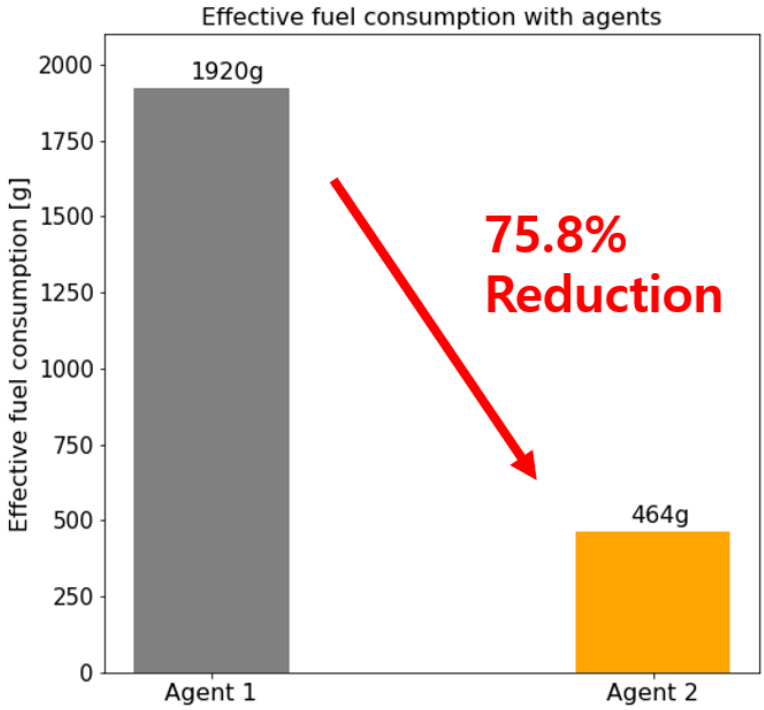


Figure 53. Effective fuel consumption of the two agents

3.5.3. Developing an improved DRL model

We developed a DRL agent that can reduce the stack degradation and the effective fuel consumption by modifying the reward and the state. However, since the DRL agent derives a continuous action value, it is difficult to effectively reduce the deterioration caused by the load change operation that dominates the deterioration. In the previous study, it was also confirmed that the agent performing energy management considering deterioration took a strategy of actively utilizing the idling operation to reduce the load change operation. In the previous study, it was also confirmed that the agent that performs energy management in consideration of the stack degradation actively utilizes the idling operation to reduce the load change operation.

We modified the configuration of the action to develop a DRL model that can more effectively cope with the stack degradation caused by the load change operation. To this end, we added a new action to the action configuration of the DRL model that determines whether to take the action of the current step the same as the action of the previous step. Figure [54] shows the structure of the actor network for the newly constructed DRL model. The newly configured DRL model differs from other DRL models in that the number of units of the output layer is two. In this study, the DRL model with the structure shown in Figure [54] was defined as "agent 3". The action of agent 3 is derived as Eq (46) based on the two output-values a_1 and a_2 derived from the output layer. In Eq (46), $\hbar_{step}(x)$ is defined as a binarized step function that becomes 1 when x is greater than or equal to 0.5, and 0 when x is less than 0.5, as in Eq (47).

$$a = \hbar_{step}(a_1) \cdot a_{prev} + [1 - \hbar_{step}(a_1)] \cdot (a_2 \cdot j_{max}) \tag{46}$$

$$\hbar_{step}(x) = \begin{cases} 1, & x \geq 0.5 \\ 0, & x < 0.5 \end{cases} \tag{47}$$

Therefore, a_1 can be viewed as an output value that determines whether to use the action of the previous step as the current action value. If a_1 is greater than 0.5, the current action is selected as the action value of the previous step, and if a_1 is less than 0.5, a new action value calculated based on a_2 is taken.

As another model, we used the Deep Q-Network (DQN) model to derive the discrete action [25]. Since the load change operation has a profound effect on the stack degradation and the effective fuel consumption, we judged that an effective power distribution strategy could be derived through the DQN model that derives the discrete actions.

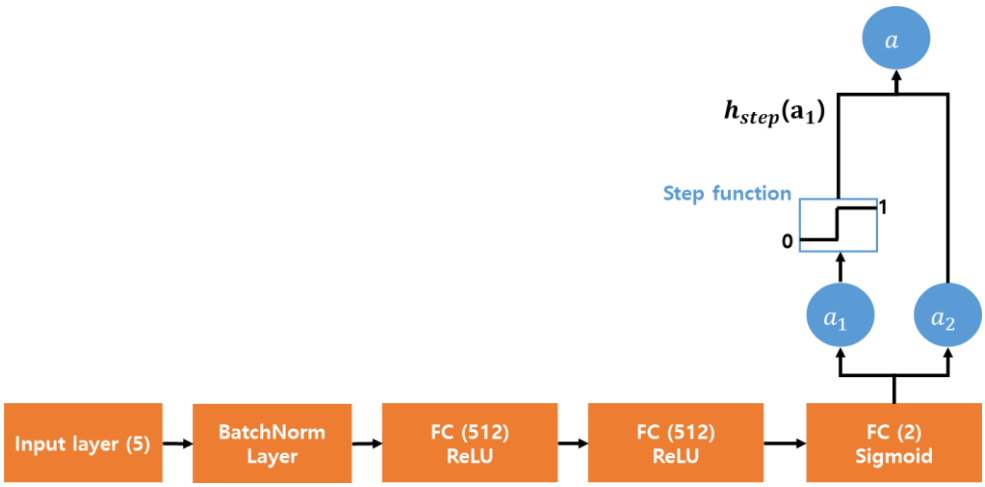


Figure 54. Architecture of the DRL model with two actions

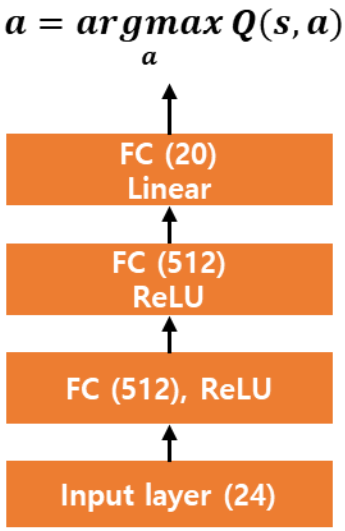


Figure 55. Structure of the DQN model that derives discretized action

Figure [55] shows the architecture of the DQN model. In this study, the discrete action size was set to 20. Therefore, the number of units of the output layer in the DQN model is 20, and each output value is an estimated value of the Q-value corresponding to each action. And the DQN model selects the action that expects the

largest Q-value among the estimated Q-values as in Eq (48) as an action. Another point to note in the DQN model is that the size of the input layer is 24. Since we express the discretized action of the previous step in one-hot encoding format, the action of the previous step is expressed in the form of a vector of size 20. Therefore, the state input to the DQN model has a total size of 24 by adding the action of the previous step in a vector format of size 20 to the basic state of size 4. In this study, the DQN model that derives these discrete actions was named "agent 4".

$$a = \underset{a}{\operatorname{argmax}}\, Q(s, a; \theta) \tag{48}$$

Table11 shows the main features of the agent 3, which has an actor-critic structure that derives two actions, and the agent 4, which is based on a DQN model with 20 action sizes. The reward factors of the agent 3 and the agent 4 derived through the training were selected as 25.4 and 20.49, respectively.

As a result, the agent 1, which establishes power distribution strategy without considering the stack deterioration, and the agent 2, the agent 3, and the agent 4, which establish power distribution strategy considering the stack degradation were developed through this study. Similar to the previous study, we made 100 validation cycles using the MDP-driver model for the systematic comparison of the 4 agents and we conducted the performance comparison experiments of the 4 agents on 100 validation cycles. The performance of the agent was evaluated in three aspects including the fuel consumption, the voltage-drop due to degradation, and the SOC-sustainability.

Table XI. Main features of the agent 3 and the agent 4

	Agent 3	**Agent 4**
Algorithm	Actor-Critic	DQN
State configuration	$[P_{dmd}, \Delta SOC, j_{min}, j_{max}, a_{prev}]$ Size of the state = 5	$[P_{dmd}, \Delta SOC, j_{min}, j_{max}, a_{prev}]$ Size of the state =24
Action configuration	$a = h_{st}(a_1) \cdot a_{prev} + (1 - h_{st}(a_1)) \cdot a_2$	$a = \underset{a}{argmax}\, Q(s, a)$
Reward configuration	$R = -(\dot{m}_{eff} + \gamma \lvert \Delta SOC \rvert)$	$R = -(\dot{m}_{eff} + \gamma \lvert \Delta SOC \rvert)$
Reward factor	25.4	20.49

Figure [56] is a matrix-type scatter plot comparing four agents in terms of the fuel consumption. The matrix has a size of 4×4, and the 1st, 2nd, 3rd, and 4th columns of the matrix correspond to agent 1, agent 2, agent 3, and agent 4 respectively. And, the 1st, 2nd, 3rd, and 4th rows of the matrix corresponds to agent 4, agent 3, agent 2, agent 1 respectively. For example, the comparison result of the fuel consumption with the agent 2 and the agent 3 is shown in the scatter plot corresponding to (2, 2) or the scatter plot corresponding to (3, 3) in the matrix. In the scatter plot corresponding to the (2, 2) element in the matrix, the x-axis represents the fuel consumption for the agent 2 and the y-axis represents the fuel consumption for the agent 3. It can be seen that the points are clustered on the right based on the decision boundary of the scatter plot, which means that the agent 3 has superior performance in terms of the fuel consumption compared to the agent 2. On the contrary, the x-axis of the scatter plot corresponding to the (3, 3) element in the matrix represents the fuel consumption for the agent 3, and the y-axis represents the fuel consumption for the agent 2. Since the points are distributed on the left based on the decision boundary, the agent 3 is more efficient than the agent 2. In this study,

a matrix-type scatter plot is named “scatter plot matrix”. In this study, this matrix type scatter plot is named “scatter plot matrix”.

The scatter plot for the first column of the matrix, that is, the elements (1, 1), (2, 1), and (3, 1), shows that the points are distributed on the left side based on the decision boundary. In other words, the agent 1 shows superior performance in terms of the fuel consumption compared to the other three agents. The specific results of the comparative experiments of the four agents related to the fuel consumption are represented through Table 12 and Table 13.

Table 12 shows the rankings recorded by each agent during 100 experiments in terms of the fuel efficiency. Table 12 shows that for all experiments, agent 1 exhibits superior performance in terms of the fuel consumption compared to the other agents. And except for the agent 1, the agent 3, the agent 2, and the agent 4 have good fuel efficiency in the order.

Table 13 shows the relative difference of the average fuel consumption between the four agents. The relative difference between the i-th row and the j-th column of Table 13 is calculated as Eq (49), where F_{agent_j} represents the performance value for the “agent j”, and F_{agent_i} represents the performance value of the “agent i”. In the case of Table 13, the performance value is the fuel consumption. Column 1 of Table 13 shows the relative difference of the fuel consumption between agent 1 and the remaining agents. It can be seen that the agent 1 uses about 20% less fuel than the agent 4 and the agent 1 uses about 6% less fuel than the agent 3. Considering that the agent 1 is trained regarding the fuel consumption and the SOC-sustainability without including the degradation, it is reasonable that it shows higher performance in terms of the fuel consumption compared to other agents.

$$\Delta = \frac{F_{agent_j} - F_{agent_i}}{\max\left(F_{agent_j},\ F_{agent_i}\right)} \qquad (i \neq j) \tag{49}$$

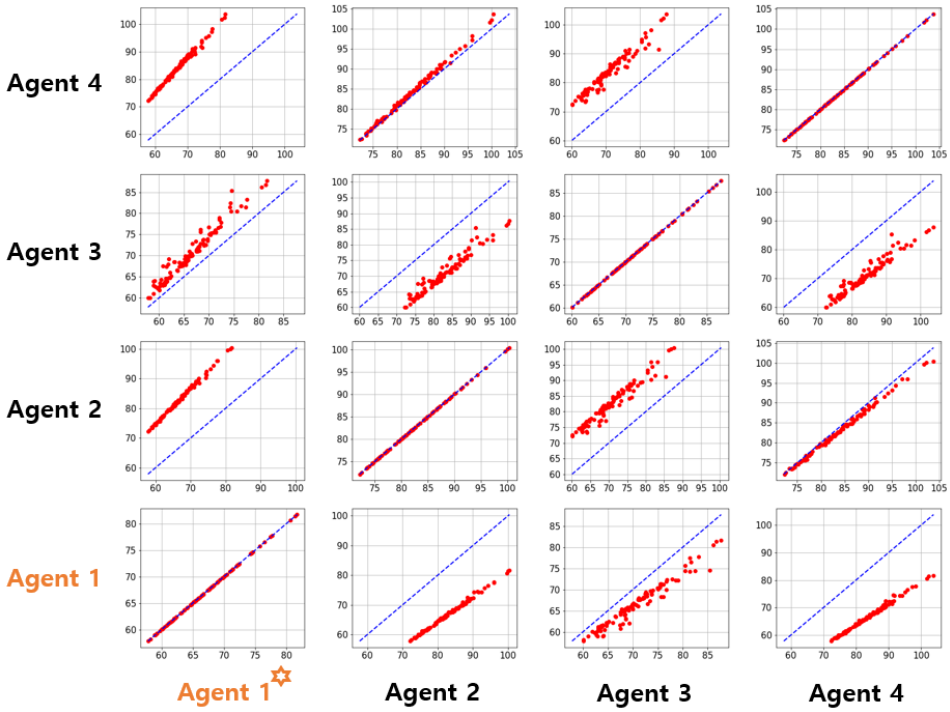


Figure 56.Matrix of the scatter plots comparing four agents for the fuel consumption

Table XII. Ranking for the fuel consumption

	Agent 1	Agent 2	Agent 3	Agent 4
# of 1st	100 / 100	0 / 100	0 / 100	0 / 100
# of 2nd	0 / 100	0 / 100	100 / 100	0 / 100
# of 3rd	0 / 100	97 / 100	0 / 100	3 / 100
# of 4th	0 / 100	3 / 100	0 / 100	97 / 100

Table XIII. Relative difference for the average fuel consumption

	Agent 1	Agent 2	Agent 3	Agent 4
Agent 1	-	+ 19.4	+ 6.00	+ 20.3
Agent 2	- 19.4	-	- 14.31	+ 1.16
Agent 3	- 6.00	+ 14.31	-	+ 15.30
Agent 4	- 20.3	- 1.16	- 15.30	-

And we compared the performance of the four agents in terms of the stack degradation. As when comparing the fuel consumption, the results of the comparative experiment were shown through the scatter plot matrix, the table related to the ranking, and the table for the relative difference. Figure [57], Table 14, and Table 15 show the results of voltage-drop due to the stack degradation.

Figure [57] shows the scatter plot matrix for voltage-drop. The scatter plot corresponding to the elements (1, 1), (2, 1), and (3, 1) related to the 1st column shows that the points exist on the right side of the decision boundary. This means that the agent 1 is inferior to the other three agents in terms of the stack degradation. On the other hand, the scatter plot for the 3rd column related to the agent 3 shows that the points are distributed on the left side based on the decision boundary, which means that the agent 3 shows superior performance in terms of the degradation compared to other agents.

Table 14 shows the rankings recorded by four agents during 100 test cycles for the voltage-drop. Through Table 14, we can confirm that the ranking of the agents for the stack degradation is clearly classified. In other words, the agent 1, the agent 2, the agent 3, and the agent 4 ranked 4th, 2nd, 1st and 3rd respectively in all 100 validation cycles. Table 15 shows the relative difference between the agents for the degradation. Through column 3 related to the agent 3 of Table 15, we can confirm that the agent 3 has an overwhelming advantage over other agents in terms of the degradation. When compared with the agent 1, the agent 3 shows the result of reduction for the degradation by about 91%, and the agent 3 shows the result of the degradation is reduced by about 52% compared with the agent 2. On the other hand, if you look at the 1st column related to the agent 1 in Table 15, it is apparent that the agent 1 does not cope well with the deterioration. Compared to the agent 2, the agent 1 shows 81.4% more degradation of the stack, and the agent 1 shows 80.5% more degradation compared to the agent 4, as shown in 1st column of Table 15.

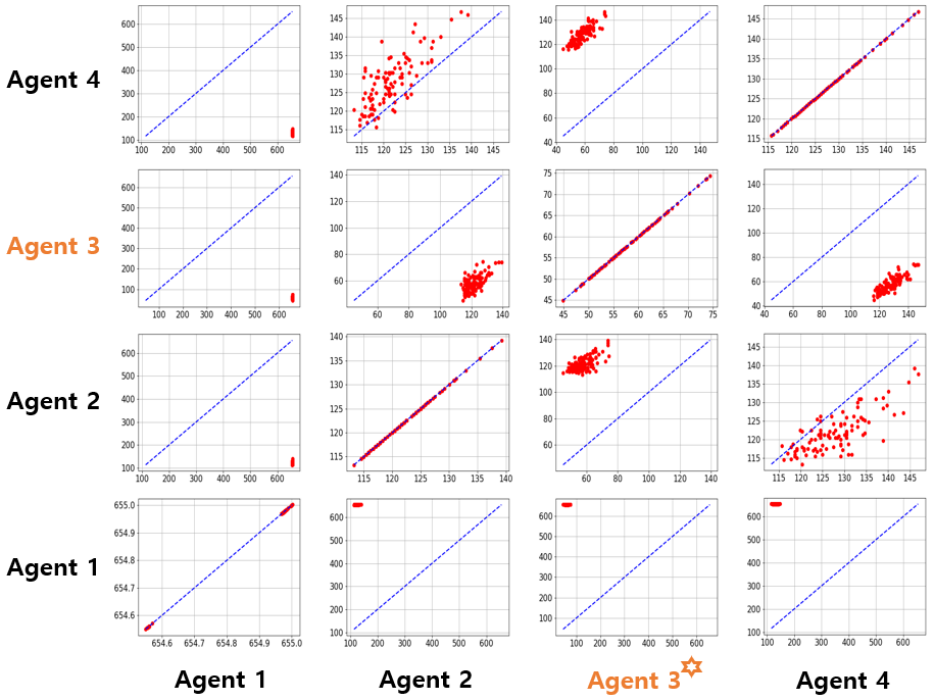


Figure 57. Matrix of the scatter plots comparing four agents for the voltage-drop due to the stack degradation

Table XIV. Ranking for the stack degradation

	Agent 1	Agent 2	Agent 3	Agent 4
# of 1st	0 / 100	0 / 100	100 / 100	0 / 100
# of 2nd	0 / 100	91 / 100	0 / 100	9 / 100
# of 3rd	0 / 100	9 / 100	0 / 100	91 / 100
# of 4th	100 / 100	0 / 100	0 / 100	0 / 100

Table XV. Relative difference for the average voltage-drop due to the stack degradation

	Agent 1	Agent 2	Agent 3	Agent 4
Agent 1	-	- 81.4	- 91.1	- 80.5
Agent 2	+ 81.4	-	-52.3	+ 4.74
Agent 3	+ 91.1	+ 52.3	-	+ 54.6
Agent 4	+ 80.5	- 4.74	- 54.6	-

Finally, we compared the performance of the four agents regarding the charge-sustainability. The charge-sustainability is defined as the absolute value of the difference between the final SOC and the initial SOC as in Eq (50).

$$\Delta SOC = \left| SOC_{final} - SOC_{initial} \right| \tag{50}$$

The experiment on the charge-sustainability was analyzed in a similar manner to the previous two performance experiments. Figure [58] shows the scatter plot matrix for the SOC-sustainability. The scatter plot in the 4th column related to the SOC-sustainability of the agent 2 shows that the points are generally distributed on the left side based on the decision boundary. Through this, it can be seen that agent 2 secures higher SOC-sustainability than other agents. Based on this fact, we can see that the agent 2 secures high SOC-sustainability compared to other agents.

From Table 16 and Table 17, it is confirmed that from the viewpoint of the SOC-sustainability, the agent 2, the agent 3, the agent 4, and the agent 1 show excellent performance in order.

And Figure [59] is the scatter plot showing the SOC-sustainability for the agent 1. Therefore, both x-axis and y-axis for Figure [59] correspond to the SOC-deviation for the agent 1, and all points exist on the decision boundary. Figure [59] shows that most of the deviation values between the initial SOC and the final SOC are distributed within approximately 0.025 in 100 experiments for the agent 1, which has the lowest performance for the SOC-sustainability. Therefore, the SOC-sustainability is somewhat different between the agents, but all four agents have excellent performance regarding the SOC-sustainability.

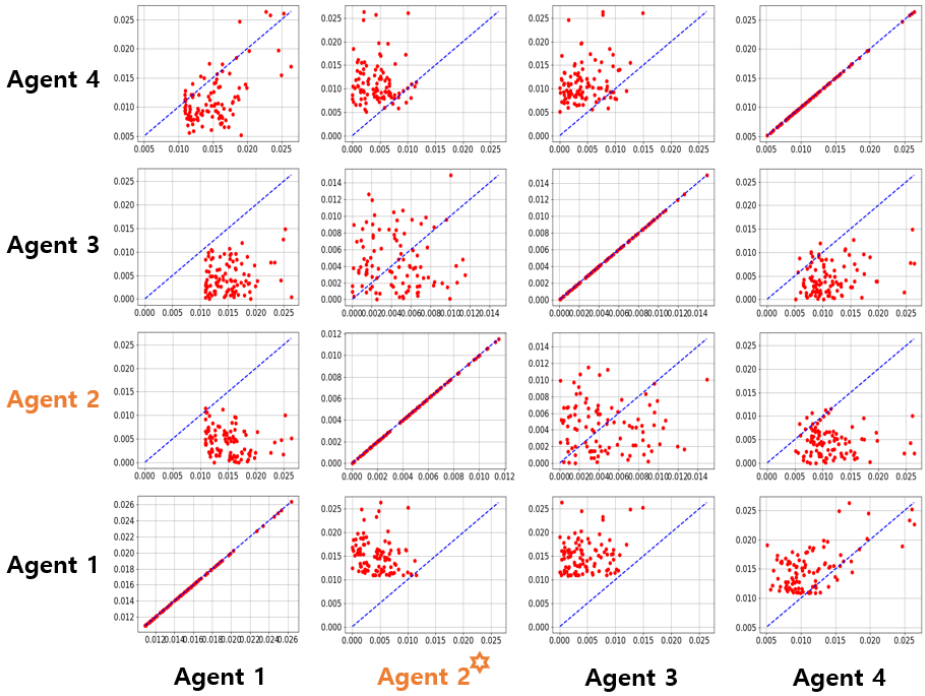


Figure 58. Matrix of the scatter plots comparing four agents for the SOC-sustainability

Table XVI. Ranking for the SOC-sustainability

	Agent 1	Agent 2	Agent 3	Agent 4
# of 1st	0 / 100	51 / 100	48 / 100	1 / 100
# of 2nd	1 / 100	43 / 100	45 / 100	11 / 100
# of 3rd	16 / 100	5 / 100	7 / 100	72 / 100
# of 4th	83 / 100	1 / 100	0 / 100	16 / 100

Table XVII. Relative difference for the average voltage-drop due to the stack degradation

	Agent 1	Agent 2	Agent 3	Agent 4
Agent 1	-	- 70.7	- 69.8	- 25.2
Agent 2	+ 70.7	-	+ 2.97	+ 60.8
Agent 3	+ 69.8	- 2.97	-	+ 59.6
Agent 4	+ 25.2	- 60.8	- 59.6	-

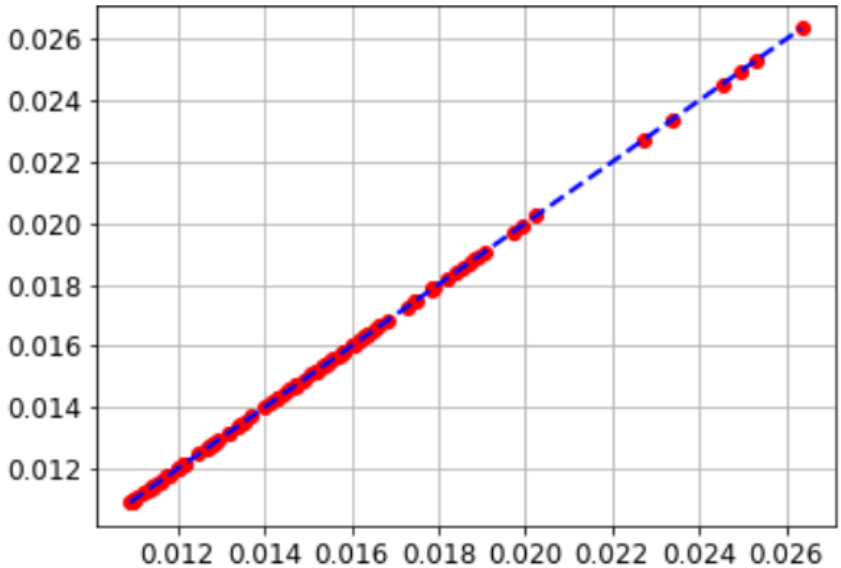


Figure 59. Scatter plot for the SOC-sustainability of the agent 1

We created a random cycle and checked the action profile of each agent for that cycle. Figure [60] shows the generated validation cycle, and Figure [61] shows the action profiles of the agents for the validation cycle. In Figure [61], (a), (b), (c), and (d) represent the action profiles corresponding to the agent 1, the agent 2, the agent 3, and the agent 4.

Figure [61] shows the secret that the agent 3 was able to significantly reduce the degradation compared to other agents. We can see from Figure [61] that the agent 3 minimizes the load change operation while maintaining the previous action.

The average effective fuel consumption for 100-drivings of the four agents is shown in Figure [62]. Figure [62] shows that the control strategy that minimizes the load change operation of the agent 3 has higher efficiency than other agents in terms of the effective fuel consumption. These findings suggest that the effective power distribution strategies for the FCHEV can be developed based on the DRL model with the same structure as the agent 3.

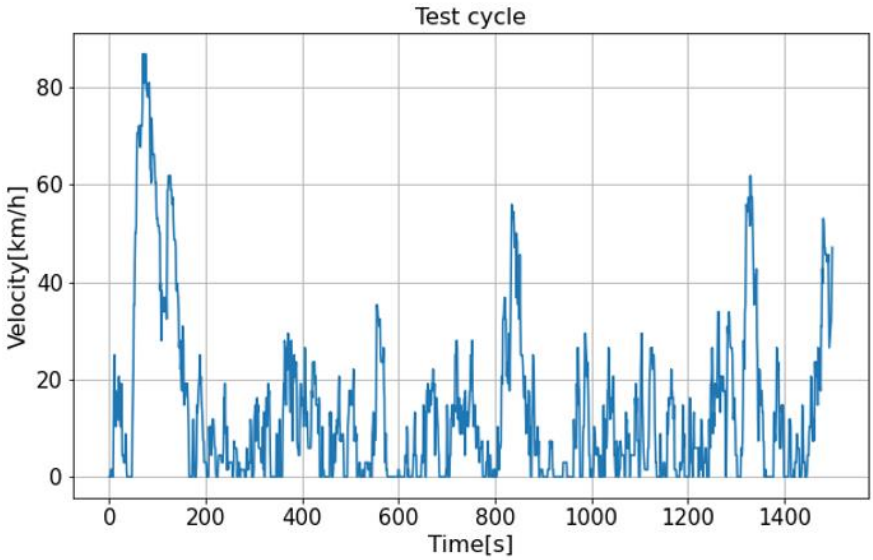


Figure 60. Driving cycle generated from MDP-driver model

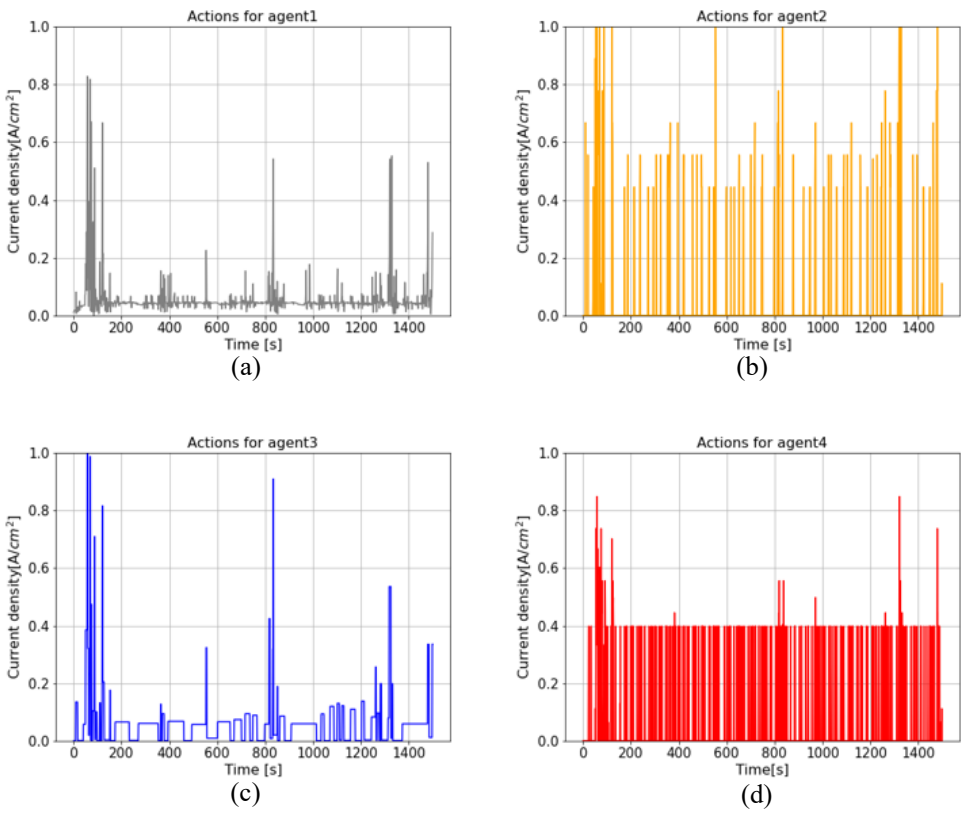


Figure 61. Action profiles with the agents: (a) action profile for the agent 1, (b) action profiles for the agent 2, (c) action profiles for the agent 3, (d) action profiles for the agent 4

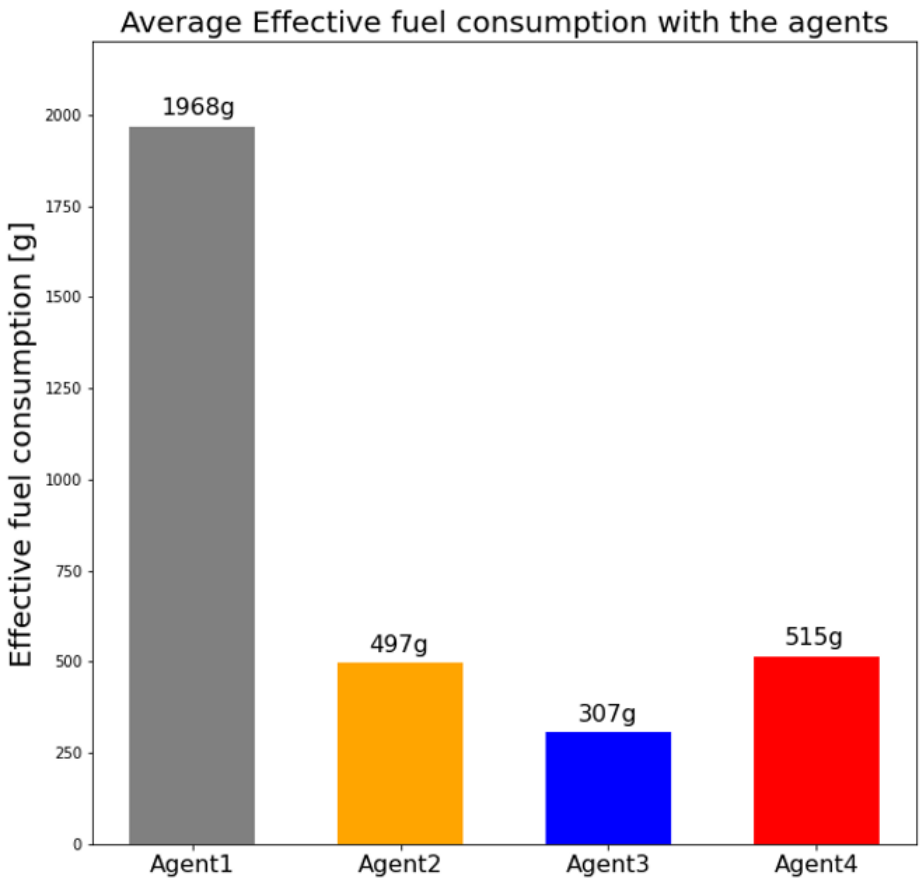


Figure 62. Comparison of the effective fuel consumption for the four agents on the validation cycle

3.6. Development of the Methodology for the Online-Learning on the DRL Model

In the previous study, the trained network was not updated according to the multiple driving cycles, and the fixed network was continuously used. However, the DRL algorithm has the advantage of optimizing the network and easily responding to system changes through the online-learning framework based on the recent experiences. In this study, we developed an online-learning methodology for the power distribution strategy of the FCHEV by taking advantage of the DRL algorithm, which is easy to apply online-learning.

Considering that the FCHEV's stack degradation occurs at the start of driving and has a great influence on the entire system, it is important to develop the online-learning methodology for the control strategies. In this study, we developed the online-learning methodology for the DRL model. And we compared the performance of the DRL agent that is applied the online-learning algorithm and the reference model that is not applied the online-learning algorithm in the FCHEV's stack degradation simulations.

This chapter consists of three parts. 3.6.1 describes the online-learning methodology for the DRL agent. And the process and results of the two FCHEV degradation simulations are described in 3.6.2 and 3.6.3. In 3.6.2, the DRL model to which the online-learning algorithm is applied and the DRL model to which the online-learning algorithm is not applied are compared under the FCHEV simulation with the fuel cell stack in which the fixed voltage-drop occurs. And in 3.6.3, the DRL model to which online-learning is applied and the DRL model to which online-learning is not applied are compared under the simulation conditions in which real-time deterioration occurs due to the stack operations.

3.6.1. Online-learning framework for the DRL model

We developed the online-learning methodology for the DRL agent considering the limited computation power of the FCHEV. In order to develop the online-learning algorithm that considers limited computational power, we do not train the agent while the FCHEV is driving, and when the driving is finished, we extract a small number of mini-batch from the replay memory so that the online-learning for the DRL agent. Figure [63] schematically shows the online-learning process of the DRL model in this study. The online-learning process developed in this study is divided into three parts: “Driving”, “Training” and “Validation”.

In the “Driving” phase in which the FCHEV is driven, the power distribution of the FCHEV is performed based on the policy of the existing trained DRL model, and experiences according to the driving are newly stored in the replay memory and at the same time, the old experiences disappear from the replay memory. And after the driving-phase is finished, the training-phase begins. Considering the low computation power for the vehicle, it is important to secure enough time for the training of the agent. Therefore, we defined the time when the driving-phase ends and the training-phase starts as the time when the vehicle is completely turned off and the vehicle is parked.

In the training-phase, the existing DRL model is trained by a small amount of the experiences from the replay memory in consideration of the limiting computation power of the vehicle. In this study, the existing agent is trained through 100 mini-batches consisting of 64 experiences.

Finally, in the validation phase, it is decided whether to replace the existing DRL agent with the newly trained DRL agent by comparing the performance of the newly trained DRL agent to the existing DRL agent. Since the DRL model constructed based on the deep neural network has a catastrophic forgetting problem in which previously trained information disappears during training, it is necessary to

check whether the training of the DRL model is conducted correctly in the online-learning process. We coped with the catastrophic forgetting problem of the DRL agent by confirming the effectiveness of the newly trained agent through the validation-phase. The agent derived through the online-learning algorithm is compared with the existing agent based on the driving cycle that the FCHEV drove in the driving-phase immediately before in the validation-phase. In the validation-phase, the effectiveness of the newly derived DRL model is judged in terms of the SOC-sustainability and the effective fuel consumption. If the DRL agent derived by the online-learning algorithm guarantees the SOC-sustainability on the driving cycle driven in the previous driving-phase and at the same time shows superior performance in terms of the effective fuel-consumption than the existing agent, the existing agent is replaced with the agent derived by online-learning method. On the other hand, if the DRL agent trained by online-learning algorithm does not guarantee SOC-sustainability or does not show superior performance in terms of the effective fuel-consumption than the existing agent, the energy management strategy based on the existing DRL agent will be maintained.

3.6.2. Comparative experiment 1: Static degradation simulation

In the first comparative experiment, the performance of the online-learning model and the DRL model without the online-learning were compared under a simulation condition where a certain amount of the voltage-drop occurred initially and the real-time voltage-drop according to the driving was not considered. In this study, the experiment was performed assuming that the voltage-drop of the cell occurs as much as 0.03V.

Figure [64] shows the process of the first comparative experiment. The reference model in Figure [64] represents the DRL model with the structure of the

agent 3 derived from the previous study. In addition, the DRL model applying online-learning algorithm also initializes the network weights with the weights of the reference model. The online-learning model and the reference model run the same driving cycle derived from the MDP-driver model, and the initial SOC of the next driving is not initialized to 0.6, but to the final SOC value of the previous driving. Through this simulation design, we tried to check whether the SOC-sustainability of the online-learning model and the reference model is guaranteed even in a situation similar to the actual driving.

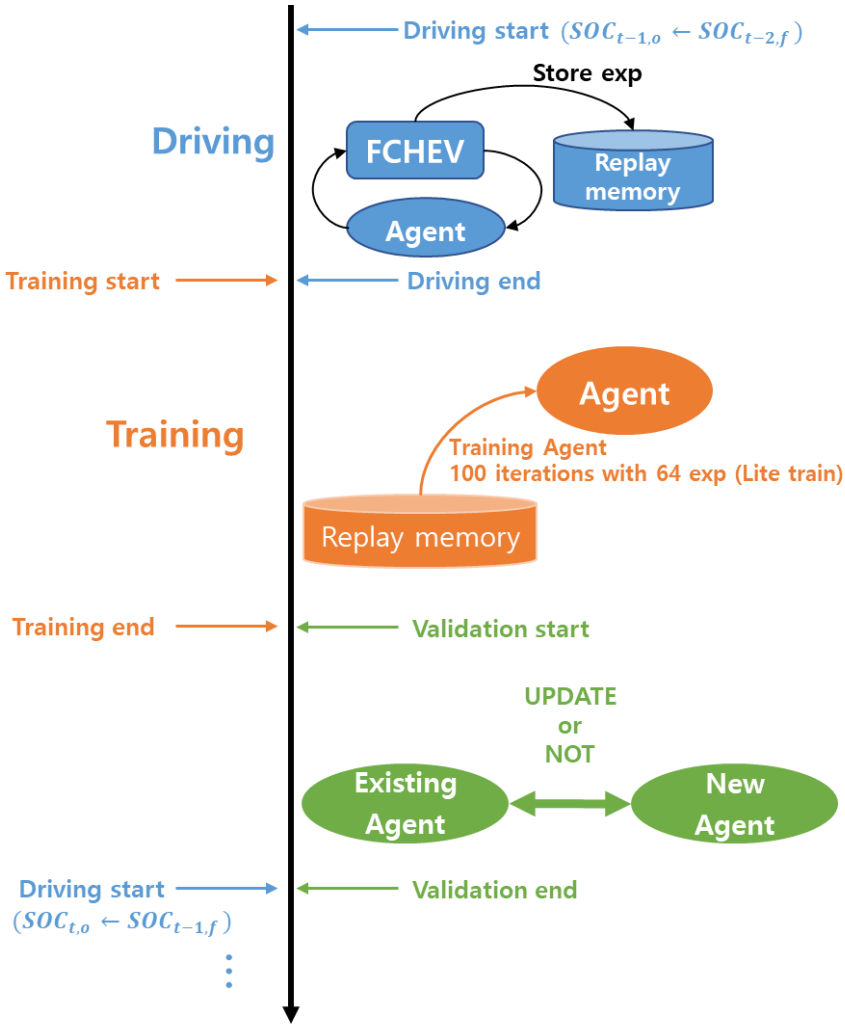


Figure 63. Schematic diagram for the online-learning process of the DRL model

The biggest difference between the online-learning model and the reference model is in the rest-phase that exists between the driving-phases. In the case of the online-learning model, training and validation process is progressed in the rest-phase based on the driving cycle derived in the previous driving-phase. On the other hand, in the case of the reference model, training is not performed in the rest-phase and the power distribution strategy based on the reference model is maintained.

Figure 64. Schematic diagram of the comparative experiment process between the online-learning model and the reference model

We compared the online-learning model and the reference model based on the driving-cycle created through the MDP-driver model. The experiment was conducted through a total of 100 generated driving cycles in this study. And the analysis of the comparative experiment between the two models was performed based on the effective fuel consumption, which includes information on the amount of fuel consumption and the voltage-drop due to deterioration, similar to the previous studies.

Figure [65] shows the final SOC distribution of the online-learning model and reference model on 100-driving cycles under the condition that the initial stack degradation of the FCHEV occurred as much as 0.03V. It can be seen that the difference between the reference SOC set to 0.6 and the final SOC is mostly distributed below 0.02, which shows that both models guarantee the SOC-sustainability.

And Figure [66] shows the difference in performance between the online-learning model and the reference model according to the 100-driving cycles. That is, Figure [66] shows the effective fuel consumption of both models according to the number of driving. Figure [66] shows that the online-learning model can perform more efficient control strategy than the reference model through the continuous learning. It can be seen that the online-learning model can reduce the effective fuel consumption by about 3950g compared to the reference model. From these facts, it can be confirmed that the online-learning methodology developed in this study can help the DRL agent optimize performance and adapt to the system changes.

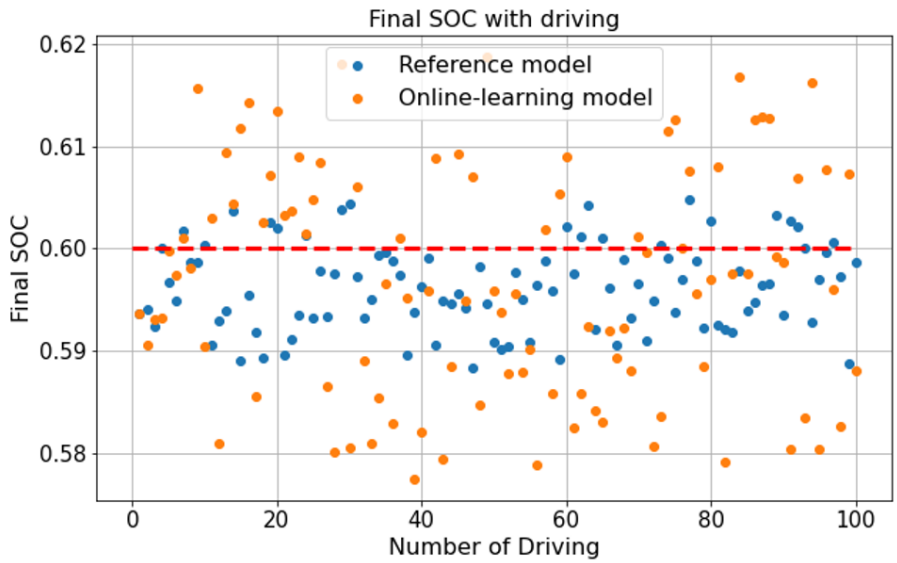


Figure 65. Final SOC distribution in 100-drivings of both models

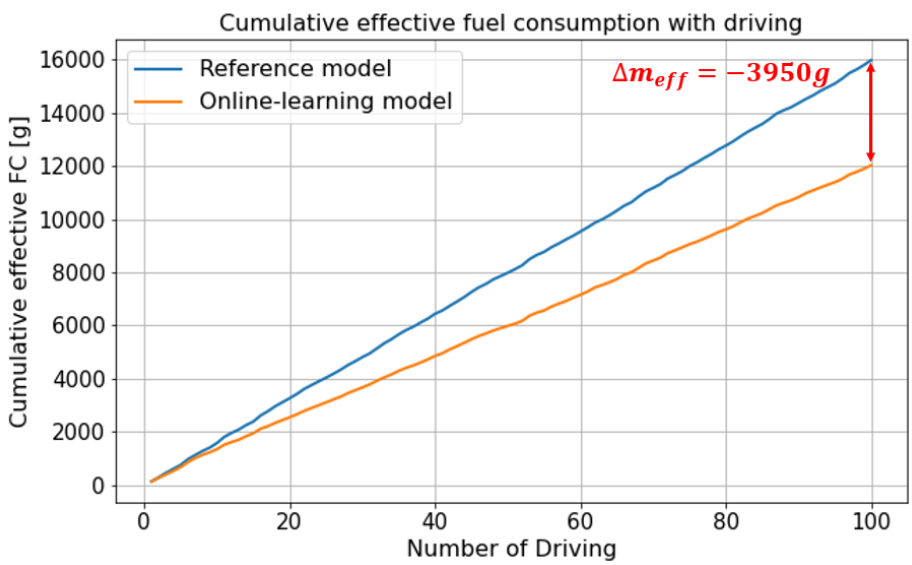


Figure 66. Cumulative effective fuel consumption in 100-drvings of both models

$$m_{eff} = \underbrace{m_{fc@SOC_{ref}}}_{\text{Fuel Consumption}} + \underbrace{\sigma \Delta V + M \frac{NAj}{2FV_{cell}} \delta V}_{\text{Degradation}}$$

Figure 67. Components of the effective fuel consumption

The effective fuel consumption is divided into the fuel consumption related term and the voltage-drop related term as shown in Figure [67]. We compare the online-learning model and the reference model in terms of the fuel consumption and the voltage-drop. Figure [68] shows the results of comparing the online-learning model and the reference model in terms of fuel consumption, voltage-drop, and effective fuel consumption. The (a) of Figure [68] shows the difference between the two agents for the fuel consumption on the 100 driving cycles, and the difference between the two agents for the fuel consumption of the two agents is defined as Eq (51). In Eq (51), FC_{online} means the fuel consumption of the online-learning model, and $FC_{reference}$ means the fuel consumption of the reference model.

$$\Delta FC = FC_{online} - FC_{reference} \tag{51}$$

Therefore, the red dotted line in Figure [68] means the decision boundary. If points are distributed below the red dotted line, it means that the performance of the online-learning model is superior to that of the reference model. If the points are distributed above the red dotted line, it means that the performance of the reference model is better than that of the online-learning model. Figure [68]-(a) shows that points are evenly distributed up and down around the decision boundary, which means that there is not much difference in the performance between the two models in terms of the fuel consumption. However, it can be seen that the reference model generally shows higher efficiency than the online learning model.

Figure [68]-(b) shows the difference in the voltage-drop due to the stack deterioration between the two models. The difference in the voltage-drop is expressed as Eq (52), where ΔV_{online} represents the voltage-drop of the online-learning model, and $\Delta V_{reference}$ represents the voltage-drop of the reference model. The points in Figure [68]-(b) are distributed below the decision boundary, which

means that the online-learning model shows superior performance regarding the voltage-drop caused by the stack degradation compared to the reference model.

$$\Delta(\Delta V) = \Delta V_{online} - \Delta V_{reference} \tag{52}$$

Figure [68]-(c) shows the difference between the two agents for the effective fuel consumption. In a similar manner to the previous case, the effective fuel consumption is expressed as Eq (53), where $FC_{eff,online}$ represents the effective fuel consumption for the online-learning agent, and $FC_{eff,reference}$ represents the effective fuel consumption for the reference agent.

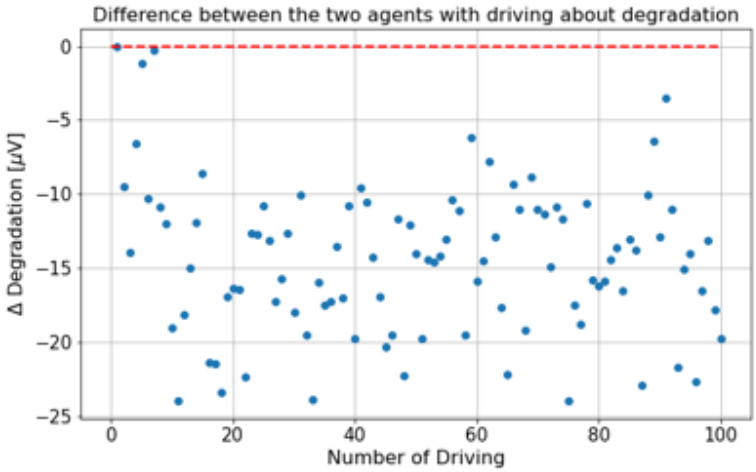
$$\Delta FC_{eff} = FC_{eff,online} - FC_{eff,reference} \tag{53}$$

Figure [68]-(c) shows that most of the points are distributed below based on the decision boundary. From this fact, it is confirmed that the performance of the existing agent can be improved through the online-learning algorithm.

Table 18 shows detailed information on the total amount of the fuel consumption, the total amount of the voltage-drop occurs during the simulation, and the total amount of the effective fuel consumption consumed by the two models on the 100-driving cycles. As shown in Figure [68], it can be seen that the online-learning algorithm is effective in reducing the voltage-drop caused by the stack degradation and the effective fuel consumption.

(a)

(b)



(c)

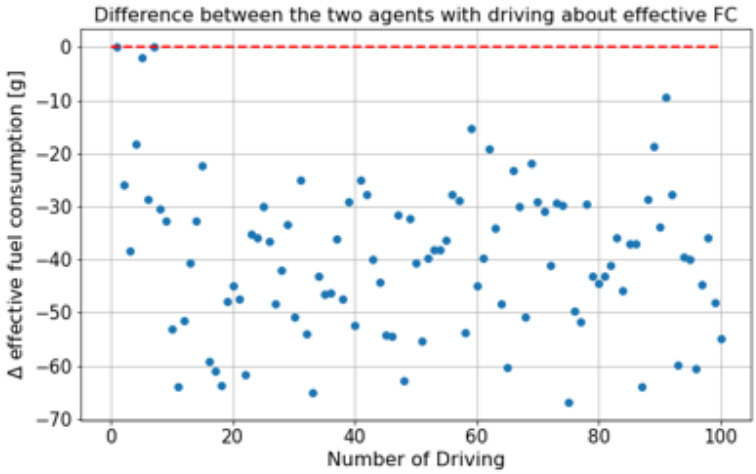


Figure 68. Difference between the online-learning model and the reference model according to driving: (a) Difference for the fuel consumption, (b) Difference for the voltage-drop, (c) Difference for the effective fuel consumption

Figure [69] shows the action profile of the online-learning model and the reference model for 5 driving cycles. Figure [69]-(a) shows five different generated driving profiles, and Figure [69]-(b) shows the action profile for the online-learning model and the action profile for the reference model on the five driving profiles. We can see from Figure [69] that the action profile of the online-learning model has a stronger tendency to maintain the previous action than the tendency of the reference model. In other words, the fact that the online-learning model shows better performance in terms of the degradation than the reference model can be seen as a difference that appears because the online-learning model has a stronger tendency to maintain the previous action.

Table XVIII. Simulation results for the two models on the 100-driving cycles

	Reference Agent	**Online-learning Model**	**Difference**
Total Fuel consumption [g]	7087	7175	**+1.2 %**
Total Voltage-drop [μV]	3115	1666	**-46.5 %**
Total Effective FC [g]	15980	12030	**-24.7 %**

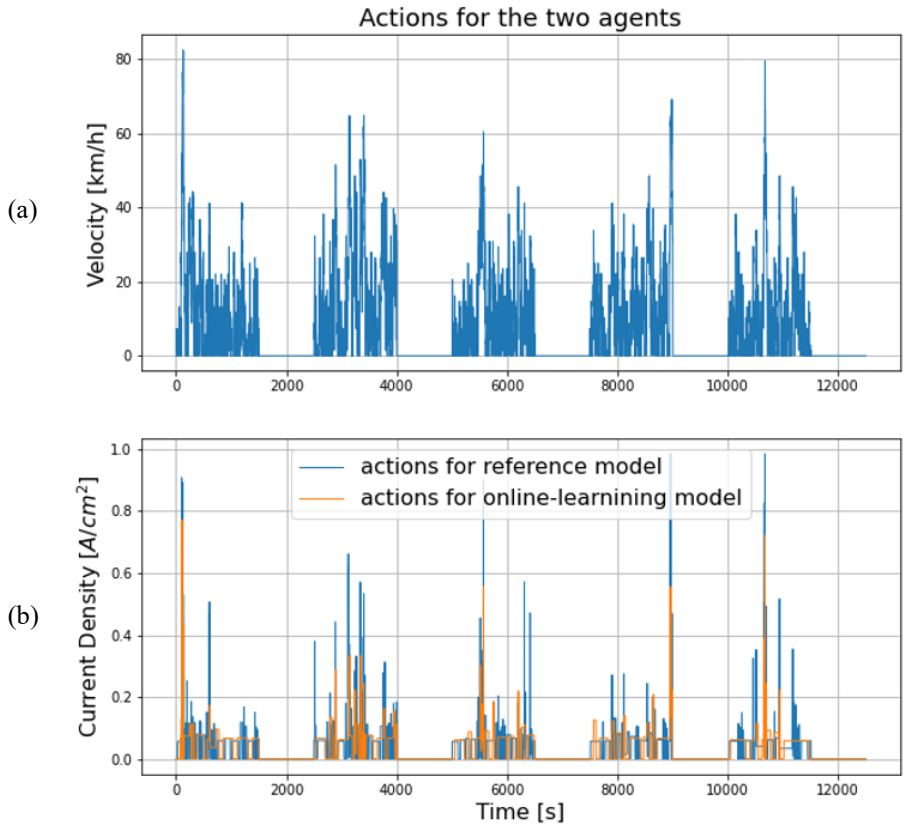


Figure 69. Action profiles of both models on 5-driving cycles: (a) driving cycles generated from the MDP-driver model, (b) action profiles of both models

3.6.3. Comparative experiment 2: Dynamic degradation simulation

In the second experiment, we conducted the comparative experiment between the online-learning model and the reference model on the simulation where the initial degradation and the real-time degradation from the stack operation occurs. We assumed that the initial voltage-drop due to the degradation of 0.015V occurred in this comparative experiment, and the experiment was designed to stop the simulation when the voltage-drop of 0.03V occurs due to the stack deterioration in any of the two models due to stack operation.

Figure [70] shows information on the cumulative voltage-drop according to the number of driving cycles of the online-learning model and the reference model. The fuel cell stack for the reference model deteriorates faster than the stack for the online-learning model, and as a result, the stack voltage-drop of 0.03V occurs in the reference model due to the driving for 468 driving-cycles. In this experiment, it is also confirmed that the online-learning model effectively reduces the voltage-drop compared to the reference model. Figure [70] shows that when the voltage-drop for the reference model occurs as much as 0.03V, the voltage-drop in the online-learning model occurs as much as about 0.023V.

Figure [71] shows the distribution of the final SOC of the two models for the driving about 500 driving-cycles. Figure [71] shows that in both models, the deviations between the initial SOC set to 0.6 and the final SOC are distributed within 0.03 in about 500-drivings. In other words, we can confirm that both the online-learning model and the reference model guarantee the SOC-sustainability from Figure [71].

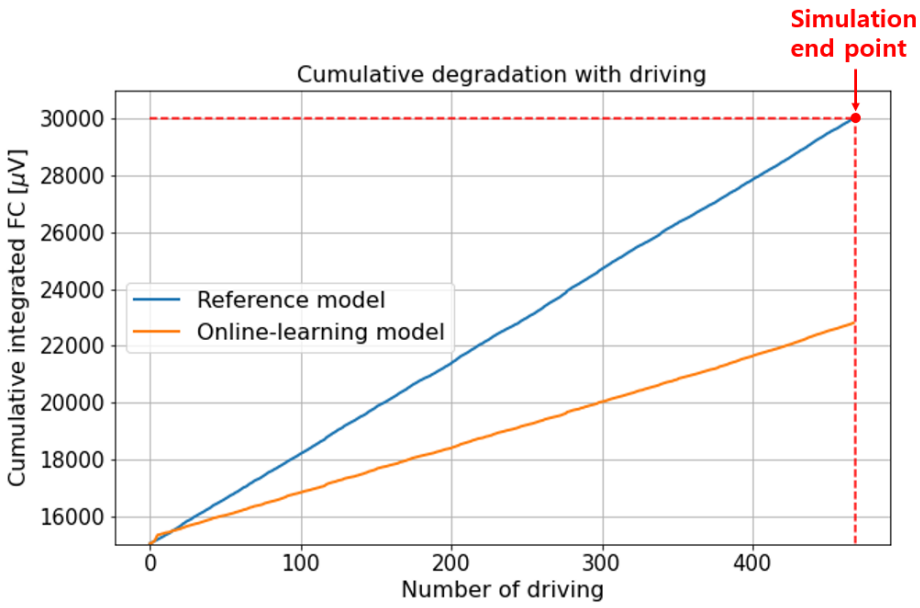


Figure 70. Cumulative voltage-drop for the two agents

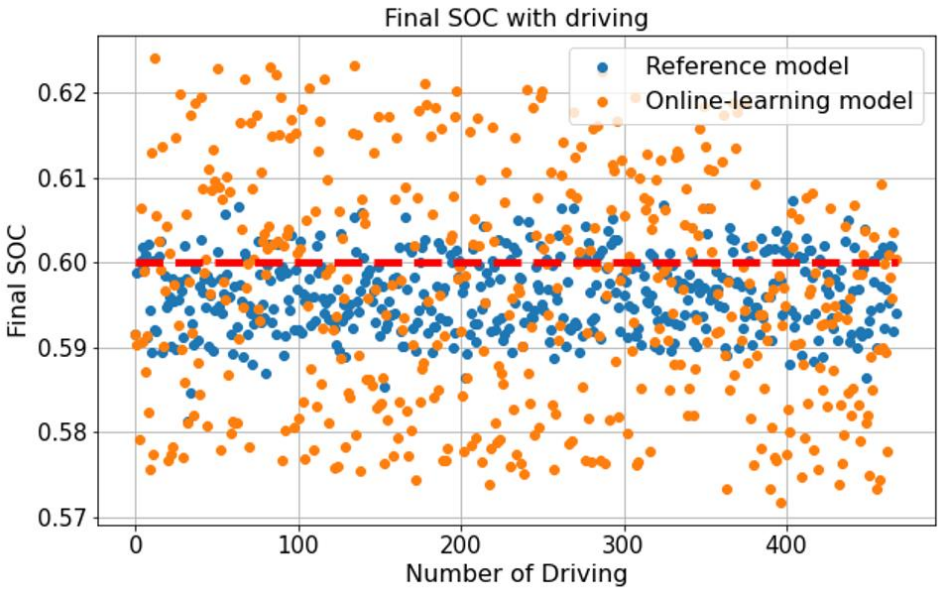


Figure 71. Final SOC distribution in 989-drivings of both models

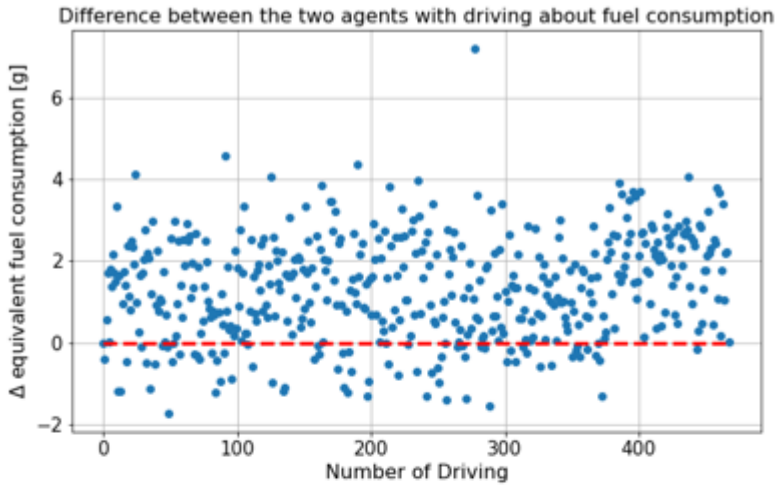
Figure [72] shows the difference in performance between the online-learning model and the reference model according to the driving-cycle. (a) shows the difference regarding the fuel consumption, (b) shows the difference in terms of the voltage-drop, and (c) shows the difference in terms of the effective fuel consumption. Like Figure [68], the red dotted line in Figure [72] means the decision boundary. If points are distributed under the decision boundary, it means that the performance of the online-learning model is superior to that of the reference model and if the points are distributed above the decision boundary, it means that the performance of the reference model is superior to that of the online-learning model. In terms of the fuel consumption, the reference model generally shows excellent performance, while the online-learning model shows generally excellent performance in terms of the voltage-drop and the effective fuel consumption. In the early driving cycles, the online-learning model shows poorer performance than the reference model in terms of the effective fuel consumption. It means that the DRL agent is trained in the direction of the poor performance through the online-learning algorithm. This phenomenon occurs because the result verified through one driving cycle in the

validation-phase cannot be said to be valid for all other driving. However, if too many driving cycles are used in the validation-phase, it may cause the computational problems and memory problems in the validation-phase. Therefore, in the validation-phase, the principle of judging the training effectiveness of the DRL agent based on the driving cycle which is recorded through the previous driving-phase was maintained.

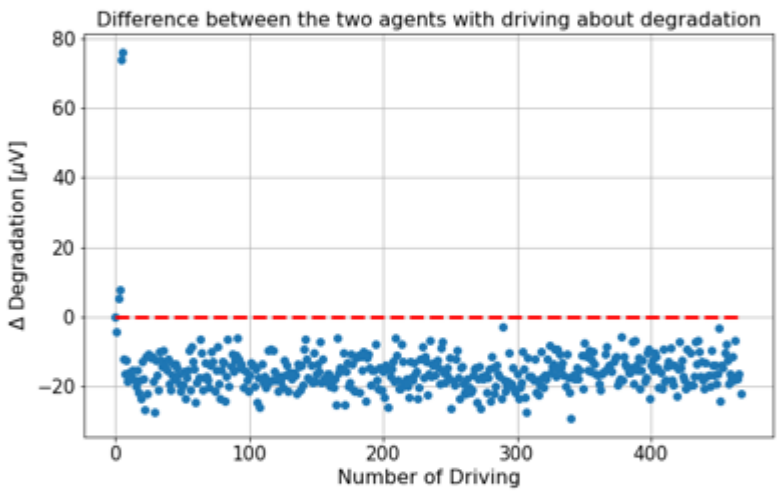
Although there is a section where the online-learning model is less efficient than the reference model, the online-learning model generally shows superior performance compared to the reference model regarding the effective fuel consumption. Table 19 shows the fuel consumption, the cumulative voltage-drop occurs during the simulation, and the effective fuel consumption consumed in about 500-drivings of the online-learning model and the reference model. Although the online-learning model recorded about 2% more fuel consumption than the reference model, it reduced the voltage-drop as much as 48% compared to the reference model. As a result, the online-learning model shows a high efficiency of about 26% compared to the reference model in terms of the effective fuel consumption.

Figure [73] shows the trend of the accumulative effective fuel consumption of the online-learning model and the reference model for about 500 driving-cycles. Figure [73] shows that the power distribution strategy based on the DRL model for the actual FCHEV where real-time stack degradation occurs can improve the performance and respond effectively to the changes in the FCHEV system through the online-learning algorithm.

(a)



(b)



(c)

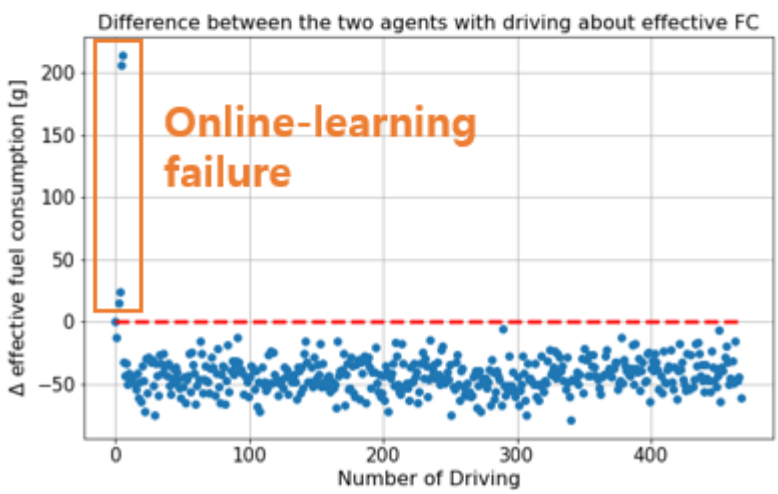


Figure 72. Difference between the online-learning model and the reference model according to driving: (a) Difference for the fuel consumption, (b) Difference for the voltage-drop, (c) Difference for the effective fuel consumption

Table XIX. Simulation results for the two models on the 486-driving cycles

	Reference Agent	Online-learning Model	Difference
Total Fuel consumption [g]	33440	34060	**+ 1.82 %**
Total Voltage-drop [μV]	15000	7800	**-48.0 %**
Total Effective FC [g]	75350	55820	**-25.9 %**

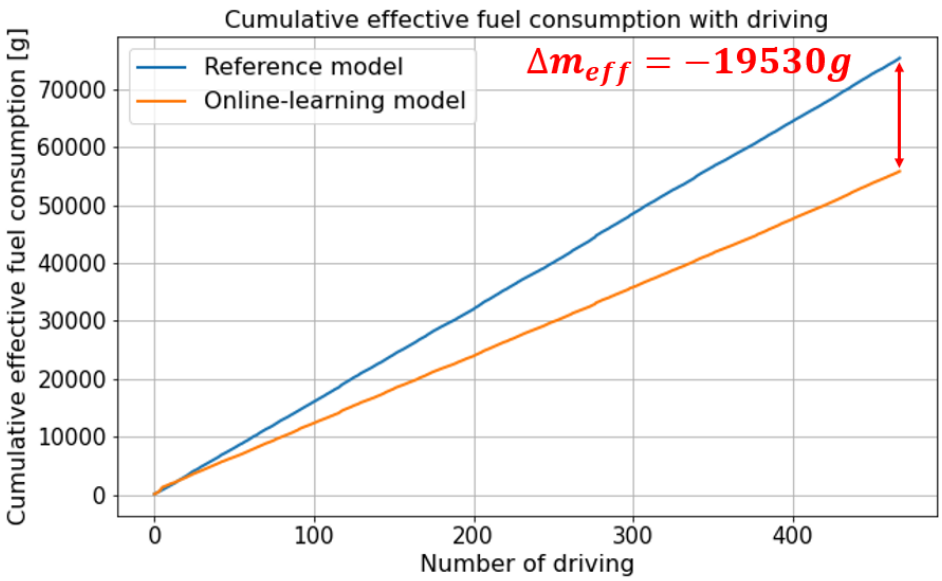


Figure 73. Cumulative effective fuel consumption with the driving for the two models

4. Conclusion and Achievement

We developed the DRL-based power distribution strategy for the FCHEV through this study. In developing the DRL-based power distribution strategy, we considered four aspects: “generalization”, “stack degradation”, “scalability” and “online-learning application”.

One of the most important factors in developing the energy management strategy is that the developed energy management strategy ensures the generalization performance. In other words, any power distribution strategy must be effective in all driving conditions, not only in specific driving conditions. The DRL-based power distribution strategy has a great advantage in ensuring the generalization performance in that the energy management control is made based only on the current driving information without future driving information. In this study, the MDP-driver model that can generate countless validation cycles was developed. And it was confirmed that the DRL-based control strategy can achieve high performance while guaranteeing generalization performance through comparison of the optimal control theory-based control strategy and the DRL-based control strategy.

Studies to develop control strategies in terms of the scalability are hardly in progress in the field of research related to the energy management of the HEV. However, considering the fact that many types of structures and systems for the HEV are being developed continuously, the development of the power distribution strategy that can guarantee scalability is very important from an industrial perspective. We developed the methodology related to the state normalization and the reward factor selection for the development of the control strategy for the FCHEV that can guarantee the scalability. It was confirmed that the DRL model-based power distribution strategy could be developed on the same training framework in the problem of considering only the fuel consumption, even if it is extended to the problem that requires additional consideration of the stack degradation.

One of the biggest issues of the FCHEV is the durability of the fuel cell stack. In this study, a study was conducted by developing a deterioration model of the fuel cell stack with reference to previous studies. In this study, the fuel cell stack degradation model was developed with reference to the previous studies in order to develop the control strategy considering the stack degradation. In addition, the equivalent factor that equalizes the voltage-drop due to the degradation to the fuel consumption rate was derived based on the economic analysis, and the reward factor was reconstructed based on the concept called effective fuel consumption. Since the load change operation has a great influence on the degradation of the fuel cell stack, we have effectively coped with the deterioration due to the load change operation by changing the action and the state configuration for the DRL model.

DRL is basically composed so that the agent is trained through own experiences, so it is easy to optimize the existing model by applying the online-learning concept. Also, since the FCHEV is sensitive to the stack degradation, it is necessary to cope with the changed system through the development of the online-learning methodology. In this study, the online-learning algorithm was developed in consideration of the limited computing power and memory of the FCHEV. We conducted the experiments comparing the online-learning model and the existing reference model under the two stack degradation conditions of the FCHEV. We confirmed that the developed online-learning algorithm can help the existing DRL model improve and adapt to system changes through the two experiments.

The DRL algorithm can play a big role in developing the power distribution strategy for the HEV that ensures the generalization performance and the scalability. In particular, power system for the FCHEV changes over time due to the stack degradation that occurs in real time. Therefore, the DRL algorithm that is easy to apply the online-learning concepts is effective in developing the power distribution strategy for the FCHEV. However, few studies to develop the energy management

strategy for the FCHEV using the DRL algorithm have been carried out yet. We hope that this study will be helpful in research related to the development of the DRL-based power distribution strategy for the FCHEV in the future.

5. Future works

So far, we have conducted research on the assumption that there is no disturbance in the state that the agent receives. However, the state that the agent is entered may be different from the actual vehicle behavior since there are many disturbing factors in the actual vehicle. We applied the disturbance to the required power, one component of the state, to find out the effect of the noise on the trained agent. We have assumed that the disturbance is applied only when the demanding power is positive. Figure [74] shows the required power in the ideal environment without the disturbance and the required power with the disturbance on an arbitrary driving cycle.

Required power with the noise is expressed as Eq (54), where P_{req} represents the ideal required power, and P_{noise} represents the noise for the required power.

$$P_{req,noise} = P_{req} + P_{noise} \tag{54}$$

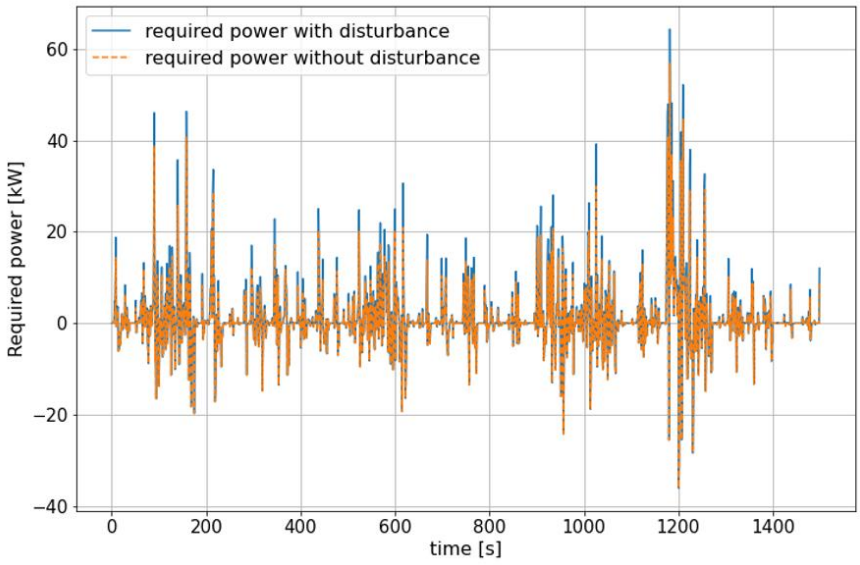


Figure 74. Comparison between the required power with the disturbance and the required power without the disturbance

And the value for the noise is expressed as Equation (55), where τ is a real number between the lower limit, α and the upper limit, β, which is extracted by uniform distribution. We fixed the lower limit of τ at 10% and adjusted the size of the disturbance by changing the upper limit of τ to 20%, 30%, 40%, and 50%.

$$P_{noise} = \tau \cdot P_{req}$$
$$\because \tau \sim U(\alpha, \beta) \quad where \ \alpha < \tau < \beta \tag{55}$$

We compared the effective fuel consumption of the trained agent by varying the upper limit of the disturbance on the multiple driving cycles. The comparison experiment was conducted according to the magnitude of the noise based on the 35 generated cycles. Figure [75] shows the trend of the effective fuel consumption according to the magnitude of the disturbance.

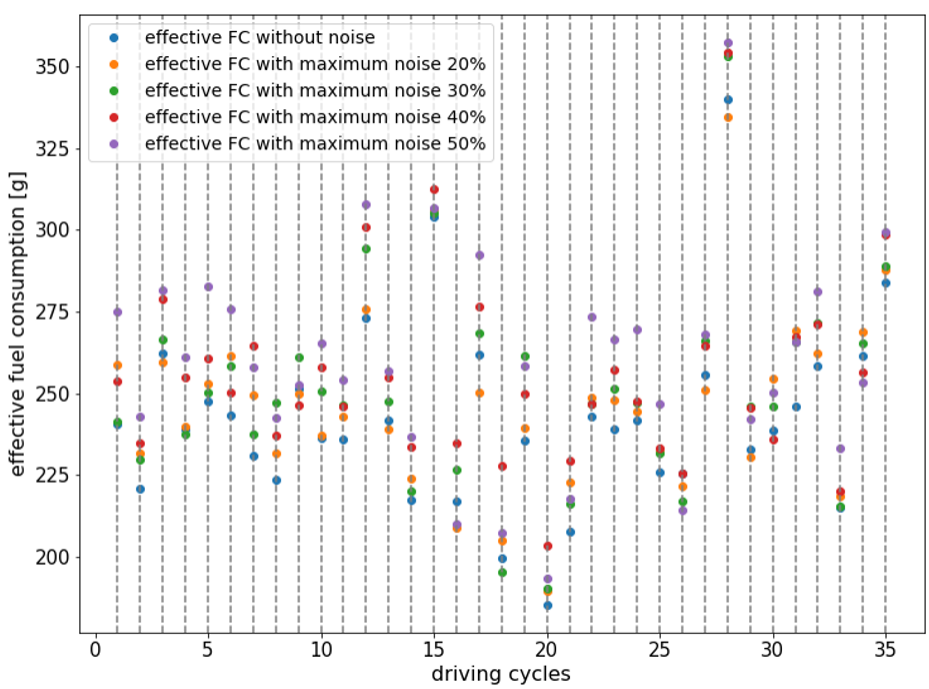


Figure 75. The effective fuel consumption according to the amplitude of the noise

As expected, it can be seen from Figure [75] that the effective fuel consumption tends to increase as the amplitude of the disturbance increases. Table 20 shows the average effective fuel consumption according to the amplitude of the disturbance. From the Table 20, it can be clearly seen that as the upper limit of τ increases, the average effective fuel consumption increases. When the upper limit is 50%, the average effective fuel consumption increases by 6.8% compared to the case without the disturbance.

It can be seen from the experimental results that the trained agent controls the FCHEV properly even when a considerable amount of the disturbance occurs. However, considering that the performance of the agent is reduced by the noise and the fact that there are many disturbance factors in the actual vehicles, it is very important to develop a methodology that can cope with the disturbance. If a DRL agent that can cope with the disturbances is developed through additional research, we expect that the applicability of the DRL agent to actual vehicles will be greatly improved.

Table XX. comparison results with the disturbance for the demanding power

	Noise (τ) Interval	**Average effective fuel consumption [g]**	**Difference [%]**
Reference case	-	242.3	**-**
Disturbance case 1	[0.1, 0.2)	247.1	**+ 1.98**
Disturbance case 2	[0.1, 0.3)	250.4	**+ 3.34**
Disturbance case 3	[0.1, 0.4)	255.2	**+ 5.32**
Disturbance case 4	[0.1, 0.5)	260.0	**+ 6.80**

We conducted the study to normalize the state and to find the optimal reward factor by understanding the learning pattern of the DRL model in order to create the scalable DRL-based power distribution strategy.

However, the DRL model developed in this study cannot automatically derive hyper parameters and optimal network structure. Since the knowledge and the experience of the model developer is required to derive the hyper parameters or the optimal network structure, it is necessary to develop a methodology that can automatically find the hyper parameters and the network architecture. We think that autoML (machine learning) technology can be of great help in implementing automation for the hyper parameters and model structure retrieval. Therefore, we plan to develop a methodology for automatically deriving the hyper parameters and the network architecture through the autoML technology in the future research.

We developed the MDP-driver model and created the virtual driving cycles to confirm the generalization performance of the DRL model. Since the MDP-driver model is derived through the stochastic characteristics of the reference driving profiles representing the general driving situations, there is a limit to generating driving information about unexpected situations that occur in the actual driving. In a future study, the effectiveness of the DRL-based power distribution strategy derived in this study will be tested through actual driving data of the vehicle. In the future study, the effectiveness of the DRL-based power distribution strategy will be tested through the actual driving data of the FCHEV.

Bibliography

1. Retrieve from https://landtransportguru.net/european-emission-standards/

2. H. Son and H. Kim, “Development of near optimal rule-based control for plug-in hybrid electiric vehicles taking into account drivetrain component losses”, Energies, vol. 9, 2016

3. C. Geng, X. Jin, X. Zhang, “Simulation research on a novel control strategy for fuel cell extended-range vehicles”, Hydrogen Energy, vol. 44, pp. 408-420, 2019

4. C. Zheng, Y. Park, W. S. Lim, and S. W. Cha, “Comparison of rule-based power management strategy and optimal control strategy in fuel cell hybrid vehicles”, Trans. Korean Society of Automotive Eng., vol. 20, pp. 103-108, 2012

5. R. Zhang, J. Tao, “GA-based fuzzy energy management system for FC/SC-powered HEV considering H_2 consumption and load variation”, IEEE Trans. Fuzzy Syst., vol. 26, pp. 1833-1843, 2018

6. H. Lee, C. Kang, Y. Park, and S. W. Cha, “Study on power management strategy of HEV using dynamic programming”, World Elec. Veh. J., vol. 8, pp. 274-280, 2016

7. W. Zhou, L. Yang, Y. Cai, and T. Ying, “Dynamic programming for new energy vehicles based on their work modes Part II: Fuel cell electric vehicles”, J. Power Sources, vol. 407, pp. 92-104, 2018

8. L. Perez, G. Garcia, et al, “Optimization of power management in a hybrid electric vehicle using dynamic programming”, Mathematics and Computers in Simulation, vol. 73, pp. 244-254, 2006

9. R. Wang, S. Lukic, “Dynamic programming technique in hybrid electric vehicle optimization”, 2012 IEEE Int. Elec. Veh. Conf., 2012

10. A. Ali, A. Ghanbar, and D. Soffker, “Optimal control of multi-source electric vehicles in real time using advisory dynamic programming”, IEEE Trans. Veh. Tech., vol. 11, pp. 10394-10405, 2019

11. K. Ou, W. Yuan, Y. Kim, et al, “Optimized power management based on adaptive-PMP algorithm for a stationary PEM fuel cell/battery hybrid system”, Hydrogen Energy, vol.43, pp.15433-15444, 2018

12. H. Li, A. Ravey, A. Djerdir, et al, “Online adaptive equivalent consumption minimization strategy for fuel cell hybrid electric vehicle considering power sources degradation”, Energy Convs. Manag., vol. 192, pp. 133-149, 2019

13. N. Kim, S. Ha, J. Jeong, and S. W. Cha, “Sufficient conditions for optimal energy management strategies of fuel cell hybrid electric vehicles based on Pontryagin’s minimum principle”, Proc. Inst. Mech. Eng. Part D: J. Automobile Eng., vol. 230, pp. 202-214, 2015

14. J. Jeong, D. Lee, S. W. Cha, et al, “Development of PMP-based power management strategy for a parallel hybrid electric bus”, Int. J. Precision. Eng. Manuf., vol. 15, pp. 345-353, 2014

15. S. Zhang, R. Xiong, C. Zhang, “Pontryagin’s minimum principle-based power management of a dual-motor-driven electric bus”, Appl. Energy, vol. 159, pp. 370-380, 2015

16. S. Onori and L. Tribioli, “Adaptive Pontryagin’s minimum principle supervisory controller design for the plug-in hybrid GM Chevrolet Volt”, Appl. Energy, vol. 147, pp. 224-234, 2015

17. N. W. Kim, D. H. Lee, S. W. Cha, et al, “Realization of PMP-based control for hybrid electric vehicles in a backward-looking simulation”, Int. J. of Automotive Technology, vol. 15, pp. 625-635, 2014

18. R. Xiong, J. Cao, and Q. Yu, “Reinforcement learning-based real-time power management for hybrid energy storage system in the plug-in hybrid electric vehicle”, Appl. Energy, vol. 211, pp. 538-548, 2018

19. H. Lee, C. Song, N. Kim, and S. W. Cha, “Comparative analysis of energy management strategies for HEV: dynamic programming and reinforcement learning”, IEEE Access: Artificial Intelligence Tech. Elec. Power Syst., vol. 8, pp. 67112-67123, 2020

20. X. Lin, Y. Wang, M. Pedram, et al, “Reinforcement learning based power management for hybrid electric vehicles”, 2014 IEEE/ACM Int. Conf. ICCAD, 2014

21. T. Liu, X. Hu, D. Cao, et al, “Reinforcement learning optimized look-ahead energy management of a parallel hybrid electric vehicle”, IEEE/ASME Trans. Mechatronics, vol. 22, pp. 1497-1507, 2017

22. Y. Zou, T. Liu, D. Liu, and F. Sun, “Reinforcement learning-based real-time energy management for a hybrid tracked vehicle”, Appl. Energy, vol. 171, pp. 372-382, 2016

23. T. Liu, B. Wang, C. Yang, “Online Markov Chain-based energy management for a hybrid tracked vehicle with speedy Q-learning”, Energy, vol. 160, pp. 544-555, 2018

24. H. Lee, C. Kang, S. W. Cha, et al, “Online data-driven energy management of a hybrid electric vehicle using model-based Q-learning”, IEEE Access: Artificial Intelligence Tech. Elec. Power Syst., vol. 8, pp. 84444-84454, 2020

25. V. Mnih, K. Kavukcuoglu, D. Hassabis, et al, “Human-level control through deep reinforcement learning”, Nature, vol. 518, pp. 529-533, 2015

26. O. Vinyals, I. Babuschkin, D. Silver, et al, “Grandmaster level in StarCraft II using multi-agent reinforcement learning”, nature, vol. 575, pp. 350-354, 2019

27. T. Lillicrap, J. Hunt, D. Silver, et al, “Continuous control with deep reinforcement learning”, arXiv: 1509.02971, 2015

28. P. Wang, Y. Li, F. Northrop, et al, “A deep reinforcement learning framework for energy management of extended range electric delivery vehicles”, 2019 IEEE Intelligent Veh. Symposium, 2019

29. P. Zhao, Y. Wang, X. Lin, et al, “A deep reinforcement learning framework for optimizing fuel economy of hybrid electric vehicles”, 2018 Asia and Pacific Design Automotive Conf., 2018

30. J. Wu, H. He, Z. Li, et al, “Continuous reinforcement learning of energy management with deep Q network for a power split hybrid electric bus”, Appl. Energy, vol. 222, pp. 799-811, 2018

31. R. Liessner, C. Schroer, B. Baker, et al, “Deep reinforcement learning for advanced energy management of hybrid electric vehicles” Conf. Agent and Artificial Intelligence, 2018

32. Y. Wu, H. Tan, H. He, et. al, “Deep reinforcement learning of energy management with continuous control strategy and traffic information for a series-parallel plug-in hybrid electric bus”, Appl. Energy, vol. 247, pp. 454-466, 2019

33. X. Han, H. He, Y. Li, et al, “Energy management based on reinforcement learning with double deep Q-learning for a hybrid electric tracked vehicle”, Appl. Energy, vol. 254, 2019

34. H. Tan, H. Zhang, Y. Wu, et al, “Energy management of hybrid electric bus based on deep reinforcement learning in continuous state and action space”, Energy Convers. Manag., vol. 195, pp. 548-560, 2019

35. Z. Zhu, Y. Liu, and M. Canova, “Energy management of hybrid electric vehicles via deep Q-networks” 2020 American Control Conf., 2020

36. Y. Hu, W. Li, C. Li, et al, “Energy management strategy for a hybrid electric vehicle based on deep reinforcement learning”, Appl. Sci., vol 8, 2018

37. M.Jouin, R.Gouriveau, D.Hiseel, et al., Prognostics and health management of PEMFC state of the art and remaining challenges, Hydrogen Energy, vol. 38, pp. 15307-15317, 2013

38. M. Mayur, M. Gerard, P. Schott, and W. Bassler, “Lifetime prediction of a polymer electrolyte membrane fuel cell under automotive load cycling using a physically-based catalyst degradation model”, Energies, vol. 11, 2018

39. H. Chen, P. Pei, and M. Song, “Lifetime prediction and the economic lifetime of proton exchange membrane fuel cells”, Appl. Energy, vol. 142, pp. 154-163, 2015

40. X.Zhang, D. Yang, M.Luo, Load profile based empirical model for the lifetime prediction of an automotive PEM fuel cell, Hydrogen Energy 2017, 42, 11868-11878

41. C. Liu and L. Liu, “Optimal power source sizing of fuel cell hybrid vehicles based on Pontryagin's minimum principle”, Hydrogen Energy, vol. 40, pp. 8454-8464, 2015

42 C. H. Zheng, G. Q. Xu, S. W. Cha, et al, “Prolonging fuel cell stack lifetime based on Pontryagin’s minimum principle in fuel cell hybrid vehicles and its economic influence evaluation”, J. Power Sources, vol. 248, pp. 533-544, 2014

43. Z. Hu, J. Li, G. Kou, et al, “Multi-objective energy management optimization and parameter sizing for proton exchange membrane hybrid fuel cell vehicles”, Energy Convers. Manag., vol.129, pp. 108-121, 2016

44. D. Fares, R. Chedid, R. jabr, et al, “Dynamic programming technique for optimizing fuel cell hybrid vehicles”, Hydrogen Energy, vol. 40, pp. 7777-7790, 2015

45. L. Xu, M. Ouyang, J. Hua, et al, “Application of Pontryagin's minimal principle to the energy management strategy of plugin fuel cell electric vehicles”, Hydrogen Energy, vol. 38, pp. 10104-10115, 2013

46. R. O’Hayre, S. W. Cha, W. Colella, F. B. Prinz, “Fuel Cell Fundamentals (3rd edition)”, Wiley, 2016

47. S. Ioffe, C. Szegedy, “Batch normalization accelerating deep network training by reducing internal covariate shift”, arXiv: 1502.03167, 2015

48. A. Wilson, G. Kleen, and D.Papageorgopoulos, Fuel cell system cost – 2017, DOE Hydrogen and Fuel Cells Program Record, 2017

49. Y. Park, Retrieved from https://zdnet.co.kr/view/?no=20190428134647

50. R. Bird, W. Stewart, and E. Lightfoot, “Transport Phenomena (2nd edtion)”, John Wiley and Sons, 2002

Appendix 1-1

Proton Exchange Membrane Fuel Cell (PEMFC) Modeling

In this study, the PEMFC model was developed by referring to flux balance fuel cell modeling from S. W. Cha, F. B. Prinz et al. [46]. Figure [A-1] shows the 1-D PEMFC model, and Table A-1 shows the notation of variables and values of the variables related to the PEMFC modeling. The voltage of the fuel cell is affected by three losses like Eq (A-1), where η_{act}, η_{ohmic}, and η_{conc} means activation loss, ohmic loss, and concentration loss respectively.

$(\cdot)|_a, (\cdot)|_b, (\cdot)|_c$, and $(\cdot)|_d$ represent the parameters in the fuel cell interface. "a", "b", "c", and "d" represent anode-inlet interface, anode-membrane interface, cathode-membrane interface and cathode-inlet interface respectively.

$$V = E_{thermo} - \eta_{act} - \eta_{ohmic} - \eta_{conc} \qquad \text{A-1}$$

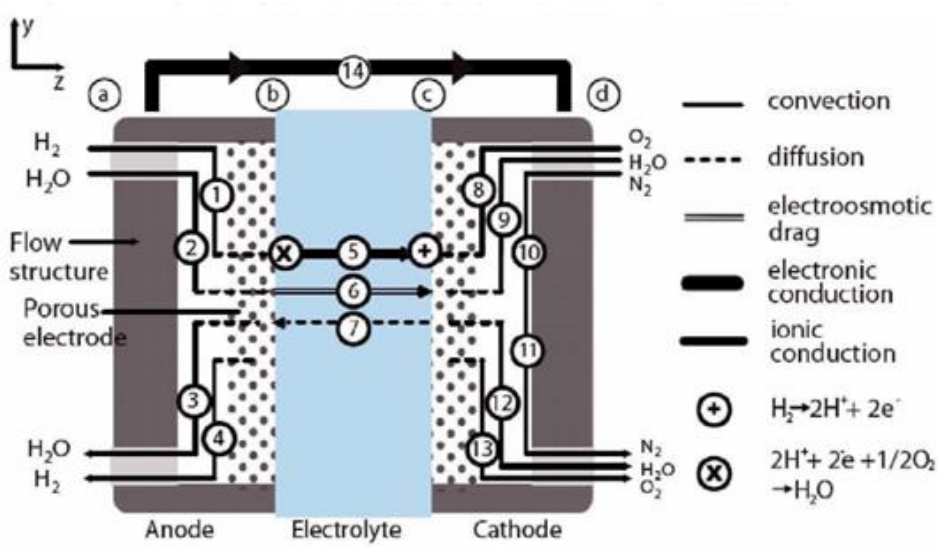


Figure A- 1. Schematic for the 1-D PEMFC model [46]

Table A- I. Values and Notation for properties in PEMFC model

Physical Properties	Notation	Values
Temperature (k)	T	343
Hydrogen mole fraction at anode	$x_{H_2}\|_a$	0.9
Oxygen mole fraction at cathode	$x_{O_2}\|_d$	0.19
Water mole fraction	x_{H_2O}	0.1
Cathode pressure (atm)	P^C	3
Anode pressure (atm)	P^A	3
Water diffusivity in Nafion (cm^2/s)	D_λ	3.18×10^{-6}
Transfer coefficient	α	0.5
Exchange current density (A/cm^2)	j_0	0.0001
Electrolyte thickness (μm)	t^M	125
Anode thickness (μm)	t^A	350
Cathode thickness (μm)	t^C	350
Electro-osmotic drag coefficient	n_{drag}^{SAT}	2.5
Nafion equivalent weight (kg/mol)	M_m	1.0
Limiting current density (A/cm^2)	$j_{L,cathode}$	3.0

Therefore, it is necessary to develop a mathematical model that derives the ohmic loss, the activation loss, and the concentration loss for the PEMFC modeling. The ohmic loss is the most difficult to derive among the three losses. From the flux balance relation, the mole fraction for H_2O on the anode side can be derived as Eq (A-2), and the mole fraction for H_2O in the anode-membrane interface is expressed as Eq (A-3). R is the gas constant, F is the Faraday number, and D_{H_2,H_2O}^{eff} is the effective diffusivity between H_2 and H_2O.

$$x_{H_2O}(z) = x_{H_2O}|_a - z\frac{\alpha^* jRT}{2FP^A D^{eff}_{H_2,H_2O}} \quad \text{A-2}$$

$$x_{H_2O}|_b = x_{H_2O}|_a - t^A\frac{\alpha^* jRT}{2FP^A D^{eff}_{H_2,H_2O}} \quad \text{A-3}$$

The diffusivity between two substances, i and j, is expressed as Eq (A-4), T_c, P_c, and M denote critical temperature, critical pressure, and molecular weight respectively. $(\cdot)_i$ and $(\cdot)_j$ mean parameters for substance i and j. In addition, a was set to 3.64×10^{-4} and b was set to 2.334 [50].

$$P \cdot D_{i,j} = a\left(\frac{T}{\sqrt{T_{ci}T_{cj}}}\right)^b (P_{ci}P_{cj})^{\frac{1}{3}}(T_{ci}T_{cj})^{\frac{5}{12}}\left(\frac{1}{M_i}+\frac{1}{M_j}\right)^{\frac{1}{2}} \quad \text{A-4}$$

Effective diffusivity is expressed as Eq (A-5), where ε means porosity, and the porosity of the fuel cell electrode has a value of around 0.4.

$$D^{eff}_{i,j} = \varepsilon^{1.5} D_{i,j} \quad \text{A-5}$$

The mole fraction for H_2O at the cathode side is equal to Eq (A-6), and the mole fraction for H_2O at the cathode-membrane interface is equal to Eq (A-7).

$$x_{H_2O}(z) = x_{H_2O}|_d + z\frac{(1+\alpha^*)jRT}{2FP^c D^{eff}_{O_2,H_2O}} \quad \text{A-6}$$

$$x_{H_2O}|_c = x_{H_2O}|_d + t^c\frac{(1+\alpha^*)jRT}{2FP^C D^{eff}_{O_2,H_2O}} \quad \text{A-7}$$

And Nafion's water content, λ, is expressed as Eq (A-8) through the analytic solution.

$$\lambda(z) = \frac{11\alpha^*}{n_{drag}^{SAT}} + C \cdot exp(\frac{jM_m n_{dry}^{SAT}}{22F\rho_{dry}D_\lambda} z) \qquad \text{A-8}$$

Therefore, the water content on the anode-membrane interface and the cathode-membrane interface are expressed as Eq (A-9) and Eq (A-10).

$$\lambda|_b = \lambda(0) = \frac{11\alpha^*}{n_{drag}^{SAT}} + C \qquad \text{A-9}$$

$$\lambda|_c = \lambda(t^M) = \frac{11\alpha^*}{n_{drag}^{SAT}} + C \cdot exp(\frac{jM_m n_{dry}^{SAT}}{22F\rho_{dry}D_\lambda} t^M) \qquad \text{A-10}$$

Water content on the Nafion can also be expressed as Eq (A-11) through experimental data, and a_w represents water activity. The water activity is expressed as Eq (A-12) and the water activity is a function of the partial pressure of the water vapor, P_w and the vapor saturation pressure, P_{SAT} expressed as Eq (A-13).

$$\lambda = \begin{cases} 14a_w & for \quad 0 < a_w \leq 1 \\ 10 + 4a_w & for \quad 1 < a_w \leq 3 \end{cases} \qquad \text{A-11}$$

$$a_w = \frac{P_w}{P_{SAT}} \qquad \text{A-12}$$

$$\log_{10} P_{SAT} = -2.18 + 0.03T - 9.18 \times 10^{-5}T^2 + 1.45 \times 10^{-7}T^3 \qquad \text{A-13}$$

Therefore, the water content on the anode-membrane interface with low water activity and the cathode-membrane interface with high water activity are expressed as Eq (A-14) and Eq (A-15).

$$\lambda|_b = 14a_w|_b = 14\frac{P^A}{P_{sat}}(x_{H_2O}|_a - t^A\frac{\alpha^* jRT}{2FP^C D^{eff}_{H_2,H_2O}}) \quad \text{A-14}$$

$$\lambda|_c = 10 + 4a_w|_c = 10 + 4\frac{P^C}{P_{sat}}(x_{H_2O}|_d + t^C\frac{(1+\alpha^*)jRT}{2FP^C D^{eff}_{O_2,H_2O}}) \quad \text{A-15}$$

Through the system of equations of Eq (A-9), Eq (A-10), Eq (A-14), and Eq (A-15), we can derive the unknown parameters α^* and C as Eq (A-16).

$$\begin{bmatrix} \alpha^* \\ C \end{bmatrix} = \boldsymbol{A^{-1}}b$$

$$\because A = \begin{bmatrix} \frac{11}{n^{SAT}_{drag}} + 14\frac{P^A}{P_{sat}}\frac{t^A jRT}{2FP^A D^{eff}_{H_2,H_2O}} & 1 \\ \frac{11}{n^{SAT}_{drag}} - 4\frac{P^C}{P_{sat}}\frac{t^C jRT}{2FP^C D^{eff}_{O_2,H_2O}} & \exp\left(\frac{jM_m n^{SAT}_{drag}}{22F\rho_{dry}D_\lambda}t^M\right) \end{bmatrix} \quad \text{A-16}$$

$$\because b = \begin{bmatrix} 14\frac{P^A}{P_{sat}}x_{H_2O}|_a \\ 10 + 4\frac{P^C}{P_{sat}}x_{H_2O}|_d + 4\frac{P^C}{P_{sat}}\frac{t^C jRT}{2FP^C D^{eff}_{O_2,H_2O}} \end{bmatrix}$$

The conductivity of the Nafion has a lot of correlation with the temperature and the water content, and the conductivity of the Nafion is mathematically expressed as Eq (A-17) through the experimental data.

$$\sigma(z) = \sigma_{303k}(\lambda)\, exp\left[1268\left(\frac{1}{303} - \frac{1}{T}\right)\right] \quad \text{A-17}$$

$$\because \sigma_{303k}(\lambda) = 0.005193\lambda - 0.00326$$

By substituting Eq (A-8) into Eq (A-17), the conductivity of the Nafion can be expressed as in Eq (A-18).

$$\sigma(z) = \left(0.005193\left(\frac{11\alpha}{n_{drag}^{SAT}} + Cexp\left(\frac{jM_m n_{drag}^{SAT}}{22F\rho_{dry}D_\lambda}z\right)\right) - 0.00326\right) \times exp\ (1268\left(\frac{1}{303} - \frac{1}{T}\right) \quad \text{A-18}$$

Therefore, the area specific resistance in the membrane and the ohmic loss for the fuel cell are derived through Eq (A-19) and Eq (A-20).

$$ASR_m = \int_0^{t^M} \frac{dz}{\sigma(z)} \quad \text{A-19}$$

$$\eta_{ohmic} = jASR_m \quad \text{A-20}$$

The second loss to be considered is the activation loss. The activation loss is represented as Eq (A-21) because most of the activation loss is caused on the cathode side, and P_0 is the reference pressure, which corresponds to 1atm.

$$\eta_{act,\,cathode} = \frac{RT}{4\alpha F} ln \frac{jP_0}{j_0 P^C x_{O_2}|_c} \quad \text{A-21}$$

Through the flux balance relation, the mole fraction for O_2 at the cathode side is equal to Eq (A-22), and the mole fraction for O_2 at the cathode-membrane interface is expressed as Eq (A-23).

$$x_{O_2}(z) = x_{O_2}|_d - z\frac{jRT}{4FP^c D^{eff}_{O_2,H_2O}} \quad \text{A-22}$$

$$x_{O_2}|_c = x_{O_2}|_d - t^c\frac{jRT}{4FP^c D^{eff}_{O_2,H_2O}} \quad \text{A-23}$$

That is, the activation loss is represented as Eq (A-24).

$$\eta_{activation} = \frac{RT}{4\alpha F}\ln\left[\frac{jP_o}{j_0 P^C\{x_{O_2}|_d - t^c jRT/4FP^C D^{eff}_{O_2,H_20}\}}\right] \quad \text{A-24}$$

The last loss to be calculated is the concentration loss. Concentration loss, like activation loss, is mostly caused by the cathode side. The concentration loss can be expressed as Eq (A-25).

$$\eta_{conc} = \frac{RT}{4F}\cdot ln(1 - \frac{j}{j_{L,cathode}}) \quad \text{A-25}$$

Abstract in Korean

연료전지 스택의 열화를 고려한 연료전지 하이브리드 차량 대상의 Actor-critic 알고리즘을 활용한 진보한 실시간 동력분배전략의 개발

갈수록 강화되는 차량의 배출가스 규제가 심화됨에 따라서 차량 제조사는 연비효율을 높이기 위한 대안으로 하이브리드 차량의 개발 및 생산에 대한 노력을 기울이고 있다. 하이브리드 차량은 두 가지 이상의 동력원을 지니는 차량으로 정의된다. 개별 동력원을 효율적인 작동점에서 운용할 수 있다는 장점을 통해서 하이브리드 차량은 일반 내연기관 차량에 비해서 높은 효율성을 보이게 된다. 하지만 하이브리드 차량의 높은 효율성은 오직 유효한 동력분배전략이 확보될 때에만 보장할 수 있다.

하이브리드 차량의 효율성에 미치는 동력분배전략의 중요성으로 인하여 그 동안 동력분배전략 개발과 관련하여 많은 연구가 진행되어 왔다. 많은 관련 연구들에서는 규칙기반제어, 최적제어이론, 강화학습 이론 등을 통해서 하이브리드 차량의 동력분배전략을 개발해오고 있다. 최적제어이론 기반의 동력분배전략은 높은 연비효율을 달성할 수 있다는 장점을 지니지만 미래의 주행정보를 고려해야 한다는 점에서 실차적용성과 일반화 성능이 낮다는 단점이 있으며 규칙기반제어과 강화학습 미래의 주행정보를 필요로 하지 않는다는 점에서 실차적용성과 일반화 성능이 높지만 연비 효율이 상대적으로 낮다는 단점이 있다.

현재 관련 연구에서는 각각의 동력분배전략이 지닌 단점을 보완하여 일반화 성능과 효율성 모두 우수한 동력분배전략을 개발하는 데 초점이 모아지고 있다.

대다수의 하이브리드 차량의 동력분배전략은 내연기관과 배터리 조합의 구조로 구성된 일반적인 하이브리드 차량을 대상으로 개발되어 왔으나 최근에는 연료전지 차량의 대중화와 맞물려 연료전지 차량을 대상으로한 동력분배전략 개발에 대한 연구가 증가하는 추세이다. 연료전지 차량은 보통 연료전지 스택과 배터리의 조합으로 동력원이 구성되며 일반 하이브리드 차량과 비교하여 전혀 배기가스를 배출하지 않고 파워트레인의 구조를 간소화할 수 있으며 높은 효율성을 달성할 수 있다는 장점을 가진다. 하지만 연료전지 차량의 연료전지 스택은 내구성 문제에 취약하다는 문제점 때문에 연료전지 스택의 열화를 고려하여 동력분배전략을 개발할 필요가 있다. 따라서 연료전지 하이브리드차량은 연료전지 스택의 열화와 연료소모율을 모두 고려하여 최적화 문제를 해결해야하는 다중 목적 문제 (multi-objective problem)에 속하기 때문에 연료소모율만을 고려하는 일반적인 하이브리드 차량에 대한 동력분배전략의 개발에 비해 문제의 복잡도가 높다. 그럼에도 불구하고 연료전지 하이브리드차량 대상의 동력분배전략의 개발은 이미 대중화가 이루어진 내연기관과 배터리의 조합으로 이루어진 일반 하이브리드 차량에 비해 많은 연구가 이루어지고 있지 않은 실정이다.

본 연구에서는 강화학습을 활용하여 연료전지차량 대상의 동력분배전략을 개발하였다. 강화학습은 최근에 심층인공신경망과의 융합을 통해서 큰 발전을 이루었다. 우리는 Actor-Critic 알고리즘 기반의 심층강화학습을 활용하여 연료전지차량의 열화와 연료소모율을 고려한

동력분배전략을 개발하였다. 심층강화학습은 현재의 state를 기반으로 제어전략을 도출하므로 일반화 성능이 높다는 장점이 있으며 확장성 측면에서도 매우 우수하다. 심층강화학습의 높은 확장성을 활용한다면 동일한 학습 프레임워크를 통해 하이브리드 차량의 시스템 변화나 최적화 문제의 복잡도의 변화에도 쉽게 대응이 가능하다. 그리고 심층강화학습은 온라인 학습을 통해서 실시간으로 발생하는 연료전지 스택의 열화에 대해 대응할 수 있다는 장점을 지닌다. 본 연구에서는 위와 같은 심층강화학습의 장점을 최대한 활용할 뿐 아니라 온라인 학습을 통해 연료전지 스택의 열화에 대응하면서 동시에 일반화 성능과 확장성을 확보한 연료전지 하이브리드차량 대상의 동력분배전략을 개발하였다.

감사의 글

설렘과 꿈을 가지고 재생에너지 변환 연구실에 입학한지 벌써 6년이라는 시간이 지났네요. 6년의 대학원 생활은 돌이켜 생각해보면 저에게 감사함을 품을 수밖에 없는 시간이었습니다. 이러한 짧은 글을 통해 그동안 많은 분들께 받은 호의와 도움을 표현할 순 없겠지만 최대한 제 감사의 마음이 잘 전달될 수 있도록 글을 써봅니다.

우선, 입학 당시에 프로그래밍도 모르는 저를 믿고 제자로써 받아주신 차석원 교수님께 무엇보다 깊은 감사의 마음을 전달 드립니다. 차석원 교수님의 폭 넓은 지식과 융합적 사고 덕분에 다양한 분야에 대한 경험과 지식을 얻을 수 있었습니다.

바쁜 와중에도 제 학위논문 심사위원을 맡아 주시고 좋은 조언을 해주신 윤병동 교수님과 송한호 교수님께 감사드립니다. 그리고, 제 학위논문을 지도해주시고 저의 심사를 위해 외부에서 찾아와 주신 김남욱 교수님과 조구영 교수님께 큰 감사를 드립니다.

6년의 대학원 생활동안 힘들고 어려운 시기도 있었지만, 재생에너지 변환 연구실의 동료 및 선후배가 곁에 있어서 그 어려운 시기를 헤치고 학업을 별 탈없이 마무리할 수 있었습니다. 저와 동기처럼 허물없이 지냈던, 원종이, 승완이, 창범이형, 본현이형, 그리고 예근이에게 감사한 마음을 드립니다. 또, 저에게 많은 가르침을 주셨던 호원이형과 기영이형, 종대형 그리고 희윤이형에게 감사함을 드립니다. 그리고, 김남욱 선배님, 이대흥 선배님, 정춘화 선배님께도 깊은 감사함을 전합니다. 선배님들의 노력과 시간 덕분에 저의 지금의 연구가 가능할 수 있었습니다. 팀은 다르지만 전공 외의 지식과 경험을 공유해주신 익황이형, 윤호형,

구영이형, 태현이형, 준호형, 상훈이형에게 감사함을 전달드립니다. 그리고 저의 업무를 많이 도와주었던 후배님들께도 감사함을 표합니다. 부족한 방장 때문에 때때로 혼란과 동요가 있긴 했었지만.. 묵묵히 맡은 바를 다해준 동환이, 상훈이, 상봉이, 유성이, 경현이, 원엽이, 인원이, 명석이, 성현이, 재원이에게 모두 감사함을 표현합니다. 그리고 앞으로 새로운 출발을 하는 양재와 현준이도 훌륭하신 교수님과 좋은 동료들과 함께 알찬 대학원 생활을 보내길 바랍니다.

항상 저를 믿어 주시고 묵묵이 응원해주시는 부모님과 저희 가족에게도 감사를 드립니다. 항상 제 멋대로인 아들을 30년이 넘게 지원해주시고 계시는 부모님께 표현은 하진 않지만 언제나 감사한 마음을 가지고 있습니다. 부모님의 투자가 헛되지 않도록 노력하고 있으니 믿어 주시기 바랍니다. 조만간 자랑스러운 아들이 될 거라고 생각합니다.

덤벙대는 성격이라 저에게 많은 도움과 호의를 주셨음에도 미처 적지 못한 많은 분들이 있을 것이라고 생각합니다. 누락되신 분이 있으시다면 양해 부탁드립니다.

저 혼자라면 이루지 못했을 일을 여러분들께서 도와주신 덕분에 지금의 제가 있을 수 있었습니다. 이제는 제가 받은 이러한 큰 도움과 호의를 되돌려줄 수 있는 존재가 되도록 언제나 끊임없이 발전하고 노력하겠습니다.

모든 분들께 감사합니다.

송창희 올림