# REINFORCEMENT LEARNING BASED ENERGY MANAGEMENT STRATEGY FOR FUEL CELL HYBRID VEHICLES

by
ZEKERIYA ENDER EĞER

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfilment of
the requirements for the degree of
Master of Science

Sabancı University
December 2022

**REINFORCEMENT LEARNING BASED ENERGY MANAGEMENT STRATEGY FOR FUEL CELL HYBRID VEHICLES**

Approved by:

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

Date of Approval: ..................................

# Abstract

REINFORCEMENT LEARNING BASED ENERGY MANAGEMENT STRATEGY FOR FUEL CELL HYBRID VEHICLES

Zekeriya Ender Eğer

Mechatronics Engineering, Master's Thesis, December 2022

Thesis Supervisor: Assist. Prof. Dr. Tuğçe Yüksel
Thesis Co-Supervisor: Prof. Dr. Serhat Yeşilyurt

Keywords: reinforcement learning, fuel cell electric vehicles, energy management, optimization, fuel cell, battery, DC-DC converter

There is an increasing concern on the usage of vehicles powered by internal combustion engines due to their high emission levels. The demand for cleaner energy technologies have led to research and development of electric and hybrid vehicles. Among these, fuel cell vehicles have started to draw attention due to the fact that it is clean, sustainable and it has high energy density. Thus, fuel cell hybrid vehicles have the potential to compete with vehicles powered by internal combustion engine in the future, yet there are challenges for fuel cell such as slow dynamics requiring that their operation together should be managed favorably.

The primary objective of the thesis is to address the problem of energy management in fuel cell vehicles. For this purpose, first a model of the powertrain is developed. Then, in order to achieve an efficient energy management, a model free reinforcement learning algorithm called deep deterministic policy gradient (DDPG) is employed. The energy management strategy focuses on running the fuel cell in its high efficiency range while limiting the deviation of state of charge of the lithium-ion battery from a target value. It is found that the DDPG agent trained simply with step power inputs can achieve up to 2.7% less energy consumption compared to commonly used rule-based energy management strategies while maintaining the state of the charge of the battery within a certain interval. Our results show that DDPG algorithm shows promising potential to be utilized in such applications.

# Özet

HİDROJEN YAKIT HÜCRELİ ARAÇLAR İÇİN PEKİŞTİRMELİ ÖĞRENMELİ ENERJİ KONTROL STRATEJİSİ

Zekeriya Ender Eğer

Mekatronik Mühendisliği ,Yüksek Lisans Tezi, Aralık 2022

Tez Danışmanı: Dr. Öğr. Üyesi Tuğçe Yüksel
Tez Eş Danışmanı: Prof. Dr. Serhat Yeşilyurt

Anahtar Kelimeler: Pekiştirmeli öğrenme, yakıt hücreli elektrikli araçlar, enerji kontrol stratejisi, optimizasyon, yakıt hücresi, batarya, DC-DC dönüştürücü

Son yıllarda içten yanmalı motorların emisyonları sebebiyle çevreye etkisi problem olmaya başlamıştır. Daha temiz enerji teknolojilerilerine olan talep elektrikli ve hibrit araçların araştırma ve geliştirme sürecinin başlamasına sebep olmuştur. Ancak bataryaların karakteristik özellikleri yüzünden elektrikli araçların menzil, şarj etme süresi ve maliyet gibi bazı dezavatajları bulunmaktadır. Buna alternatif olarak bataryalar başka enerji kaynaklarıyla birlikte çalışmaktadır. Diğer enerji kaynakları konusunda yakıt hücreleri temiz, sürdürülebilir ve yüksek enerji yoğunluğuna sahip olması sebebiyle dikkat çekmeye başlamıştır. Yakıt hücreli hibrit araçların, bu sebeplerle içten yanmalı motorla çalışan araçlara alternatif olma potansiyeli vardır, ancak yakıt hücrelerinin yavaş dinamiğe sahip olması gibi bazı dezavantajları olması nedeniyle, bu tarz araçlardaki enerji kaynaklarının birlikte çalışması önem arz etmektedir.

Bu tezin öncelikli amacı enerji kontrol stratejisi sorununa odaklanmaktır. Öncelikle aracın güç sistemi modellenmiş olup efektif bir enerji kontrol sistemi tasarlanmıştır. Öğrenme algoritmalarının farklı problemlere uygulanabilir hale gelmesinin avantajı kullanılarak kontrol sistemi modelden bağımsız çalışan derin deerministik ilke türevi (DDPG) algoritması ile eğitilmiştir. Enerji kontrol stratejisinin amacı yakıt hücresini en verimli bölgelerde çalıştırmak ve bunu yaparken bataryanın şarj seviyesindeki sapmaları en aza indirmektir. DDPG algoritmasıyla basit basamak güç girişleriyle eğitilen kontrol sisteminin performansı, farklı sürüş çevrimleri altında denenmiştir ve enerji tüketiminde % 2.7 ye kadar varan bir azalış gözlemlenmiştir. Bu sonuçlar DDPG algoritmasının bu tarz uygulamalar için potansiyeli olduğunu göstermektedir.

# Acknowledgements

I would like to express my deepest gratitude to my advisor Dr. Tuğçe Yüksel for her guidance, support and particularly patience during my master's years. It was a remarkable experience working with her and I had the opportunity to develop myself in research benefiting from her experience.

I appreciate the support of moy co-advisor Prof. Dr. Serhat Yeşilyurt as he guided me wherever I was lacking the key knowledge in the whole process of working on the thesis. I am really thankful that I had the opportunity to work with him and benefit from his experience.

I would also like to thank Prof. Dr. Ayşe Berrin Yankıoğlu, Dr. Melih Türkseven and Dr. Vahid Tavakol Aeghei for accepting to take place in the jury comittee of my thesis defense. Their valuable comments have helped me improve the thesis.

Farzad Rokhsar Talabazar has been a valuable friend for me as he always supported me whenever I struggled with the pressure. I would like to thank him also for his encouragement and motivational talks he gave in important times.

I would like to thank my dear friends Ece Kurt and Can Alper Önol for their great friendship and making my master's years more enjoyable.

My lab mates Kazi Sher Ahmed and Peyman Khandar Shahabad have always been wonderful towards me and I would like to thank them for their support.

I would also like to thank my friends Mehmet Alper Okur and Doğay Kamar for helping me understand concepts of the topics I worked on.

Special thanks to my genuine group of friends Erdem Çalışkan, İzzet Can Bekar, Oğulcan Güdülü, Şükrü Tarık Kostak and Emre Uzun for standing with me during my bachelor years. My deepest appreciation for my friends goes to Özen Yılmaz and Dilara Toprakhisar for their long and warm friendship.

I would like to thank my dear family for their encouragement and unconditional support throughout my life. None of this would be possible without their guidance and support.

*To my parents,*
for their love and patience.

# Table of Contents

# List of Tables

# List of Figures

# List of Algorithms

# Chapter 1

# Introduction

Chapter 1 introduces the hybrid vehicles and modelling of the energy sources. Then it discusses the application of rule based, deterministic and stochastic optimization-based methods on the fuel cell hybrid electric vehicle in order to minimize energy consumption while satisfying the power demand. The discussion is followed by a brief overview of reinforcement learning methods and their utilization on such a problem that is the most important aim of the thesis.

## 1.1 Hybrid Vehicles

Since the industrial revolution, fossil fuels have been the main energy source of vehicles and evolved into mostly gasoline or diesel due to the several advantages it offers, such as long range, high power and energy density, fast replacement, easy storage. However extensive exploitation of those fuels caused shortage in fossil energy as they can only be formed in thousands of years. Furthermore it started to affect environment and human health adversely due to its hazardous emissions. Thus, the demand for vehicles utilizing by alternative energy sources has increased recently.

In order to achieve less or zero emission and also decrease the dependency on fossil fuels vehicles propelled by electric motors are offered instead of combustion engines. However, although recently we see a lot of improvements, there have always been concerns about this vehicle mainly about limited range, long charge duration and lack of charging infrastructure. On the other hand in hybrid electric vehicles where different sources are utilized together are offered to eliminate these issues. The most popular HEVs (Hybrid Electric Vehicle) are categorized as such: Parallel hybrid, series hybrid and power-split hybrid. As there are several different configurations of these type of vehicles, classifications cannot draw a certain line and be conclusive. A hybrid vehicle configuration where an internal combustion engine (ICE) and an electric motor (EM) both partake in the propulsion of the vehicle mechanically is classified under the category of parallel hybrid vehicle. To achieve such an architecture an extra electric motor operating as a generator is not required since the standalone electric motor is able to serve the purpose when necessary. The most important advantage of such a system is that it provides the opportunity to run the engine and the motor in their efficient ranges creating an optimization problem. Unlike parallel hybrid if the ICE is not involved in propelling the vehicle mechanically, such a vehicle falls into the category of series hybrid vehicles. The ICE is only required to supply power to charge the battery that the EM uses. In this configuration since the only purpose of the ICE is charging the battery, it is possible to operate it in its most efficient ranges. Since EM is the only source of thrust, the control problem is not a complicated one. In series hybrid, hybridisation of the ICE and EM is established via electrical connection whereas parallel hybrid benefits from a mechanical connection. Power split hybrid combines those connections with the assistance of an extra electric motor. The vehicle can be powered by only ICE or only EM or both of them at the same time.

So far the hybrid vehicles in question were the ones employing two machines that are ICE and EM. Another hybrid type that is similar to a series hybrid system can be accomplished with adoption of EM only whose energy is supplied by two energy sources. Alongside the battery that is the most conventional energy source particularly in hybrid vehicles, fuel cell has gained popularity and found itself a place as the secondary or primary energy source. In that case hybridisation is on a different level and an optimization of two different energy sources must be achieved.

Fuel cell hybrid electric vehicles (FCHEV) are the most promising hybrid vehicle type in terms of achieving zero emission. The system does not produce any hazardous end product after the chemical reactions occur when producing the energy. The application of fuel cell the technology in transportation is relatively new and has potential to be improved. It has advantages over traditional vehicles powered by ICE, PHEV, HEV and pure EV. The range problem of EV is eliminated in FCHEV thanks to similarity of operation as in HEV or conventional vehicles with ICE. It requires a hydrogen tank that will not take more than a few minutes to be refilled resulting in a continuous travel. Since no combustion occurs unlike ICE, the efficiency of the fuel cell can go up to %60. The range of such vehicles is also not a problem. There are challenges of utilizing fuel cell as the only energy source, as in the initial start-up it might take some time due to the structural requirement of the system. Using it with a battery on the other hand can eliminate this problem. There are other challenges as well such as the cost, hydrogen supply to customer, energy management strategy, safety and reliability, but in this study we will focus on minimizing the energy consumption of such vehicles [9]. Figure 1.1 shows the structure of a FCHEV. Hydrogen fuel tank and the oxygen in the air are used by the fuel cell stack that comprises several fuel cells in order to generate electricity alongside the battery pack. Via the DC-DC converted the electricity is transmitted in to the EM. Also there must be a cooling system as the sources generate heat as well as a side product.

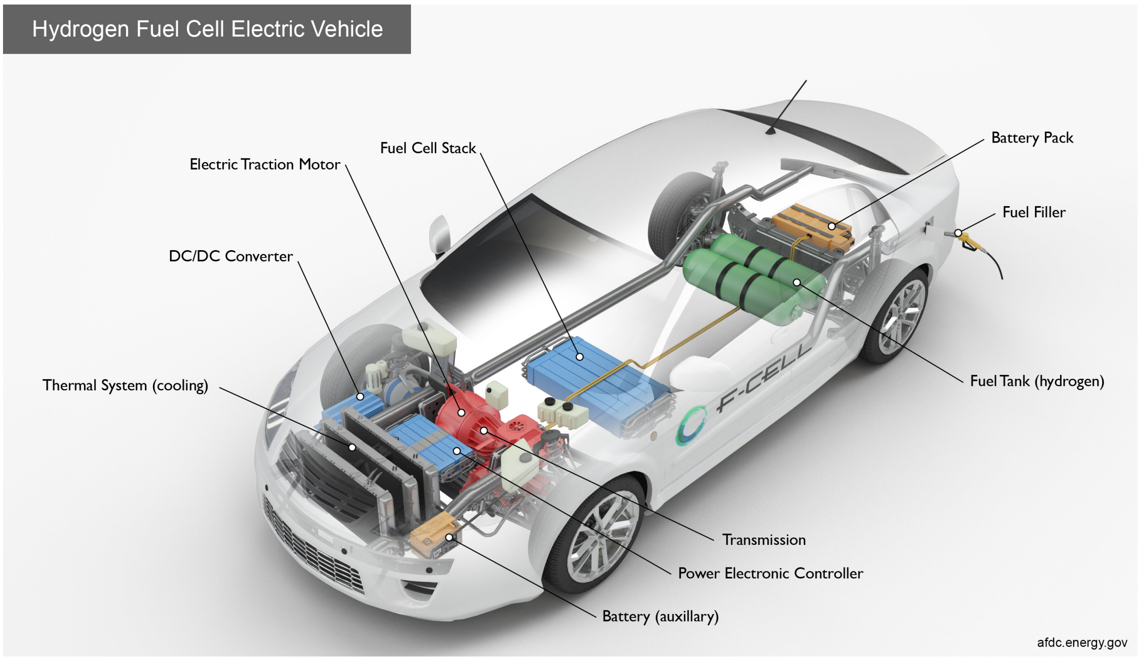

Figure 1.1: Architecture of a FCHEV [2]

The most common classification of FCHEVs shows that there are two types of configuration in these vehicles depending on the size [10]. Either the fuel cell or the battery is the main power source. If the battery size is greater then the fuel cell is used as a range extender. On the other hand, if the fuel cell is the main power source then the battery supports the fuel cell. As fuel cell hybrid vehicles (FCHEV) is a relatively a new type of hybrid vehicles it is not common in industry compared to HEVs and only three brands Toyota, Hyundai and Honda have manufactured such vehicles so far. Several studies are present on the topic of energy management of fuel cell vehicles utilizing optimization or rule based methods. On the other hand reinforcement learning applications on energy management are not that common.

Another important element in FCHEVs is the DC-DC converter as the energy management system decisions are exerted on the switches of the converter. If there is only one converter in the system then there is no need for external control as the ratio of the voltage of the motor to voltage of the source will set the value of the switches with a simple closed loop model. However if there are two converters ensuring a better control scheme, then there should be another loop for setting the current. Then it is possible to control the power sharing.

## 1.2 Energy Storage and Conversion Systems in FCHEV

Energy storage in transportation is clearly an important subject as any vehicle is obligated to move without any external energy supply at least for a certain amount of time. Traditionally energy is stored as a fuel in a fuel tank, whose range is limited by the capacity of the tank. Combustion engines are still the most common energy converters today and it requires gasoline, diesel or kerosene as the fuel. On the other hand, these vehicles utilize batteries not necessarily as an energy source for propulsion but as a source for accessory purposes. However as the emission problem of these vehicles are becoming an environmental issue, alternative energy sources started to replace the conventional petroleum products used by combustion engines. The transition from ICE vehicles to battery electric vehicles (BEV) on the other hand has not been straightforward since the energy and power density of the traditional batteries were not sufficient for a vehicle to travel in long distances and needed abundant of time to be recharged. Developments in battery technologies have made it possible for them to be used only in HEV and PHEV but also BEV.

Lithium-ion battery is a rechargeable type of battery that has started to be used extensively in all types of electric vehicles to replace Nickel Metal Hydrate (NiMH),Nickel Cadmium (NiCd) and Lead Acid batteries. The most important advantages the lithium-ion batteries offer are that they are able to hold high energy density, they have higher Coulombic efficiency and do not self-discharge easily [11]. As shown in figure 1.2, compared to its predecessors lithium-ion batteries cover the area corresponding to a higher energy and power density region and even have more potential in this aspect with selection of different material in cathode.

The lithium ion batteries consist of several cells. The cells are combined together to form battery modules. The plant used in vehicles is in the structure of battery packs. Thus the cells can be configured to satisfy the demands of a converter in terms of maximum voltage and current hence the capacity.

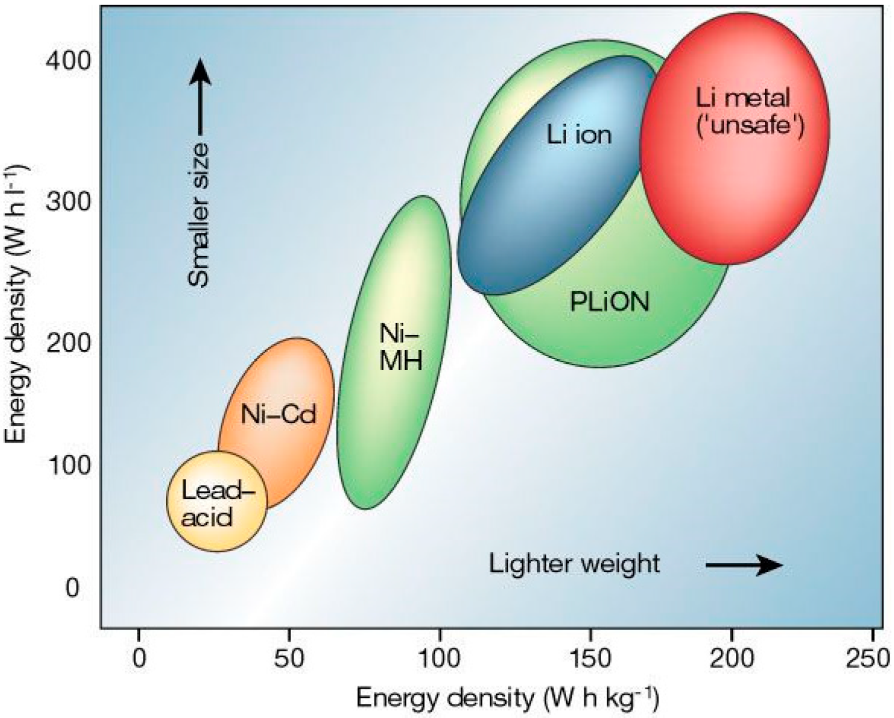

Figure 1.2: Power/Energy Density Diagram of Different Battery Technologies [11]

Another alternative energy source is the fuel cell that is relatively a new technology. It converts the chemical energy stored in hydrogen to electricity. The fact that the end products of reactions occurring in the fuel cell are water and electricity, makes it a perfect candidate to replace the conventional technologies. As a plus it requires oxygen alongside hydrogen that has clearly ease of access.

There are variety of fuel cell types listed in the Figure **??** featuring different characteristics and used in different applications. Thanks to its advantages such as high efficiency, operation in low temperatures and less susceptibility to corrosion PEM fuel cells are preferred in transportation systems.

### 1.3 Energy Management in Hybrid Vehicles

As there is not a single energy source or a power converter in hybrid vehicles, a control method is required regardless of its being an HEV, PHEV or a FCHEV. Different machines have different efficiency characteristics and optimal operating points. The general purpose is to minimize the total energy consumption. Even though HEVs and PHEVs are more common in the market resulting in the fact

Table 1.1: Fuel Cell Types [1]

| FC Type | Operating Temperature and Efficiency | Typical Stack Size | Automotive Applications | Advantages | Disadvantages |
|---|---|---|---|---|---|
| Polymer Electrolyte Membrane (PEM) | <120°C<br>50-60% | 1 - 100 kW | -Backup power<br>-Portable power<br>-Transportation | -High power density<br>-Low temperature<br>-Quick start-up<br>-Quick load following<br>-Solid electrolyte reduces corrosion | -Sensitive to fuel impurities<br>-Expensive catalysts |
| Solid Oxide (SOFC) | 500-1000 °C<br>60% | 1 kW-2 MW | -Auxillary power<br>-Electric Utility | -Tolerance to fuel impurities<br>-Fuel flexibility<br>-High efficiency | -Long start-up<br>-Slow dynamic load behaviour<br>-High temperature<br>-Corrosion of components |
| Molten Carbonate (MCFC) | 600-700°C<br>50% | 0.3 - 3MW | -Electric Utility | -High efficiency<br>-Fuel flexibility | -Low power density<br>-High temperature<br>-Long start-up time |
| Alkaline (AFC) | <100 °C<br>60% | 1-100 kW | -Military<br>-Space<br>-Backup power<br>-Transportation | -Low cost components<br>-Low temperature<br>-Quick start-up | -Sensitive to CO2 in fuel and air<br>-Electrolyte management<br>-Electrolyte conductivity |
| Phosphoric Acid (PAFC) | 150-200 C<br>40% | 5-400 kW | -Distributed generation | -Suitable for CHP<br>-Increased tolerance to fuel impurities | -Expensive catalysts<br>Long start-up time<br>-Low power density |

that energy management strategies (EMS) are extensively researched, there are still studies on EMS in FCHEVs as well. The EMS can be divided into two main categories as optimization based and rule based. The optimization methods on the other hand can be applied in a way that either a global optimum point is found with the driving cycle data which is known beforehand or a sub optimal point is found with not only the past information but also present and future information. The distinction between the methods is that the optimality in the first one cannot be updated whereas in the latter is indeed a dynamic one and adaptive as well.

**Rule-based** strategies require a set of conditions to be checked in each time instance. Those rules are derived heuristically and cannot guarantee an optimal operating point. However it is widely used today due to the fact that it is practical, easy to apply and works fast. Based on those rules controller decides how to share the power demand of the vehicle between the energy sources. The strategy is applied to vehicle systems [12] and also in the form of fuzzy logic [13]. The same strategy is applied to a FCHEV as well [14] and the adaptation will be used for comparison in this study.

**Real time optimization** (RTO) methods predicts the optimal output within a process and keeps measuring the the real data. By doing so instead of finding a global optimum point, several optimization problems are created to be solved at each time step. The method is developed in order to address the uncertainty of the real-time interaction of the controller with the environment. It is aimed for the controller to respond properly in the case of existence of a disturbance. The method provides a framework for not only past but also present and the future information to be utilized. The most popular methods applied for RTO are equivalent cost minimization strategy (ECMS) [15] [16] and model predictive control (MPC) [17] [18].

**Global Optimization** methods tries to find the optimum point of a given objective function using a set of data most often driving cycles in our case. Mostly a combination of different drive cycles are fed into the model and best set of decisions are made in order to optimize the given function. Dynamic programming (DP) is widely used for that purpose [19] [20] [21] that is a method benefiting from the principal of optimality idea of Bellman equation. Linear programming [22] and convex optimization [23] are also used for that purpose however DP is still the most common approach as it almost ensures that global optimum is found. The only drawback in terms of accuracy stems from the discretization of control input which can be avoided largely for the expense of simulation time. Such methods are obviously excellent for comparing the performance of any other method as it sets the best achievable target for the objective function. Moreover the result of the other optimization or even rule based methods can be updated in order for it to be closer to the best possible outcome [24].

There are also stochastic optimization methods some of which is only involved in the offline optimization seeking to find the global optimum point of the objective function. Those methods utilizes a Markov Decision Process (MDP) along with the Bellman equation [7]. The most important concepts to comprehend an MDP are state, action, probability of state transition, reward and policy. States are variables in a model that is helpful to observe the behaviour of the system. Actions are the variables that interacts with the model/environment and cause a state transition. For each transition there is probability and the action gets the reward. Lastly the policy is the map of action variables for each state and the objective is to find the policy that maximizes the reward.

Lately as the reinforcement learning algorithms started to become a promising technique and being applied to many control problems, thanks to development of the model-free algorithms that can be applied to any environment defined as an MDP. Q learning and Deep Q learning (DQN) are the most common model-free algorithms in the studies focusing on HEV or PHEV that are similar to FCHEV. In Q learning or also denoted as Q table learning a random action is selected causing a state transition and a reward is obtained. The sum of future and immediate rewards are collected in the Q table consisting of state and action values. In every step if the current reward is greater than the previous reward, the table containing the value of the reward corresponding to that state and action value is updated. The Q learning algorithm is applied to an FCHEV [25, 26], presented as a favorable method, results of which are compared to the conventional control methods. There are also studies of adapting a derivative of DQN, which is a method similar to Q learning yet utilizing a deep network for mapping, for FCHEV. In [26], the focus is on the effect of initial state of charge and they prove the applicability of reinforcement learning algorithm. Fuel cell degradation is another important phenomena that is studied in [27] along with the energy minimization with DQN. Also importance of selecting the best objective function is emphasized in [28] by comparing the performance with the outputs obtained by dynamic programming as well as comparing them with Q-learning based EMS. In almost all of these studies the converter is only modelled as an efficiency map or just a single efficiency value. There are also applications of DQN in transportation applications whose energy is supplied by the combination of fuel cell and battery such as railway vehicles and ships [29] [30, 31]. DQN application is also common in HEV [32–36] and PHEV [37, 38], several studies tried to explore the application of the method. DQN utilizes neural networks instead of a table and is proposed as a faster algorithm applicable to simulations that are considered to last longer. Instead of updating the values inside the table, parameters inside the network are revised.

There are also combinations of real time and global optimization that aim to exploit the advantages of these methods [33]. As the speed of training process of the algorithms is crucial several studies attempted to modify the algorithms in order to increase the convergence rate [25, 39]. Another algorithm called as Dyna-H is proposed [33] and modified it in order to obtain better performance. Deep Deterministic Policy Gradient (DDPG) is another algorithm of reinforcement learning evolved from the DQN algorithm. The most important advantage of the DDPG algorithm over DQN is that it can take a continuous action space as an input unlike DQN in which the action space must be discretized. The DDPG algorithm is applied for a HEV [40] and also for a hybrid electric bus (HEB) [41] showing that it is able to perform better in terms of energy consumption.

## 1.4 Thesis Objectives and Contributions

The rule based energy management strategy is the most common method used in industrial application since it is easy to implement, practical and operates in high speed. However since it is not an optimization method, the best energy efficiency cannot be achieved. On the other hand global optimization methods require massive amount of time and cannot guarantee the best performance when applied in real-time. Finally real-time applications on its own may not find the optimum results since it divides the main objective function in time domain. Hence we aim to construct a reinforcement learning framework using a cutting edge algorithm to improve efficiency of the fuel cell where SOC deviation is limited compared to the most common strategies by getting as close as possible to the global optimum. Moreover we aim to benefit from the fact that the neural networks trained with prior knowledge, can be updated with online traffic information. Thus, creating a base controller trained with only the data acquired before hand, to be further improved by real-time applications is our purpose. We also intend to accomplish creating the framework by controlling the duty cycles of the switches in the DC-DC converter as it would be in the real-life application.

These guide the the objectives of this thesis to **propose a novel energy management strategy where the purpose is reducing the energy consumption of a FCHEV model involving the energy sources and DC-DC converters by improving the efficiency of the fuel cell and maintaining the state of charge of the battery within certain limits, by using the reinforcement learning algorithm DDPG**.

To meet the objectives, the main contributions of this thesis are focused on the modelling and optimization of the energy management system of FCHEV by using the DDPG algorithm and demonstrate that it outperforms the rule-based approaches in terms of energy consumption. Specifically:

A. Development of a realistic FCHEV power unit model and control system as well.

B. Application of a powerful reinforcement learning algorithm DDPG on the minimization of energy.

C. Comparing the results of the DDPG algorithm with commonly used DQN algorithm and rule-based strategies.

## 1.5 Thesis Outline

The thesis is organized as follows:

- The power unit modelling of the FCHEV is one of the most important segments in the study. **Chapter** 2 describes the modelling of the energy sources and the DC-DC converter as well. It further explains the control method of the two separate DC-DC converters individually connected to the energy sources and both connected to the electric motor requiring both current and voltage control.

- Since the purpose of the study is to show that reinforcement learning algorithms can bu utilized in energy management in FCHEV, general descriptions of the approach are given and the algorithms to be employed are explained in **Chapter** 3 It further discloses the implementation of DQN and DDPG algorithms in our model.

- **Chapter** 4 provides the simulation and the methods that we aim to compare our approach with. Their implementation is explained and the performances of all EMS strategies are evaluated based on several drive cycles.

- Finally, the thesis is concluded in **Chapter** 5 with recommendations on the training and validation of the algorithms for future work.

# Chapter 2

# Fuel Cell Hybrid Vehicle Power Unit Modelling

A fuel cell hybrid vehicle is powered by an electric motor whose energy is supplied by the battery and the fuel cell. A converter is required to establish the electrical connection between the energy sources and the electric motor. The schematic of the simulation model is given in figure 2.1.

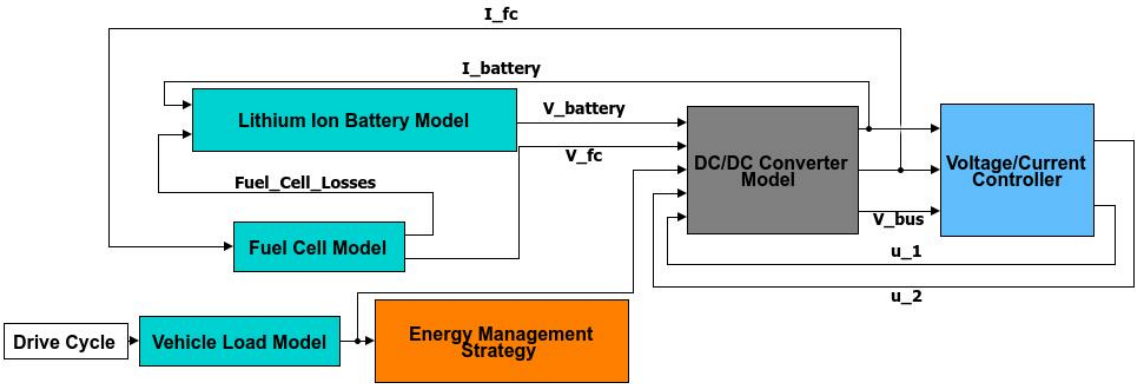

Figure 2.1: Simulation Model Overview

When the power demand is set by the driver, duty cycle of the switches in the converters are controlled based on the energy management strategy, in order to split the power between the energy sources. Those components can be combined in different combinations as every topology have their own advantages. They are divided in four categories as summarized in figure 2.2. The fourth topology is selected in this study as it provides the opportunity to control the power flow from both of the sources.

The main power source in the vehicle is the fuel cell and the battery supports the fuel cell when necessary. Thus, the purposes of the battery can be listed as accumulating the energy from braking, power the vehicle when the demand is relatively low, and help the fuel cell operate within an efficient range when possible. This chapter

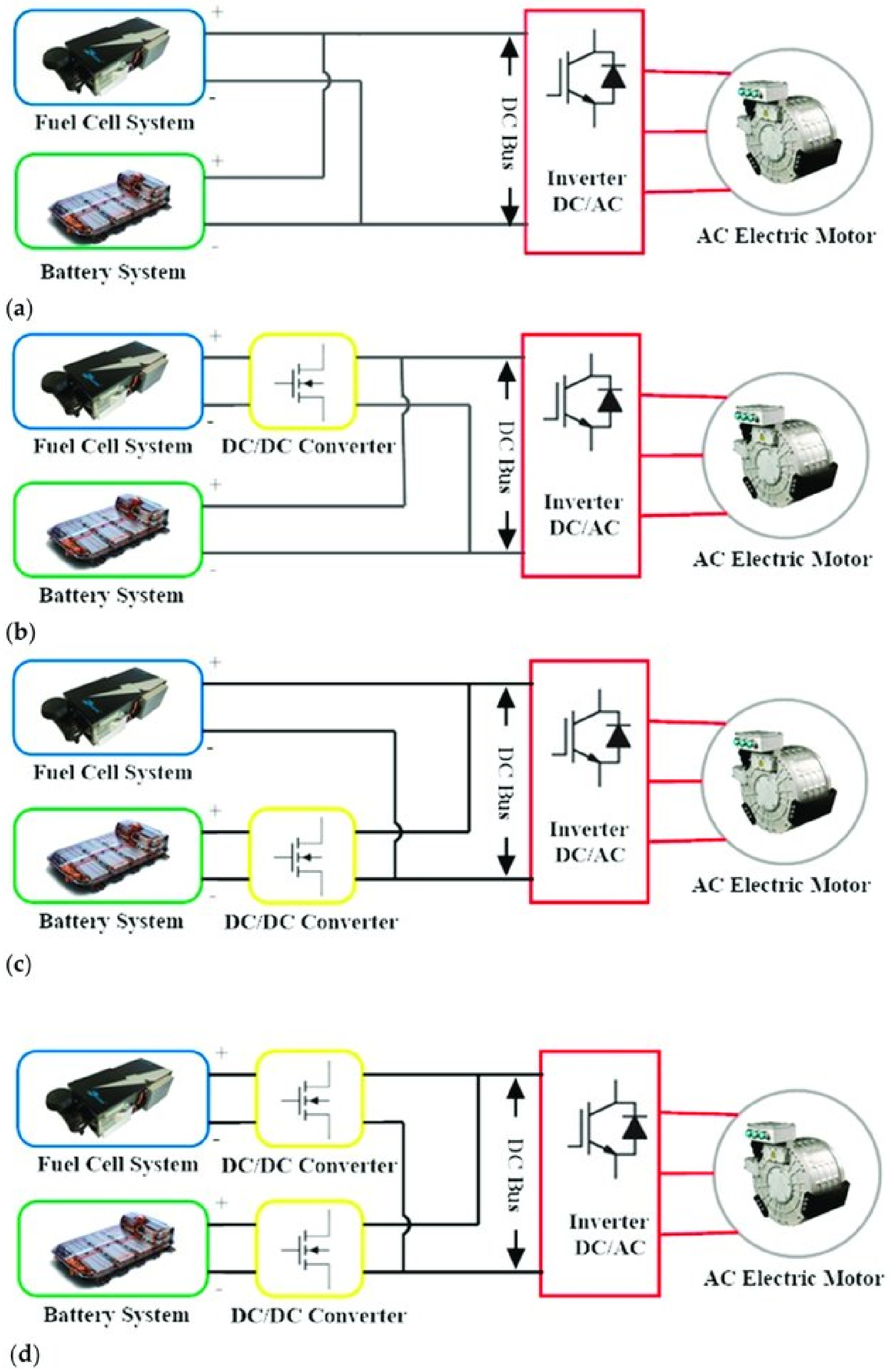

Figure 2.2: Topologies of FCHEV power unit [3]

describes the modelling of the fuel cell, battery and the DC-DC converter. The energy management strategy will be discussed in the next chapter.

## 2.1 Battery Model

One of the power sources of FCHEV is the lithium-ion battery. A lithium-ion battery consists of two current collectors, an anode and a cathode, separator and electrolyte as seen in figure 2.3. When discharging the battery, the anode material that is usually graphite whose surface contains lithium ions, releases those after half chemical reactions resulting in formation of electrons. As the separator allows only the lithium ions to pass through, electrons are forced to use another channel to move towards the cathode side where a metal oxide compound is present. As a result electricity is produced and supplied to the load connected to current collectors. Both of the anode and cathode materials have lithium in their structure. Electrons and the lithium ions combine again in the cathode, intercalating into the cathode material.

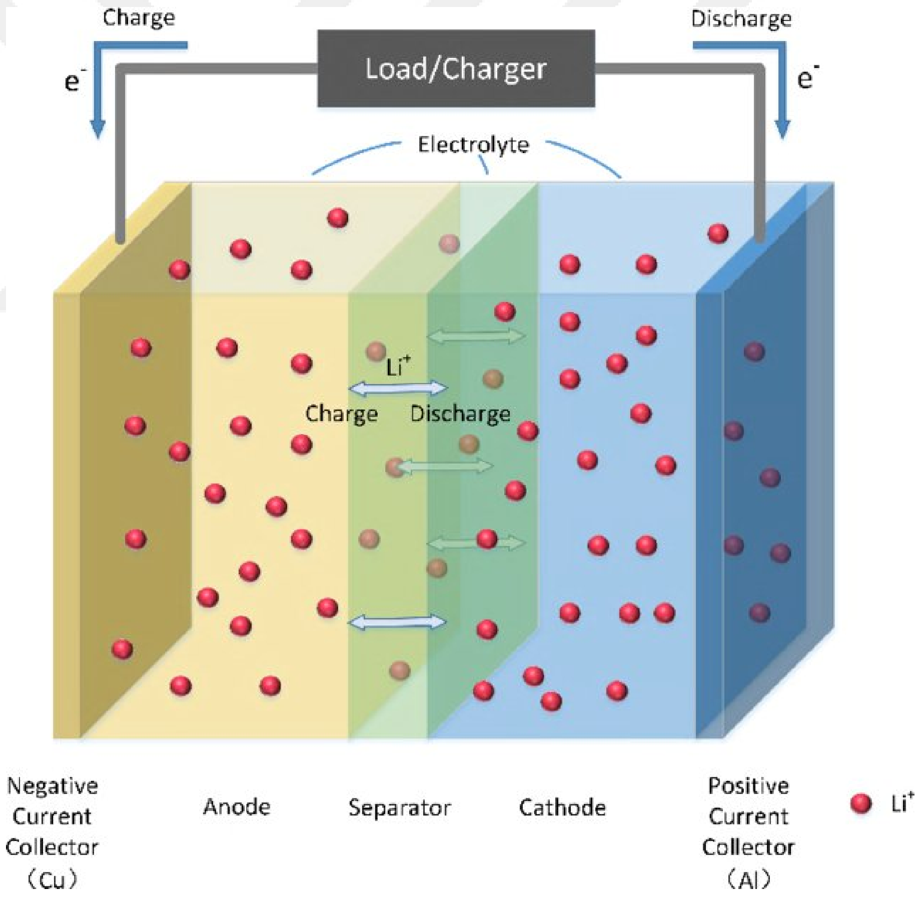

Figure 2.3: Schematic of a Lithium-ion Battery [4]

There are two common approaches to model the behaviour of a lithium ion battery cell under charge and discharge. The first one is a mathematical model where the behaviour is approximated by using common circuit elements such as resistor and capacitors, derived after observing the data collected in testing where it is charged or discharged with different currents and State of Charge (SOC) values [42]. It is rather easy to compose these equations, use them in real time control operations and able to demonstrate the behaviour of the battery within an acceptable margin of error. The other approach is physics based modeling of the charge and mass conservation of the solid particles and electrolyte, and also the movement of lithium between phases. Sharp accuracy is the purpose of this approach as it involves complex chemical equations used to model what is really happening inside the battery, whereas for the same reason it is not suitable for real-time applications. In this study due to its practical benefits, the lithium ion battery is modelled with equivalent circuit model as in figure 2.4. The model is developed in order to simulate the response of the battery under different SOC and currents. The procedure of the derivation of the model starts from discharging a battery under controlled conditions followed by the observation of the collected SOC and voltage data. Those outputs are then stored and used to form a lookup-table. It provides the voltage output of the battery pack, with respect to the current input drawn by the DC-DC converter. The values of resistors and capacitors in the model defines the characteristics of the battery. The purpose of the resistors is to demonstrate the voltage drop when current is drawn and capacitors are crucial to show the voltage recovery behaviour of the battery. Their values change with SOC and the temperature of the battery. Furthermore the voltage that the source provides changes with respect to SOC and when there is no load on the battery, the voltage that the battery has is called open circuit voltage (OCV). Since one of the important aspects of the study is to maintain the SOC level within a narrow margin, the temperature dependency of OCV, resistances and the capacitances is neglected. In addition constant resistance and capacity values are adopted, although it causes the model to be slightly less accurate.

The SOC of the battery changes when it supplies power to the motor via the converter. This change can be calculated as follows:

$$\frac{dSOC}{dt} = -\frac{i_{bat}}{Q_{bat}} \tag{2.1}$$

where $i_{bat}$ is the current drawn from the battery in amperes (A) and $Q_{bat}$ is the energy capacity of the battery in ampere-hours (Ah). $i_{bat}$ is positive when charged. The change in the SOC is then used to determine the charge state of the battery in percentage.

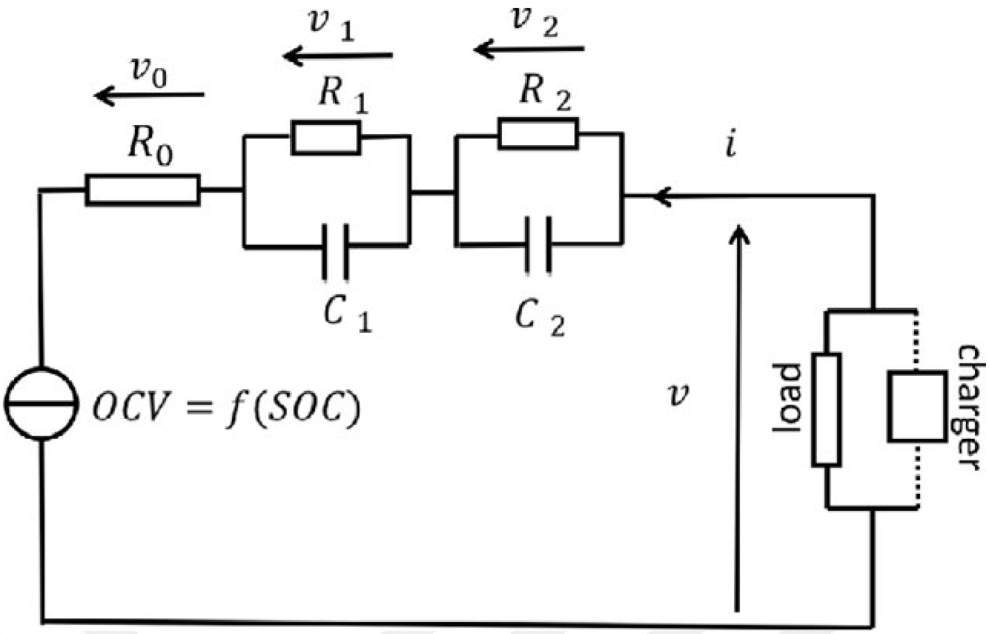

Figure 2.4: Equivalent Circuit Model [5]

The current provided by the battery controlled by the DC- DC converter causes voltage loss from OCV. The battery voltage under a load can be calculated using equation 2.2 shown below:

$$V_{bat} = V_{OCV} - V_1 - V_2 - V_0 \tag{2.2}$$

In this equation, $V_0$ represents the instant loss when current drawn from the battery and found by the multiplication of this current with $R_0$. Since the voltage loss of a real battery develops over time $V_1$ and $V_2$ are required to represent that type of behaviour. $V_{OCV}$ refers to open circuit voltage, is a function of SOC only and can be obtained by the look-up tables recorded after charge and discharge test applied to batteries. The current passing through the resistors R1 and R2 ($i_{R1}$ and $i_{R2}$) can be found by equations 2.3 and 2.4 respectively.

$$\dot{i}_{R_1} = \frac{i_{bat} - i_{R_1}}{R_1 C_1} \tag{2.3}$$

$$\dot{i}_{R_2} = \frac{i_{bat} - i_{R_2}}{R_2 C_2} \tag{2.4}$$

## 2.2 Fuel Cell Model

The main power source of the vehicle considered in this study is the polymer electrolyte membrane fuel cell. A fuel cell uses hydrogen and oxygen to generate electricity, heat and water. It is similar to the batteries in some aspects such that it has elements as anode, cathode, electrolyte and separator resulting in a similar architecture. The main difference is that fuel cells are not energy storage devices but energy conversion devices. They can generate electricity as long as the fuel ($H_2$) is flown into the anode. In the anode, particularly the catalyst later, hydrogen is split into $H^+$ ions and electrons through the chemical reactions. Hydrogen ions are allowed to pass through the exchange membrane as electrons are not and instead they move to the outer circuit supplying electricity for the load. On the other side of the fuel cell in the cathode, oxygen flow is performed capturing the electrons and the $H^+$ ions producing water as summarized in figure 2.5. Hydrogen is stored in a tank in a pressurized form whereas oxygen flow is conducted via a compressor which takes the air as the input.

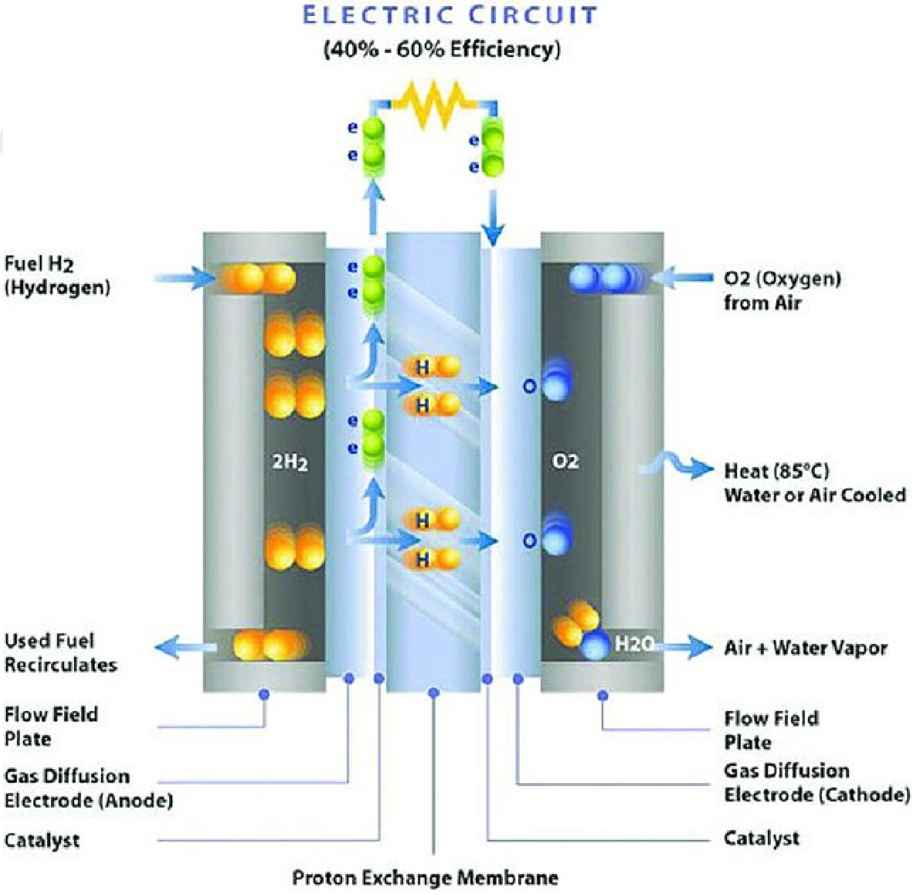

Figure 2.5: Schematic of a PEM Fuel Cell [6]

When there is no load on the fuel cell the cell voltage is equal to ideal Nernst voltage. There are three main phenomena which are ohmic ($V_{ohm}$), activation ($V_{act}$) and concentration ($V_{con}$) losses causing voltage drop when current is drawn. The output voltage is expressed in equation:

$$V_{cell} = E_{nernst} - V_{ohm} - V_{conc} - V_{act} \tag{2.5}$$

where $V_{cell}$ represents the cell voltage and $E_{nernst}$ is the potential voltage the cell can provide. It can be calculated by equation 2.6. Hydrogen partial pressure $\mathrm{P_{H_2}}$ and temperature of the cell $\mathrm{T_{st}}$ are the variables in the equation.

$$\begin{split} E_{nernst} = 1.229 - 0.85 \times 10^{-3}(T_{st} - 298.15) + 4.3085 \\ \times 10^{-5}[ln(P_{H_2}) + 0.5 \times ln(P_{O_2})] \end{split} \tag{2.6}$$

Ohmic Loss: In a fuel cell due to the resistance of current collectors, polymer electrolyte membrane for ion exchange and electrodes, sudden voltage drop occurs as soon as the current starts flowing. $\mathrm{i_{fc}}$ is the current drawn from the fuel cell where $\mathrm{R_M}$ is the resistance of the membrane.

$$V_{ohm} = i_{fc}(R_M) \tag{2.7}$$

The resistance of the membrane is proportional to the membrane thickness $t_m$ and the conductivity $\sigma_m$ :

$$R_M = \frac{t_m}{\sigma_m} \tag{2.8}$$

In order to calculate the membrane conductivity, water content denoted as $\lambda_m$ and the temperature of the membrane are used as input. The temperature of the stack is assumed to be equal to membrane temperature.

$$\sigma_m = (b_1\lambda_m - b_2)exp(b_3(\frac{1}{303} - \frac{1}{T_{st}})) \tag{2.9}$$

The coefficients $b_1$ $b_2$ and the membrane thickness are derived from Nafion 117 membrane and $b_3$ is used as a fitting coefficient [43].

Concentration Losses: Concentration polarization effect occurs as the fuel cell is discharged with high current. When the current demand is high, $H_2$ and the oxidants decrease in high rates at the gas channels. However, in the inlet portion of the fuel cell the concentration of these reactants stays high. The difference causes supply voltage to fall. The loss is modeled as in equation 4. j is the current density [44].

$$V_{conc} = -\frac{RT_{st}}{2F} ln(1 - \frac{j}{j_{max}}) \tag{2.10}$$

Activation Losses: An electrochemical reaction is similar to a chemical reaction. In each step reactants have to overcome a certain threshold of activation energy to form a product. Even though there are several factors in the electrochemical reaction resulting in activation loss, the voltage drop can be expressed with the equation.

$$V_{act} = \frac{RT_{st}}{F} ln(\frac{i_{fc}}{i_0}) \tag{2.11}$$

Like many energy and power source fuel cell cannot maintain the same level of efficiency throughout different power demands. Particularly when the power demand is low, the efficiency is very low as well due to the characteristics of the PEM fuel cell. After the peak efficiency point, the efficiency starts decreasing as the power demand increases further. Figure 2.6 shows the efficiency of the fuel cell with respect to power demand.

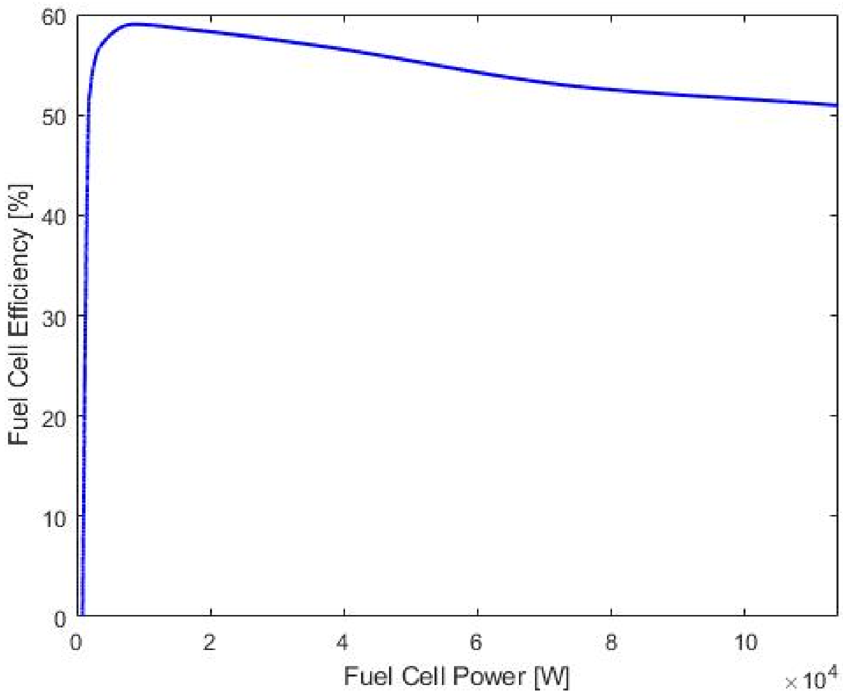

Figure 2.6: Efficiency of a PEM Fuel Cell [6]

### 2.2.1 Compressor Model

In order to produce electricity using hydrogen and oxygen as in a fuel cell, they have to be in high pressure in the reaction. Assuming hydrogen is supplied steadily from a tank , oxygen flow only can be achieved by a compressor. It causes some of the energy to be consumed even before the fuel cell produces electricity. Mass flow rate of oxygen reacting in the cathode is calculated in the equation:

$$\dot{m}_{ca,rec} = \frac{M_{O_2} n_{fc} i_{fc}}{4F} \tag{2.12}$$

where $n_{fc}$ is the number of cells in a stack and $i_{f}c$ is the fuel cell current. Using the excess ratio $\lambda_{O_2}$ ,which is the proportion of flow of oxygen into the cathode to the flow mass of oxygen reacted, and also taking oxygen proportion in the air into consideration that is denoted by $\phi_{air-oxygen}$, flow into the cathode is calculated in equation 25.

$$\dot{m}_{comp} = \dot{m}_{ca,rec} \lambda_{O_2} \phi_{air-oxygen} \tag{2.13}$$

The power output of the compressor is then calculated by equation 2.16 whose derivations are shown in steps in the equation 2.14 and 2.15, where P is the pressure and $\nu$ is the specific volume.

$$W_{comp} = \dot{m}_{comp} \int \nu dP \tag{2.14}$$

$$W_{comp} = \dot{m}_{comp} \frac{kR(T_{out} - T_{in})}{k-1} \tag{2.15}$$

$$W_{comp} = \dot{m}_{comp} \frac{kRT_{in}}{k-1} [(\frac{P_{out}}{P_{in}})^{(k-1)/k} - 1] \tag{2.16}$$

## 2.3 DC-DC Converter Model

DC-DC converters are power converters designed to raise or lower the voltage. Since voltage of the energy sources and operating voltage of the electric machines usually mismatch, they cannot be connected to each other directly. If the converter raises the voltage, then it works in boost mode whereas if it lowers the voltage then it operates in buck mode. The voltage transformation is achieved by the switches inside the circuit of the converter and usually they are one of metal–oxide–semiconductor fieldeffect transistor (MOSFET), insulated-gate bipolar transistor (IGBT) and bipolar junction transistor (BJT). The voltage is increased due to the coil in the circuit which is loaded by the source during the phase that the circuit is a closed one. If the switch is open, then the load on the coil releases to the motor side. The frequency of the opening and closing the switch adjusts the amount of voltage to be raised or lowered. These switches are controlled by pulse width modulation (PWM) which is simply square waves taking the value of 0 or 1. Some converters can work only in one mode and some of them are capable of achieving both. In terms of electric current direction, there are two types of DC-DC converters one of them is bidirectional and the other one is unidirectional. The first one can operate in either directions and the latter can only work in one direction.

Due to the fact that unlike fuel cell, battery pack can be charged, a bidirectional converter that adjusts the current accordingly in case of charging, has to be connected in series with the battery, on the other hand fuel cell requires a unidirectional converter. When the power flow direction is to the motor both of the converters act as a boost converter since the voltage of the battery pack is lower than the required voltage at the electric motor. When the battery is charged the bidirectional converter acts as a buck converter. Since boost and buck are accomplished by the IGBT switch, energy management is indeed performed by the PWM duty cycle fed into it. The schematic of the model is presented in figure 2.7.

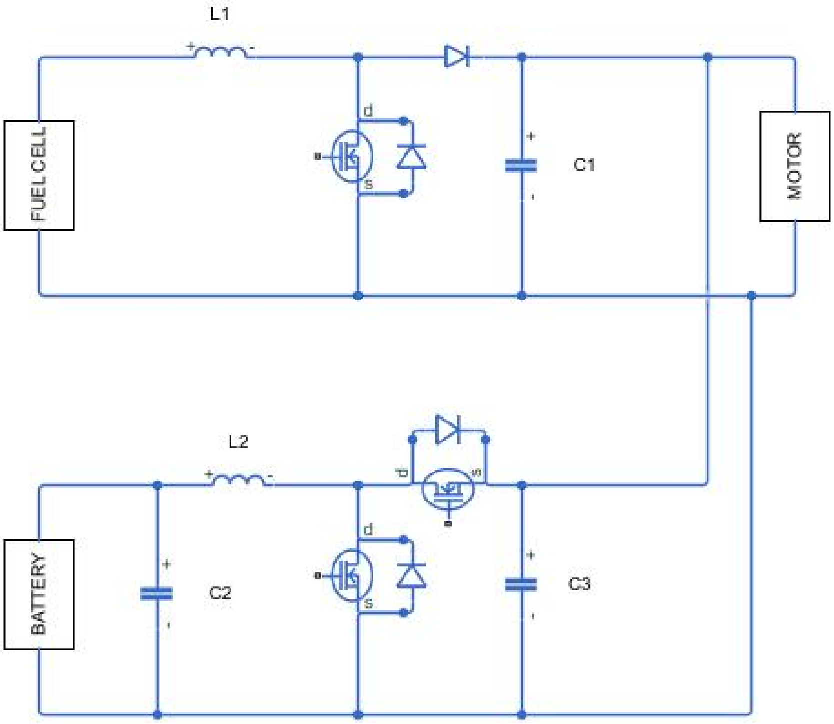

Figure 2.7: Schematic of a DC-DC Converters

$u_1$ and $u_2$ are the duty cycles of the switches used to boost the voltage and regulate the current drawn from fuel cell and battery respectively.

$$u_1 = K_p[i_{L_1} - F_{fc} * [K_p(V_{busref} - V_{bus}) + K_i \int (V_{busref} - V_{bus})]] + K_i \int [i_{L_1} - F_{fc} * [K_p(V_{busref} - V_{bus}) + K_i \int (V_{busref} - V_{bus})]] \quad (2.17)$$

$$u_2 = K_p[i_{L_2} - F_{bat} * [K_p(V_{busref} - V_{bus}) + K_i \int (V_{busref} - V_{bus})]] + K_i \int [i_{L_2} - F_{bat} * [K_p(V_{busref} - V_{bus}) + K_i \int (V_{busref} - V_{bus})]] \quad (2.18)$$

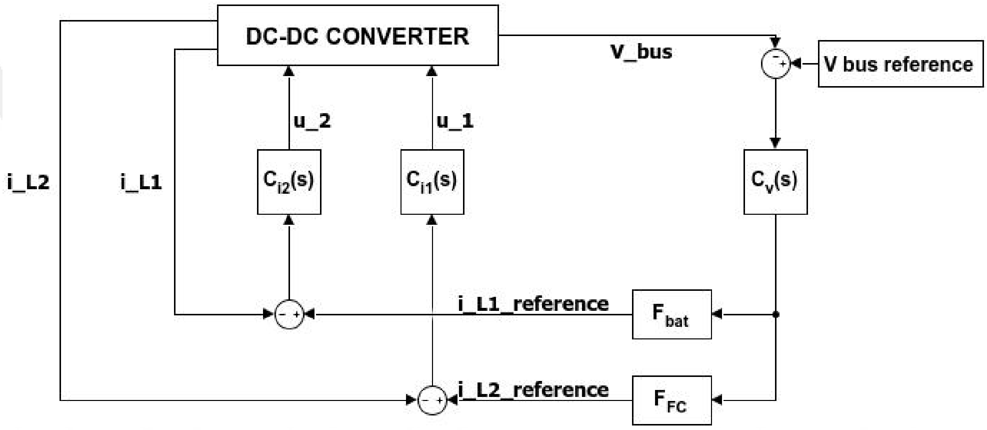

Figure 2.8: Active Current Sharing Control Scheme

$F_{bat}$ and $F_{fc}$ are the gains that the energy management algorithm adjusts in order to decide the proportion of the power demand that is to be supplied by each power source [45]. As there are two separate converters one of which is connected to the battery and the latter to fuel cell, aiming to keep the motor voltage constant when at the same time setting the current demands according to source voltages, their control must involve both current and voltage closed loop systems summarized figure 2.8. Difference between the bus/motor voltage and the reference is multiplied with gains denoted as F after being fed into the voltage controller. The difference is defined as the error and is given as the input of PI controller that behaves as a voltage to current converter. The output signal sets the target of currents for both sources. A similar logic is applied as it was in voltage-loop, difference between the current values is fed into the current controller and converted into the duty cycle. Again this loop is controlled by a PI controller. Based on the duty cycle of the switches, current drawn from the battery and fuel cell is calculated in equation 2.19. $i_L$ is the current flowing through the inductor and $L$ is the inductance of it. $V_{bus}$

and $V_{source}$ are the voltages of the bus and the source respectively.

$$V_L = V_{bus}u - V_{bus} + V_{source} \tag{2.19}$$

$$\dot{i}_L = \frac{V_{bus}u - V_{bus} + V_{source}}{L} \tag{2.20}$$

The effect of input current and shared currents on bus voltage can be calculated as in equation 2.21. $i_C$ and $C$ are the current and the capacitance of the capacitor connected parallel to the bus/electric motor respectively.

$$i_C = i_L - i_{bus} - i_L u \tag{2.21}$$

$$\dot{V}_{bus} = \frac{i_L - i_{bus} - i_L u}{C} \tag{2.22}$$

Since the converter is able to operate in buck and boost modes, when the power flow is in reverse, from wheel to the source particularly when regenerative braking occurs, buck mode is activated whose equations are:

$$\dot{i_L} = \frac{V_{bus}u_3 - V_{bat}}{L_1} \tag{2.23}$$

$$\dot{V_{bus}} = \frac{u_3 i_{bus} - i_L}{C_1} \tag{2.24}$$

## Chapter 3

# RL-Based Energy Management Strategy

Energy management strategies in HEV aim to minimize total energy consumption and keep SOC between certain limits by allocating the power demand to different sources while at the same time meeting that demand. When the power is supplied by battery it will then require to be charged by either fuel cell or regenerative braking since it cannot be charged externally. A basic approach for this problem would be extracting power from fuel cell where the efficiency of the source is close to maximum. As the power demand increases the fuel cell start to run relatively less efficiently, in that case battery can be of service supplying power up to a point that is limited by its SOC. If the power demand increases even more, since the main priority is to ensure that vehicle tracks the velocity profile, there is not much of an option causing fuel cell to run on with low efficiency. The other concern that is to maintain SOC within an interval. As external charging is not possible in an HEV unlike a PHEV, it must be ensured that SOC level must be maintained between certain limits by the fuel cell or regenerative braking during vehicle operation.

In order to accomplish these goals, rule-based and the optimization-based approaches are commonly employed. The optimization based approaches also fall into two categories: deterministic and stochastic optimization methods. The application of the latter is the focus of this study as the popularity and applicability of the model-free reinforcement learning methods have increased recently. In this chapter reinforcement learning will be introduced first and the specific algorithms that are applied to this problem will be defined afterwards.

## 3.1 Reinforcement Learning

Reinforcement learning is considered one of the three paradigms in machine learning alongside supervised and unsupervised learning and applied extensively in many areas. It is a method in which an agent learns how to form the relation between the states and the actions based on the reward function. A random action selected starting from the initial time step for a certain state causing a state transition, action interacts with the environment and as a result a reward value is gained for every single time step. The relation is illustrated in figure 3.1. There are several algorithms serving the purpose, using Markov Decision Process (MDP) which contains state, action, reward and the next state $(S_t, A_t, R_t, S_{t+1})$, as formalization of the problem. In order to comprehend the interactions, elements of the reinforcement learning algorithms, states, actions and rewards, must be introduced.

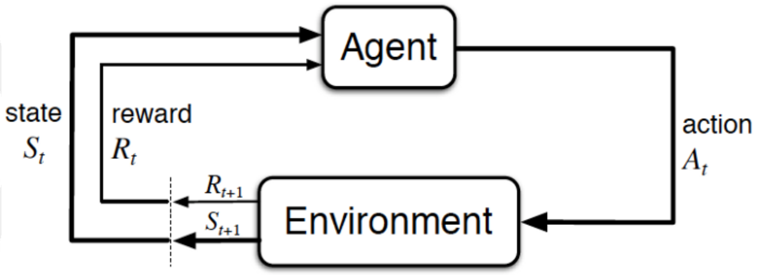

Figure 3.1: The agent environment interaction in reinforcement learning [7]

**States** in reinforcement learning can be considered as an input to the agent consisting of neural networks. It provides the information of the environment after an action interaction. It must contain enough variable for the agent to completely understand the system. If the number of variables in the state increases then it will take extensive amount of time for the training process to be concluded and reward to be maximized. Systems must be investigated thoroughly and minimum number of state variables must be defined because of the curse of dimensionality.

**Actions** are the outputs of the agent and inputs for the environment. The agent selects those randomly and feeds into the environment then checks the situation via states. As they are the only feature the agent can control, variables in that space must chosen carefully. Compared to state selection it is simpler to choose action variables in any case.

For a state an agent takes an action and it return it gets a **reward**. Since the main purpose of the reinforcement learning is to maximize reward, it must be somewhat similar to the objective function. It is not always easy to select the right reward function. Setting it as a similar function to objective function is a simple yet an inefficient approach. The agent sometimes requires some extra encouragements when it is choosing an action in the right direction. Apart from the function itself that is supposed to produce higher rewards when an action serves its purpose, extra rule based implementations might steer the agent to the right direction in a shorter time. Another factor is the numerical range of reward output. If the difference between two rewards is too high then it is possible for the network to be updated drastically causing divergence. Another issue is that the reward might be deceptive for some episodes. This is particularly a problem of the initialization process. If the range of initial variables of the system is too large, a reward similar to the objective function is likely to fail since it will never be clear which actions are actually good. For instance initialization might start somewhere close to the target and an action even though it is completely inaccurate might take more reward than an action simulated in the system initialized far from the target and that is indeed is the best of all. Furthermore it is possible and even certain that the agent will take some actions that will result in deviation from the goal particularly in the first steps of training. In that case a penalty should be defined in order to discourage the agent to take those actions again.

As the simulation progresses the rewards are not accumulated directly, instead future rewards are multiplied by a discount factor to ensure that in a long horizon total reward converges to a value. The value of the discount factor is crucial for both continuous tasks where the problem defined in a time horizon cannot be divided into sub-simulations and episodic tasks where simulation time can be set thus dividing the complete process into sub-groups. In episodic tasks as in our problem until the terminal state is reached, in most case it is the last time step of the simulation or the time when the simulation is stopped as a punishment, target of the value function is updated as in equation 3.1.

$$y_i = R_i + \gamma \max_{A'} Q_t \left( S_i', A' \mid \phi_t \right) \tag{3.1}$$

The target is the sum of the immediate reward and the expected future rewards. As the discount factor approaches to zero target value will always be zero meaning that in every step the next state will be considered as a terminal state and only the immediate reward will be taken into account. On the other hand a discount ratio equal to 1 will cause the future rewards as equally as important compared to the immediate reward.

For several steps at the beginning action selection is always random, but all state transitions, the actions inducing that and the resulting return is stored in the experience buffer. This stored information is used to select the next action sets. The length of the time until when the actions are completely randomized is called mini-batch. That is to say selection of actions, the **policy**, the mapping between the states and the actions are updated as the time hits the mini-batch size. Every once time step reaches to a multiplier of the mini-batch, the parameters in the mapping, in the policy that is formed by the neural networks are altered based on the experience. As the size of the mini-batch increases, for one update of the networks, more experience is used resulting in the increase in the probability that the new values of the parameters in the network are more accurate. On the other hand that leads to a time-consuming process. If the mini-batch size is too small, then the update is conducted based on the information provided by only a few time steps. Thus a good balance must be stroke. Another factor effecting the update is the learning rate that is multiplied with the average gradient over the experience. A high learning rate will cause the average gradient have more effect on the next update, hence the value is expected to be low.

In the final step the loss function is calculated. As mentioned above a value function was calculated with the immediate and expected rewards. The loss function is the square of the difference between value function target and the current value of the value function as in equation 3.2.

$$L = \frac{1}{M}\sum_{i=1}^{M}\left(y_i - Q\left(S_i, A_i \mid \phi\right)\right)^2 \tag{3.2}$$

It is a metric of how close the value function is to the target or how small the values of the expected rewards are. The update of the mapping occurs at this step, after minimizing the loss function.

In order to apply any model-free algorithm of reinforcement learning the problem must be defined. Unlike model-based algorithms, the model-free algorithms as the name suggests can be applied to any model as long as the problem definition includes the elements to be described are action and state space and reward function. Before introducing those, the objective function should be explained as it will demonstrate itself the reason behind the selection of the elements.

### 3.2 Problem Formulation

The objective function set for this problem involves two main variables and they are related to the energy sources of the vehicle. In order to minimize the energy that the fuel cell provide, it must produce power as efficient as possible. The fuel cell efficiency used in this study demonstrates that as long as the power output of the source is close the 10 kW, the purpose of energy management will be satisfied. On the other hand that condition is naturally contradictory for battery usage as an optimization based on solely fuel cell battery might supply so much power that it might exceed the maximum power of the source and also shorten the life of the battery. Minimization of the energy consumption of the battery is the second variable implemented as the change of SOC. Thus the objective function takes the form of:

$$J = \int_{t_0}^{t} \{\alpha[H_{2,eff} - H_{2,effmax}] + \beta[SOC - SOC_{desired}]^2\}dt \tag{3.3}$$

The purpose is to ensure that fuel cell efficiency is close to the maximum and SOC variance is maintained in a narrow range. It is mentioned that SOC loss means that battery is discharged and it is supplying power to the electric motor. The increase in SOC, on the other hand means that the battery is charged and it can only be performed by the fuel cell or the regenerative braking system. Since the case that fuel cell charges the battery might have advantages or disadvantages, in the training process it is the situation that is evaluated. The latter case is assumed to occur at all times meaning that whenever the vehicles brakes, the regenerative braking energy is captured completely by the battery.

The model-free reinforcement learning algorithms used in this study are DQN and DDPG. They slightly differ from each other. The common approach in the literature is DQN or DQN-based algorithms as discussed in the introduction. DDPG is relatively a more recent concept and it has not been implemented widely for problems similar to the one in this study. The convergence for those algorithms is achieved with different states and initialization. Thus the selection of action, state and reward will be defined after the algorithms are explained individually.

### 3.3 Deep Q Learning (DQN)

Before explaining DQN method, first the most basic reinforcement learning algorithm called Q-learning must be introduced. In Q learning the Q-value is stored in a Q-table in which the dimensions are state and action. The most common practice is to use the temporal difference method that is integrated into the Bellman equation. The equation is the founding base of the learning algorithm and may be slightly modified in different algorithms. Q value calculation with temporal difference is given:

$$Q^{\text{new}}\ (s_t, a_t) \leftarrow Q\,(s_t, a_t) + \alpha \cdot \Big(r_t + \gamma \cdot \max_a Q\,(s_{t+1}, a) - Q\,(s_t, a_t)\Big) \tag{3.4}$$

The table is updated in each step with the learning rate multiplier denoted as $\alpha$ and ends when the Q-value cannot increase anymore supposedly because the best action sets for each state is found. It is also possible that actions are stuck because of their greediness. The trade-off between exploration and exploitation that is defined by $\epsilon$ is the most common problem. When $\epsilon$ that is defined between 0 and 1 increases exploration rate increases as well resulting in more random action selection thus giving priority to the future rewards. When the number is close to 0 the algorithms becomes greedy and approaches to the immediate reward. The most striking downside of the algorithm is that it requires a Q-table whose number of elements is the multiplication of the number of states and the actions.

DQN utilizes deep neural network (DNN) instead of a Q table and makes it possible for problems involving with a larger state-action space to be solved in shorter time or solvable at all. DNN consists of several layers first of which is called the input layer and it ends with the output layer; it is derived from artificial neural networks that only consists of an input, hidden and the output layer. DNN on the other hand benefits from several hidden layers in order to provide the opportunity for more complex correlations to be found. Those layers have nodes and all the nodes in each layer are connected to each other. Depending on the application they the connections may differ however in its basic form it has a feed-forward structure as observed in figure 3.2.

In each node there is an activation function whose parameters are updated in order to map the input to the output correctly. The accuracy of mapping or fitting is naturally affected by the number of nodes and the layers. Even though there is not a straightforward guideline to find the numbers that will produce the best results,

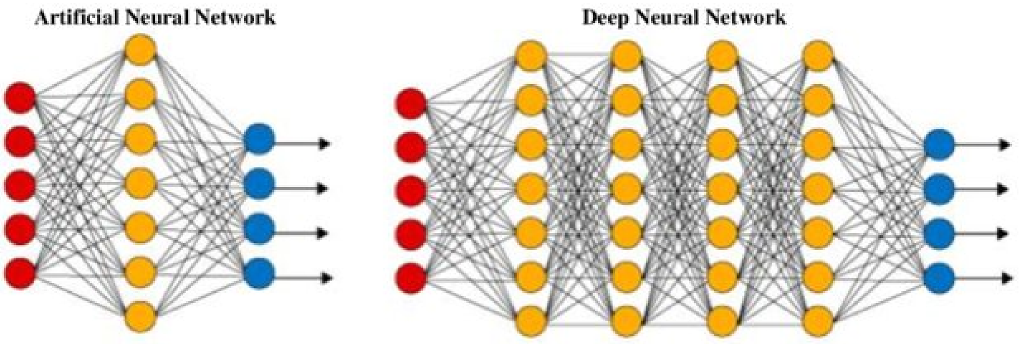

Figure 3.2: Artificial and Deep Neural Network [8]

the basic approach is to build the network as simple as possible and try to see if the fitting has acceptable error. When the layer and node number increases it takes a lot of computation time and that might be misleading in terms of convergence. It is possible that in such a case it requires vast amount of time that is not predicted by the user. However it is not simple to define the level of complexity.

The DQN algorithm as mentioned before utilizes this approach and below is the pseudo-code of the algorithm [46]:

**Algorithm 1** Deep Q-learning with Experience Replay

```
Initialize replay memory $\mathcal{D}$ to capacity $N$
Initialize action-value function $Q$ with random weights
for episode $= 1, M$ do
    Initialise sequence $s_1 = \{x_1\}$ and preprocessed sequenced $\phi_1 = \phi(s_1)$
    for $t = 1, T$ do
        With probability $\epsilon$ select a random action $a_t$
        otherwise select $a_t = \max_a Q^* (\phi(s_t), a; \theta)$
        Execute action $a_t$ in emulator and observe reward $r_t$ and image $x_{t+1}$
        Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$
        Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in $\mathcal{D}$
        Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from $\mathcal{D}$
        Set $y_j = \begin{cases} r_j \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) \quad \text{for terminal } \phi_{j+1} \\ \qquad \text{for non-terminal } \phi_{j+1} \end{cases}$
        Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ according to eq. 3
    end for
end for
```

Instead of using temporal difference method in the value function, and storing this q value in a table, DQN uses a target value function in which only the immediate and future rewards are summed. The network is then updated with the gradient descent of the loss function. In addition instead of evaluating every case one by one, experience is utilized as explained in the reinforcement learning section.

DQN is applied to our problem and the action to be taken is a form of how much of the demand power will be supplied by the battery. The power flow is controlled by the switches in the converter. $F_{bat}$ is the gain in the converter model that is selected as the action. Instead of choosing $F_{FC}$ and $F_{bat}$ as the action variables it is decided that only one of them will be included. Their sum is constant, one is dependant and the other one is independent variable. The equation below shows the relation between the gains and the battery current.

$$I_{bat} = \frac{F_{bat}}{F_{bat} + F_{FC}} * I_{in} \tag{3.5}$$

As long as sum of $F_{bat}$ and $F_{FC}$ is greater than one the system works robustly. It can be concluded from the computational experiments that it is observed that changing this sum improves system response thus it is picked as 4 .

$$a = \{F_{bat}\} \ where \ F_{FC} = 4 - F_{bat} \tag{3.6}$$

The range of the action is selected as below after trial and error, as the DQN algorithm requires discrete actions the range is split into 16 steps with a step size of 0.4.

$$-2 < F_{bat} < 4 \tag{3.7}$$

Table 3.1 shows the modes of power sharing.

Table 3.1: Operation Modes

| Mode | $F_{FC}$ | $F_{bat}$ |
|---|---|---|
| FC charges battery and supplies power | 6 | -2 |
| Only FC supplies power | 4 | 0 |
| FC and battery supplies equal current | 2 | 2 |
| Only Battery supplies power | 0 | 4 |

State variable candidates in the problem are $P_{demand}$,$P_{bat}$,$P_{fc}$,$Fc_{eff}$,$SOC$,$SOC - SOC_{desired}$ that are power demand, power supplied by the battery and fuel cell, fuel cell efficiency, state of charge and deviation of the state of charge respectively. They can be defined in a different form however those are the main variable candidates. In the training process several combinations are tried and finally reward maximization achieved. Those state variables for DQN are:

$$s = \{SOC - SOC_{desired}, P_{demand}\} \tag{3.8}$$

As the state $P_{demand}$ is indeed the input of the system, there cannot be any limitations. On the other hand the first state $SOC - SOC_{desired}$ is limited as:

$$-SOC_{difference,limit} < SOC - SOC_{desired} < SOC_{difference,limit} \tag{3.9}$$

Reward function given below is a form of the objective function whose final form is obtained after a trial and error process.

$$r = -w_{SOC} * (SOC - SOC_{desired})^2 - w_{H_2} * (H_{2,effmax} - H_{2,eff}) \tag{3.10}$$

The simulation stops if the state exceeds the limit shown in equation 3.9 and the agent is given a large penalty to avoid such an action in the future.

## 3.4 Deep Deterministic Policy Gradient (DDPG)

DDPG algorithm is similar to DQN however differs when it comes to updating the network parameters. DDPG is a member of the actor-critic algorithms though DQN has only one network structure. Actor-critic approach resembles the relation between a child and a mother. When an action is decided and interacts with environment the critic guides the actor in the right direction whereas in DQN there is only one network and it is led by the value function only. The algorithm of DDPG method is demonstrated in algorithm 2 [47].

**Algorithm 2** DDPG algorithm

```
Initialize critic network $Q\left(s,a \mid \theta^{Q}\right)$ and actor $\mu\left(s \mid \theta^{\mu}\right)$ with weights $\theta^{Q}$ and $\theta^{\mu}$.
Initialize target network $Q'$ and $\mu'$ with weights $\theta^{Q'} \leftarrow \theta^{Q}, \theta^{\mu'} \leftarrow \theta^{\mu}$
Initialize replay buffer $R$
for episode $=1,\mathrm{M}$ do
    Initialize a random process $\mathcal{N}$ for action exploration
    Receive initial observation state $s_1$
    for $\mathrm{t}=1,\ \mathrm{T}$ do
        Select action $a_t=\mu\left(s_t \mid \theta^{\mu}\right)+\mathcal{N}_t$ acc. to the policy and exploration noise
        Execute action $a_t$ and observe reward $r_t$ and observe new state $s_{t+1}$
        Store transition $\left(s_t,a_t,r_t,s_{t+1}\right)$ in $R$
        Sample a random minibatch of $N$ transitions $\left(s_i,a_i,r_i,s_{i+1}\right)$ from $R$
        Set $y_i=r_i+\gamma Q'\left(s_{i+1},\mu'\left(s_{i+1} \mid \theta^{\mu'}\right) \mid \theta^{Q'}\right)$
        Update critic by minimizing the loss: $L=\frac{1}{N}\sum_i\left(y_i-Q\left(s_i,a_i \mid \theta^{Q}\right)\right)^2$
        Update the actor policy using the sampled policy gradient:
            $\nabla_{\theta^{\mu}}J \approx \frac{1}{N}\sum_i \nabla_a Q\left(s,a \mid \theta^{Q}\right)\Big|_{s=s_i,a=\mu(s_i)} \nabla_{\theta^{\mu}}\mu\left(s \mid \theta^{\mu}\right)\Big|_{s_i}$
        Update the target networks:
                        $\theta^{Q'} \leftarrow \tau\theta^{Q}+(1-\tau)\theta^{Q'}$
                        $\theta^{\mu'} \leftarrow \tau\theta^{\mu}+(1-\tau)\theta^{\mu'}$
    end for
end for
```

DDPG utilizes the actor critic approach and those two different networks are updated with different methods. Critic network is updated in the same way networks are updated in DQN. On the other hand the update of the actor network is conducted by the gradient descent. Similar approach is observed in the way that parameters are updated. However by doing so it is possible to define the action space continuously and decreases the errors caused by the discretizetion. The gradient is calculated as:

$$\nabla_{\theta_{\mu}}J \approx \frac{1}{N}\sum_i \nabla_a Q\left(s,a \mid \theta^{Q}\right)\Bigg|_{s=s_i,\alpha=\mu(s_i)} \nabla_{\theta_{\mu}}\mu\left(s \mid \theta^{\mu}\right)\Bigg|_{s_i} \tag{3.11}$$

The gradient of the critic with respect to action and the actor output with respect to the actor parameters is multiplied in order to find the gradient. DDPG is a more complex algorithm and the training process takes longer time compared to DQN.

However the fact that it does not require discrete action space is a huge advantage. The action is selected according to the current policy and is distorted with a noise function that is decaying throughout the process in order to increase exploration.

The action space defined for DDPG algorithm is the same that of DQN with one difference, they are not discrete. State space on the other hand is slightly different. Again after several training episodes the state variables are chosen as:

$$s = \{SOC, H_{2,eff}\} \tag{3.12}$$

As in DQN state variables are limited in DDPG as well. The limitation for the variables $SOC$ and $H_{2,eff}$ respectively are:

$$SOC_{min} < SOC < SOC_{max} \tag{3.13}$$

$$H_{2,effmin} < H_{2,eff} < H_{2,effmax} \tag{3.14}$$

Efficiency of the fuel cell is directly related to the power supplied by the fuel cell. As the agent observes the efficiency value without knowing how much power is supplied by the fuel cell, it takes a reward in that state. The idea here is that the agent does not need to know the fuel cell power but only the efficiency curve. So it is not important that if the power is sliding left or to the right as the focus is on the efficiency. The sign of the power difference is obtained by the other state variable $SOC$. Reward function is also similar with one little difference and defined as:

$$r = -w_{SOC} * (SOC - SOC_{desired})^2 - w_{H_2} * (H_{2,effmax} - H_{2,eff})^2 \tag{3.15}$$

If the state limits are exceeded then the simulation is stopped and the agent gets a penalty.

## 3.5 Training Process

Implementation of both reinforcement learning algorithms commences with the training process. The model set up apart from the energy management block stays same except for the variables to be initialized randomly for each episode of the training. For DDPG algorithm training episodes last for 15 seconds with a step size of 0.1 second. In each episode at the beginning initial value of the SOC and the power demand is randomly generated within a certain interval. Power demand is kept constant during the episode. An example of the power demand is presented in figure 3.3. On the other hand DQN is trained with a a portion of a driving cycle shown in figure 3.9. The training power input is selected as it cover different power levels within a wide range. Since DQN is a learning technique relatively faster and less accurate, it allowed for the duration of an episode to be longer. An episode in DQN training lasts for 35 seconds with the same step size of 0.1 second. As the cycle possesses random power demand, an initial power demand randomization was not employed. Also as it was possible for the agent to explore more SOC values it was also not randomized.

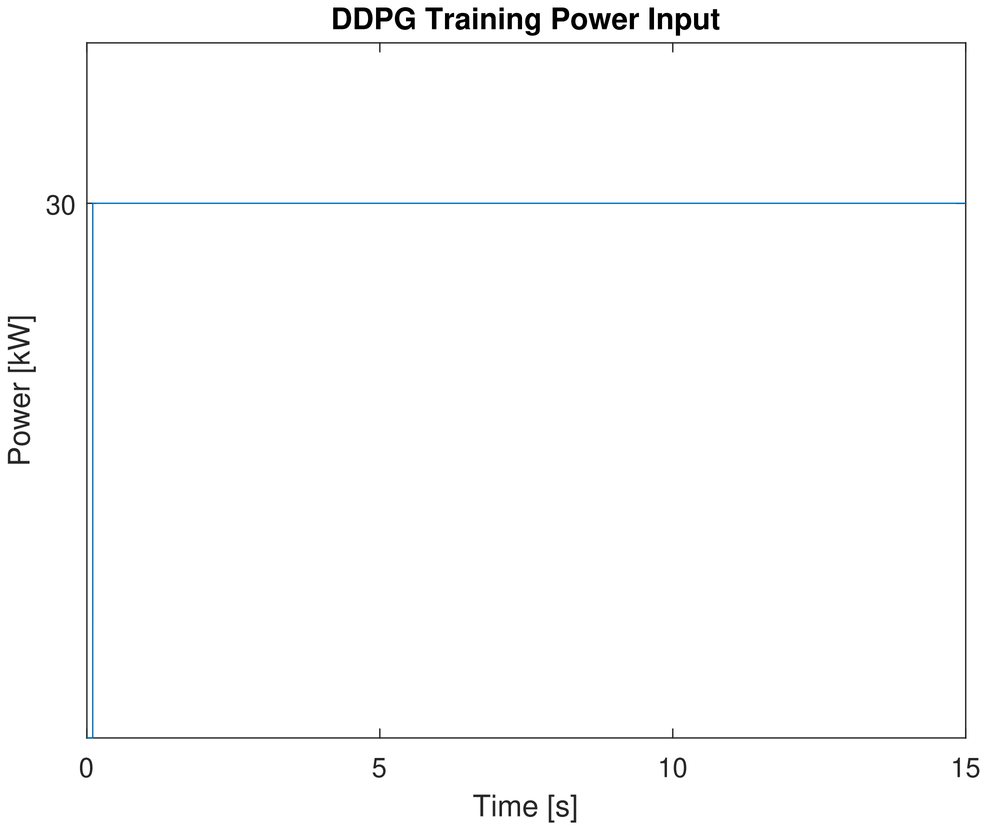

Figure 3.3: Training Power Input for DDPG

In the training process hyper-parameters are the parameters that should be tuned in order to control the learning that are given in table 3.2. The table shows the values of these parameters tuned for the DDPG algorithm.

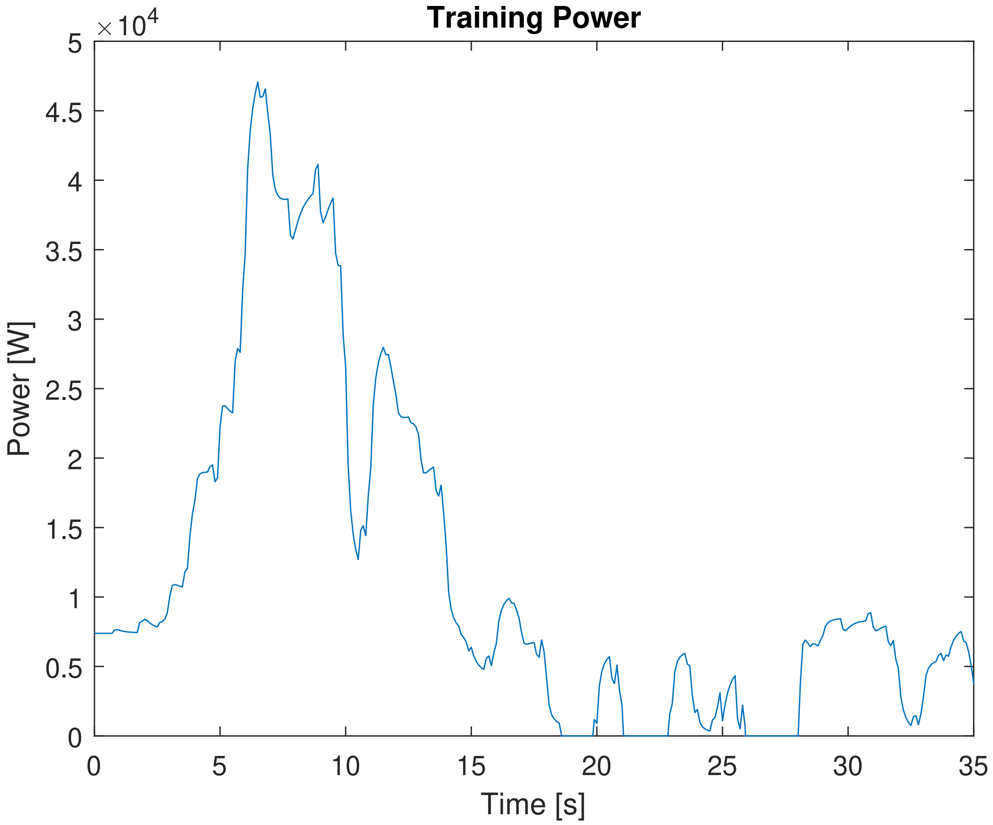

Figure 3.4: Training Power Input for DQN

Table 3.2: DDPG Training Hyper-Parameters

| Parameters | Values |
|---|---|
| Actor Learning Rate | 1e-4 |
| Critic Learning Rate | 1e-4 |
| Discount Factor | 0.99 |
| Mini Batch Size | 128 |
| Experience Buffer Length | 1e6 |
| Agent Noise Variance | 0.1 |
| Agent Noise Variance Decay Rate | 1e-3 |
| Sample Time | 0.1 |
| Simulation Time | 15 |

The first two hyper-parameters are the learning rates. They are chosen as same in this case. Higher learning rates cause divergence mostly whereas lower ones slows down the procedure. In our case if it is high (higher than 1e-4) it is observed that actor approaches either upper or lower limit. Discount factor that has a lower limit of 0 and an upper limit of 1 determines how much of the future rewards agent will take into account. If it is large it considers the possibility of future rewards for actions and if it is small it will only care about the immediate rewards. In many examples this value is chosen to be between 0.9 and 0.99. Before defining mini batch size, full batch learning and stochastic learning must be introduced. In full batch learning network parameters $\theta$ are updated according to the sum of calculated gradients of each episode set. In other words in full batch exact answer is provided with respect to optimum gradient. In stochastic learning however gradient is updated in every single step. Even though full batch learning is bound to converge by maximizing reward, the disadvantage is that it takes a lot of time. Stochastic learning on the other hand is fast as long as it converges. Since the gradients are calculated very often, there could be examples of misleading experience and the maximization of reward might not be achieved. Mini-batch size is the trade off parameter between these two methods. If it is higher it is close to full-batch, if it is lower it is close to stochastic learning. Experience buffer length opens up a space in the computer storage in order to store experience information. Agent exploration within its range is encouraged by the exploration noise variance. Variance decay rate determines how far the variance effect will go in terms of samples. Network structure is shown in figure 3.6 in which there are total of 3 hidden layers and a relu layer containing 300 nodes each. Relu layer has an activation function that converts negative values into 0. Actor network on the other hand has 3 hidden layers directly as a string and the connections are provided by relu layers. The final layer is tanh layer which has an activation function used to output values squeezed between -1 and 1.

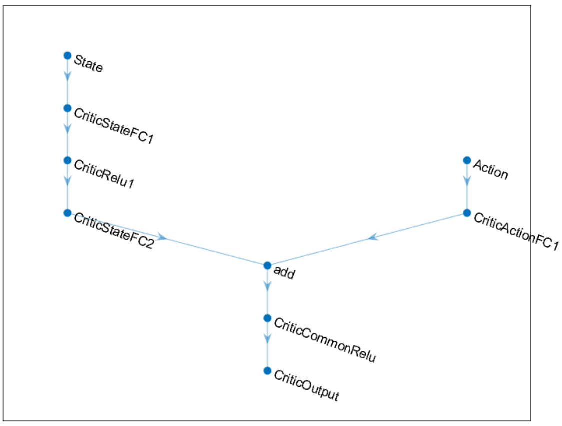

Figure 3.5: Critic Network Structure of DDPG Algorithm

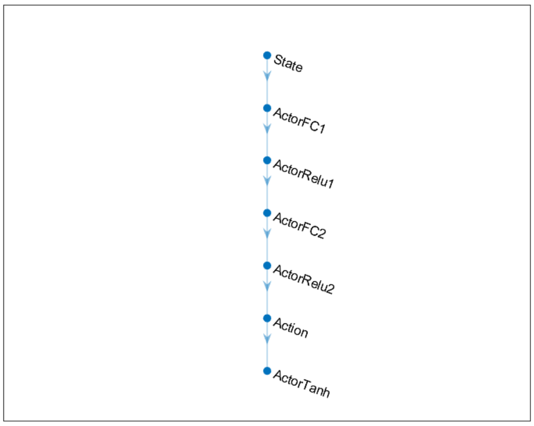

Figure 3.6: Actor Network Structure of DDPG Algorithm

Maximization of the reward function under the given hyper-parameters and the initial conditions started to occur around 7000th episodes as shown in figure 3.7 where blue points shows the reward of the episode and red points shows the average reward over 20 episodes. Until then it can be said that the agent was still exploring the environment and after the 8500th episode the reward started to increase even further to the values close to 0. An agent that achieved such a reward on the episodes particularly between 9000 and 10000 is selected and afterwards used as the controller.

In the DQN training the hyper-parameters that are set are given in table 3.3. As the importance of them are emphasized for DDPG training process already, only lack of hyper-parameters should be pointed out. Since there is no actor-critic scheme in DQN, only one learning rate should be defined. Also as it is not continuous actor space randomization is not required thus there is no parameter about actor variance.

Table 3.3: DQN Training Hyper-Parameters

| Parameters | Values |
|---|---|
| Learning Rate | 1e-3 |
| Discount Factor | 0.9 |
| Mini Batch Size | 2048 |
| Experience Buffer Length | 1e6 |
| Sample Time | 0.1 |
| Simulation Time | 35 |

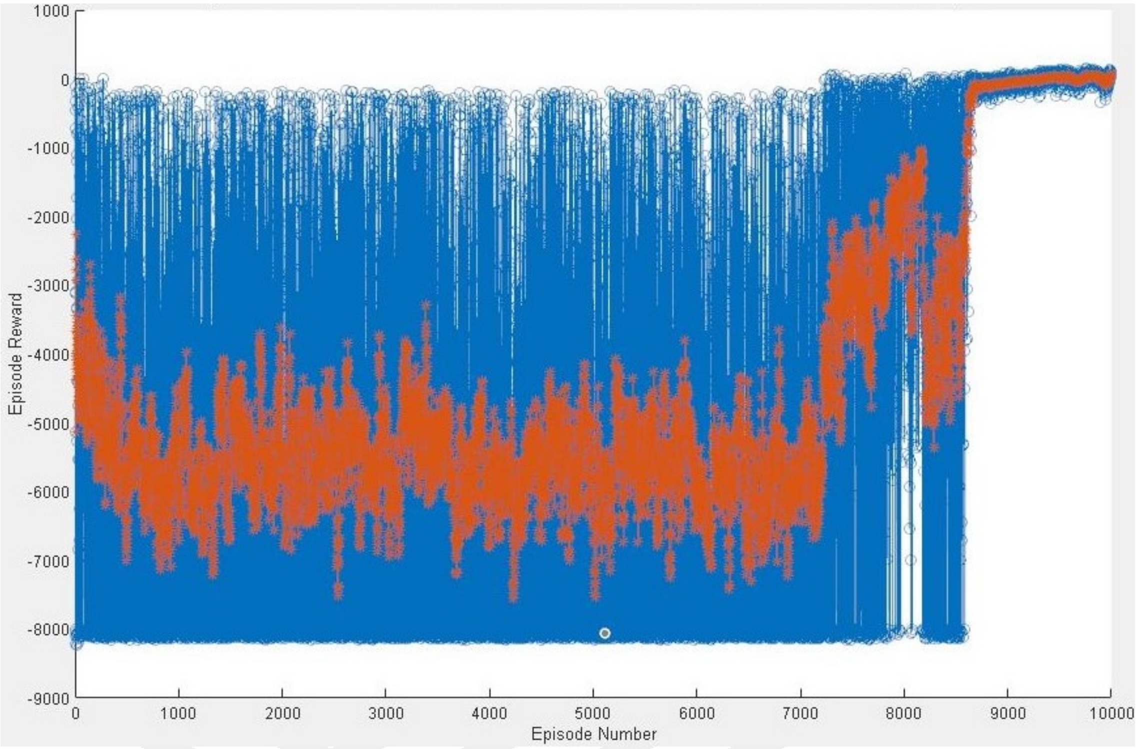

Figure 3.7: Training with DDPG Algorithm

Network structure of the DQN is similar to the critic network of DDPG as shown in figure 3.8. There are total of 8 layers 3 of which are relu layers. In each layer 24 nodes are present. The structure is simpler than that of DDPG and it is obtained after several iterations.

With the right setting, by that it is meant with the right hyper-parameters, network structure, state, action and reward selection the algorithm maximizes the reward in few episodes as seen in figure 3.9. It takes only 65 episode to maximize the reward. Red points as it was the case for DDPG, show the average reward and the blue points show the reward of that particular episode.

In both algorithms the agents with the maximum rewards are selected and their performance are checked. Extensive performance comparison is given in the next chapter.

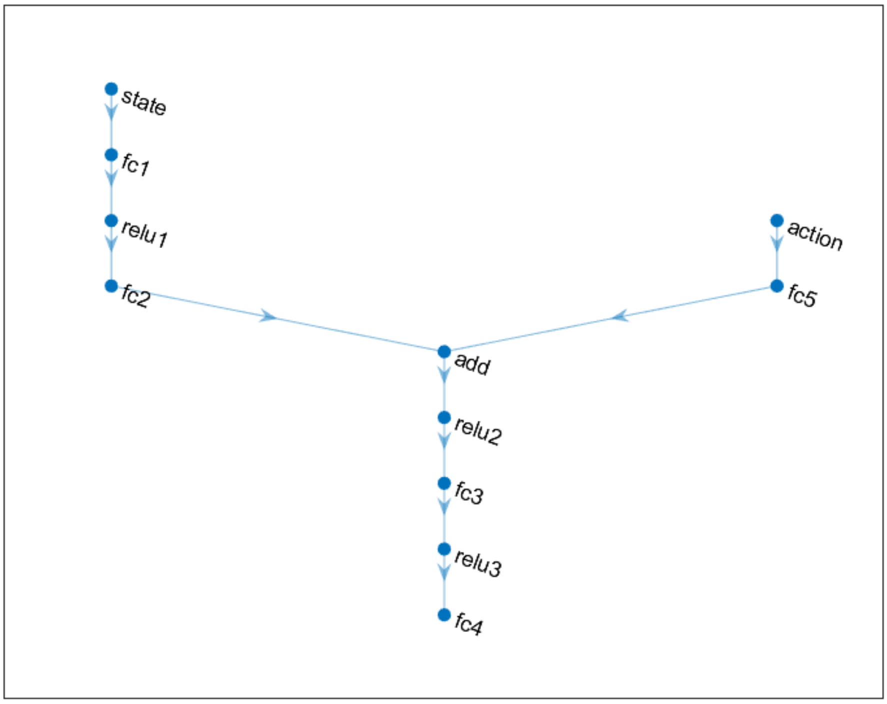

Figure 3.8: Network Structure of DQN Algorithm

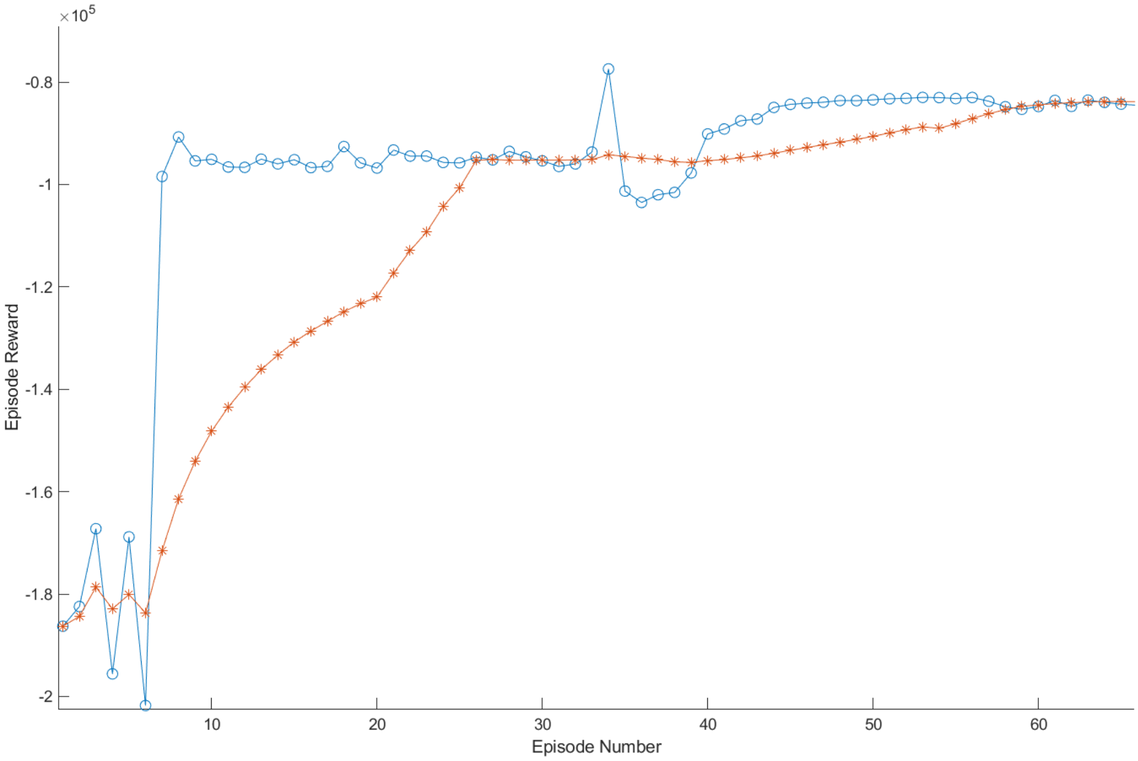

Figure 3.9: Training with DQN Algorithm

## Chapter 4

# Simulation and Results

This chapter describes the simulation set-up, rule based and deterministic optimization method implementation. First the system modelled in Simulink will be introduced then energy management implementation will be explained. Reinforcement learning methods will be followed by rule based and deterministic optimization methods that are used in order to compare the results obtained with reinforcement learning.

### 4.1 Performance of RL Based Energy Management

There are 5 main blocks in the model named as Fuel Cell Model Lithium-ion Battery Model, DC-DC Converter Model, Voltage-Current Controller and Energy Management System as demonstrated in figure 2.1.

Simulation starts with the drive cycle input and it is fed into both energy management block and the DC-DC converter. Based on the decision made in energy management block, the voltage/current controller sets the current demands ($i_{bat}$ and $i_{fc}$) for both converters. Getting the target current and voltage values, the converter block outputs the currents to be drawn from the energy sources and after they interact with these blocks they return the voltage values that is captured by the DC-DC converter again.

Drive cycle is a set of points that represents the velocity of a vehicle with respect to time. It is commonly used in vehicle testing on dynamometers particularly to check the level of emission that a vehicle produces. On the other hand as it provides useful information about the behaviour of a driver or vehicle the application of those drive cycles are extended. They are also benefited in this study to test and validate our model. These cycles are converted into power demand of the EM by Autonomie [48] that is a commercial software in order to feed it into the energy management system and the DC-DC converter as well.

There are several types of drive cycles in order to manifest driving characteristics of the drivers depending on different traffic conditions and also for different regions. For example a driving cycle named HWFET (Highway Fuel Economy Test) is created to represent the behaviour of the vehicle on a highway. On the other hand US06 (High Acceleration Aggressive Driving Schedule) is a cycle useful to observe the capability of the vehicle for situations requiring high acceleration. UDDS (Urban Dynamometer Driving Schedule) is another cycle used commonly to simulate the vehicle driven in an urban environment where the vehicle speed is low and repetitive stop-start occurs [49]. Those three cycles are used to verify the model and also compare the performance of different algorithms. The speed profile of these cycles is given in figure 4.1.

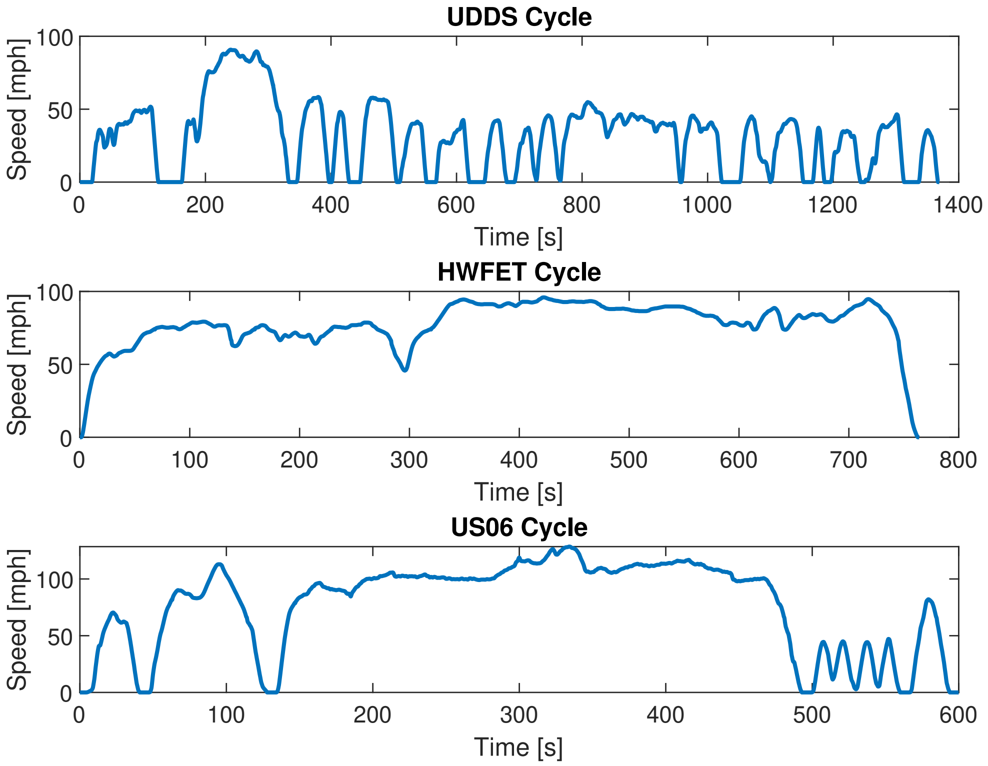

Figure 4.1: Speed Profile of the Selected Drive Cycles

## 4.2 Comparison With Other Energy Management Strategies

Energy management block is where the power sharing decision is made and even though it is not explicitly shown in the model, decision variables are sent into the voltage/current controller block. In this study we aim to compare different control approaches which we describe in the next section.

The first and foremost concept considered before designing a controller is to set up conditions and breakpoints, then modify the action subject to those constraints.

**A rule based strategy** decides on the energy management based on a set of rules. We implemented a rule based energy management strategy in our model. We base our rule-based strategy to Autonomie. Autonomie is a vehicle simulation tool developed by Argonne National Laboratory. It focuses on mainly two variables that were also mentioned in reinforcement learning section. It checks the SOC level and the power demand. Based on max power capacity of the components and the current SOC level, they satisfy the required power and ensure that power gained by regenerative braking is accurately. For that particular reason, those controllers aim to deploy the battery as soon as it is recharged, only then it will be able to recharge the battery later on in case of braking without deviating from the target SOC level largely.

Unlike the rule based strategies optimization methods do not requires several conditions to check and are able to decide more precisely as simply for each state values, they are able to change the actions. In order to find the best possible outcome as long as the algorithm is not stuck in a local minimum or maximum, that kind of method must be used. We have implemented a deterministic optimization method solely with the purpose to compare our results obtained with learning algorithms and demonstrate how close it can become to the best possible solution.

**Brute force search algorithm** is the most general approach to any optimization problem and the algorithm searches for all possible solutions. For each possible action in each state it gets the value of the function. In every single state the action providing the minimum value of the function is stored. Then a lookup table is generated. The method is time consuming and not preferred to solve optimization problems that are complex. In our application for the very same issue, the optimization problem is reduced to a simple one. Evaluation is conduction for only a single time step making the optimum solution short-sighted. However it is still able represent a deterministic optimization approach and used for comparison.

In this study for the brute force search algorithm state vectors are power demand and SOC. They are discretized and the action space is the same that of DQN.

## 4.3 Results

This section compares the results of the algorithms implemented in the model that are DDPG, DQN, Rule-based and brute force under different drive cycles. Brute force findings provides a target for the best control actions even though it is limited by the discretization of variables. We present that learning techniques are able to produce better outcome than rule-based method and close to brute force algorithm results.

### 4.3.1 UDDS Cycle

The comparison will be made based on total energy consumption, average fuel cell efficiency and SOC deviation. Under the UDDS cycle the agents trained with DDPG and DQN algorithms are able to keep the SOC level between certain limits and the deviation from the target that is set as % 50 is not large. At the end of the cycle it still has an acceptable value and restarting the cycle from that point will not cause any significant change of SOC behaviour as shown in figure 4.2. In addition efficiency of the fuel cell is high and the system tracks the power demand within a very small margin of error. Since DQN action space is discrete, sudden action changes cause slight overshoots. A similar SOC behaviour is also achieved by the rule based strategy, however figure 4.3 shows that many changes in the SOC are sharper than that of the DDPG-trained controller. Another point is that it is possible for the SOC level to drop slightly from the target level, if it is necessary, a situation that cannot be observed in rule-based EMS.

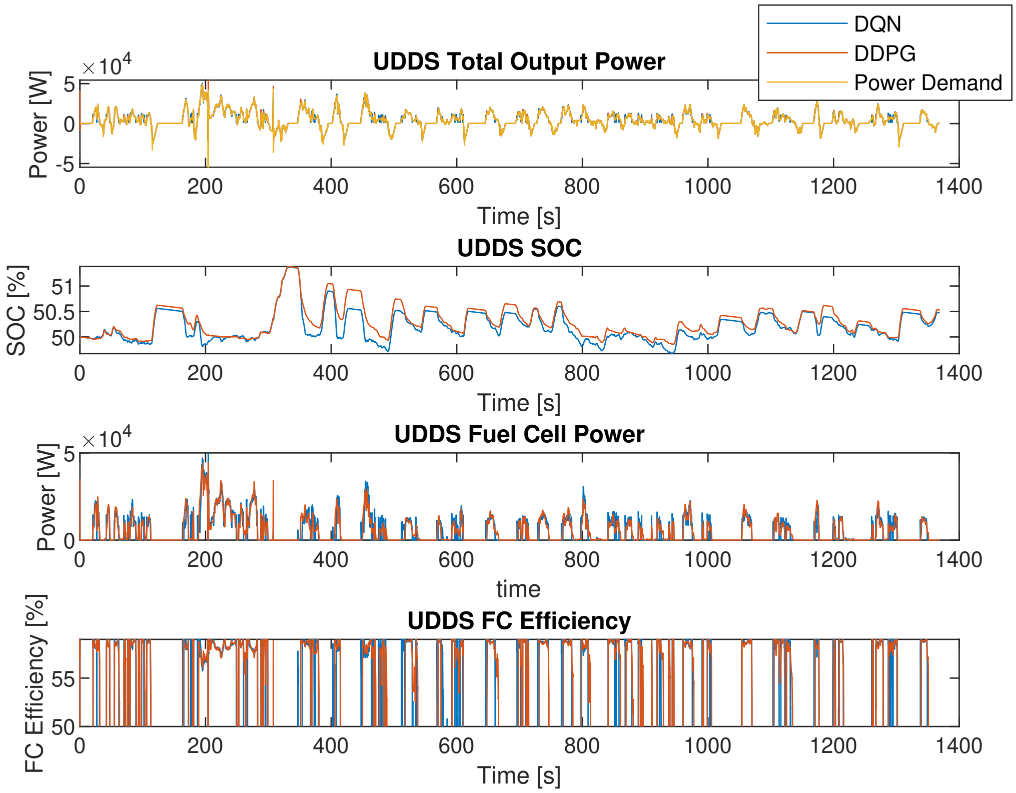

Figure 4.2: Behaviour of the RL-Based EMS trained with DDPG and DQN under the UDDS drive cycle

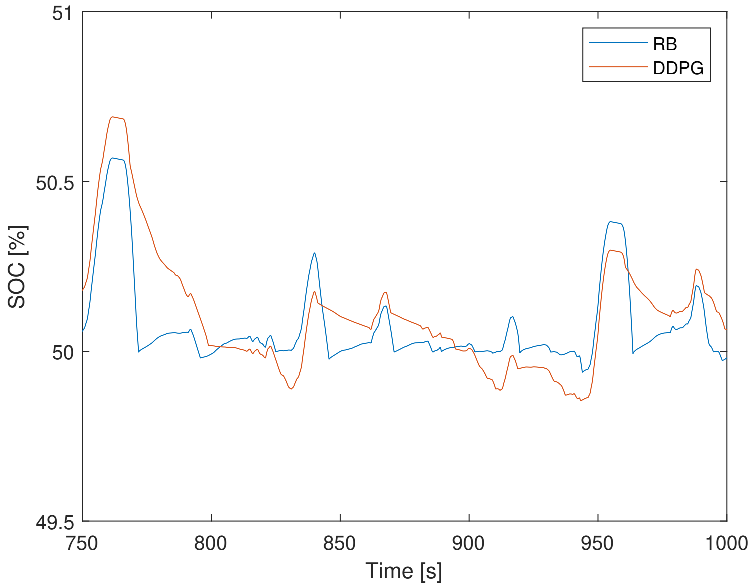

Figure 4.3: Comparison of SOC behaviour of the controller developed with Rule-Based EMS and RL-Based EMS trained with DDPG

SOC levels and efficiency of the fuel cell under different EMS should be discussed together since it is obvious that there is a trade-off between these two phenomena. Several instances are picked in order to compare the performance of the EMS. As it is pointed out before, in the region where the SOC becomes less than % 50 in DDPG-based EMS, the battery is used instead of the fuel cell. On the other hand rule-based EMS is dedicated to prevent SOC fall below the target, as a result even though the power demand at that time range is low, fuel cell is activated causing it to run on low efficiency as shown in Figure 4.4. This is crucial in the way that as long as SOC level is not significantly low, it is treated as a secondary objective and fuel cell efficiency is prioritized.

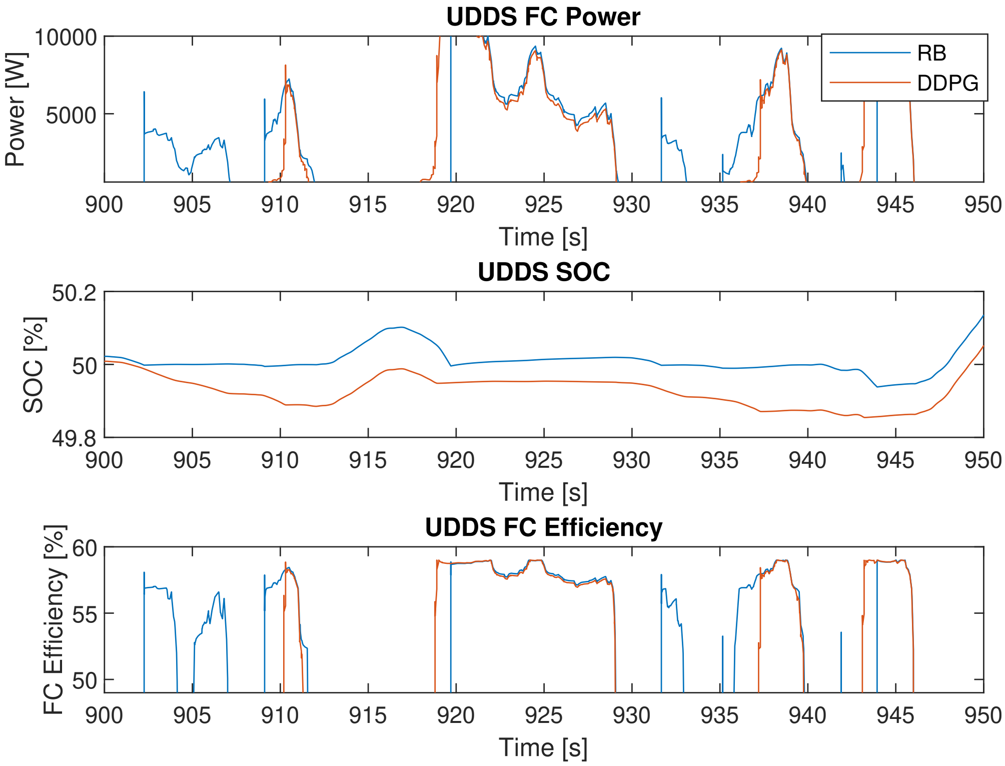

Figure 4.4: Comparison of low power region performances of controller developed with Rule-Based EMS and RL-Based EMS trained with DDPG

Figure 4.5 shows the case where SOC level is slightly higher than the target. DDPG-based EMS starts operating the fuel cell almost where it is the most efficient and using the battery, it tries to keep running the fuel cell on higher efficiency region whereas rule-based EMS under-performs and starts running the fuel cell slightly later causing the SOC fall sharper.

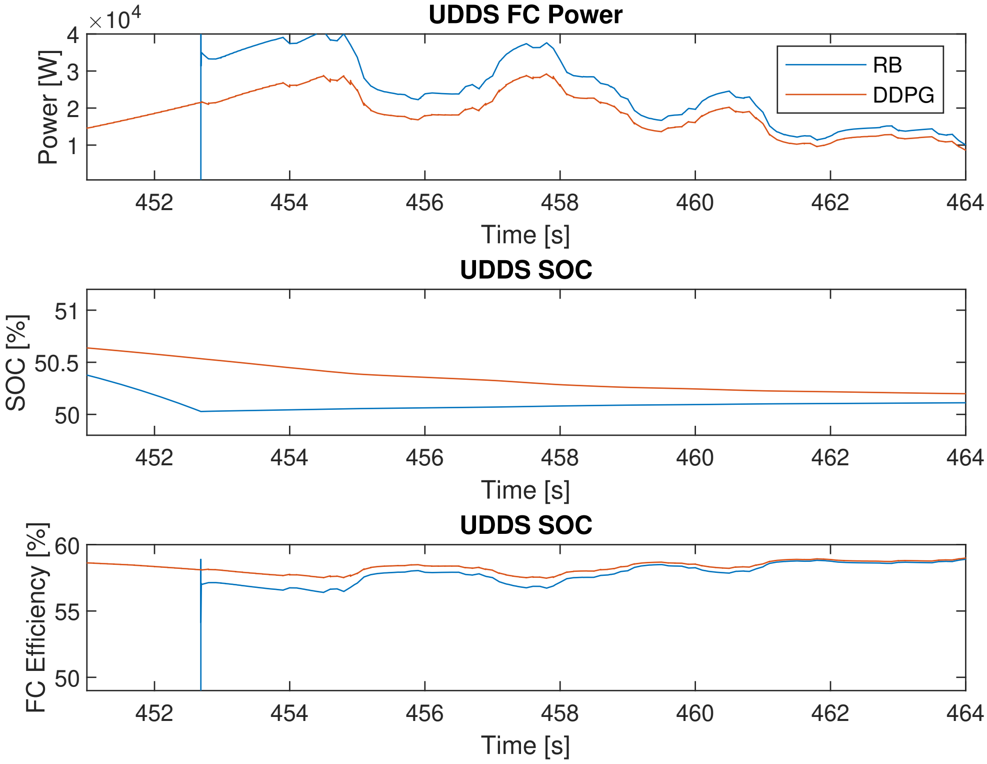

Figure 4.5: Comparison of mid power region performances of controller developed with Rule-Based EMS and RL-Based EMS trained with DDPG

It should be pointed out that the most important contribution of DDPG-algorithm particularly based on the UDDS cycle is that it does not spend the charge as soon as possible, instead it always tends to be steadier in terms of state of charge. The situation provides the opportunity to exploit the DDPG-based EMS fuel cell efficiency. In other words as long SOC level is not significantly low, the EMS will attempt to run fuel cell on close to its maximum efficiency. Since the UDDS is an urban cycle, there are similar behaviours that could be observed throughout the cycle.

DQN-based EMS on the other in some cases manages to perform better than the DDPG-based EMS in terms of efficiency, however without extra effort on smoothing the decisions taken by the DQN-controller it is inevitable that the system will suffer from the discreteness of the actions. Even though it is true that with smaller step size, a smoother performance can be obtained, it hurts the speed of the training in great scale due to curse of dimensionality. This is also important since there are PI controllers in the DC-DC converter any sharp changes will cause overshoot when the power is drawn from both of the energy sources.

Finally the table 4.1 shows several results obtained throughout the UDDS cycle. Overall energy consumption is one of the most important indicator of the performance of the strategies. DDPG and DQN-based strategies achieve a lower energy consumption with respect to the rule based EMS. For a driving cycle lasting for 1369 seconds equivalent of approximately 23 minutes, DDPG-based and DQN-based strategies consumes around 39 and 31 Wh less energy respectively compared with the rule-based EMS. Average FC efficiency of DDPG and DQN-based strategies are almost equal and higher than that of rule based EMS. Another important result is that less energy consumption is achieved with almost equal SOC deviation.

Table 4.1: Results under UDDS Cycle

| Signals | BF | DDPG | RB | DQN | Units |
|---|---|---|---|---|---|
| Motor Supply Energy In | 1564 | 1564 | 1564 | 1564 | Wh |
| Battery Energy Out | -78.47 | -71 | -96.6 | -87.03 | Wh |
| Fuel Cell Energy Out | 1738 | 1731 | 1756 | 1746 | Wh |
| Fuel Cell Energy In | 2978 | 2972 | 3038 | 2997 | Wh |
| Fuel Economy | 84.3 | 84.58 | 82,66 | 83.75 | mpge |
| Fuel Economy | 83.3 | 83.58 | 81.68 | 82.8 | mile/kg(H2) |
| Total Energy Consumption | 2900 | 2902 | 2941 | 2910 | Wh |
| Total Battery Capacity Out | -27.39 | -43.89 | -23.33 | -39 | Ah |
| Battery Energy Consumption | 600.2 | 485.8 | 541.7 | 581.9 | Wh |
| Fuel Cell Charging Energy | -128.4 | -22.82 | -87.75 | -134.6 | Wh |
| Regen Braking Energy | -534.4 | -534.4 | -534.4 | -534.4 | Wh |
| Average FC Efficiency | 0.583 | 0.582 | 0.578 | 0.582 | |
| Final SOC | 50.33 | 50.52 | 50.28 | 50.5 | % |
| Minimum SOC | 49.75 | 49.85 | 49.94 | 49.67 | % |
| Maximum SOC | 51.35 | 51.38 | 51.33 | 51.37 | % |

### 4.3.2 HWFET Cycle

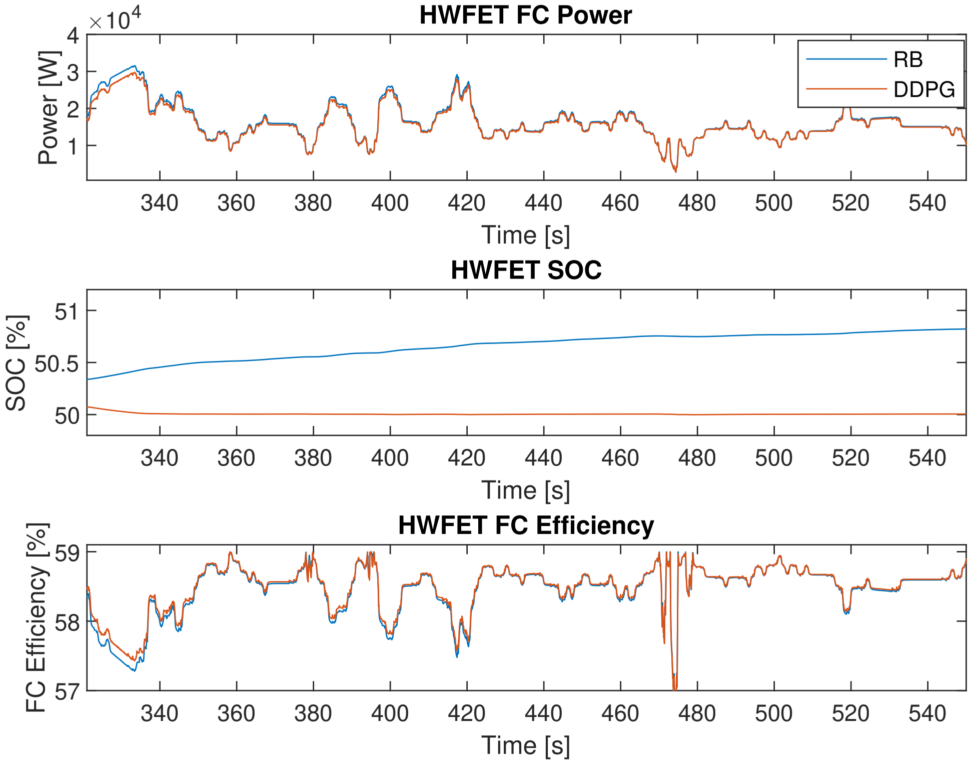

Figure 4.6: Comparison of near constant power region performances of controller developed with Rule-Based EMS and RL-Based EMS trained with DDPG

HWFET cycle represents the behaviour of vehicle in a highway. Thus it is a cycle where the shifts in power are not likely to be sudden. The significant portion of this cycle is the duration where the power is always required and no regenerative braking occurs as in figure 4.6. In this interval rule based EMS tries to charge the battery whereas no such attempt is observed for the DDPG and DQN-based strategies. Charging the battery in such a condition seems like a good opportunity, on the other hand, since the power demand exceeds the maximum efficiency region of the fuel cell, drawing more power from this source causes a less average fuel cell efficiency. Table 4.2 shows that energy consumption of DQN and DDPG-based strategies are almost equal and is 23 Wh less than that of rule-based EMS. The cycle takes 764 seconds that is approximately 13 minutes.

Table 4.2: Results under HWFET Cycle

| Signals | BF | DDPG | RB | DQN | Units |
|---|---|---|---|---|---|
| Motor Supply Energy In | 2491.3 | 2491.3 | 2491.3 | 2491.3 | Wh |
| Battery Energy Out | -78.99 | -73.1 | -96.89 | -77.67 | Wh |
| Fuel Cell Energy Out | 2636 | 2630 | 2653 | 2635 | Wh |
| Fuel Cell Energy In | 4512 | 4508 | 4555 | 4512 | Wh |
| Fuel Economy | 76.59 | 76.68 | 75.89 | 76.62 | mpge |
| Fuel Economy | 75.68 | 75.77 | 74.99 | 75.71 | mile/kg(H2) |
| Total Energy Consumption | 4433 | 4435 | 4458 | 4435 | Wh |
| Total Battery Capacity Out | -109.2 | -109.7 | -124.2 | -101.8 | Ah |
| Battery Energy Consumption | 195.2 | 88.39 | 154.5 | 230.4 | Wh |
| Fuel Cell Charging Energy | -117 | -4.67 | -94.12 | -151.3 | Wh |
| Regen Braking Energy | -156.9 | -156.9 | -156.9 | -156.9 | Wh |
| Average FC Efficiency | 0.584 | 0.583 | 0.582 | 0.583 | |
| Final SOC | 51.32 | 51.32 | 51.55 | 51.235 | % |
| Minimum SOC | 49.82 | 49.95 | 49.98 | 49.76 | % |
| Maximum SOC | 51.32 | 51.32 | 51.55 | 51.235 | % |

### 4.3.3 US06 Cycle

The drive cycles that were discussed had one common feature which is the power demand never exceeded 40 kW. However US06 is a very aggressive driving cycle literally pushing the vehicle to its limits from time to time even though it takes 600 seconds that is exactly 10 minutes. So far even though the maximum power capacities of the energy sources were never introduced for both of the DQN and DDPG-based algorithm particularly in the reward function, power demand in every time step of the cycle was satisfied. It should be stated that overshoots provoked by the PI controllers results in slight differences between the demand and the supply. For a commercial vehicle (Toyota Mirai) the maximum power the battery and the fuel cell can provide is around 40 and 114 kW respectively. So if a case where the power demand is more than 40 kW and somehow EMS decides to use the battery, occurs then no further action can be taken and the conclusion is that the power unit cannot satisfy the demand completely for that interval. Such a situation is observed for particularly the DQN-based strategy even though DDPG-based strategy is slightly affected. After obtaining power from regenerative braking for 25 seconds, SOC of the battery increases as shown in figure 4.7. It must also be stated that regenerative braking is limited and cannot exceed 20 kW. Since the all strategies does not control regenerative braking at all, the charge is captured totally. Reinforcement learning based strategies consider this situation as an opportunity and use the stored energy as soon as possible in order for the regenerative braking to be captured again without concern for high deviation from the target SOC. However the power demand in the next following seconds is highly aggressive and it goes from 0 to 80 kW in just 5 seconds. It can be observed that both DQN and DDPG attempts to use the battery, however as it tries to exceed the maximum power limit, power demand of the cycle is not satisfied. It also can be observed that DDPG-based strategy starts using the fuel cell only a second after the mismatch occurs. However DQN-based strategy fails to do so and misses the target for almost 8 seconds.

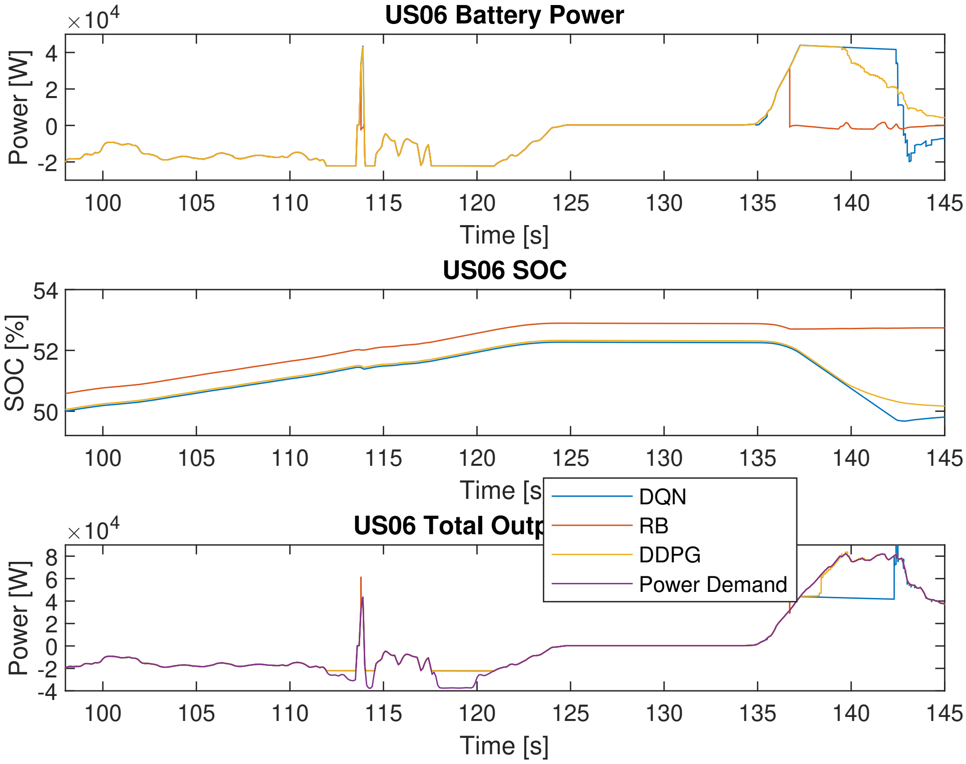

Figure 4.7: Comparison of cycle tracking performances of controller developed with RL-Based EMS trained with DDPG and DQN

Apart from that as shown in figure 4.8 average efficiency of the fuel cell of the DDPG-based strategy is higher than that of rule based EMS in an interval that requires high power. In that instance rule based EMS tends to charge the battery slightly. Since the efficiency of the fuel cell decreases in higher rates as the power demand increases after the peak efficiency point, such an action causes the efficiency to be lower.

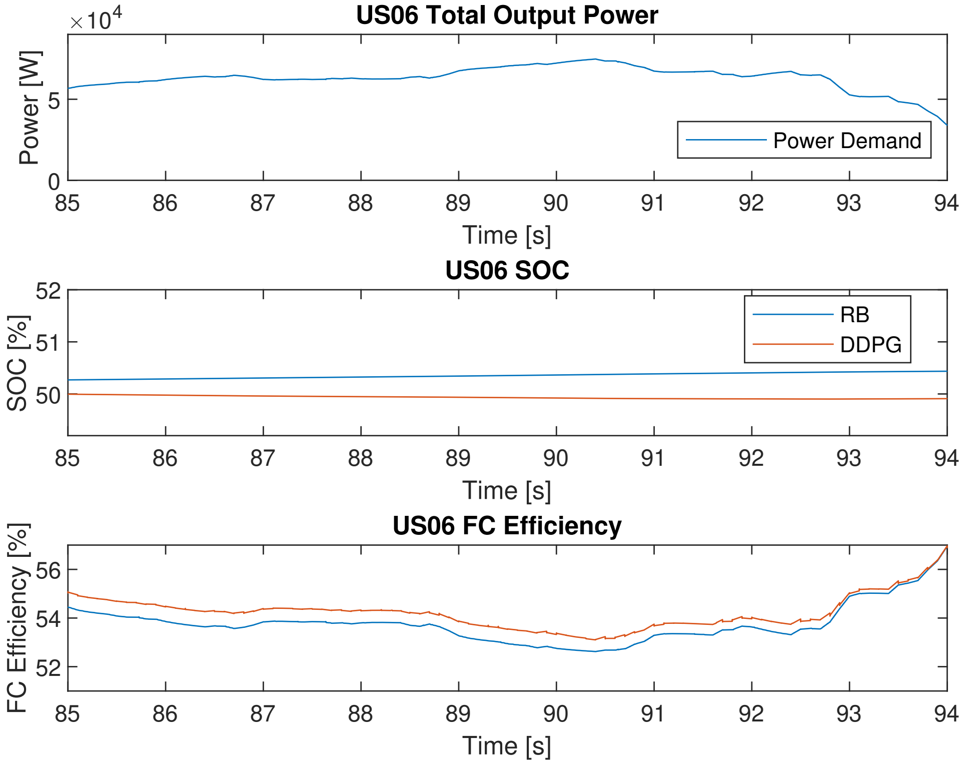

Figure 4.8: Comparison of high power region performances of controller developed with Rule-Based EMS and RL-Based EMS trained with DDPG

In the table 4.3 DDPG-based approach achieves less energy consumption than the rule based EMS by 140 kW and the DQN-based approach looks even better. However it must be taken into account it cannot meet the power demand for 10 seconds and misses approximately 60 Wh whereas DDPG based strategy only misses 3 Wh. Deviation from the target SOC is acceptable for all EMS. The general behaviour of RL-based approaches is that they try to utilize the regenerative braking energy so that fuel cell is running on high efficiency regions. DDPG-based approach do not charge the battery from fuel cell as much as the other methods, thus contributing the lifetime of a battery.

Table 4.3: Results under US06 Cycle

| Signals | BF | DDPG | RB | DQN | Units |
|---|---|---|---|---|---|
| Motor Supply Energy In | 2682.65 | 2682.65 | 2682.65 | 2682.65 | Wh |
| Battery Energy Out | -176.8 | -148 | -259 | -207.5 | Wh |
| Fuel Cell Energy Out | 2971 | 2968 | 3080 | 2984 | Wh |
| Fuel Cell Energy In | 5242 | 5232 | 5489 | 5271 | Wh |
| Fuel Economy | 51.48 | 51.57 | 49.16 | 51.2 | mpge |
| Fuel Economy | 50.87 | 50.96 | 48.57 | 50.59 | mile/kg($H_2$) |
| Total Energy Consumption | 5065 | 5084 | 5224 | 5063 | Wh |
| Total Battery Capacity Out | -82.5 | -86.44 | -284.2 | -62.26 | Ah |
| Battery Energy Consumption | 418.6 | 355.8 | 349.8 | 685.7 | Wh |
| Fuel Cell Charging Energy | -98.11 | -9.1 | -102.9 | -204.8 | Wh |
| Regen Braking Energy | -494.7 | -494.7 | -495.7 | -494.7 | Wh |
| Average FC Efficiency | 0.566 | 0.567 | 0.561 | 0.566 |  |
| Final SOC | 51 | 51.05 | 53.5 | 50.75 | % |
| Minimum SOC | 49.9 | 49.9 | 49.9 | 49.55 | % |
| Maximum SOC | 52.34 | 52.33 | 53.5 | 52.25 | % |

## Chapter 5

# Conclusion and Recommendations

The studies presented in this thesis are concluded here along with the recommendations for future work.

### 5.1 Concluding Remarks

This thesis focuses primarily on three areas: (**1**) modelling of fuel cell hybrid vehicle power unit, (**2**) the implementation of model free reinforcement learning algorithms DDPG and DQN, (**3**) comparison of the results under different drive cycles with rule based and optimization based approaches.

A fuel cell vehicle model is developed whose power unit including fuel cell, lithium-ion battery and DC-DC converter is modelled. The vehicle load model is obtained from Autonomie software which converts speed input to the power demand from converter all the way from the wheels to the electric motor. The fuel cell and the lithium-ion battery models produce outputs as the voltage ($V_{fc}$ and $V_{bat}$) when the currents are given as inputs. The voltage becomes the input for the DC-DC converters which sends the current signals ($i_{fc}$ and $i_{bat}$) into the energy sources depending on the difference between the source and bus voltage. How much current to be drawn from the sources is decided by the energy management system that outputs gains ($F_{bat}$ and $F_{fc}$) fed into the voltage current controller. Based on the gains and voltage differences the controller then sends the duty cycle of switches into the converter.

Implementing reinforcement learning algorithms in the energy management strategy is the main purpose of this study. In that aspect it is concluded that apart from Q-learning, DQN algorithm is the most common model-free reinforcement learning algorithm that is applied energy management systems in HEV, PHEV and FCHEV. The algorithm is implemented and after training the agent with drive cycles, the agent with the maximum reward is selected and its performance is investigated under different drive cycles. Then another algorithm named as DDPG is implemented and a similar procedure is followed. It is an algorithm with the advantage of having continuous action space that is highly promising particularly in the applications such as fuel cell hybrid vehicles. As a disadvantage it is bound take longer for the agent to be trained. In the training process instead of drive cycles, random step power inputs are used.

Once the agents are trained their performance under drive cycles are compared with the energy management strategies where rule-based and optimization-based approaches are applied. We based our rule based approach on the Autonomie software and the optimization based method is selected as the brute force search algorithm. The evaluation criterias are total energy consumption, the fuel cell efficiency and the SOC of the battery. UDDS, HWFET and US06 cycles are selected as they represent different type of driver behaviours.

It is found that in all of these drive cycles, energy management strategies based on DDPG and DQN are able to consume less energy than the rule based approach while achieving a similar SOC behaviour and small deviation in SOC. Their performances are slightly worse than BF method. DQN particularly in the HWFET cycle is very close to DDPG and slightly worse than DDPG in UDDS cycle, nevertheless it faces to the issue of not being able to track the power demand in US06 cycle for a very short time. It can be deduced that the DDPG algorithm has the potential to be used to find the global optimum whose learning process can continue in real-time applications.

### 5.2 Recommendations for Future Work

For the next steps including the realization of the fuel cell hybrid electric vehicle model and the energy management strategy, a few recommendations are listed below.

- The lithium-ion battery that is modelled with ECM should be improved in order to increase the accuracy.
- Fuel cell compressor model should be improved. In this study it is only implemented as a steady state however a model that will represent the slow

dynamics of the source is recommended.

- Global optimization methods will be able to give the best possible outcome. It is recommended that a dynamic programming algorithm should be implemented for each of the cycles individually. This will show the quality of general approaches in terms of finding a global optimum. In this the global optimization method that is applied is also a general approach. By general approaches it is meant that they are optimized not solely focused on one specific cycle or condition, instead they seek to find the best result under several different conditions.

- The network structure, hyper-parameter the states and the rewards are selected after a trial and error process. Thus they can always be improved. Indeed several cases should be created and the validation should be carried out by the comparison with the results found by dynamic programming.

- Real-time application potential is also recommended. Since an agent can still learn even after the initial validation, it makes learning techniques even more appealing.

# Bibliography

[1] "Comparison of Fuel Cell Technologies." [Online]. Available: https://www.energy.gov/eere/fuelcells/comparison-fuel-cell-technologies

[2] "Alternative Fuels Data Center: Fuel Cell Electric Vehicles." [Online]. Available: https://afdc.energy.gov/vehicles/fuel_cell.html

[3] I.-S. Sorlei, N. Bizon, P. Thounthong, M. Varlam, E. Carcadea, M. Culcer, M. Iliescu, and M. Raceanu, "Fuel cell electric vehicles—a brief review of current topologies and energy management strategies," *Energies*, vol. 14, no. 1, p. 252, 2021.

[4] J. Zhang, L. Zhang, F. Sun, and Z. Wang, "An overview on thermal safety issues of lithium-ion batteries for electric vehicle application," *Ieee Access*, vol. 6, pp. 23 848–23 863, 2018.

[5] C. Zhang, W. Allafi, Q. Dinh, P. Ascencio, and J. Marco, "Online estimation of battery equivalent circuit model parameters and state of charge using decoupled least squares technique," *Energy*, vol. 142, 10 2017.

[6] W. Nsour, T. Taa'mneh, O. Ayadi, and J. Al Asfar, "Design of stand-alone proton exchange membrane fuel cell hybrid system under amman climate," *Journal of Ecological Engineering*, vol. 20, no. 9, 2019.

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

[8] B. Mostafa, N. El-Attar, S. Abd-Elhafeez, and W. Awad, "Machine and deep learning approaches in genome: Review article," *Alfarama Journal of Basic Applied Sciences*, 08 2020.

[9] N. Sulaiman, M. Hannan, A. Mohamed, E. Majlan, and W. Wan Daud, "A review on energy management system for fuel cell hybrid electric vehicle: Issues and challenges," *Renewable and Sustainable Energy Reviews*, vol. 52, pp. 802–814, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032115007790

[10] N. Briguglio, L. Andaloro, M. Ferraro, and V. Antonucci, *Fuel Cell Hybrid Electric Vehicles*, 09 2011.

[11] Y. Miao, P. Hynan, A. von Jouanne, and A. Yokochi, "Current li-ion battery technologies in electric vehicles and opportunities for advancements," *Energies*, vol. 12, no. 6, 2019. [Online]. Available: https://www.mdpi.com/1996-1073/12/6/1074

[12] X. Li, L. Xu, J. Hua, X. Lin, L. Jianqiu, and M. Ouyang, "Power management strategy for vehicular-applied hybrid fuel cell/battery power system," *Journal of Power Sources*, vol. 191, pp. 542–549, 06 2009.

[13] N. J. Schouten, M. A. Salman, and N. A. Kheir, “Energy management strategies for parallel hybrid vehicles using fuzzy logic,” *Control Engineering Practice*, vol. 11, no. 2, pp. 171–177, 2003, automotive Systems. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967066102000722

[14] R. K. Ahluwalia, X. Wang, and A. Rousseau, “Fuel economy of hybrid fuel-cell vehicles,” *Journal of Power Sources*, vol. 152, pp. 233–244, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378775305000996

[15] G. Paganelli, T.-M. Guerra, S. Delprat, J.-J. Santin, M. Delhom, and E. Combes, “Simulation and assessment of power control strategies for a parallel hybrid car,” *Int. J. of Automobile Engineering*, vol. 214, pp. 705–717, 07 2000.

[16] C. Musardo, G. Rizzoni, Y. Guezennec, and B. Staccia, “A-ecms: An adaptive algorithm for hybrid electric vehicle energy management,” *European Journal of Control*, vol. 11, no. 4, pp. 509–524, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0947358005710487

[17] Y. Zhou, A. Ravey, and M.-C. Marion-Péra, “Real-time cost-minimization power-allocating strategy via model predictive control for fuel cell hybrid electric vehicles,” *Energy Conversion and Management*, vol. 229, p. 113721, 02 2021.

[18] H. Borhan, A. Vahidi, A. M. Phillips, M. L. Kuang, I. V. Kolmanovsky, and S. Di Cairano, “Mpc-based energy management of a power-split hybrid electric vehicle,” *IEEE Transactions on Control Systems Technology*, vol. 20, no. 3, pp. 593–603, 2012.

[19] C.-C. Lin, H. Peng, J. Grizzle, and J.-M. Kang, “Power management strategy for a parallel hybrid electric truck,” *IEEE Transactions on Control Systems Technology*, vol. 11, no. 6, pp. 839–849, 2003.

[20] R. Wang and S. M. Lukic, “Dynamic programming technique in hybrid electric vehicle optimization,” in *2012 IEEE International Electric Vehicle Conference*, 2012, pp. 1–8.

[21] 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,” *Journal of Power Sources*, vol. 407, pp. 92–104, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378775318311571

[22] E. T. Stephen and S. P. Boyd, “Finding ultimate limits of performance for hybrid electric vehicles,” *SAE Paper*, 2000.

[23] X. Hu, L. Johannesson Mårdh, N. Murgovski, and B. Egardt, “Longevity-conscious dimensioning and power management of a hybrid energy storage system for a fuel cell hybrid electric bus,” *Applied Energy*, vol. 137, 01 2014.

[24] F. Odeim, "Optimization of fuel cell hybrid vehicles," Ph.D. dissertation, May 2018. [Online]. Available: https://duepublico2.uni-due.de/receive/duepublico_mods_00046123

[25] W. Li, J. Ye, Y. Cui, N. Kim, S. W. Cha, and C. Zheng, "A speedy reinforcement learning-based energy management strategy for fuel cell hybrid vehicles considering fuel cell system lifetime," *International Journal of Precision Engineering and Manufacturing-Green Technology*, pp. 1–14. [Online]. Available: https://app.dimensions.ai/details/publication/pub.1140048378

[26] N. P. Reddy, D. Pasdeloup, M. K. Zadeh, and R. Skjetne, "An intelligent power and energy management system for fuel cell/battery hybrid electric vehicle using reinforcement learning," in *2019 IEEE Transportation Electrification Conference and Expo (ITEC)*, 2019, pp. 1–6.

[27] Y. F. Zhou, L. J. Huang, X. X. Sun, L. H. Li, and J. Lian, "A long-term energy management strategy for fuel cell electric vehicles using reinforcement learning," *Fuel Cells*, vol. 20, no. 6, pp. 753–761, 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/fuce.202000095

[28] L. Guo, Z. Li, and R. Outbib, "Reinforcement learning based energy management for fuel cell hybrid electric vehicles," in *IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society*, 2021, pp. 1–6.

[29] K. Deng, Y. Liu, D. Hai, H. Peng, L. Löwenstein, S. Pischinger, and K. Hameyer, "Deep reinforcement learning based energy management strategy of fuel cell hybrid railway vehicles considering fuel cell aging," *Energy Conversion and Management*, vol. 251, p. 115030, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0196890421012061

[30] P. Wu, J. Partridge, and R. Bucknall, "Cost-effective reinforcement learning energy management for plug-in hybrid fuel cell and battery ships," *Applied Energy*, vol. 275, p. 115258, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261920307704

[31] P. Wu, J. Partridge, E. Anderlini, Y. Liu, and R. Bucknall, "Near-optimal energy management for plug-in hybrid fuel cell and battery propulsion using deep reinforcement learning," *International Journal of Hydrogen Energy*, vol. 46, no. 80, pp. 40 022–40 040, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360319921037745

[32] P. Zhao, Y. Wang, N. Chang, Q. Zhu, and X. Lin, "A deep reinforcement learning framework for optimizing fuel economy of hybrid electric vehicles," in *2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2018, pp. 196–202.

[33] G. Du, Y. Zou, X. Zhang, T. Liu, J. Wu, and D. He, "Deep reinforcement learning based energy management for a hybrid electric vehicle," *Energy*, vol. 201, p. 117591, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544220306988

[34] Y. Zou, T. Liu, D. Liu, and F. Sun, "Reinforcement learning-based real-time energy management for a hybrid tracked vehicle," *Applied Energy*, vol. 171, pp. 372–382, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261916304081

[35] J. Wu, H. He, J. Peng, Y. Li, and Z. Li, "Continuous reinforcement learning of energy management with deep q network for a power split hybrid electric bus," *Applied Energy*, vol. 222, pp. 799–811, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261918304422

[36] T. Liu, Y. Zou, D. Liu, and F. Sun, "Reinforcement learning of adaptive energy management with transition probability for a hybrid electric tracked vehicle," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7837–7846, 2015.

[37] 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," *Applied Energy*, vol. 211, pp. 538–548, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261917316707

[38] C. Liu and Y. L. Murphey, "Power management for plug-in hybrid electric vehicles using reinforcement learning with trip information," in *2014 IEEE Transportation Electrification Conference and Expo (ITEC)*, 2014, pp. 1–6.

[39] H. Shen, Y. Zhang, J. Mao, Z. Yan, and L. Wu, "Energy management of hybrid uav based on reinforcement learning," *Electronics*, vol. 10, no. 16, p. 1929, 2021.

[40] R. Liessner, C. Schroer, A. M. Dietermann, and B. Bäker, "Deep reinforcement learning for advanced energy management of hybrid electric vehicles." in *ICAART (2)*, 2018, pp. 61–72.

[41] Y. Wu, H. Tan, J. Peng, H. Zhang, and H. He, "Deep reinforcement learning of energy management with continuous control strategy and traffic information for a series-parallel plug-in hybrid electric bus," *Applied energy*, vol. 247, pp. 454–466, 2019.

[42] G. L. Plett, *Battery management systems, Volume I: Battery modeling.* Artech House, 2015.

[43] J. Pukrushpan, A. Stefanopoulou, and H. Peng, "Modeling and control for pem fuel cell stack system," in *Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301)*, vol. 4, 2002, pp. 3117–3122 vol.4.

[44] Y.-X. Wang, K. Ou, and Y.-B. Kim, "Modeling and experimental validation of hybrid proton exchange membrane fuel cell/battery system for power management control," *International Journal of Hydrogen Energy*, vol. 40, no. 35, pp. 11 713–11 721, 2015.

[45] W. Jiang and B. Fahimi, "Active current sharing and source management in fuel cell–battery hybrid power system," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 2, pp. 752–761, 2009.

[46] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, “Playing atari with deep reinforcement learning,” *arXiv preprint arXiv:1312.5602*, 2013.

[47] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous control with deep reinforcement learning,” *arXiv preprint arXiv:1509.02971*, 2015.

[48] Argonne National Laboratory, “Autonomie.” [Online]. Available: https://www.autonomie.net

[49] X. Yuan, C. Zhang, G. Hong, X. Huang, and L. Li, “Method for evaluating the real-world driving energy consumptions of electric vehicles,” *Energy*, vol. 141, pp. 1955–1968, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544217319928