# Designing Intelligent Energy Management and Cost-effective Data Acquisition for Vehicular Solar Idle Reduction Systems

by

Seyedmohammad Hosseini

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2019

## AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

# Abstract

In this study, an innovative energy management system (EMS) employing the promising reinforcement learning (RL) method is proposed. The EMS intelligently administrates the power flow between the main battery which is fed through the alternator and a solar-powered auxiliary battery which is used for the vehicle idle time reduction via providing energy for auxiliary loads which force the engine to be running, although the service vehicle is stopped. RL, which is an exquisite artificial intelligence technique, endeavors to offer a sub-optimal performance for this control problem compared to the really time consuming Dynamic Programming approach, which determines the optimal solution through exhaustive search.

A service vehicle is modeled in the Matlab/Simulink environment. Different parts of the model are described in detail, and the dynamics of the considered vehicle are discussed. The simulation results express a better functionality compared to an existing rule-based controller and the idled engine case, turning the proposed RL-based EMS into an effective method for implementation in vehicular solar idle reduction (SIR) systems. Double DQN is also utilized to come up with the continuous observation space. The results are showing that Deep-RL can be a promising method in control tasks like the EMS of vehicular systems.

Furthermore, a cost-effective and efficient data acquisition system is designed, tested, and implemented using the renowned Raspberry Pi board, and some sensors to collect voltage, current, and temperature data. The required electrical enclosures are also designed to keep the whole package safe. The validation of the system results is done and the process is discussed in detail. This data acquisition system can be employed to read the required information from vehicle and its loads, in order that the intelligent EMS system can wisely decide which action to take in a real-time manner.

# Acknowledgements

I would like to express my gratitude to my supervisors, Dr. Nasser L. Azad and Dr. John Wen, for letting me be a member of the SHEVS lab. They made the research an enjoyable experience for me by their encouragement, support and guidance during my graduate studies at the University of Waterloo.

I also wish to thank my colleagues involved in the Solar Idle Reduction project, without whose help it would have been impossible to fulfill my duties. Special thanks go to Dr. Mehrdad M. Majdabadi, and Dr. Ebrahim Moradi for their valuable insights throughout my research and providing me with their Simulink model of the system. I am deeply indebted to the fellows at the Canadian Solar Inc. and the City of Waterloo for their support and funding this research. Also I want to acknowledge the NSERC Canada for supporting the project.

Finally, I would like to express my appreciation to my beloved wife and our parents and siblings for their constant and loving motivation. Without their support, I would not have been able to achieve this stage of my life.

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1
# Introduction

## 1.1 Motivation and Challenges

No one can camouflage the fact that our beloved planet, Earth, is getting warmer. Not only it has been addressed widely in the news, but also we feel it year by year [1]. Many of us remember the days with a more pleasant weather, and it was not farther away than even 20 years ago. We are facing the Global Warming and we cannot stay on the sidelines and just watch it happen. Global warming is resulting in a destructive phenomenon called climate change. Climate change is mostly brought about as a result of our negligence and increasing carbon footprint. It concurrently results in natural disasters such as rising seas, heavier precipitation and flooding. To name more of these disasters, we can count more destructive hurricanes, or extreme weather conditions [2].

One of the main causes for this increasing carbon footprint is greenhouse gas (GHG) emissions by vehicles. It is good to know that burning just one liter of gasoline, will produce about 2.3 kg of carbon dioxide [3]. This accumulating carbon dioxide can trap more and more heat in the earth's atmosphere and end up altering the climate. In this project we aimed at emergency and operational vehicles, or more generally service vehicles which need to idle for a long time during their on-site operation. As an example Police cars are idled 60% of the time during their normal operation and use 21% of their total fuel while parked. While the engine provides 250 hp, together all of the accessories need less than 2 hp [4].

Idling of the vehicle refers to the case in which the engine is running (engine ON), but the vehicle is not moving. The engine in this case provides electrical power for the auxiliary units such as the lightings, medical units, lifts, communication means, and so on which require power to run. There exist some technologies and practices that are referred to as idle reduction, or anti-idling [5]. These solutions are taken into account to decrease the idling time of the vehicle or its effects. Different idle reduction strategies have been extensively studied and among all of them, using the clean renewable energy of sun has shown environmental benefits as well as promising results [6]. The sun transfers a huge amount of energy, many times greater than all we need for any reason, and today solar panels make it possible for us to make use of it by altering it to the electricity. Solar Idle Reduction (SIR) is the name we give to this technique.

To utilize the electrical power achieved from the solar panels and feed the auxiliary loads of the vehicle with it, we need to design an energy management system (EMS). It takes care of the power flow in different electrical parts of the vehicle. The EMS is the brain of the system which controls the connectivity and behavior of auxiliary systems along with internal vehicle systems. Various optimal controllers could be used for this project, such as Model Predictive Controllers [7], but AI methods have shown promising results in the case of designing EMS and have been widely studied recently [8-12]. Reinforcement Learning (RL) is one of these AI-based intelligent techniques which has been employed for developing the EMS for this project. It is capable of generating a look-up table for the system by which the power splitting between different vehicle parts is controlled.

Another aspect of the project is utilizing the dynamics of the vehicle in order to minimize its fuel consumption as the goal for the optimal controller. It requires a huge amount of data which cannot be represented as a look-up table. Therefore, the next step was to develop a deep network handling all the required data and making a decision based on vehicle dynamics. This method is called Deep Reinforcement Learning (DRL) which has recently been shown to be a game changing method for tasks requiring exploration of different criteria to find the optimum solution to the problem [13].

To collect the data from the sensors installed on vehicles, having access to a data acquisition (DAQ) system is mandatory. There are many commercial DAQ systems available in the market, but the problem is that they are expensive and most of the time offer some applications which are not required for a specific task. Some of these DAQ systems are found in [14] and [15]. To address these problems and overcome the necessity of having a DAQ system, a new cost-effective approach has been taken and a customized simple DAQ device has been developed and presented in this thesis.

## 1.2 Problem Statement and Proposed Approach

To address the problems caused by idling the vehicle such as increasing GHG emissions which may result in affecting global warming, a vehicular solar idle reduction (SIR) system has been installed on some service vehicles. The already installed SIR systems do not have a controller and only have a voltage-enabled battery separator switch. Also they have an expensive data acquisition system which is installed for research purposes. The problem with this system is the lack of an intelligent energy management system and the costly DAQ system which is barrier for commercializing it.

Motivated by these facts, this research focuses on the following objectives:

- Design and development of various EMS strategies for controlling the energy flow between main and auxiliary batteries of service vehicles considering minimization of the fuel consumption
- Providing a new cost-effective DAQ system aiming at reducing the cost of commercialization to replace the costly research-based DAQ system

To achieve the aforementioned objectives, the vehicle model for a 2011 FORD F550 4x2 SuperCab incorporated with the SIR system is modified and different generations of an intelligent EMS based on the promising RL method are developed and evaluated using MATLAB and Python. The input data to the model are collected from various standard and real drive cycles. Also a goal-oriented and cost-effective DAQ system as well as the required sensors for collecting the mandatory data is developed and implemented. It is then calibrated and can replace the expensive off the shelf DAQ systems.

## 1.3 Thesis Organization

After having a brief discussion about the necessity and goals of this research as an introduction the rest of this thesis is organized as follows: Chapter 2 briefs the background and includes a literature review on different EMS strategies, idling reduction methods, various types of RL and Deep-RL, and DAQ systems. Chapter 3 presents some details about the vehicle and SIR system model and also elaborates on energy management architecture and the controller design. Three different energy management scenarios are included in this chapter. Investigation of the data gathering process and implementation of the new DAQ system is done in chapter 4. Finally, chapter 5 provides the concluding remarks and some suggestions for future work.

# Chapter 2
# Background and Literature Review

This chapter is dedicated to elaborating on what leads to defining the SIR project and the required background to come up with the project. It also explains the related literature which helps understanding the whole scenario. This chapter is divided into six sections each of which aiming at giving some information to make the next chapters easier to follow. The first part introduces the global warming issue and GHG emissions for which vehicles are a main cause. It also briefs about the effects of it on our environment and then emphasizes the importance of doing something to control it. Then the chapter follows with idling reduction methods which are employed nowadays in order to eliminate the unnecessary GHG emission of the vehicles while they are idling for a purpose. The next section is assigned to review the literature on the energy management strategies in different (mostly vehicular) systems and various controller types which have been used for this goal recently. A brief yet rich background on the concept and scope of RL along with a comparison to other optimization and learning methods is presented in the section afterwards. It is accompanied with a background on various deep learning methods, especially the promising Deep-RL approaches for different cases. The next part is then reviewing the existing data acquisition systems and various data we can get from a vehicle utilizing different means. This chapter is concluded with a summary section which paves the way for the next chapters.

## 2.1 GHG Emissions and Environmental Issues

It is clear to everyone that our world is getting warmer and warmer, and this change is not as pleasant as what everybody expects in the cold winters of Canada. “Global Warming” is a phenomenon which challenges all of our lives and its intensive effects are making huge troubles for human beings and other creatures. Lots of people are dying because of the hot summer days which are not reported earlier. Lots of disasters are happening every day and one of the main causes for that is the global warming issue [16]. Some of these changes which occur as a result of climate3 change are rising sea levels, heavier precipitation and flooding, sudden changes in the weather, destructive hurricanes and so on.

The greenhouse effect refers to the situation where the temperature of a planet's surface increases to a higher value because of the atmosphere created by the radiatively active gases surrounding the planet. These gases help trap more heat in the atmosphere. Without the aforementioned radiative gases creating the atmosphere, the surface temperature would be lower. These radiatively active gases are called greenhouse gases (GHG). They include but are not limited to water vapor (H2O), carbon dioxide (CO2), methane (CH4), and ozone (O3). Sorting out these gases based on their contribution to the greenhouse effect on Earth, results in the following [17]:

- Water vapor: 36 ~ 70%
- Carbon dioxide: 9 ~ 26%
- Methane: 4 ~ 9%
- Ozone: 3 ~ 7%

This is clear that carbon dioxide can be a main cause which is a byproduct of burning fossil fuels like diesel or gasoline. Also there are other types of gases which contribute to the greenhouse effect and are produced as a result of fuel burning. Some of them are nitrogen oxides and sulphur oxides which are toxic gases and cause environmental problems and some illnesses [18]. GHG emission is dangerous after all and everybody needs to play his role in fighting it. It is a main cause of global warming and its consequential disasters which is mainly originated by human industrial activities and modern life style.

Results of a study by the United States (U.S.) Department of Transportation reveal that almost 26% of the total U.S. GHG emissions were generated by transportation in 2014. This puts transportation in the second place for largest GHG producing sources [19] in the U.S. Vehicles are polluting the air significantly and the trend shows that even with better engineered cars existing, still people eagerly look for SUVs which consume more fuel. In the U.S. market almost only 3 percent of the new vehicles is devoted to hybrid and electric vehicles from 1990 to 2014. This study also shows that the share of the market for pickup trucks and SUVs is increased which translates into more fuel consumption. It can be because of the low prices for gasoline and diesel fuel [19].

Now that the pace for replacing the conventional vehicles with hybrid and electrical ones is slow, we need to think about the scenarios in which the conventional vehicles, especially the ones with higher fuel consumption can contribute to slowing down the increment of GHG emissions. There are also lots of policies putting a limit on GHG emissions for the transport section. Therefore, GHG reduction seems to be a mandatory task with designated constraints and plans.

In the European Union the transportation section emits 30% of the total energy-related carbon dioxide emissions roughly. In 2011 the European Commission (EC) set the objective of decreasing direct carbon dioxide emissions by 60% in 2050 relative to its levels on 1990 which was 50% below current levels. In July 2016 also confirmed the same emission reduction goal [20]. In California some more aggressive GHG reduction objectives are considered because climate change can have a large impact on its economy. The transportation part is the biggest generator of GHG which made up over 40% of California's total emission in 2006. California has aimed at reducing the GHG emission levels to what it was on 1990 by 2020, and to 80% below 1990 levels by the year of 2050 [21]. About another countries in the world, including Canada, an agreement of reducing GHG emissions to make the global temperature stable at $1.5°C$ above the pre-industrial levels has been signed by 190 nations. Canada is going to meet the GHG emission reduction by 30% below 2005 levels by the year of 2030. Canada has also set an 80% reduction goal below 2005 levels which wants to achieve by 2050 [22]. Achieving all of these goals world-wide requires fundamental changes in the automotive industry as well as other energy related parts.

One of the states that vehicles burn the fuel but do not use it for the transportation purpose is the case they are idling. Idling refers to the situation in which the engine is on and consuming the fuel, but the vehicle does not move and is stopped in a traffic jam or for an on-site operation. When the vehicle is idled, it still pollutes the air. For instance if the vehicle is idled for more than 10 seconds, it burns more fuel and emits more carbon dioxide than restarting the car [23]. The City of Waterloo's anti-idling by-law limits unnecessary idling of vehicles to three minutes or less. However, some emergency vehicles are exempted. As an instance police cars are idled 60% of the time during their normal operation and use 21% of their fuel while parked. While the engine provides 250hp, together all of the accessories need less than 2hp, but still the engine has to work in order to keep the auxiliary loads running [24].

To reduce the emissions incurred because of vehicle idling, lots of efforts have been made and a bunch of strategies are developed. The next part in this chapter is dedicated to introducing some of these idling reduction technologies. It also gives some details about the method has been used for this research.

## 2.2 Idling Reduction Methods

The term “Idle reduction” refers to technologies and practices aiming at minimizing the accumulated time drivers idle their vehicles’ engines. There are some behavioral strategies which involve driver/operator training and financial incentives for idling reduction [25]. Along with these behavioral strategies, some technologies are specified for the purpose of using that vehicle which depend on the vehicle type. Passenger vehicles, operational and service vehicles, and vehicles with specific working scenarios have various idling times and also they have different accessories. Various auxiliary loads on diverse types of vehicles make the idling fuel consumption and GHG emissions different for them. Generally we can classify the idling reduction technologies and strategies into some distinct subgroups [26]. For the special case of service vehicles which mostly involves heavy duty trucks, we discuss some of the technologies and devices that have three main characteristics: (a) They are installed on a vehicle or at a location. (b) They reduce the unnecessary idling of the engine. (c) They provide special services and support the auxiliary loads that would otherwise need the main engine while the vehicle is not moving [25].

The methods which provide fuel savings and/or GHG emissions reducing benefits based on their determined designed application are classified by U.S. Environmental Protection Agency (EPA) into the following groups [25]:

### 2.2.1 Auxiliary Power Units and Generation Sets

Auxiliary power units (APUs) are devices installed on a vehicle that are used for purposes other than vehicle propulsion. They are basically used for auxiliary loads such as seat heaters, air conditioner, power steering, emergency lights and so on [27]. An example of an APU or generator set can be the use of auxiliary diesel engines for vehicle accessories other than propulsion. Another type of the APUs is a fuel cell APU which has the same responsibility and can be found in [28]. Fuel cell APUs can theoretically eliminate nearly all emissions. Heating, ventilation, and air conditioning (HVAC) system is the main auxiliary consumer of the provided energy in most vehicles, especially service trucks and buses. APUs are capable of supporting all of them as long as the truck has fuel. Their problem is that again they consume fuel, however it is way lower than idling the engine [29].

### 2.2.2 Fuel Operated Heaters

Fuel operated heaters or direct fired heaters are small lightweight devices that directly burn fuel from the main engine or a separate reservoir and provide heat only for the vehicle cab or truck engine. Their problem is that they are still using fuel for the purpose of heating and have exhaust emissions. Their fuel consumption can be a gallon per day which is really lower than idling the vehicle. They are easy to install and use, but they have limited functionality and cannot be used for cooling the vehicle [30]. This type of the idling reduction technology has been addressed in comparison with other types in [18].

### 2.2.3 Battery Air Conditioning Systems

Battery air condition system usually integrates with a fuel operated heater in order to power an independent cooling system. It deals with the cabin climate control to prevent idling the engine while the truck is parked or stopped. It may not provide enough cooling capacity when the temperature is too hot. They are powered by a set of batteries which can be charged when the vehicle is operating or from the shore power (connecting to an external electrical power source). This system is easy to operate and has zero emission while working. By the way, they have limited run time and are not suitable for hot temperatures. Also the battery needs to be replaced after some time [31].

### 2.2.4 Thermal Storage Systems

A thermal storage system collects heat energy when a truck is driven and uses it for air conditioning. Authors in [32] have addressed this type of idling reduction technology and developed a thermal storage system for electric vehicle cabin heating which its concepts can be adopted for other vehicles as well. They have used an advance phase change material which melts when the battery is charging and provides heat for the cabin while the vehicle is parked. This is not as functional as diesel APUs.

### 2.2.5 Truck Stop Electrification

Truck stop electrification refers to making the truck parking spaces capable of charging the truck battery and also providing electricity-powered devices for the driver comfort. These devices provide air conditioning and auxiliary power for the vehicle and prevent it from idling. Therefore, these stops are

called electrified parking spaces which operate independently of the vehicle's engine [25]. This solution is not well developed and there are not many of these electrified parking spaces available.

### 2.2.6 Automatic Engine Start/Stop

Automatic engine start/stop refers to a technology which based on some inputs and assessments, decides whether the engine should remain off or be switched on to provide a variety of features. This process is done without driver's attention and with a set of inputs. The controller determines whether it is safe to start the engine without any controls from anybody. Some of the features are maintaining the battery's state of charge and engine coolant temperature. This method can be adopted in combination with other technologies and does not require many additional components. By the way, it sometimes requires idling the main engine [33].

### 2.2.7 Solar Panels

The main use of solar panels for trucks is to support the HVAC system and some auxiliary loads without having batteries. The solar panels employed for the truck industry are mostly flexible panels which are lightweight and efficient. Fleet managers are attracted to utilize the green energy of the sun more and more these days. Solar idle reduction is a method that can introduce fuel savings as well as GHG emission reduction which helps protecting the environment. It can minimize the vehicle downtime and emergency jump-starts. It extends battery life by helping the main battery with maintaining a proper state of charge. It keeps the batteries charged especially when the engine is turned off. The downside of it relates to the cost of the panels and installation process in which the shape of the truck roof (especially bumps that make the installation hard even with flexible panels) can be a limiting factor in mounting solar panels. Also the amount of sun radiation and its exposure need to be maximized for the panels in order to get good results. If the weather is cloudy, it limits the sun exposure and therefore the fuel savings. The panels are required to be mostly clean of any dust and snow because these reduce the efficacy of energy conversion [34].

A good idea to get the most out of the solar panels is to provide an auxiliary battery which is capable of being charged with the solar panels as well as the vehicle alternator. This way the energy from the sun can be stored on the battery even if the auxiliary loads are not in use which provides energy for their utilization period. Also because of being connected to the alternator the reliability of such a system

is high and the loads may be covered, regardless of the green energy of the sun's availability. Although if the energy comes from the alternator it is not considered a fully green solution. The reliability it offers is an awesome feature which compensates for this downside. The SIR project is focusing on this technology accompanied with an auxiliary battery. The following chapters will elaborate on the model details and implementation of it, as well as introducing an intelligent controller for the purpose of energy management which rules on how to provide the electrical power for main and auxiliary loads and when to connect/disconnect the alternator and main battery to/from auxiliary battery.

## 2.3 Energy Management Strategies

The energy management system (EMS) is the brain of any embedded system determining the energy split between various electrically powered components. It needs to monitor, control and optimize the performance of the energy transmission in the system [35]. Energy management strategies and software are widely used in large industries like electricity generation and transmission, as well as smaller applications such as Microgrids. It is also used in smart buildings and vehicles to optimize the energy consumption/generation [7-12], [16], [35-46]. EMS can be theoretically classified into two totally distinguishable categories based on the controls concept: (a) Low-level control which is conducted through power electronics devices, and (b) High-level or supervisory control which refers to overall control of the system using logics and programming. These programs are then translated into understandable commands for low-level controllers [47]. Mathematically we can subcategorize the supervisory control for the EMS into two major groups: (a) Rule-based non-predictive ways, and (b) optimization-based predictive methods [48]. Both of these approaches have their own benefits and shortcomings making them interesting for researchers. Also there are some approaches trying to combine the advantages of each method and reduce their shortcomings and are widely noticed by researchers.

Rule-based approaches are more robust and real-time implementable [35]. These two benefits make them a good choice for practical and experimental works in which optimality is not an issue. The implementation of rule-based controllers is usually easy and they are simply designed. Of course, if someone wants a controller which acts better and works more efficiently, it is mandatory to study the system deeply and generate various conditional rules for the controller in order to make it near-optimal. It is obvious that such a controller that does not use the data from the whole data set and does not involve future information, cannot be optimum and is not able to beat an optimal controller in its

performance for the similar situation. Rule-based method can be implemented in deterministic or fuzzy scenarios which are investigated in different fields vastly [36], [41], [45-52].

Optimization-based approaches on the other hand can give us the optimal solution by minimizing a cost function. The problem with them is that they are time-consuming and need to search the whole state space to find the optimum. Therefore, they are not real-time implementable and are mostly used for research purposes to find the best scenario to compare with the real case [35], [47]. Optimization-based methods are then classified into offline and prediction-based scenarios. Various optimal control approaches have been utilized in the literature to empower the EMS, such as Model Predictive Control (MPC) and its variants, Dynamic Programming (DP), and Stochastic DP. Clearly real-time implementable approaches are more fruitful for the practical applications. Hence, offline controllers are not a good choice for implementation. Some prediction-based algorithms that can be accompanied by a real-time implementable method are attracting more and more researchers in a way that they can be used in industry as well [48]. References [42], [49], [53-54] reviewed the optimization-based approach in an informative manner.

Some new approaches which are trying to achieve the benefits of both rule-based and optimization-based controllers are also attracting attention. The promising artificial intelligence (AI) methods have been employed and some machine learning-based solutions are implemented for the purpose of enhancing the EMS technologies [55]. Reinforcement Learning (RL) which is deliberately discussed in the next part of this chapter, has been used for designing and implementing intelligent EMS systems for some applications such as different types of hybrid vehicles. The results of the studied literature demonstrate a promising future for RL-based EMS approaches [56-60]. The main goal of this research is to introduce an intelligent energy management strategy based on the RL method which is sub-optimal and can be used in a real-time manner, since it is going to manage the power flow between the main and auxiliary battery of the SIR-enabled vehicles. Figure 2.1 depicts a summary of different EMS categories and shows what the purpose of this research in terms of the EMS part is.

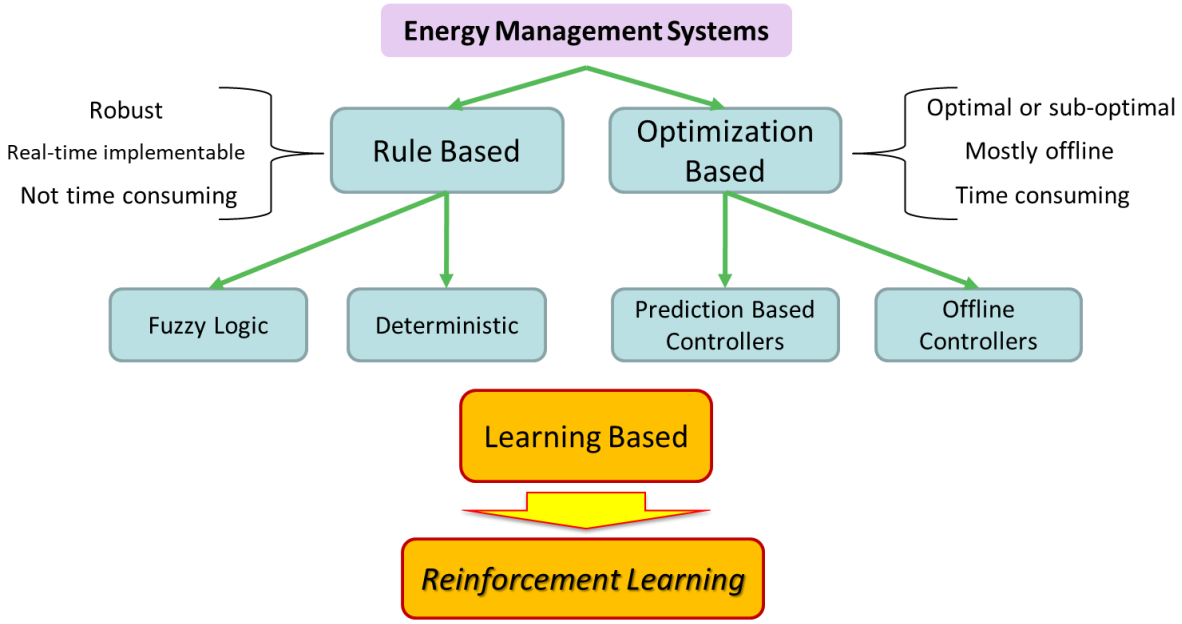


**Figure 2.1: Classification of different EMS strategies**

As it is mentioned earlier, EMS systems are widely studied for hybrid vehicles. Although there are not many articles in the literature addressing the special case of SIR-enabled vehicles or even other idling reduction methods, some sort of different EMS systems are studied by researchers for hybrid vehicles [7-8], [11-12], [16], [28], [36-43]. The idea which is adopted from these studies paves the way for designing the new EMS for this research.

## 2.4 RL and Deep Learning Background

RL is one of the most famous AI and Machine Learning (ML) techniques which has been around since 1980s [61]. It has been used widely in designing adversarial games like backgammon (TD-Gammon), and lots of trial-and-error problems [61]. For some years it was not used widely among researchers, but again in 2010s it gained the attention of professional researchers all around the world as it was showing eye-catching results when combined with deep learning methods. It has been investigated by many computer scientists and many good features are developed for it. RL approaches also have been employed for different engineering problems which involve trial-and-error behaviors. This section tries to briefly discuss machine learning methods and elaborate more on the RL approach. The RL framework is investigated and some mathematics which is the backbone of the algorithm is included.

Furthermore, some brief notes about Deep Learning and how it is combined with RL framework is presented and then followed by various Deep-RL methods compared briefly to each other. Select scenarios to go forward with for this research based on their capabilities are covered to wrap up this section of the background and literature review.

### 2.4.1 What is RL and Why to Use It?

RL refers to a method of learning which can be seen in many animals and is basically done through trial-and-error. Just assume you have a dog and want to teach it to catch the ball and bring it back to you. You can offer the dog a reward whenever it catches the ball and bring it to you successfully. RL method has a similar ideology and is shaped around rewards and punishments. RL is somehow different from two other categories of ML, Supervised Learning and Unsupervised Learning. In supervised learning algorithms, there is a training labeled data set in which the outputs are assigned to specific inputs and the responsibility of the Supervised Learning agent is to build a mathematical model based on the relations it can dig out between the inputs and corresponding outputs [62]. Therefore, there is no concept of rewarding in it and everything is clear except for the mathematical relationship model. Unsupervised Learning is distinguished from Supervised Learning in such a way that it does not have access to the labelled data which means there are no specified outputs at hand for corresponding inputs. It only has some interpretations at the beginning and tries to find the mathematical relationship model based on identifying some commonalities [62]. Also there is Semi-Supervised Learning in which some labelled data are missing and has to do something in between two previously mentioned methods.

On the contrary, Reinforcement Learning has a completely different approach for solving the problem and requires a reward function with which a rational agent tries to find the most efficient solution to the problem by maximizing a notion of the cumulative reward. It is completely distinct from having labels, annotations, classifications, or interpretations for the data. Most of the RL techniques do not consider a prior knowledge of the mathematical model and the rational agent has to interact with its environment to get information from it [63]. Figure 2.2, Figure 2.3, and Figure 2.4 from [64] summarize the scope of Supervised, Unsupervised, and Reinforcement Learning respectively and demonstrate their basic difference.

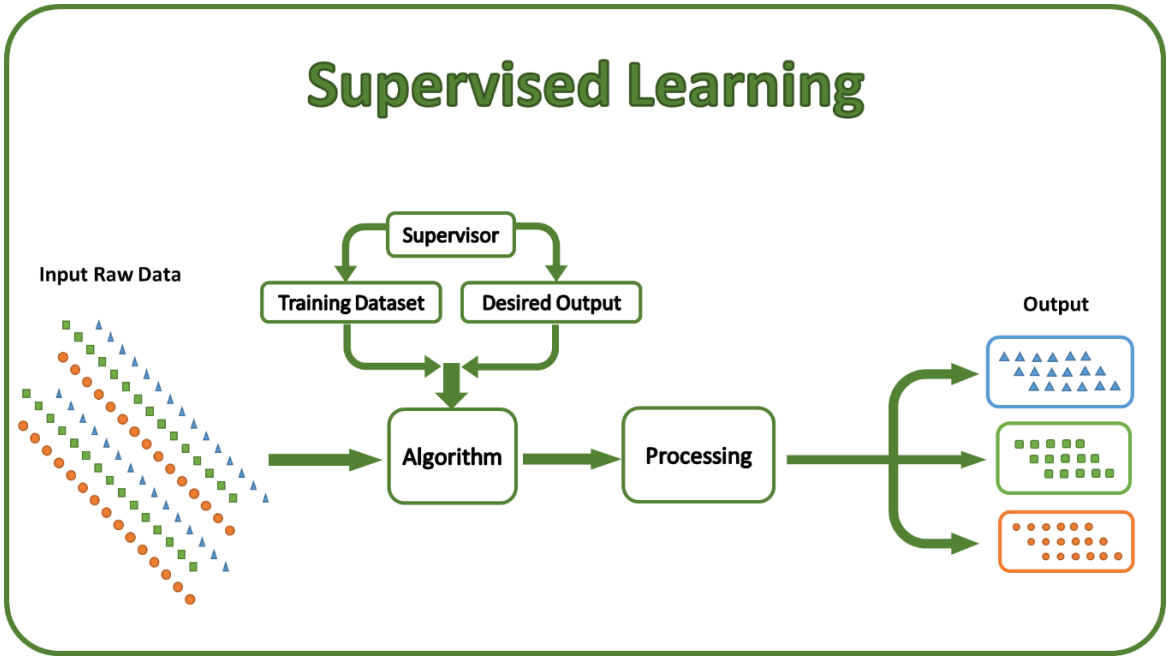


**Figure 2.2: Supervised Learning framework [64]**

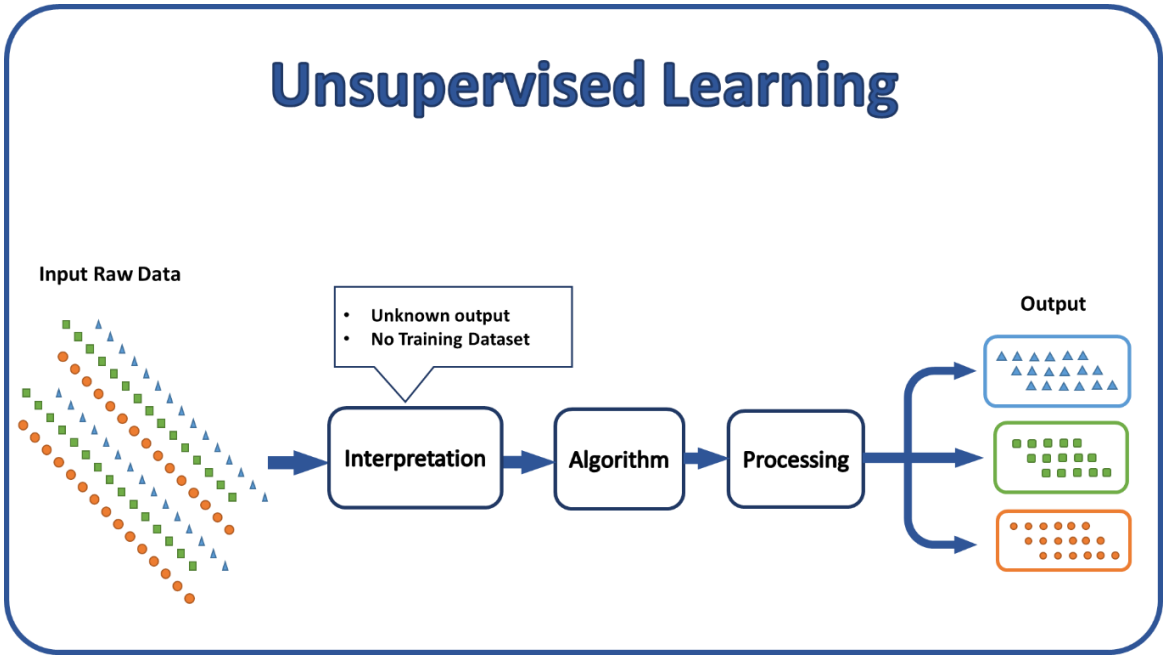


**Figure 2.3: Unsupervised Learning framework [64]**

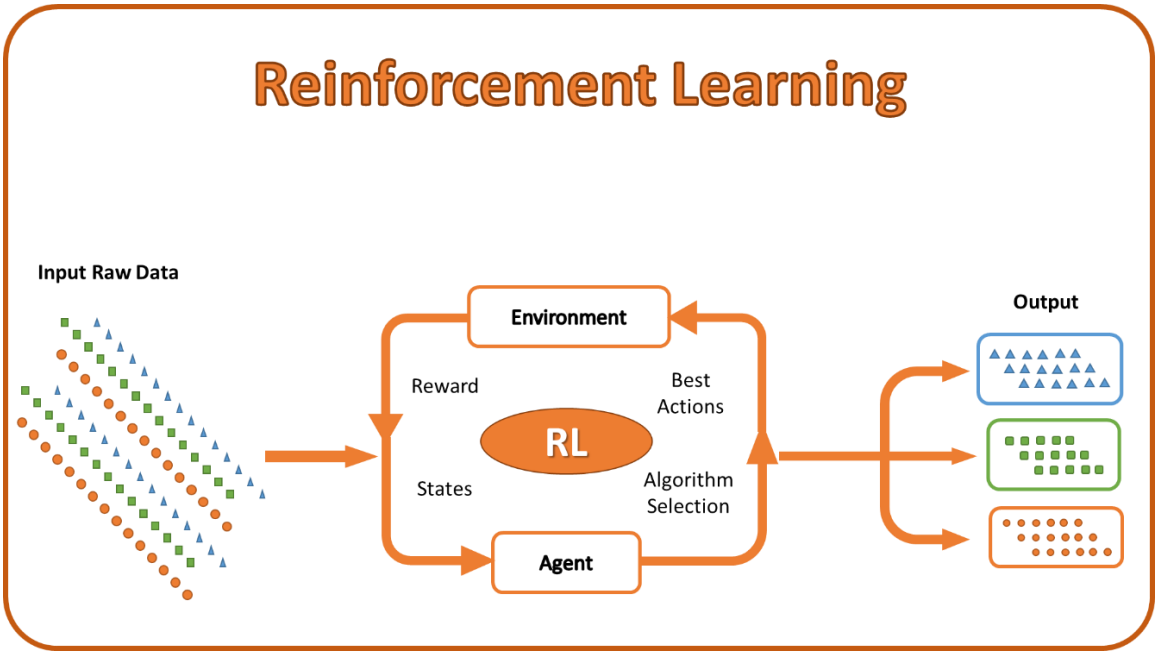


**Figure 2.4: Reinforcement Learning framework [64]**

The RL framework consists of a rational agent which is in contact with its environment in a way that senses the situations and receives some states from it and influence the environment by its taken actions. The learning process is performed through finding a map between the set of states and actions which is evaluated by allocating some notion of a reward function and generating value functions or using other operators [35]. The agent ought to explore new actions and find out which series of actions result in a higher cumulative reward for the given states [61].

Some of the most important characteristics of RL are trail-and-error feature and delayed rewards. Delayed reward refers to the reward function of a case in which actions happening at a moment can affect the situation at a later time instead of its immediate successor. Exploration and Exploitation dilemma is also a determinative scenario. The agent needs to decide whether to choose an action which maximizes the immediate reward and exploit a previously-evaluated action, or to explore newly available actions which might not produce a perfect immediate reward but introduce a chance of visiting new states that might result in a higher cumulative reward after some steps. Both of these tools are required for having a smart agent to dig the optimal solution out. The agent should exploit greedily what it thinks is fruitful based on its experience, and needs to explore the available search space to

crave for more beneficial actions which might be skipped before. Another important characteristic of RL is that the states can be partially observable which requires more sophisticated method for solving the problem. Also the RL framework allows for a life-long learning with the concept of discounting factor. Discount factor ($\gamma$) will be more discussed in the next subsection [63].

The environment of the RL framework is commonly represented as a Markov Decision Process (MDP) in which the outcome of situations are partly stochastic and partly controlled by a decision maker (rational agent). In MDP architecture, state transition probabilities are known and the value of $P_a(s, s')$ shows with which probability the decision maker takes action $a$ and goes from state $s$ to state $s'$. MDPs similarly look for a policy $\pi(s)$ that maximize the future expected reward [61]. If the state transition probabilities or rewards are not known beforehand, the problem of MDP turns into an RL problem and has its own characteristics [65]. Figure 2.5 depicts the key elements and framework of RL which is similar to the MDP, however it lacks the state transition probabilities. In controls engineering the agent acts as the controller, actions are the control signal and the environment resembles the plant [61]. The measured states and rewards can be interpreted as the input and feedback to the controller.

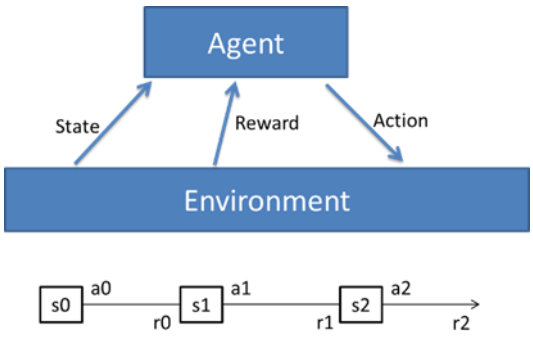


**Figure 2.5: RL framework and key parts [61]**

RL approaches can be categorized in different ways. One type of classification suggests to categorize RL methods into model-free and model-based RL. Model-based RL requires having a full mathematical model of the environment and its interactions with the agent. It is capable of learning low-dimensional and simpler tasks and also is sample-efficient. On the other hand, model-free RL can learn more complex tasks, although it is sample-inefficient. Some new methods are using a combination of both approaches to benefit from their advantages with respect to each other. Passive Learning and Active Learning are two other categories which the RL methods can fall into. In passive learning there is a

given fixed policy and the agent wants to evaluate it. Some methods of this type are Adaptive Dynamic Programming, Direct Utility Estimation, Temporal Difference Learning, and TD Lambda. On the contrary, the agent in active learning approaches must learn what to do itself and its final goal is to dig out a good policy. On-Policy Learning, Off-Policy Learning, Q-Learning, and SARSA are some sort of active learning methods. These methods are investigated mostly by computer scientists and can be found in [66-68].

The next subsection is dedicated to illustration of the RL mathematics and Q-Learning method which is an active model-free type of RL we chose to implement for this research because of its promising results and richer literature in the EMS field. In Q-Learning every state is visited infinitely often owing to the exploration characteristic of RL. Also as the time approaches infinity, the action selection becomes greedy and the agent exploits more of it experience. The learning rate also starts decreasing when the agent happens to become closer to a good policy. More details are found in the next subsection.

### 2.4.2 RL Algorithm and Mathematics

After elaborating on the RL framework and its characteristics, the mathematics which are the backbone of the RL algorithm should be clarified. For defining the equations and concepts, it is assumed that the state is observable. The agent believes that is in the state $s$ ($\epsilon S$) and wants to select an action. This action selection is modeled through this map:

$$\pi : A \times S \rightarrow [0,1] \tag{2.1}$$

A policy based on this map is defined as:

$$\pi(a, s) = P(a_t = a \mid s_t = s) \tag{2.2}$$

and shows the probability of taking action $a$ while the agent is in state $s$. The agent tries hard to find the optimal policy that maximizes the future expected reward. Also a policy can be modeled via this mapping:

$$\pi : S \rightarrow A \tag{2.3}$$

which shows that a policy determines the way the agent selects an action observing a state.

Most of the time an action might not payoff immediately. As an instance, a robot only receives the final payoff after reaching its goal at the very last action. “Expected cumulative reward” is what helps us using a planning horizon to maximize the sum of all future rewards. It is conventionally written as:

$$R_T = E\left[\sum_{\tau=1}^{T} \gamma^{\tau} r_{t+\tau}\right] \tag{2.4}$$

in which $\gamma$ is the discount factor which will be discussed shortly. If T=1 the immediate reward becomes important and we have a greedy policy. T>1 results in a finite horizon problem. An infinite T produces an infinite-horizon case which has finite reward if $\gamma < 1$.

The final sum of the rewards will be finite as long as $\gamma < 1$:

$$R_\infty \leq \ r_{max} + \gamma r_{max} + \gamma^2 r_{max} + \cdots = \frac{r_{max}}{1-\gamma} \tag{2.5}$$

The goal of planning in the MDP framework is to identify the policy which maximizes the future cumulative payoff. We denote the optimal policy as:

$$\pi^* = \underset{\pi}{\operatorname{argmax}}\, R_T^{\pi}(s_t) \tag{2.6}$$

This policy will find the plan which maximizes the expected cumulative reward. Every policy has an associated “value function”, measuring the expected value (cumulative discounted future payoff) of the specified policy. The value function for one-step optimal policy $\pi_1(s) = \underset{a}{\operatorname{argmax}}\, r(s,a)$ is defined by:

$$V_1(s) = \gamma \max_a r(s,a) \tag{2.7}$$

Then two-step optimal policy can be defined accordingly:

$$\pi_2(s) = \underset{a}{\operatorname{argmax}} \left[ r(s,a) + \int V_1(s')\, p(s' \mid a, s) ds' \right] \tag{2.8}$$

$$\pi_2(s) = \underset{a}{\operatorname{argmax}} \left[ r(s,a) + \int V_1(s')\, p(s' \mid a, s) ds' \right] \tag{2.9}$$

Following this trend, value function of T-step optimal policy will be:

$$V_T(s) = \gamma \max_a \left[ r(s,a) + \int V_{T-1}(s')\, p(s' \mid a, s) ds' \right] \tag{2.10}$$

And for the infinite horizon case we get:

$$V_\infty(s) = \gamma \max_a \left[ r(s,a) + \int V_\infty(s')\, p(s' \mid a, s) ds' \right] \tag{2.11}$$

This invariance is known as "Bellman Equation". Every $V$ that satisfies this condition, is both necessary and sufficient for the induced policy to be optimal.

One of the most researched and implemented methods of RL approach for solving the energy management and control problem is the Q-Learning algorithm which is developed by Watkins in 1989 [63]. Q-Learning is a model-free and off-policy method which does not rely on a behavior policy in order to choose actions with respect to present states. It can solve the problems containing stochastic transitions and rewards. The letter "Q" in the Q-Learning may stand for a notion of "Quality" as it is obtained from the Q-function $Q(s,a)$. The value for $Q$ can show the best possible outcome at the end of a learning episode which is obtained by performing action $a$ in the state $s$ [61].

Before beginning the learning process, $Q$ needs to be initialized to an arbitrary fixed value. Initializing the Q-value in an efficient way is something that can influence the performance and speed of the learning based on experience. Q-value is then defined for each state-action pair as a function from state and action space to a real number:

$$Q: S \times A \to \mathbb{R} \tag{2.12}$$

The update function for the Q-value is similar to the value iteration algorithm which is defined for the MDP framework. The update equation is represented as:

$$Q^{new}(s,a) \leftarrow Q(s,a) + \alpha \,.\left(r'(s) + \gamma \,.\ \max_{a} Q(s',a)\ -\ Q(s,a)\right) \tag{2.13}$$

in which the prime sign shows the values corresponding to the next time step. $r(s)$ is the reward function which is defined over the state space. The reward function could also be represented as $r(s,a)$ which means the reward would be a function of the observed state and the taken action. The phrase $\max_{a} Q(s',a)$ chooses the action with which the Q-value for the next step is maximized and returns this maximum value. If we want to clarify what happens at each update step, we need to say that after initializing $Q$ matrix at the beginning, the agent senses that has just observed state $s$ and then selects an action $a$ accordingly at each time step. For the selected action it measures the reward $r$ based on the state-action pair and then transitions into a new state $s'$. When the agent observes a final or terminating state as its $s'$, an episode ends and a new episode of the training process starts. Q-Learning is also capable of handling non-episodic tasks [61]. The simplest implementation of Q-Learning stores the data in a table. Therefore, it has some limitations on the size of data and the state and action space. For storing data in tables, the state and action space need to be discretized.

$\alpha$ in the update equation is called the "learning rate". Learning rate determines how much of the newly obtained data can override the old information. It is also called the step size. The learning rate can be between 0 and 1 ($0 \leq \alpha \leq 1$). If $\alpha = 0$ the agent literally learns nothing. It makes the second part of the update equation equal to zero and the previous Q-value is considered for the update process. It is translated into pure exploitation of the prior information. A factor of 1 is consequently referred to considering only the most recent information and ignoring the prior information. The agent in this case only focuses on exploring new possibilities which can be seen by putting $\alpha = 1$ in the update equation. The case of $\alpha = 1$ is the optimal case for a fully deterministic environment. The problem is that in the real world we do not have such a deterministic environment and the problem is usually stochastic. For these scenarios the algorithm converges only in special conditions which require the $\alpha$ to decay to zero. Practically, a constant learning rate of $\alpha = 0.1, 0.01, 0.001$, etc is used based on the specific problem we have [61], [63].

$\gamma$ is then referred to the "discount factor" which specifies the importance of upcoming rewards with respect to the immediate reward. If the discount factor $\gamma = 0$ the agent only considers the current reward $r(s)$ and disregards the estimate of future value ($\max_a Q(s', a)$). On the other hand a factor which approaches to 1 make it hungry for a higher reward obtained from a huge amount of steps. A value of $\gamma \geq 1$ may cause divergence. Even for values really close to 1 the environment histories and information become very long and the error propagation problem happens. The best scenario would be to start with a lower $\gamma$ and increase it to a final value in order to accelerate the learning process [69].

The next subsection is included to elaborate on the necessity of using function approximation or artificial neural networks in order to handle bigger state and action spaces resulting in bigger data and also in order to handle continuous spaces. It gives some information about different Deep-RL methods and states which approach fits the requirements of this research.

### 2.4.3 Deep Learning and Deep-RL Methods

As mentioned in the previous subsection, the Q-Learning generates a table to store the Q-values for future use. The table needs to include the Q-values for each pair of state and action which can possibly happen in practice or simulation. This is obvious that if the size of data grows or we have continuous state or action spaces, the table simply cannot be utilized. Therefore, there should be another way to handle these common issues.

One method to resolve the issue of big size data or continuous space is to do function approximation. For instance an adapted neural network can work as a function approximator which speeds up the learning process in finite problems. Another method is to quantize the spaces and group some similar data together. This can help with overcoming the data size problem but reduces accuracy [70].

Deep Learning algorithms are widely used among researchers who work with big datasets and try to classify these data and optimally find a solution for their problem. There are various Deep Learning methods available which are used for different tasks, such as Convolutional Neural Networks, Recurrent Neural Networks, and so on which are used for image recognition, natural language processing, etc [13]. Combining the promising concepts of deep learning with the RL framework can result in huge advancements in its performance. Also there is no need to design the state space explicitly. This approach helps resolving the issue with big size data, and also continuous state and action spaces. There have been many different deep networks developed in order to work with the

concept of RL. Table 2.1 shows some of these promising techniques along with some old-fashioned RL methods in a comparable format. There are other techniques with slight differences which are not included here but can be found in literature.

**Table 2.1: Some different RL and Deep-RL algorithms to compare**

| Algorithm | State Space | Action Space | Model | Policy |
|---|---|---|---|---|
| Q-Learning [67] | Discrete | Discrete | Model-free | Off-policy |
| SARSA [68] | Discrete | Discrete | Model-free | On-policy |
| Deep Q-Network (DQN) [13] | Continuous | Discrete | Model-free | Off-policy |
| Deep Deterministic Policy Gradient (DDPG) [71] | Continuous | Continuous | Model-free | Off-policy |
| Trust Region Policy Optimization (TRPO) [72] | Continuous | Continuous | Model-free | On-policy |
| Proximal Policy Optimization (PPO) [73] | Continuous | Continuous | Model-free | On-policy |

The works on developing an agent playing ATARI game by Google DeepMind has attracted a great deal of researchers' attention to Deep-RL methods. In 2014 DeepMind introduced the Deep Q-Network (DQN) which had some interesting features overcoming the instability occurred by nonlinearity in function approximation by artificial neural networks for RL. These features streamlined the employment of deep learning for more complex RL problems, specifically Q-Learning. One of these techniques is called the "experience replay" in which a random batch of prior actions is sampled for training instead of the most recent action [13]. This technique is also used in our research and have been discussed in chapter 3.

Another algorithm which is used in this research and has shown fruitful results in outperforming the simple DQN, is the Double Deep Q-Network or Double DQN. The idea behind that comes from the Double Q-Learning technique proposed in 2011 by Van Hasselt [74]. This approach is an off-policy

RL algorithm that considers different Q functions for action selection and value evaluation. This technique solves the overestimation issue that occurs in some simple Deep Q-Learning problems [75].

Other Deep-RL algorithms have their own configurations and benefit from different methods and opinions which are out of this research scope. As an instance, the DDPG algorithm which has continuous state and action space develops two different networks which are called actor and critic networks. Some other algorithms use “advantage” operator. Based on the requirements of this project, a Double DQN with experience replay methodology is employed.

## 2.5 Required Data and DAQ Systems

Previous sections in this chapter were dedicated to the background on the environmental motivations for doing this research and energy savings, as well as some suggested solutions for the EMS part of the system and the ways to make it an intelligent one. All of what is stated hereabove would be possible if and only if we can measure and collect the data from various parts of the system. This section tries to briefly give some ideas about required data and how to measure them, along with some existing data acquisition (DAQ) systems which can collect and process the raw data for us.

DAQ refers to selecting some samples of a real-world measured signal based on a sampling frequency and then converting it to digital values in such a way that the data can be processed and used for statistics and other purposes. These digital data are then processed through a computer. A DAQ system consists of sensors and a circuit for measuring and conditioning the analog signal, Analog to Digital Converter (ADC) for converting the measured signal to digital numeric values, and a computer with programmable software for processing the data and storing it [76], [77]. An overview of common data acquisitioning steps is depicted on Figure 2.6.

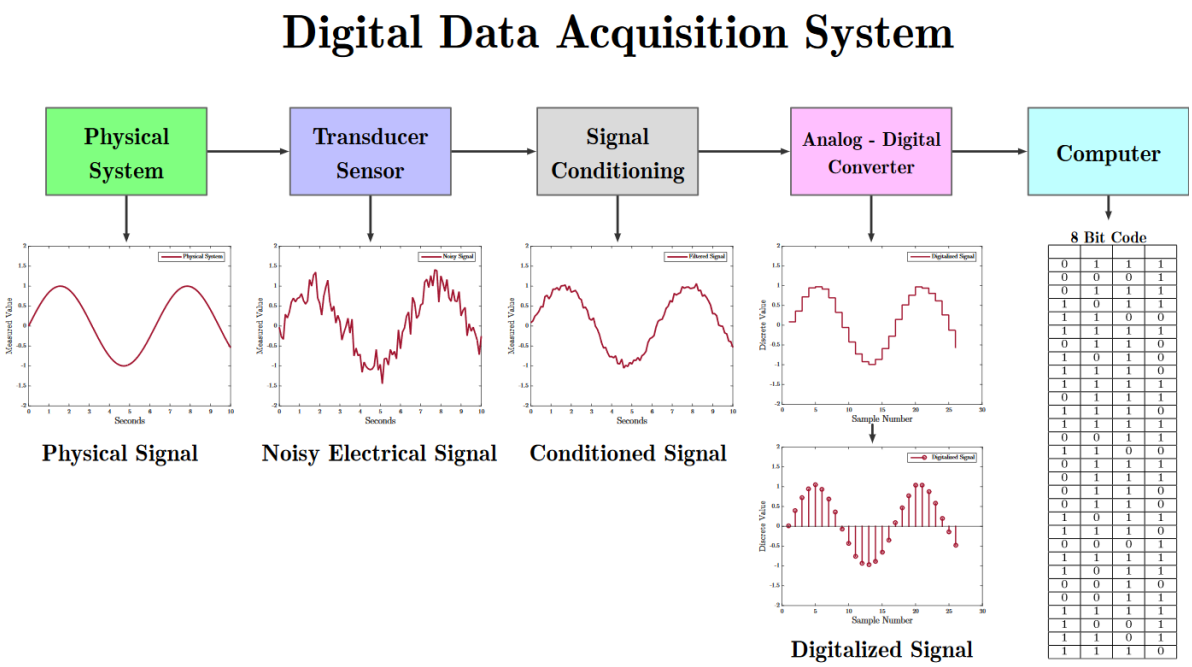


**Figure 2.6: Regular steps followed by DAQ systems [78]**

The first computers used for data acquisitioning were developed by IBM in 1963. After that a lot of effort has been made and different companies produced various DAQ systems specialized in a variety of areas. Just to give an example we can name the National Instruments company which produces specialized DAQ systems and DAQ+Controller products. It also produces modular DAQ systems which makes it perfect for customizing the performance [79]. One of these DAQ+Controller devices which has been used for the previously installed vehicular SIR systems is NI cRIO-9031 which is explained in details in chapter 4 of this thesis.

For this project in order to decrease the premium costs and benefit from a specialized system which fits the requirements of the SIR system, a new cost-effective DAQ system has been developed. The required sensors, ADC board, computer, and electrical enclosures are explained in details in chapter 4.

## 2.6 Summary

This chapter attempted to provide some supporting backgrounds on the concept which are used in the following chapters and review some literature regarding that. First off it started with giving a heads up

about the global warming issue, GHG emissions mostly by the transportation sector, and its effects on the environment, and the importance of doing something. Then it provided some ideas about the idling reduction technologies and the approach which is selected for this research. The chapter then proceeds with the energy management strategies and different types of controllers. It states the pros and cons of these methods and elaborates on the selected methods for the scope of this thesis. The next section gives a worthy background on the RL framework and its concepts as well as the methods pertaining to making it a Deep-RL framework which has huge advantages. These concepts are used in the third chapter. The final section of this chapter reviews the data acquisitioning and the existing DAQ systems which are then used for the fourth chapter.

The thing that is missing in the literature is providing an intelligent EMS for the systems designed for idle reduction. Also the concept of solar idle reduction is not well-developed and needs further research and implementation. Also developing a cost-effective DAQ system for the purpose of measuring the sensor data for idle reduction systems is not addressed recently in the literature. This thesis focuses on the aforementioned research gap and tries to come up with tenable solutions.

# Chapter 3
# Modelling and EMS Design

In the second chapter the importance of reducing the GHG emissions and lowering the fuel consumption for different type of vehicles has been reviewed. It is necessary for every fleet manager to plan for reducing these emissions in their service vehicles to meet the international goals and rules. Also different idle reduction technologies are discussed and compared together. It is stated that solar panels can be used to provide energy for the auxiliary loads of the vehicle. For the purpose of controlling the electrical power flow between the main and auxiliary battery, an energy management system (EMS) is required. The process of the intelligent EMS development for the solar idle reduction (SIR) system is explained in detail in this chapter.

This chapter presents some details about the vehicle model and its integration with the SIR system at the beginning. Some simplifications are considered and a 2-DOF bicycle model is presented which handles the vehicle dynamics. The chapter then follows with the steps of designing three different energy management strategies in the order of added complexity. First control scenario uses a simple rule-based controller, while a more intelligent and complicated approach is taken for the second EMS which is an RL-based controller. Last but not least scenario belongs to the case of Deep RL-based EMS which considers vehicle dynamics and tries to minimize the fuel consumption based on its special reward function. For each of the aforementioned energy management strategies slight modifications are applied to the model which is discussed in detail. Furthermore, the comparison of these scenarios is presented and some explanations are provided.

## 3.1 Vehicular SIR System

To understand the model of the system which resembles the vehicle dynamics and the SIR system model simultaneously, first we need to know about each and every part of the system. Figure 3.1 demonstrates a simple overview of the whole system. Solar panels are installed on the roof top of the service vehicle. The main and auxiliary batteries as well as the busbars connecting them to the main loads and auxiliary loads of the vehicle are fed through the EMS system. They are all connected through an isolator switch controlled by the EMS system which determines whether they should be connected or not. There are a variety of choices for auxiliary loads of a service vehicles. They can be different

based on the vehicle's usage. Some of the mostly used auxiliary loads can be emergency lightings, HVAC unit, audio systems and lifts. Also the main goal of the intelligent vehicular SIR system which is reducing the GHG emissions is highlighted in the figure. Some of these gases are nitrogen oxides ($NO_x$), carbon monoxide (CO), carbon dioxide ($CO_2$), and sulphur oxides ($SO_x$). A more detailed Simulink model of the system which is developed by the SIR project recent team members and is presented in [35] is reviewed in following parts.

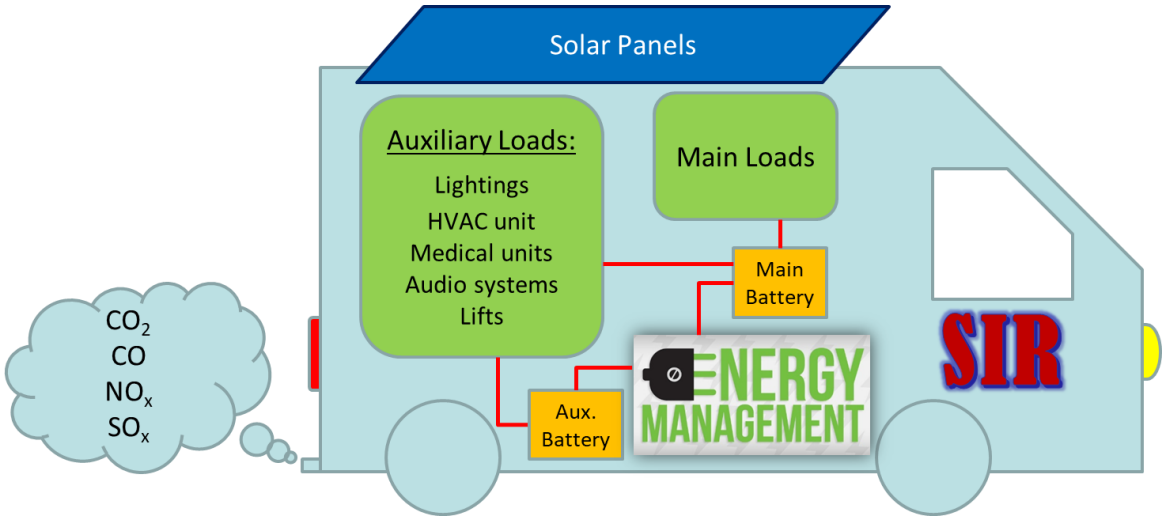


**Figure 3.1: Overall sketch of a SIR-enabled service vehicle**

Two service vehicles from the city of Waterloo' fleet has been chosen for installing the SIR system. One of them is a 2011 Ford F550 Super Duty, and the other one is a 2010 Freightliner Heavy Duty truck. Flexible solar panels are mounted on the driver cab of these vehicles and the solar charge controller, DAQ system and other components are packed in a water-resistant box on the vehicle. The auxiliary battery is also installed on these vehicles.

The only component of the implemented SIR system on the aforementioned trucks, which is acting as a simple controller is a battery isolator switch. This component works based on the voltage it senses from the connected batteries. It compares the measured terminal voltages with a reference voltage and connects/disconnects the batteries. If the circuit is closed and the batteries are connected, both of the main and auxiliary batteries which are now connected to the alternator of the vehicle are utilized to provide electrical power for the vehicle's different loads. That means the auxiliary battery is draining

the charge of the main battery in case the energy from the solar panels is not sufficient to maintain the auxiliary battery's state of charge (SOC).

Figure 3.2 shows the employed voltage-based working isolator switch by Power Fist. This battery isolator is capable of transmitting 90A currents. Its main responsibility is to prevent the main battery from discharging when no source is connected. The voltage on the fully charged vehicle batteries should be above 12.6V. When the engine is on this voltage should be 13.7 to 14.7 volts [80]. The battery isolator switch separates the main and auxiliary batteries when the voltage drops below 12.7V to prevent over-discharging of the main battery. Also when the voltage gets above 13.2V this switch connects the batteries again. When the batteries are connected via this battery isolator, if there is at least one electrical power source available, both the main and auxiliary batteries are being charged. The existing power sources considered for this research are the vehicle alternator, solar panels, and the grid (also referred as shore power).


**Figure 3.2: Voltage level enabled battery isolator switch utilized for SIR system**

Undoubtedly such a battery isolator which only works with voltage constraints is not optimum. Better results, in terms of fuel consumption and GHG emissions reduction, can be obtained if we can have control on the batteries' SOC and also consider vehicle dynamics in the controller's decision making process. This fact emphasizes the importance of designing various EMS strategies in seek for better outcomes. These new EMS strategies have been developed for this research and are presented in the

next sections. To implement the newly designed controllers, some changes are required to be made in the installed system and some components like the battery isolator need to be replaced.

## 3.2 Modeling and Prerequisites of Simulation

The MATLAB Simulink model of the SIR system along with the 2 degrees of freedom (2-DOF) vehicle dynamics model is developed by the previous researchers in the SIR project group [35]. However, for the scope of this research the model is modified for a 2011 Ford F550 Super Duty Chassis Cab 4x2, and required changes are applied for each EMS strategy. The simplest form of the model is illustrated by Figure 3.3. Four different areas are seen in the Simulink model which are explained as follows:

The "Vehicle System" basically includes various parts of a vehicle which are considered in a 2-DOF vehicle dynamics, as well as the main battery and main loads. The alternator of the vehicle is also shown separately in order to make the inputs and outputs easier to follow. The main battery is a 30 Ah lead-acid 12V automotive battery and the main loads are assumed to be 1000 W in total. The vehicle dynamics and mechanical parts block includes the required mechanical parts of the vehicle, including the diesel engine, transmission parts, final drive, and wheels' model which play a pivotal role in determining the 2-DOF bicycle model for vehicle dynamics. The vehicle dynamics equation is also integrated into the Simulink model.

The 2-DOF bicycle model for the vehicle dynamics is:

$$a = \frac{1}{M} \times \left( F_w - \frac{1}{2} C_d \rho A V^2 - C_{rr} mg \cos\alpha \times sign(V) \; - \; mg \sin\alpha \times sign(V) \right) \tag{3.1}$$

in which $M = 2100$, $C_d = 0.6$, $\rho = 1.225$, $A = 4.7$, $C_{rr} = 0.01$, $\alpha = 0$, and $g = 9.806$ for the 2011 Ford F550 service vehicle. Table 3.1 shows the required parameters for initializing the Simulink model for starting the simulation.

The fuel consumption rate is then calculated from:

$$\dot{m}_f = \alpha \omega_{eng}^2 + \beta \omega_{eng} T_{eng} \tag{3.2}$$

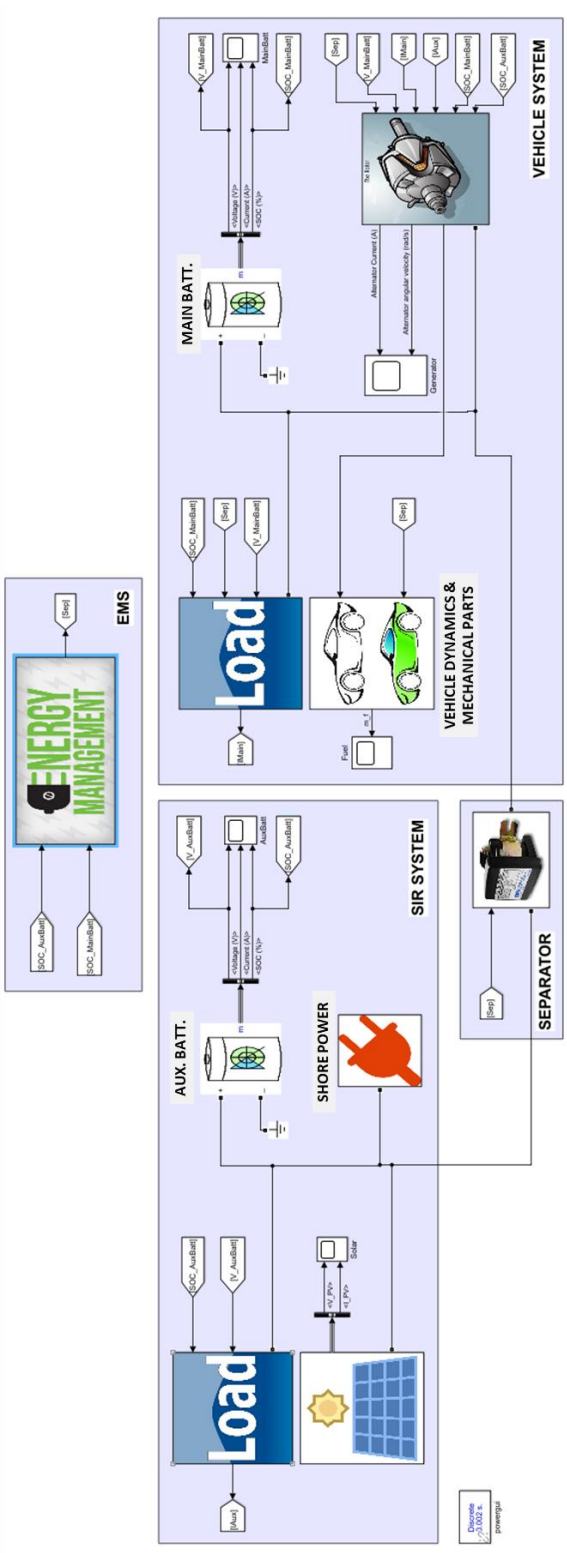


**Figure 3.3: Simulink model of the SIR system installed on 2011 Ford F550 Super Duty truck**

**Table 3.1: Required parameters for initializing the simulation**

| Parameter | Value | Parameter | Value |
|---|---|---|---|
| Solar panels' temperature | $25°C$ | Auxiliary loads' power | 750W |
| Batteries' nominal voltage | 12V | Final drive ratio | 3.73 |
| Main battery capacity | 30 Ah | Final drive efficiency | 0.94 |
| Auxiliary battery capacity | 20 Ah | Transmission efficiency | 0.5 |
| Main battery SOC | 95% | Generator ratio | 1.5 |
| Auxiliary battery SOC | 80% | $\alpha$ parameter of engine | $1.63 \times 10^{-6}$ |
| Main loads' power | 1000W | $\beta$ parameter of engine | $2.52 \times 10^{-5}$ |

The "SIR System" reveals another important part of the Simulink model. It includes the model for solar panels, and also auxiliary battery and auxiliary loads. The solar panels model receives the solar irradiance data as a lookup table. The auxiliary battery is almost similar to the main one which means a 12V lead-acid battery but with 20 Ah of capacity. Also it is assumed that the auxiliary loads are 750 W in total. The shore power is considered for this model which may charge the auxiliary battery while the vehicle is connected to the grid. The shore power is modeled as an 11.8V DC voltage source which provides electrical power to charge the batteries when the vehicle is in the parking state. Figure 3.4 demonstrates the SIR system model.

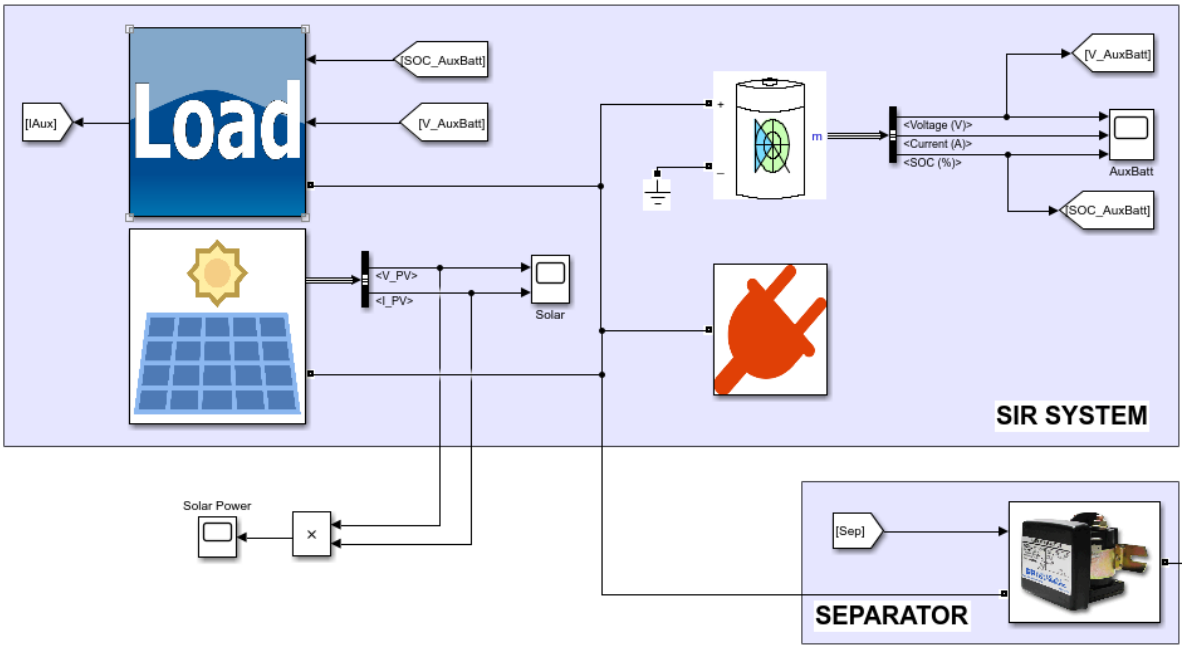


**Figure 3.4: The part of the model representing the SIR system along with the battery isolator**

There are two other parts in the Simulink model, namely "EMS" and "Separator". The separator is where all other parts are connected through. The main battery and main loads' busbar, as well as the alternator are connected to one terminal of the battery isolator which is clear from the model. The auxiliary battery and its connected loads, solar panels, and the shore power are connected to the other terminal of the battery isolator. The state of the separator switch whether it is open or closed is then determined by the EMS part of the model. The EMS block receives some inputs from other parts of the system to control the state of the battery isolator switch as its output.

Now that the simple model seems useful for going forward with, we need to specify a driving cycle for the modeled vehicle. A driving cycle acts like a data set which provides the vehicle model with the speed data over the simulation time. This helps the simulations to be more realistic because the driving cycles are mostly the recorded data from experimental real-world driving in different situations and scenarios [41]. The speed profile for each driving cycle can be in mph or km/h and will be given to the Simulink model or MATLAB code as a look-up table. The vehicle model then follows the speed profile of the driving cycle and generates the torque and the angular velocity accordingly.

The driving cycle which has been used for this research is a stack of two FTP driving cycles accompanied with some parking times and on-site operation time slots. The FTP is a city driving cycle developed through various tests performed by U.S. Environmental Protection Agency (EPA). It has a duration of 1874 seconds and the travelled distance of 11.04 miles with a 21.2 mph average speed [81]. Figure 3.5 demonstrates a specialized from of two FTP driving cycles which has been used for this research. This driving cycle is a 10,000 seconds-long cycle with 1,000 seconds at parking state at the beginning, an FTP cycle of driving to the site (1874 seconds), 4,000 seconds of on-site operation in which the vehicle does not move and the speed is zero (idling state), another FTP cycle for driving back to the parking slot (1874 seconds), and 1252 seconds at rest to finish the driving cycle with another parking state.

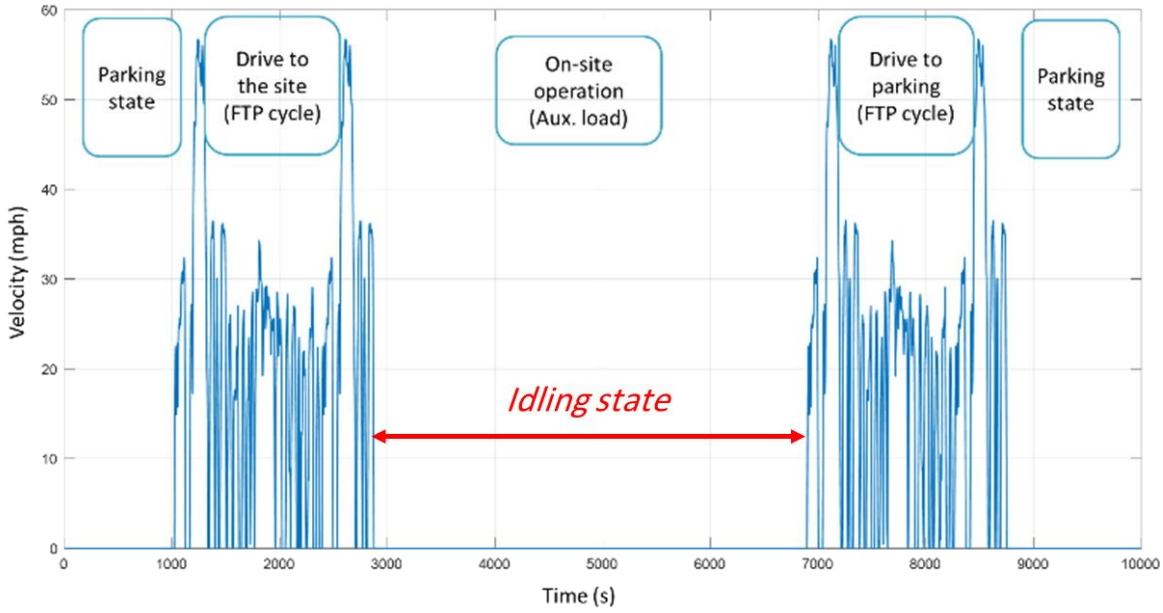


**Figure 3.5: The driving cycle being used for simulation of a service vehicle's trip during a day**

Considering all of what is stated above including the Simulink model, vehicle dynamics and fuel consumption equation, and the driving cycle, we can design different EMS systems accordingly. The next three sections of this chapter are dedicated to the Rule-based, RL-based, and Deep RL-based EMS strategies for controlling the state of battery isolator switch based on different inputs and specific scenarios. After all, a comparison of the developed EMS strategies with the case of having no SIR system at all is presented in a separate section.

## 3.3 Rule-based EMS

Based on the literature review presented in the second chapter, a Rule-based controller can be really fast and robust, and therefore real-time implementable. The only problem for that is related to the optimality issue. Even if an expert defines the best conditions for the Rule-based controller, still the optimality claim cannot be made. However, such a well-designed EMS system can be near-optimal.

The Rule-based EMS for the SIR project is defined as a Simulink function with some "if" conditions. Figure 3.6 shows the corresponding Simulink block which is located at the EMS part area. It receives the battery SOCs and checks the conditions to output a value of 0 or 1 for the separator switch. This value is stored in a memory block and is fed into the conditional function to maintain the separator state in case no conditions are violated.

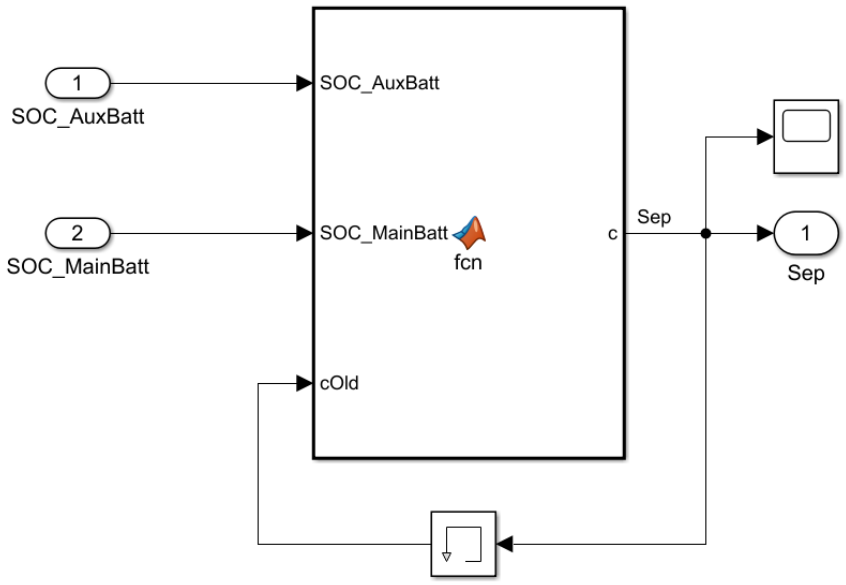


**Figure 3.6: The function block used in Simulink which forces the conditions on batteries' SOC as a Rule-based controller**

To run the simulation, the Simulink model needs to be initialized by some required parameters. These parameters are initialized from the Table 3.1. After running the simulation with the described driving cycle and aforementioned parameters, the results indicating voltage, current, and SOC of the main battery and auxiliary battery, as well as the battery isolator switch state (0: open, 1: closed) are depicted in Figure 3.7, Figure 3.8, and Figure 3.9 respectively.

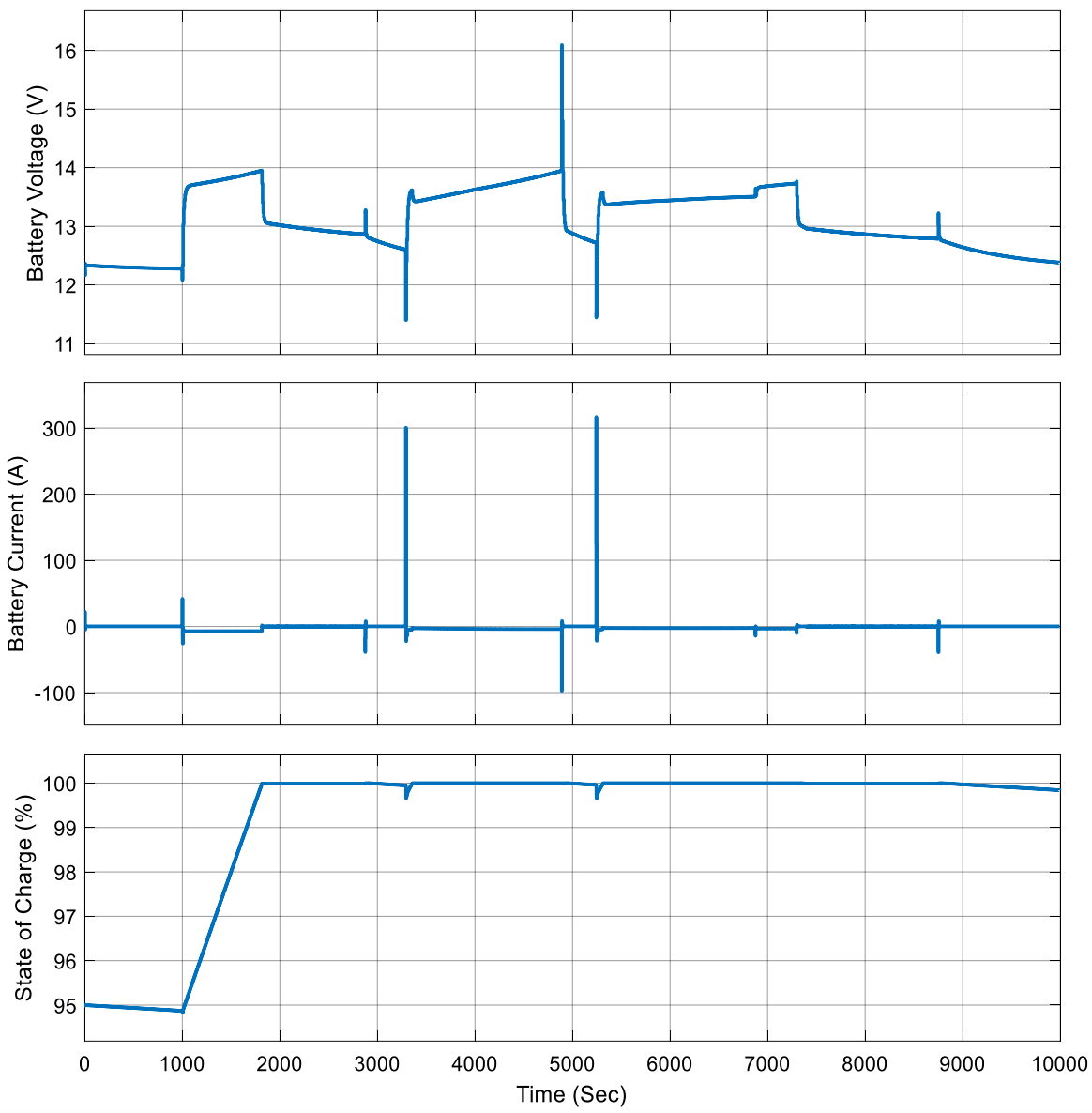


**Figure 3.7: Voltage, current, and SOC of the main battery for the Rule-based EMS**

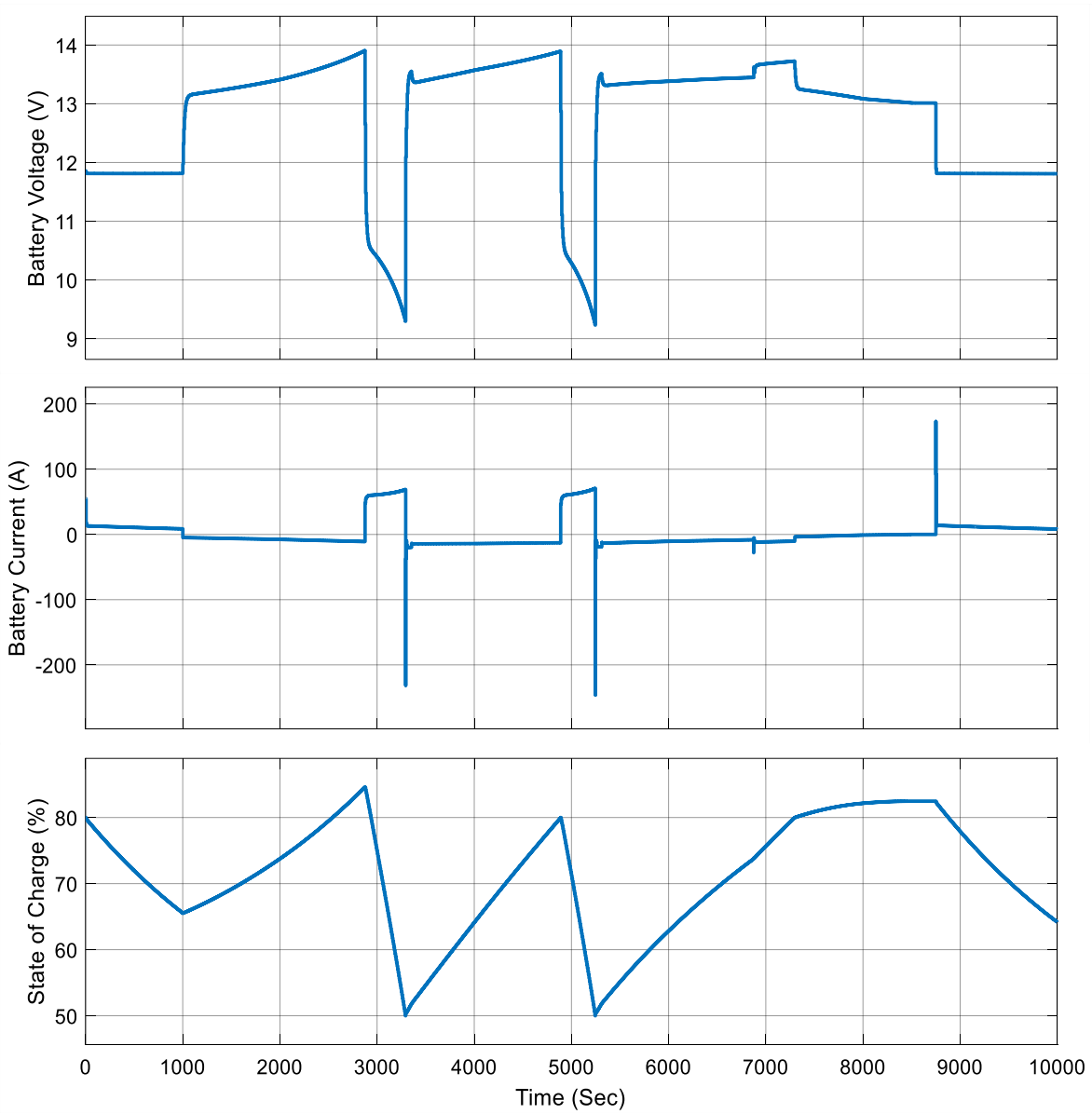


**Figure 3.8: Voltage, current, and SOC of the auxiliary battery for the Rule-based EMS**

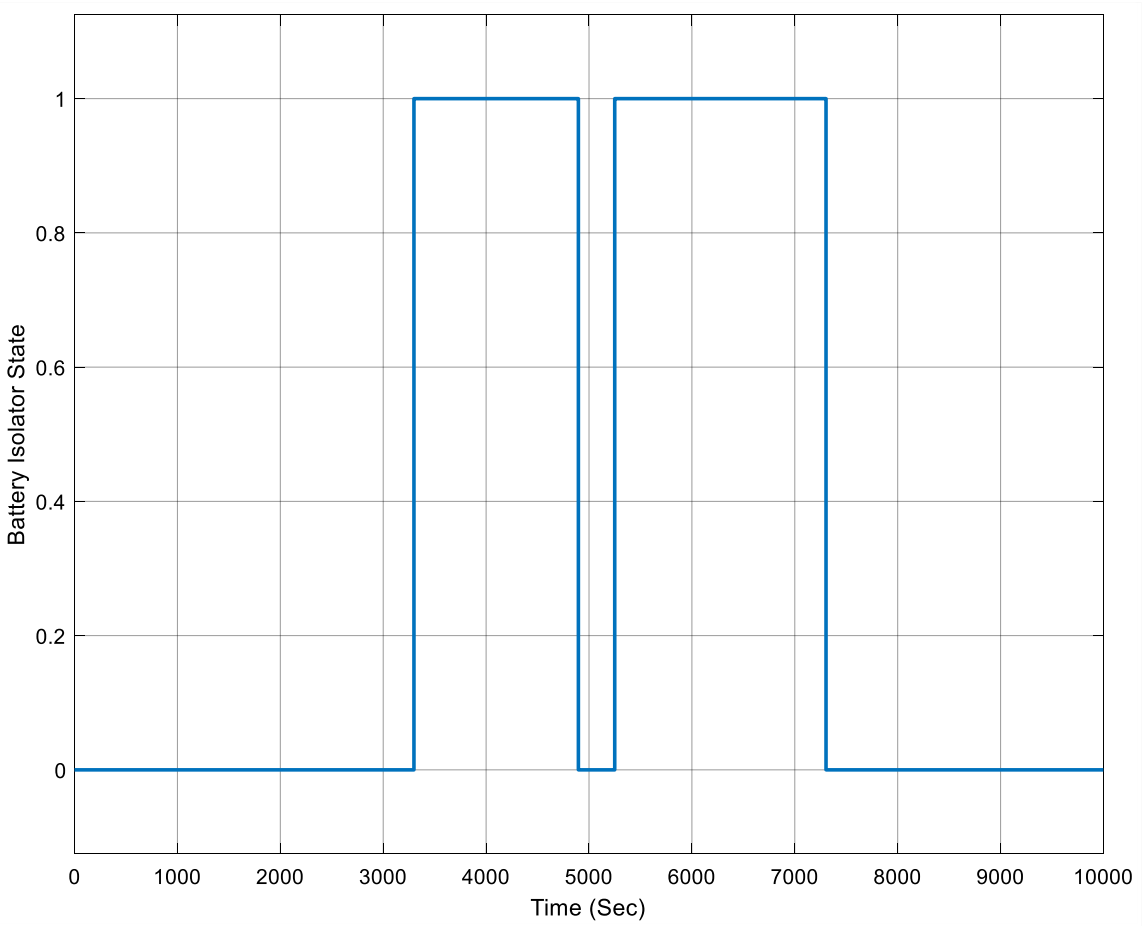


**Figure 3.9: State of the battery isolator switch for the Rule-based EMS**

## 3.4 RL-based EMS

Reinforcement Learning (RL) is the approach which is selected for this research in order to develop an intelligent EMS strategy for the vehicular SIR system. As indicated in the literature review section, RL approach is chosen as a learning-based approach which can have some benefits from both rule-based and optimization-based approaches. If a network which is trained using Q-Learning technique and the Bellman optimality criterion, handles the energy management of the SIR system, can give us the optimal solution. The limitation that is added to the problem is that the network needs to be trained before being used by the simulation. The training is also done by the Simulink model itself with the parameters which are mentioned as follows.

The states for the RL framework are considered to be the SOC of the main battery and auxiliary battery which is discretized in order to be used by the Q-Learning algorithm. The action selection is done for determining the situation of the battery isolator switch whether it is closed (1) or open (0). The reward for this framework is also defined based on different situations which can happen to the SOC of batteries for different switch (action) situations during vehicle operation. The maximum number of iterations considered for the training is $10^8$ steps for 100 episodes. The discount factor is $\gamma = 0.2$ and the learning rate is set to $\alpha = 0.1$ at the beginning which decays with respect to the number of passed steps. Also the exploration probability is $\epsilon = 0.8$.

A part of the generated Q-table is depicted on Figure 3.10 just for demonstration. The look-up table is saved by MATLAB and can be used for the simulation. The Simulink model starts with the main battery SOC of 95, and auxiliary battery SOC of 80 percent and looks up these values in the Q-table. Based on these values the action which makes the best outcome based on the Q-table is selected and the simulation proceeds with that. Figure 3.11, Figure 3.12, and Figure 3.13 demonstrate the voltage, current and battery SOC for the main and auxiliary batteries, along with the battery isolator state which refers to the action for RL-based EMS. The large voltage and current values which arise at the isolator switching points are because of the rapid changes in the state of the battery isolator which connects/disconnects the power from solar panels. The switching brings about open circuits and some sparks. These values would be possibly damped by the capacitors and inductances of the system in practice. However, we need to minimize the switching frequency to prevent multiple occurrences of these sparks and resulting large values of voltage and current.

```
val(:,:,9,30) =

    0.9322    1.0606    1.0622    0.9342    0.9345    1.0272    0.9392    1.0076    0.9797    0.6346    0.5639
    0.9833    0.9832    0.5716   -0.2993    1.0622    0.9806    0.6787    1.0880    1.0514    1.0608    1.0697
    1.0661    1.0919    0.6838    0.5717    1.0439    1.0556    0.6375    0.5398    1.0918   -0.3024    1.0487
    0.6448    0.8015    0.9623    0.9794    0.9889    0.9959    0.6143    0.9406    0.6574    1.0438    0.9753
    1.0721    0.6317   -0.3021    0.9371   -0.2935    0.8328    0.8591   -0.4357    1.0380   -0.3450    0.5497
   -0.3779   -0.4035   -0.3215   -0.4522   -0.3394    0.8014   -0.4375   -0.4342   -0.3996   -0.4623    0.8968
    0.6535   -0.3383    0.5262    0.9722   -0.4069    0.9792    0.9368    0.5641    0.6396    0.6387    0.5143
    0.6636    1.0640    0.6521   -0.2992    1.0522   -0.3632    0.8433    0.6206    0.9823   -0.3457   -0.3661
    0.5521    0.6070   -0.3516    0.6292   -0.5592   -0.3194   -0.4656   -0.3947   -0.3552   -0.3536   -0.3617
    0.6378    0.6457    0.5122   -0.3575   -0.4381   -0.4079    0.9163   -0.4230    0.5116    0.8057   -0.4317
   -0.9280   -0.9415   -0.9343   -1.0072   -0.9473   -0.9167   -0.9047   -0.9010   -0.8913   -1.0318   -1.0354
   -0.9069   -0.9252   -0.9004   -0.9392   -0.9076   -0.9270   -0.9220   -0.9019   -0.9288   -0.9620   -0.9457
   -1.0463   -0.9068   -0.9186   -0.9404   -0.8945   -0.8950   -0.9011   -0.9344   -0.9204   -0.9783   -0.9055
   -0.9529   -0.9098   -0.9433   -1.0018   -0.9297   -0.8947   -0.8995   -0.9019   -0.8945   -0.9274   -0.8956
   -0.9433   -0.8942   -0.9021   -0.9713   -0.9285   -0.9069   -1.0229   -0.9325   -0.8939   -1.0353   -0.9482
   -1.0041   -1.0124   -0.9319   -0.9172   -0.9420   -0.9157   -0.9185   -0.9489   -0.9486   -0.9042   -0.9031
   -0.8920   -0.9666   -0.9118   -0.9363   -0.9269   -0.9092   -0.9029   -1.0003   -0.9416   -0.9562   -0.9517
```

**Figure 3.10: A small section of the generated lookup table for Q-Learning, trained for the RL-based EMS**

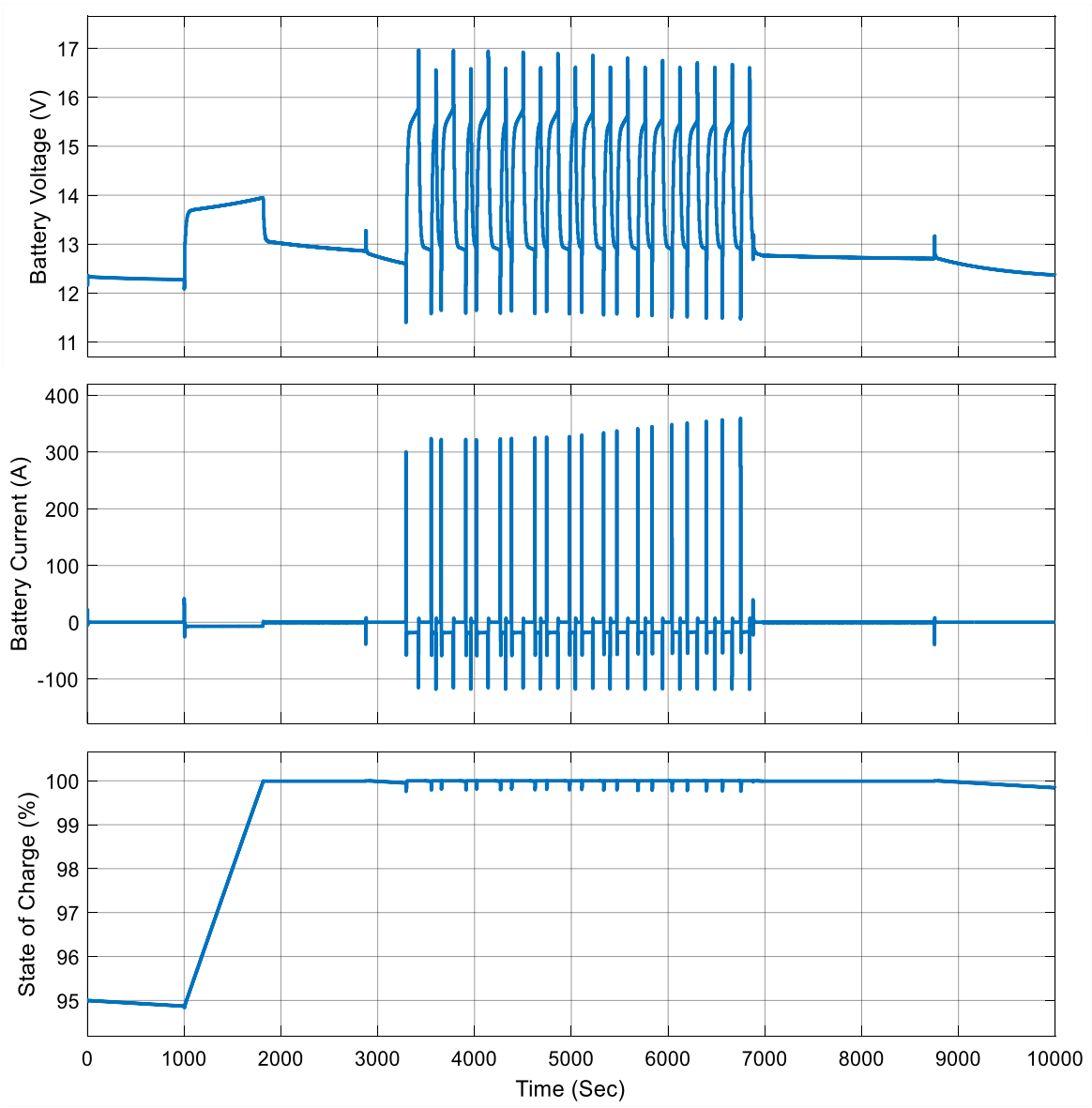


**Figure 3.11: Voltage, current, and SOC of the main battery for the RL-based EMS**

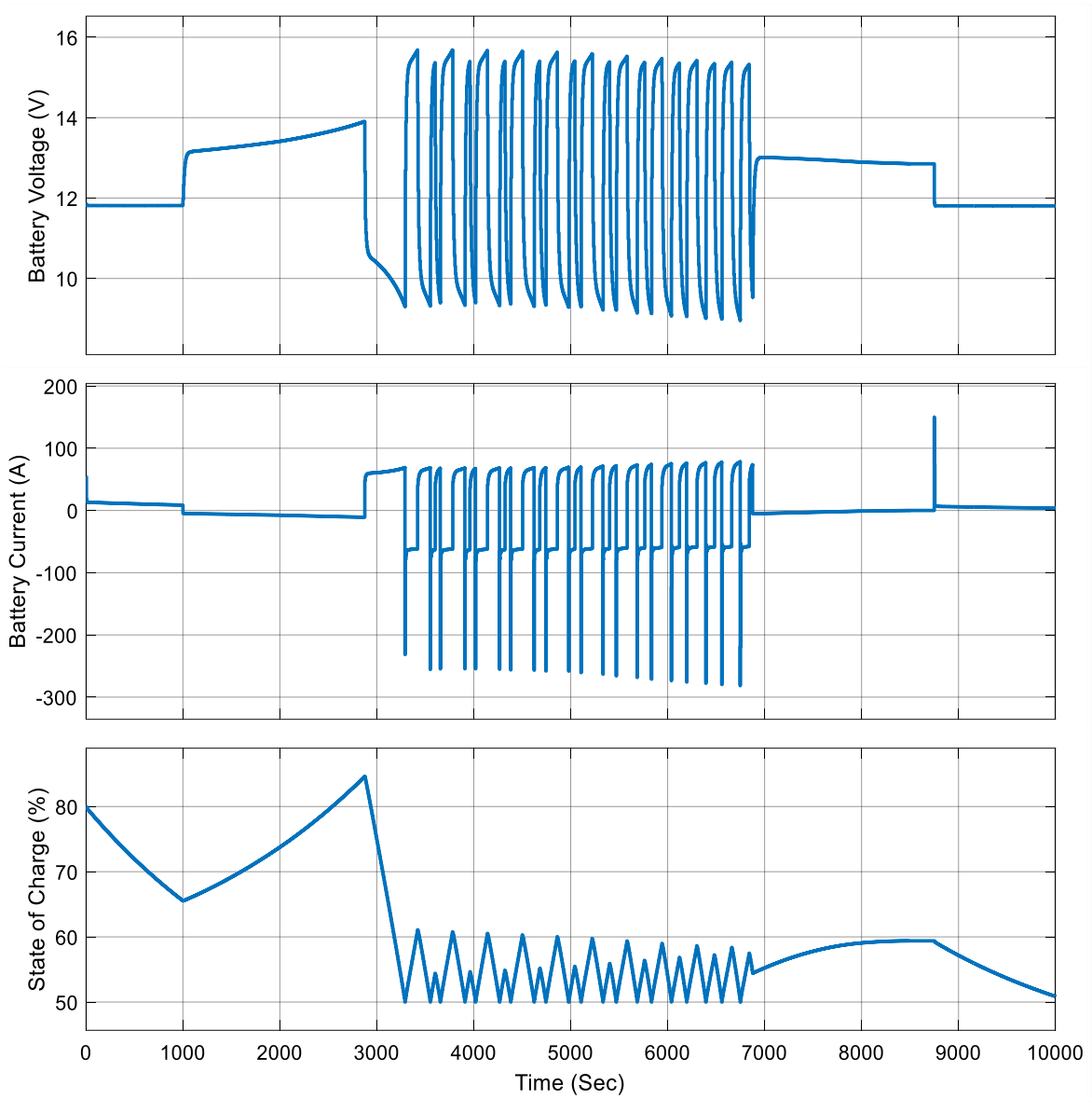


**Figure 3.12: Voltage, current, and SOC of the auxiliary battery for the RL-based EMS**

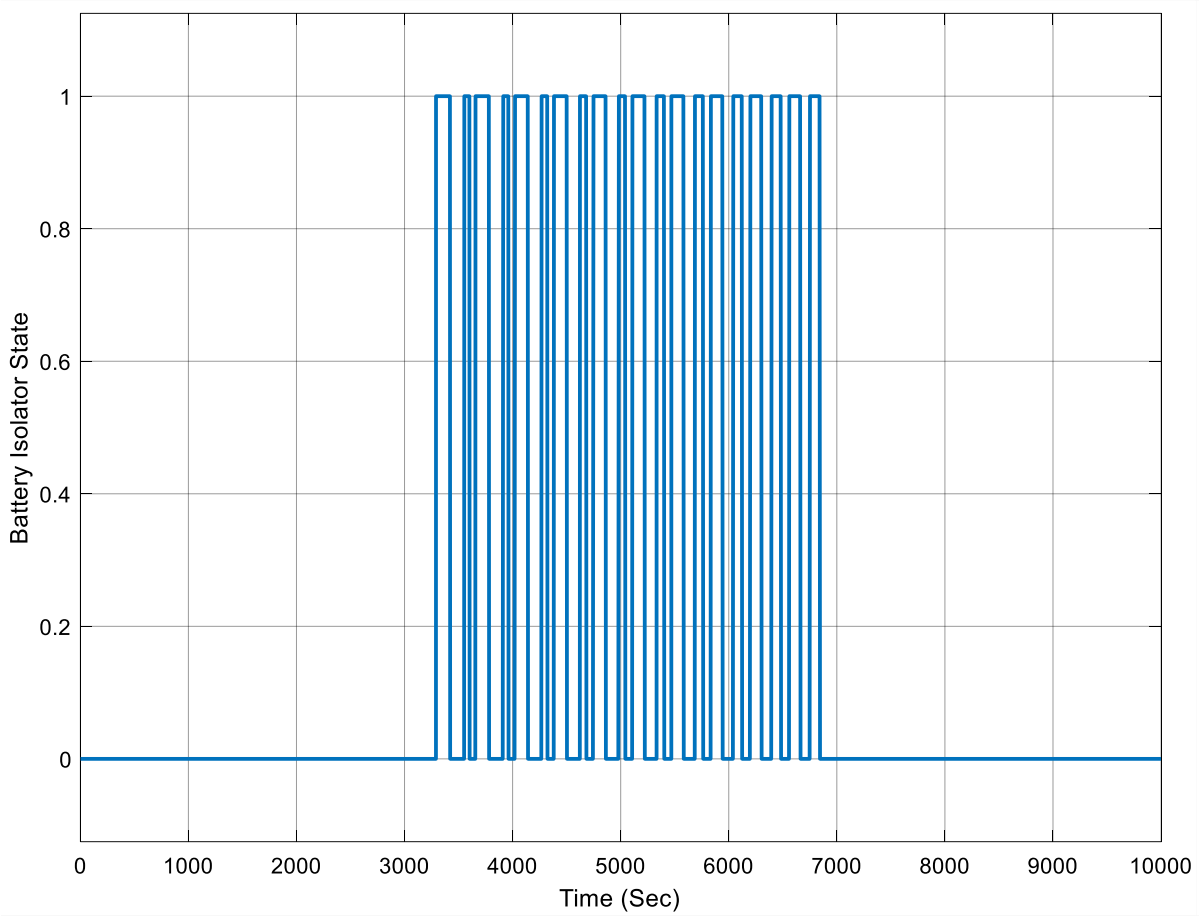


**Figure 3.13: State of the battery isolator switch for the RL-based EMS**

## 3.5 Deep RL-based EMS

One problem with the simple RL-based EMS which makes limitation for its performance and does not allow it to beat other methods is the limitation on the data size and requiring discrete data. Q-Learning approach generates a look-up table and if the state or action space has huge amount of discrete data or any of them is a continuous space, the Q-Learning approach becomes useless. That is where the promising Deep Learning approaches can help the Q-Learning method solve its dimensionality problem.

Another limitation of the designed RL-based EMS which is caused by its data size limits relies on the lack of vehicle dynamics in the state space. Because of not having any information about the vehicle dynamics during the training process of the Q-Learning based EMS, it only relies on the SOC of

batteries and does not consider the parameters affecting the fuel consumption. This aspect makes the RL-based EMS more similar to a Rule-based controller and not much analogous to an optimization-based strategy.

Developing a Deep RL-based EMS system can address these issues considering a continuous state space for increasing the accuracy of the computations, and also including some parameters of the vehicle dynamics which play a role in the fuel consumption equation for reward shaping. Based on the literature review which is done in chapter 2, a Double DQN has been implemented which handles the EMS part of the system. The process of developing this new intelligent EMS strategy is discussed in details in this section.

This is a trend among the researchers working with Deep Learning methods to write their codes in the Python language. Python provides some unique features and packages for Deep Learning which makes it incomparable. Some of these packages are Tensorflow, Theano, and Keras just for instance. These packaged with some prebuilt functions and classes can make it a lot easier to write the codes for an intelligent Deep RL-based EMS system. Also for the purpose of designing the environment for the RL framework, Python makes it possible to use the OpenAI Gym which provides transferrable classes and tools for building an environment. Consequently it was determined to write the code for this EMS scenario in Python. It requires some changes in the Simulink model and the process of training the network.

First off the previous EMS part of the model in Simulink model has to be completely removed because the training is done in Python and the resulting network also is called from Python. It is obvious that in order to send and receive data between Simulink and Python we need a communication means. One way to do that is to call the MATLAB engine in Python and try to set the MATLAB commands in Python. This is completely doable, but the issue is that this process is really time consuming. The MATLAB interpreter takes a lot of time to react to the commands set by Python code. Therefore we decided to develop a communication part in Simulink model and also configure the same communication means in Python as well.

There are different types of serial communication which has pre-defined blocks in the Simulink library. Based on the requirements of this research and the simplicity of the work, it was determined that the User Datagram Protocol (UDP) communication can be the best fit for this purpose [82]. Figure 3.14 depicts the new “EMS” part of the model including the communication blocks which is added to the Simulink model. It packs the “double” data type into “uint8” and sends the battery SOCs,

engine torque, and engine angular velocity to the Python core and also waits for the resulting “action” to be received from Python.

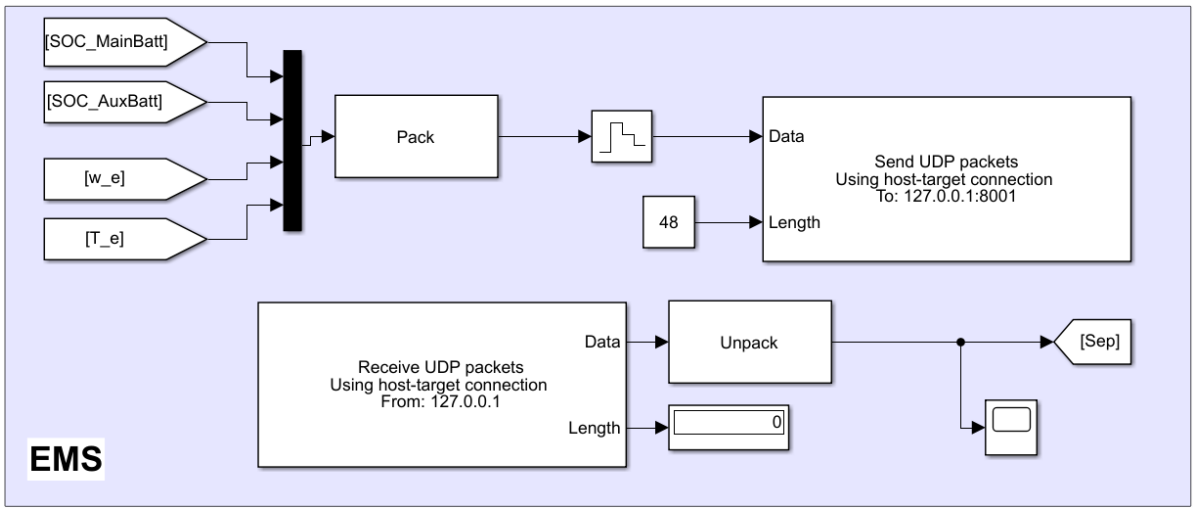


**Figure 3.14: The new EMS system including the UDP communication blocks for the Deep RL-based scenario**

The updated Simulink model for this scenario is then graphed in Figure 3.15.

To have the UDP communication set up in Python as well, we need to import the required libraries such as “socket” for that. After that similar to what we did in the Simulink model, we need to configure the send and receive ports and IPs and make them similar to what is set in Simulink. Also data packing and unpacking which requires the “struct” library is required to change the data type from “uint8” into “double” and vice versa.

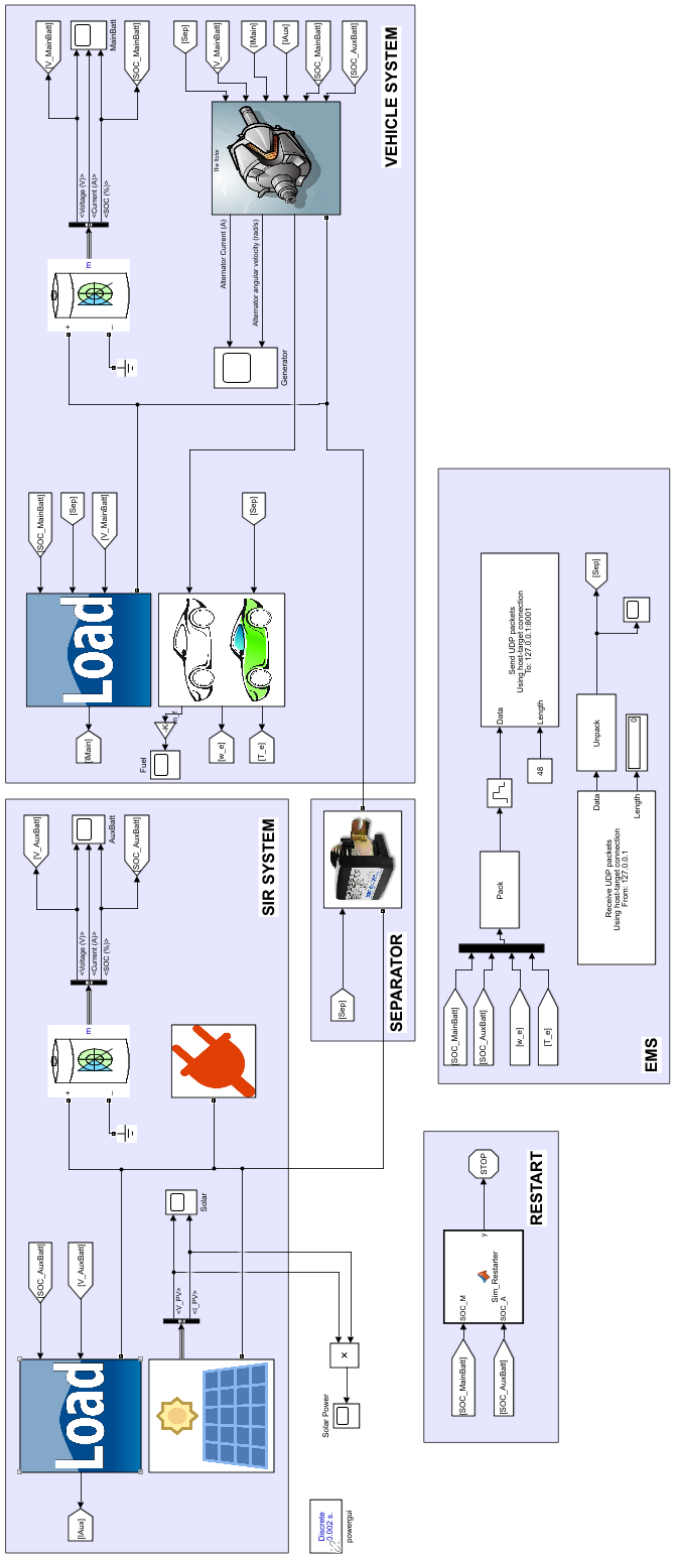


**Figure 3.15: Updated Simulink model of the service vehicle for the Deep RL-based scenario**

The Python code uses the packages from the Tensorflow [83] and Keras [84] libraries which are mainly developed for the Deep Learning problems. Also the RL environment is designed using the tools provided by OpenAI Gym [85]. The network architecture which is selected for this research is a recursive neural network with three dense layers. The "Sequential" model of the Keras library is used with the "Adam" optimizer. It has three dense layers with this configuration:

1. Dense layer; input size: state size; output size: 24; activation function: Relu
2. Dense layer; input size: 24; output size: 24; activation function: Relu
3. Dense layer; input size: 24; output size: action size; activation function: Sigmoid

The network configuration outputted by Python is depicted in Figure 3.16. This is a classification problem which wants to determine the action must be either 0 or 1.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 24)                72
_________________________________________________________________
dense_2 (Dense)              (None, 24)                600
_________________________________________________________________
dense_3 (Dense)              (None, 2)                 50
=================================================================
Total params: 722
Trainable params: 722
Non-trainable params: 0
```

**Figure 3.16: The Double DQN network configuration consisting of three dense layers from the sequential model of Keras, generated by Python**

Before running the training session, we need to define a loss function for the network compiler. The Huber loss function is used for the training of this network which is introduced by Peter Huber in 1964 [86]. Also it is necessary to initialize and set some hyper-parameters for the network. These hyper-parameters are presented in Table 3.2.

**Table 3.2: some hyper-parameters for training the Double DQN**

| Parameter | Value | Parameter | Value | Parameter | Value |
|---|---|---|---|---|---|
| Learning rate | 0.0001 | Exploration ($\epsilon$) | 1.0 (decays) | Episodes | 150 |
| Discount factor | 0.2 | Minimum $\epsilon$ | 0.05 | Steps at each episode | 3 |
| Huber loss delta | 1.0 | Decay of $\epsilon$ | 0.999 | Epochs at each step | 1 |

For the training process, firstly the UDP communication port is enabled on Python and then the Double DQN core is configured and run to be ready for receiving data from Simulink. Afterwards the Simulink model including the UDP communication blocks is initialized through MATLAB and then run to provide Python core with the states and other packaged data. Python receives the sent packaged data from Simulink, unpacks them and uses them for training the network. At the next step it updates the first Q-Network and generates the action from the other Q-Network which is then packed and sent back to the Simulink model through UDP communication. This process is done over and over so that the trial-and-error behavior behind the RL concept improves itself.

The "Restart" part in the Simulink model which is clearly seen in Figure 3.15 is basically an assertion function controlling whether the practical limitations on the SOC of batteries are not violated during the training process. If the SOC of any of the batteries wants to go above 100 percent which means overcharging the battery, or below 40 percent which drops the battery charge and causes damage to the battery, this system stops that episode and a new episode of the training is started again. This also prevents the accumulation of tremendous negative rewards in order to omit underestimation of Q-values in the training process.

There are many parameters influencing the performance quality of the Q-Network. Some of these are referred to as hyper-parameters which were set before. We can change this hyper-parameters in order to improve the performance of the training. The network architecture plot generated by the Tensorboard library for our training network is depicted on Figure 3.17 in which the loss functions,

optimizers, and additional dense layers which are generated by the Keras interpreter for testing are shown.

After training on 150 episodes (complete simulations) which took almost 18 hours on a computer with an Intel Core i7 – 4790 CPU at 3.60 GHz, a 1.0 GB AMD Radeon R5 240 Graphic Card, and 16 GB of installed memory (RAM), a somehow optimal Q-Network is obtained which can be used for any further simulation. For running the simulation again we need to configure the UDP communication port on Python and initialize the Simulink model. Then the saved network is loaded on Python and the Simulink model can be run. The Q-Network receives the states and based on its evaluation presents the action to the Simulink model. Figure 3.18 shows a block of the running commands in Python interpreter while the simulation is running.

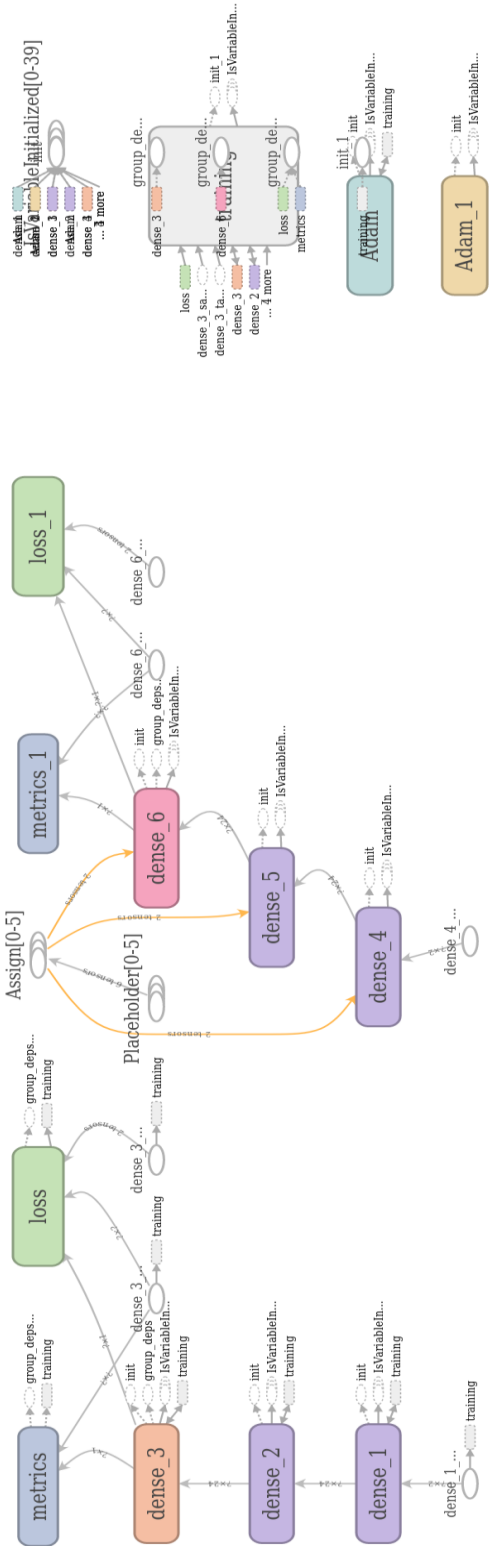


**Figure 3.17: Training and testing network architecture generated by the Tensorboard library**

```
System ready on 127.0.0.1

States are:  95.0 80.0

Sep is: 0

States are:  94.97605364487664 76.8119619046021

Sep is: 0

States are:  94.95285547781832 73.97754022771257

Sep is: 0
```

**Figure 3.18: A small section of the Python output while the simulation is running**

The Deep RL-based EMS which has been designed using a Double DQN in this part of the research provides us with the results shown in Figure 3.19, Figure 3.20, and Figure 3.21 for the main battery auxiliary battery and the battery isolator successively.

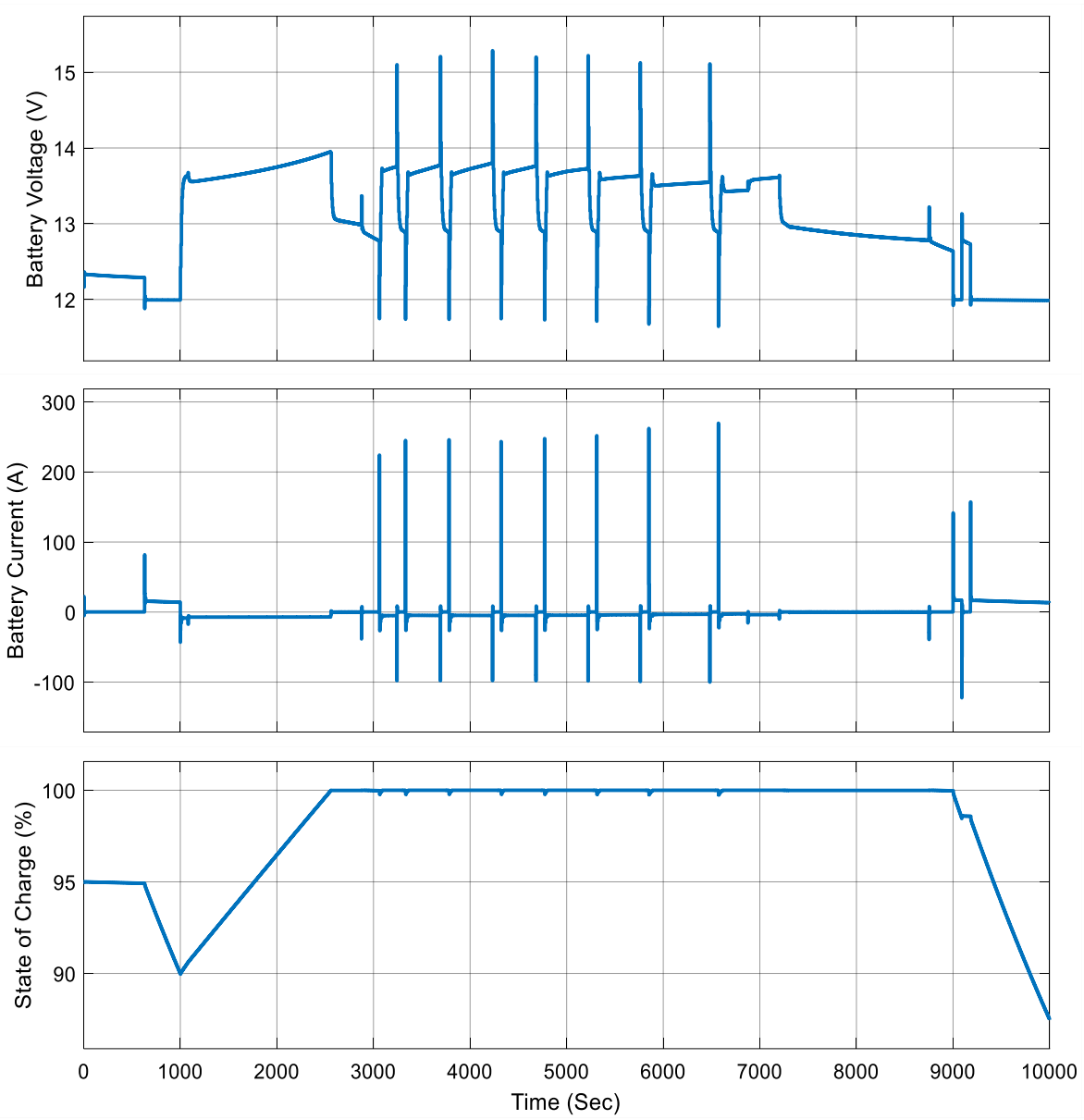


**Figure 3.19: Voltage, current, and SOC of the main battery for the Deep RL-based EMS**

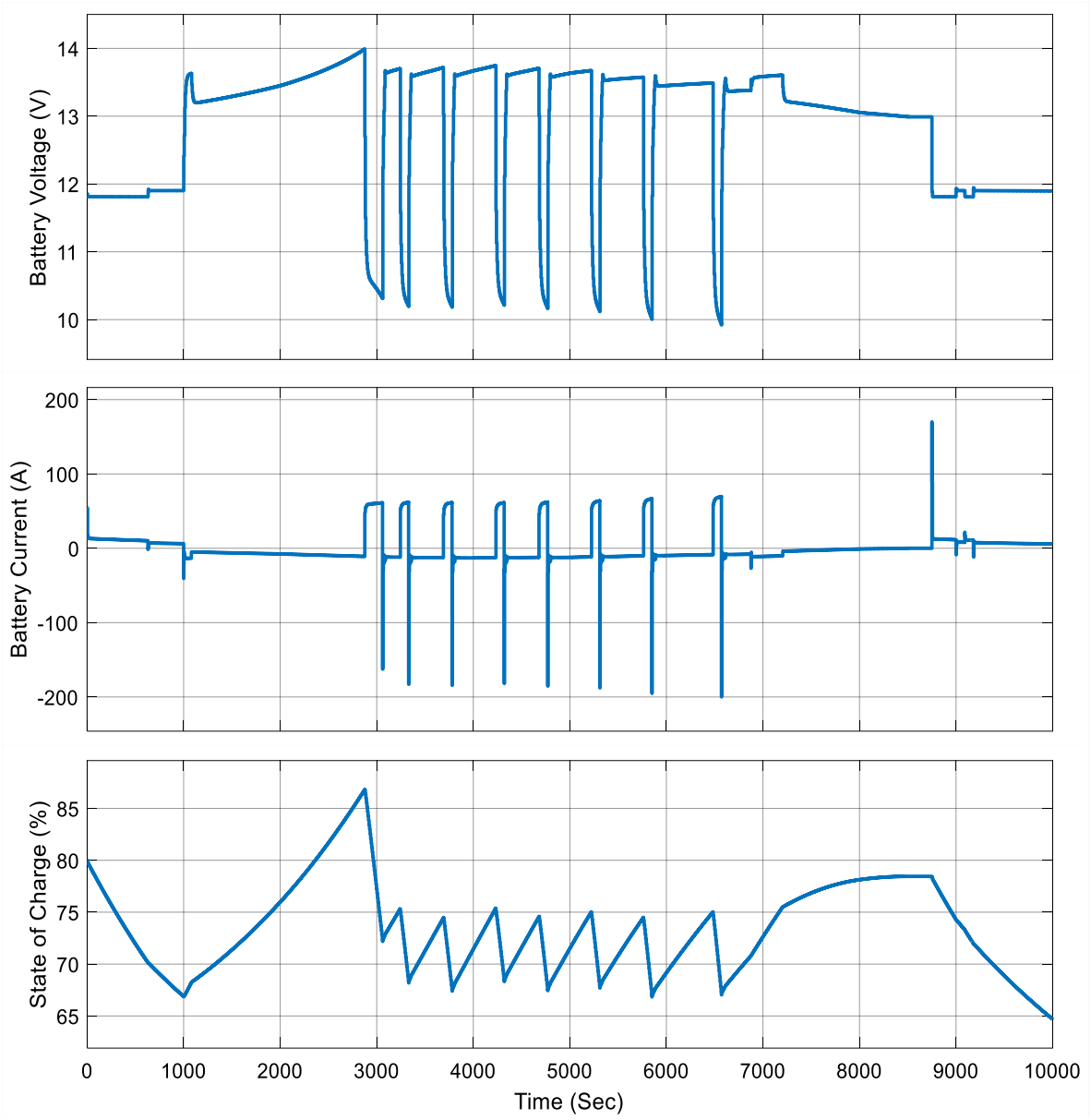


**Figure 3.20: Voltage, current, and SOC of the auxiliary battery for the Deep RL-based EMS**

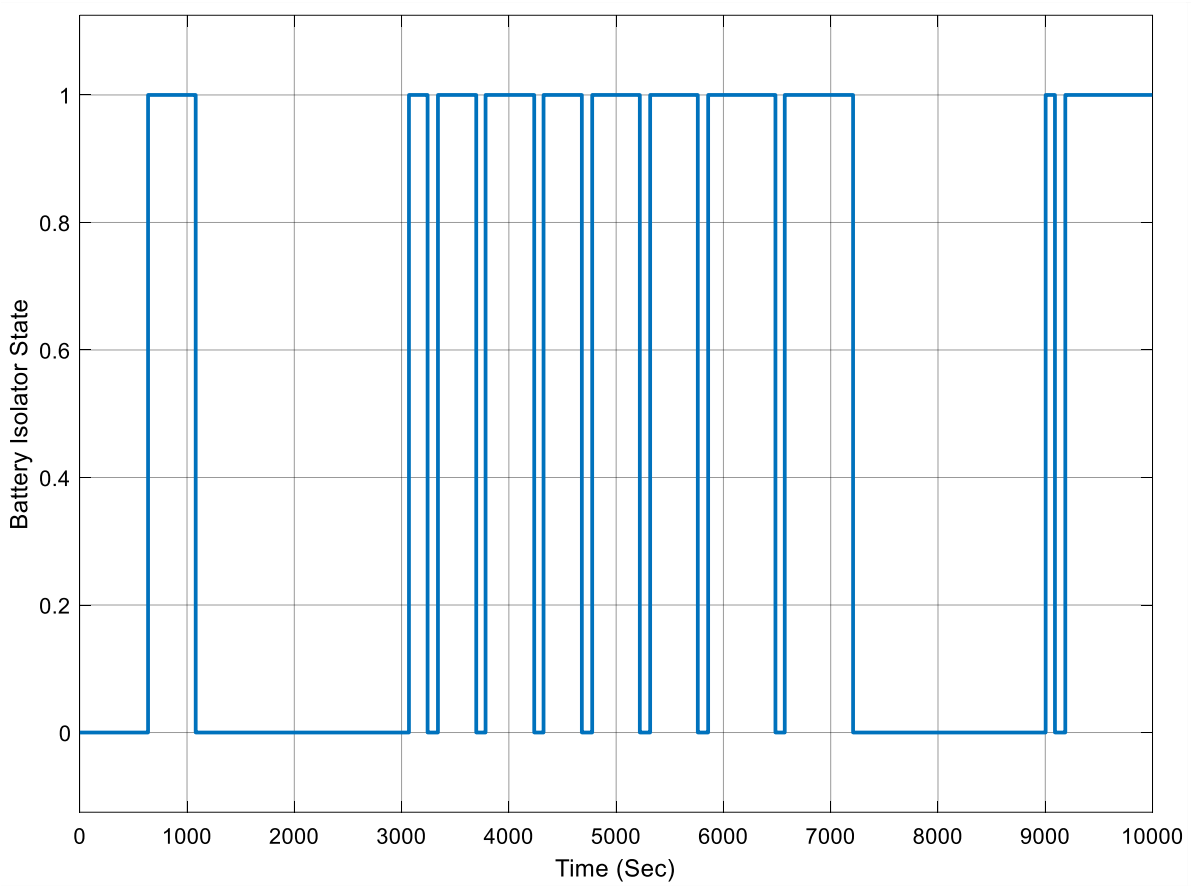


**Figure 3.21: State of the battery isolator switch for the Deep RL-based EMS**

## 3.6 Comparison of Different Scenarios

Three different variants of EMS system have been developed and explained in previous sections of this chapter. The rule-based EMS follows simple rules and is written in a MATLAB Simulink function block based on some conditions defined on the SOC of main and auxiliary battery. The RL-based scenario aims at automating the EMS in a way that no direct rules or conditions are required for the controller. The control of the battery isolator switch is done via searching through a generated Q-table obtained from the training process. The Deep RL-based EMS strategy addresses the size problem of the Q-Learning approach which is the RL method used for the RL-based EMS. Also it solves the problem of requiring explicitly designed discrete state space and accepts continuous state space which is a profit of using Deep Learning approaches to empower Q-Learning. A Double DQN has been used for the purpose of developing the Deep RL-based EMS which benefits from two different networks for

choosing actions and action evaluation, as well as the experience replay approach which speeds up the learning process and omits overestimation of the results.

The results pertaining to the voltage, current and SOC of the main and auxiliary batteries were included and discussed for each of the developed EMS strategies. Each of these approaches introduce some limitations which also were explained. Now if we want to compare the results in terms of the fuel consumption index for these three scenarios with the case that SIR system is not considered for and the engine needs to stay on and run, the following results are obtained which are demonstrated in Figure 3.22 and Figure 3.23.

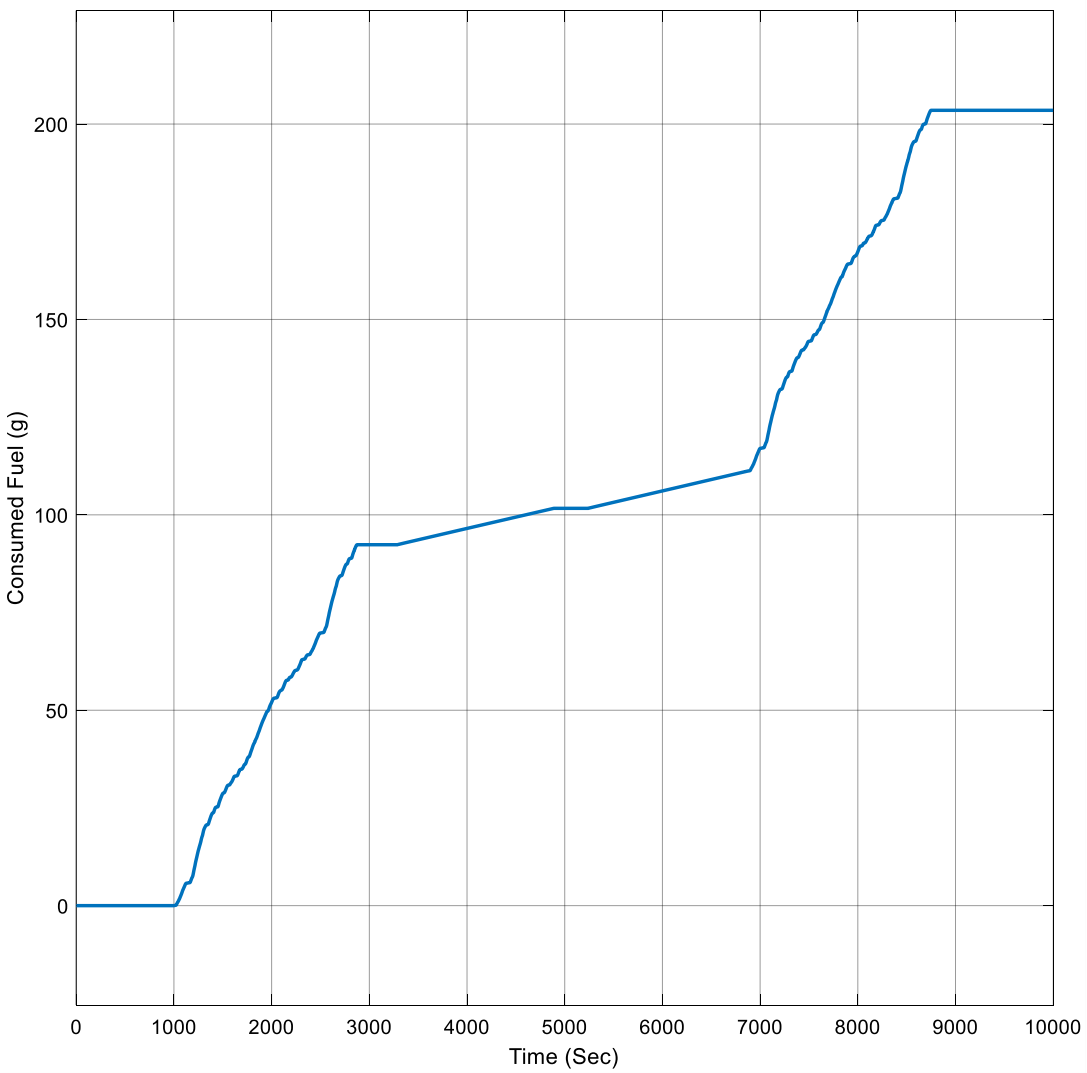


**Figure 3.22: Accumulated amount of the consumed fuel for the case of idled engine**

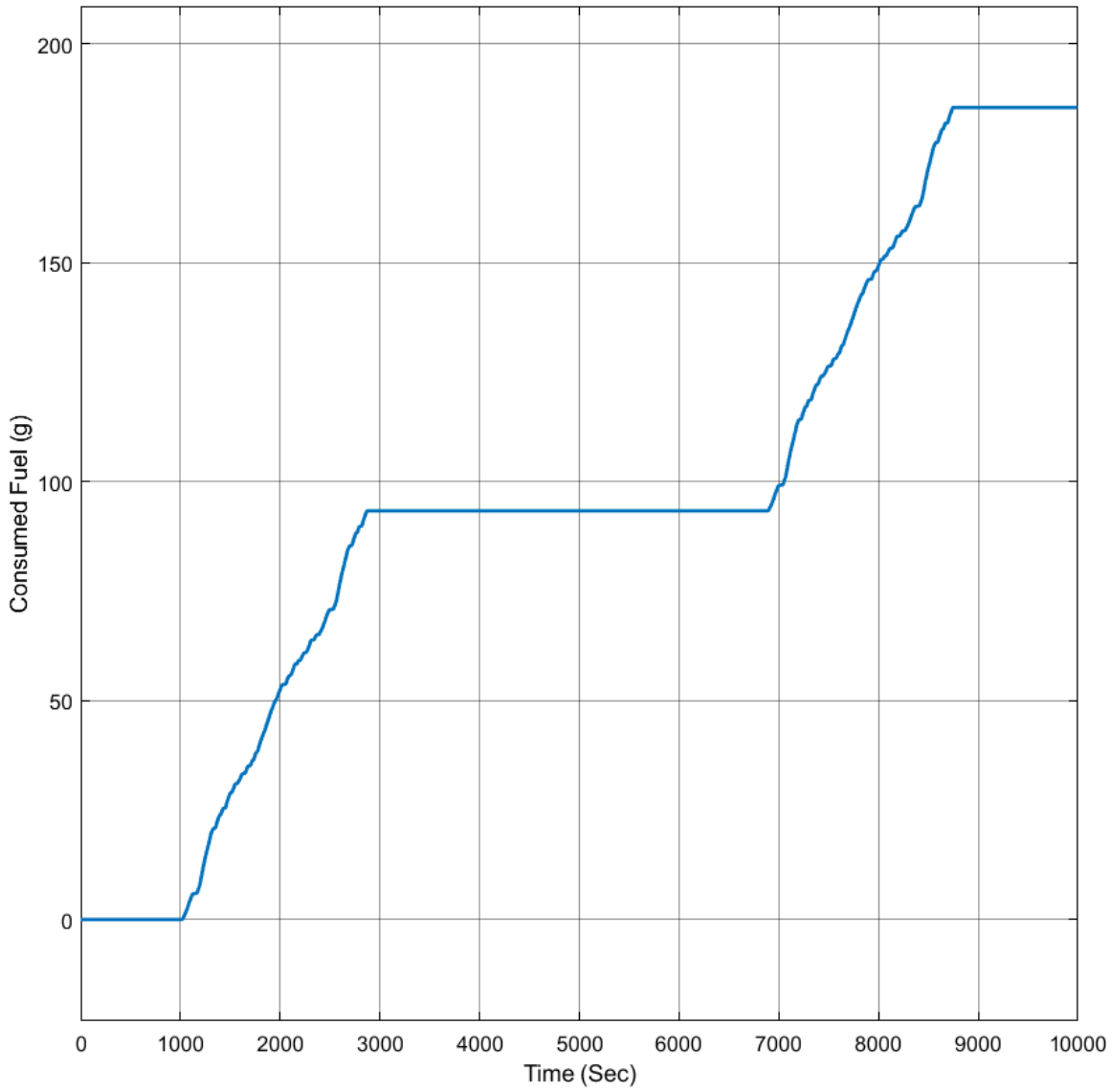


**Figure 3.23: Accumulated amount of the consumed fuel for the case in which the engine is off and SIR system provides the auxiliary loads with required electrical power**

The results showing that employing our Deep RL-based EMS system results in a 9.34% improvement in fuel consumption compared to the case the engine remains idled. Also the flat line in the fuel consumption plot which corresponds to the on-site operation of the vehicle shows that during this time the engine can be completely turned off and all the required electrical power for the auxiliary loads is

provided by the main and auxiliary batteries. Also it is worth mentioning that when the batteries are connected together (or equally the battery isolator is closed), both of them are being charged by the solar panels or the shore power for the cases the vehicle engine is off and the alternator does not charge them.

The developed Deep RL-based EMS has also been tested with two other driving cycles for comparison and to prove its capability for being used in different driving scenarios. The results show that the developed system can be extended to other driving cycles without any modifications. The next section is dedicated to review the results from HWFET and HD-UDDS driving cycles and comparing them through some figures.

## 3.7 Results for HWFET and HD-UDDS Driving Cycles

The developed intelligent Deep RL-based EMS was tested with a combined idling state and FTP drive cycle and the results were presented in the third chapter of this thesis. This appendix focuses on the results from testing the developed EMS system for two other driving cycles, namely HWFET which is for highway driving and HD-UDDS which is specifically designed for high duty vehicles in urban driving. The combined HWFET driving cycle is shown in Figure 3.24. It has two parking states at the beginning and the end, two driving stages, and on-site operation.

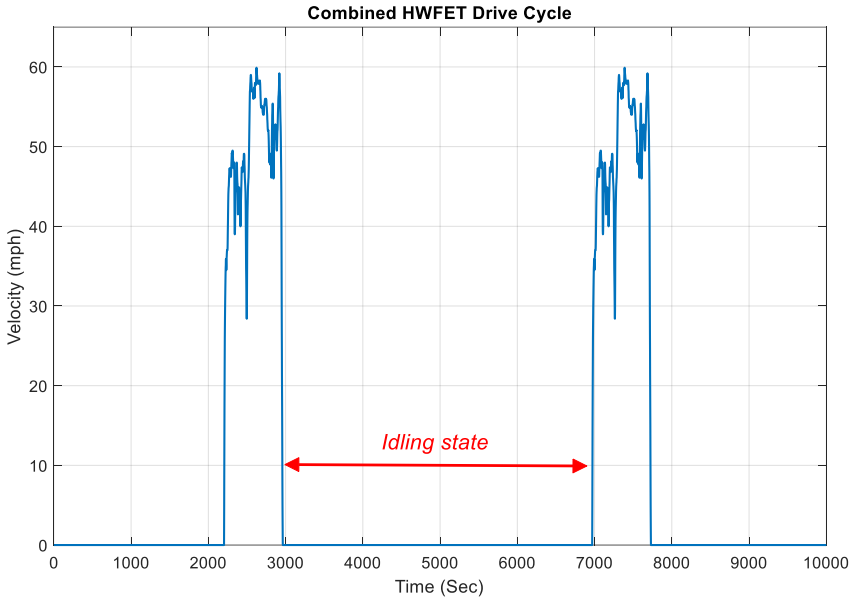


**Figure 3.24: Combined HWFET driving cycle for a service vehicle**

Also the combined HD-UDDS driving cycle can be found in Figure 3.25. It follows the same states with HD-UDDS drive cycle.

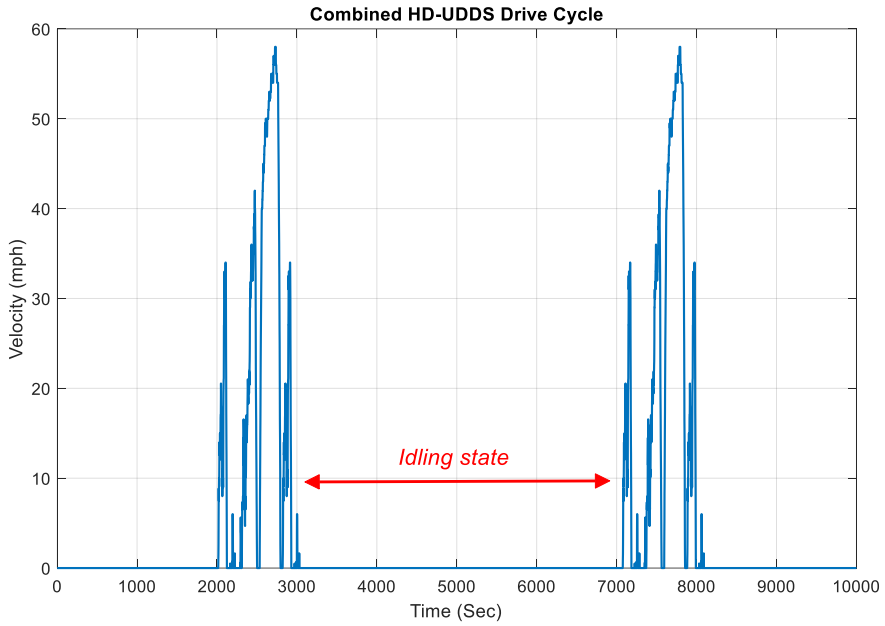


**Figure 3.25: Combined HD-UDDS driving cycle for a service vehicle**

Testing the developed Deep RL-based EMS with the HWFET driving cycle results in Figure 3.26 and Figure 3.27 for the main battery and auxiliary battery respectively.

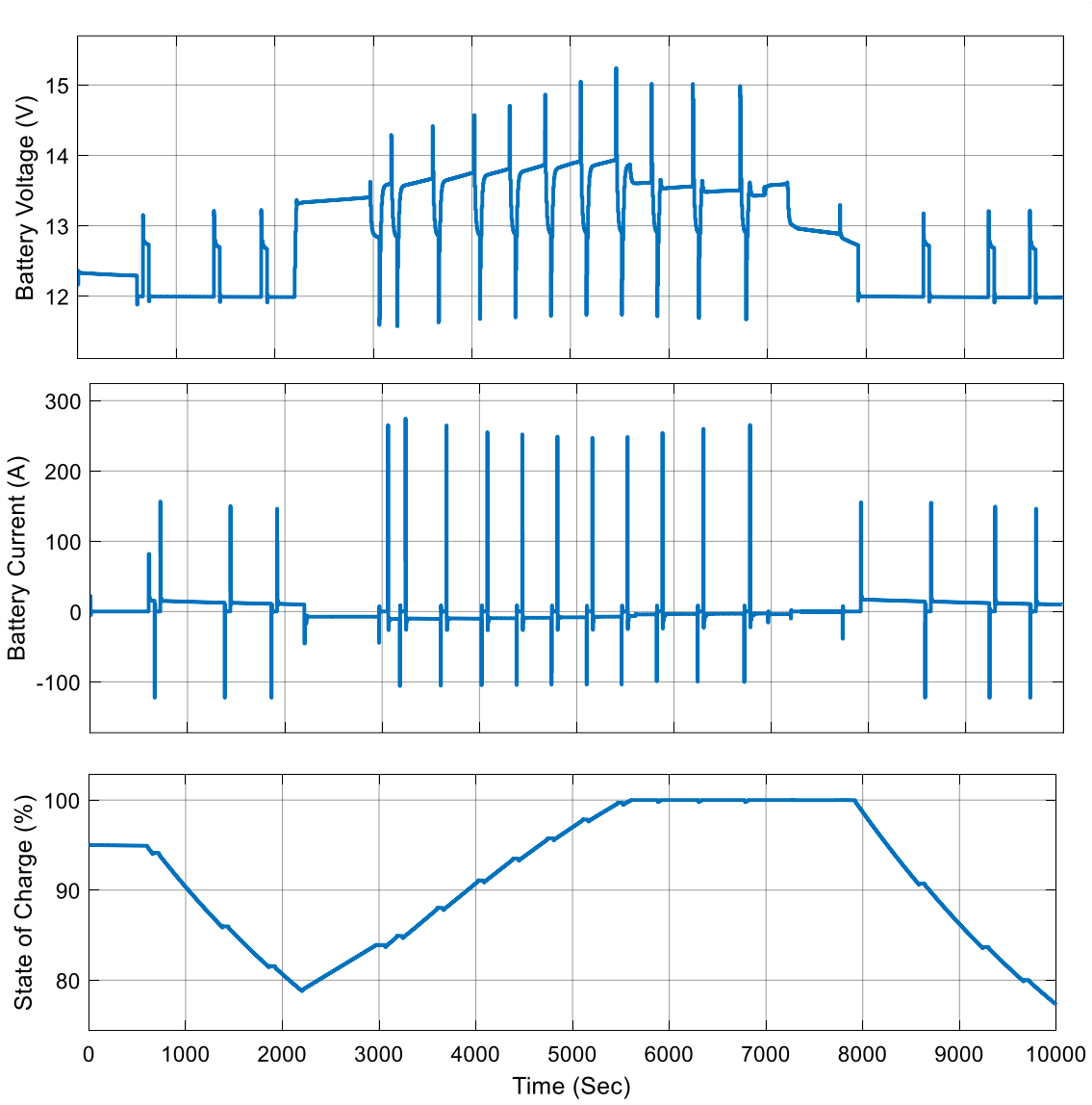


**Figure 3.26: Voltage, current, and SOC of the main battery for the Deep RL-based EMS in HWFET driving cycle**

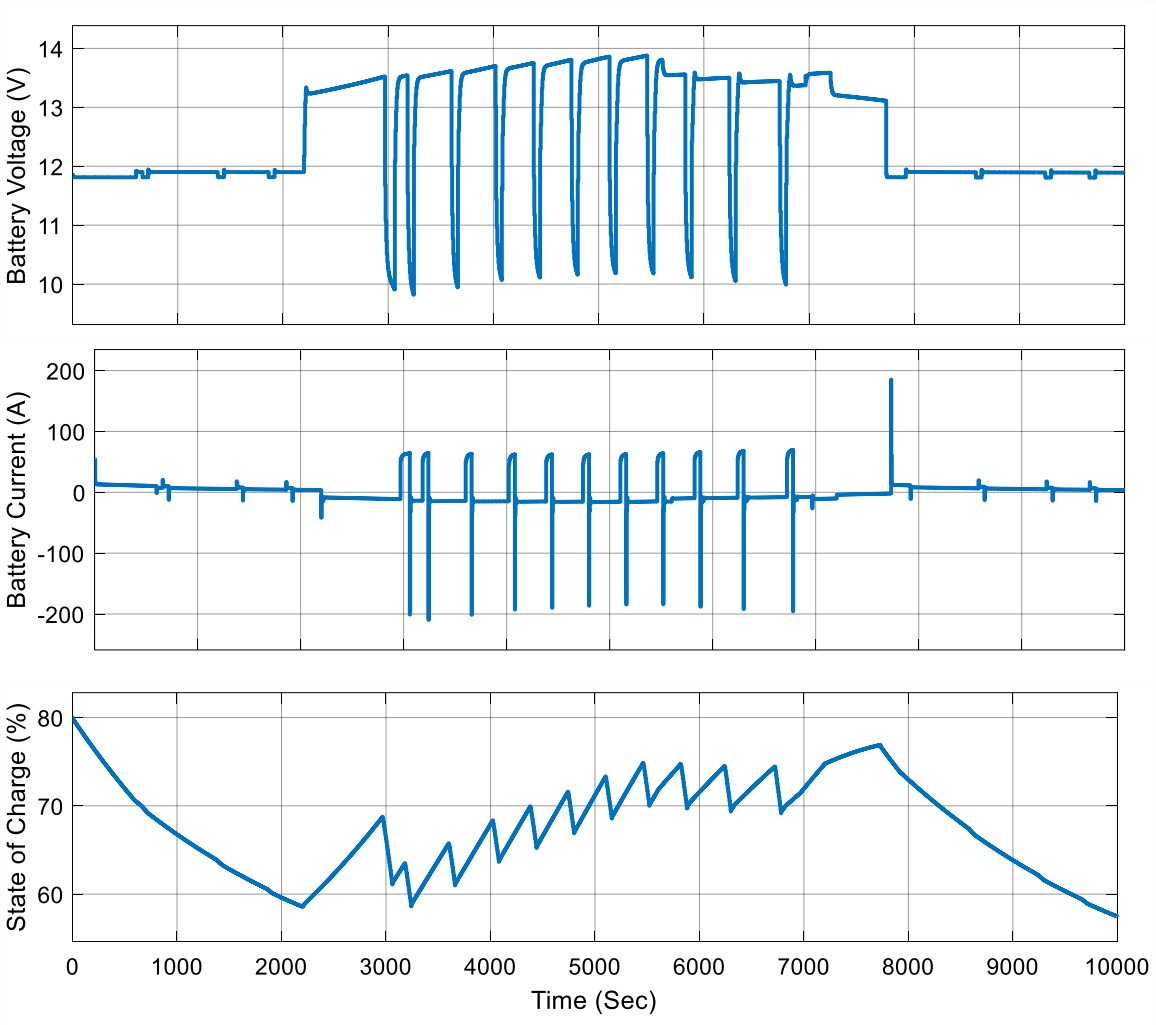


**Figure 3.27: Voltage, current, and SOC of the auxiliary battery for the Deep RL-based EMS in HWFET driving cycle**

Also the battery isolator state (open or close) as well as the consumed fuel in grams is depicted in Figure 3.28 and Figure 3.29.

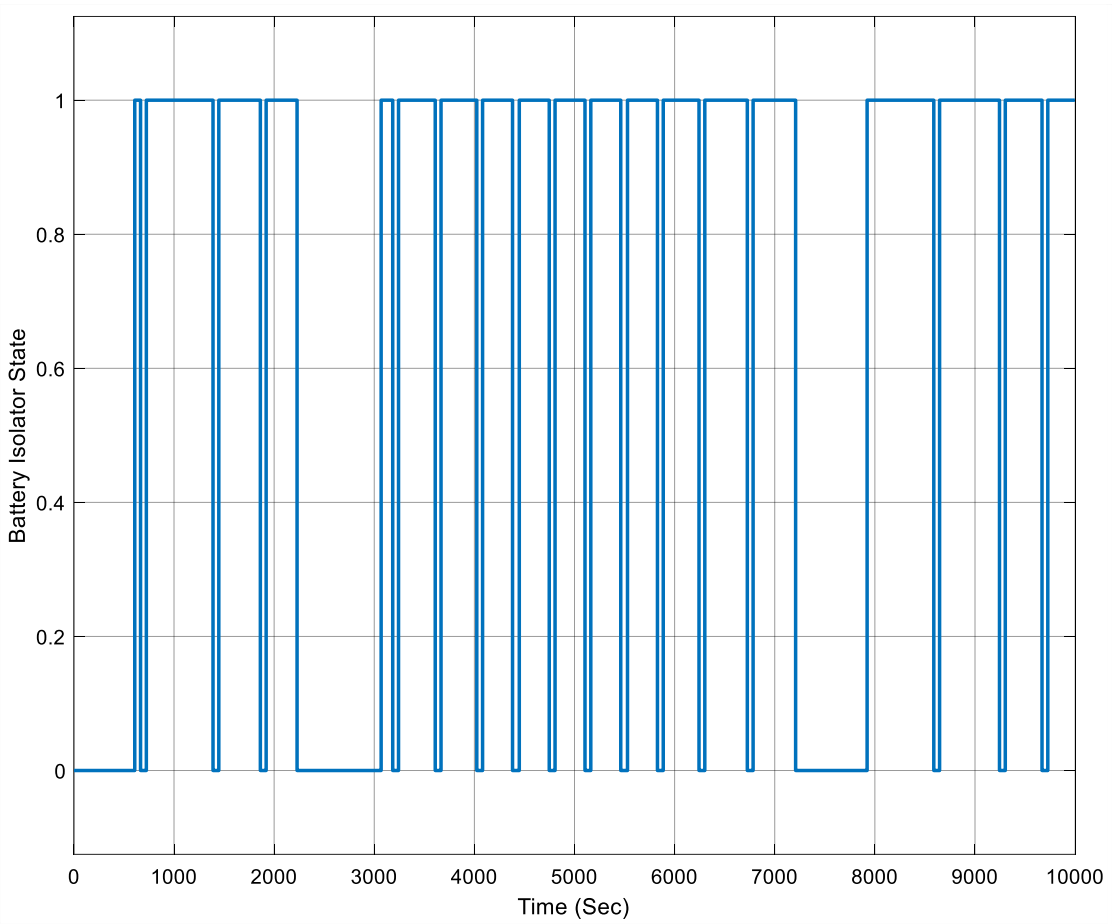


**Figure 3.28: State of the battery isolator switch for the Deep RL-based EMS in HWFET driving cycle**

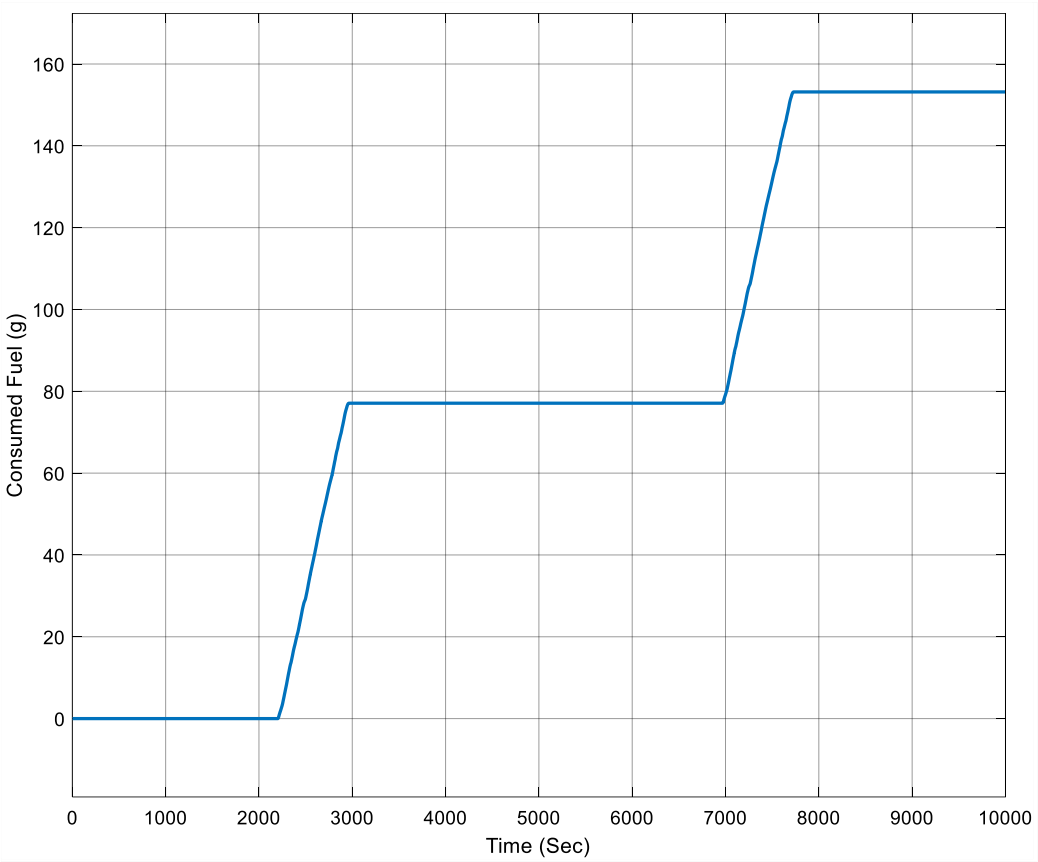


**Figure 3.29: Accumulated amount of the consumed fuel for the Deep RL-based EMS in HWFET driving cycle**

For comparison, the figures relating to the main battery, auxiliary battery, battery isolator and consumed fuel are demonstrated respectively in Figure 3.30, Figure 3.31, Figure 3.32, and Figure 3.33 for the case of having the engine idled or what we call “No SIR”. The fuel consumption for the Deep RL-based EMS is 11.3% lower than the case with idled engine which shows a great performance.

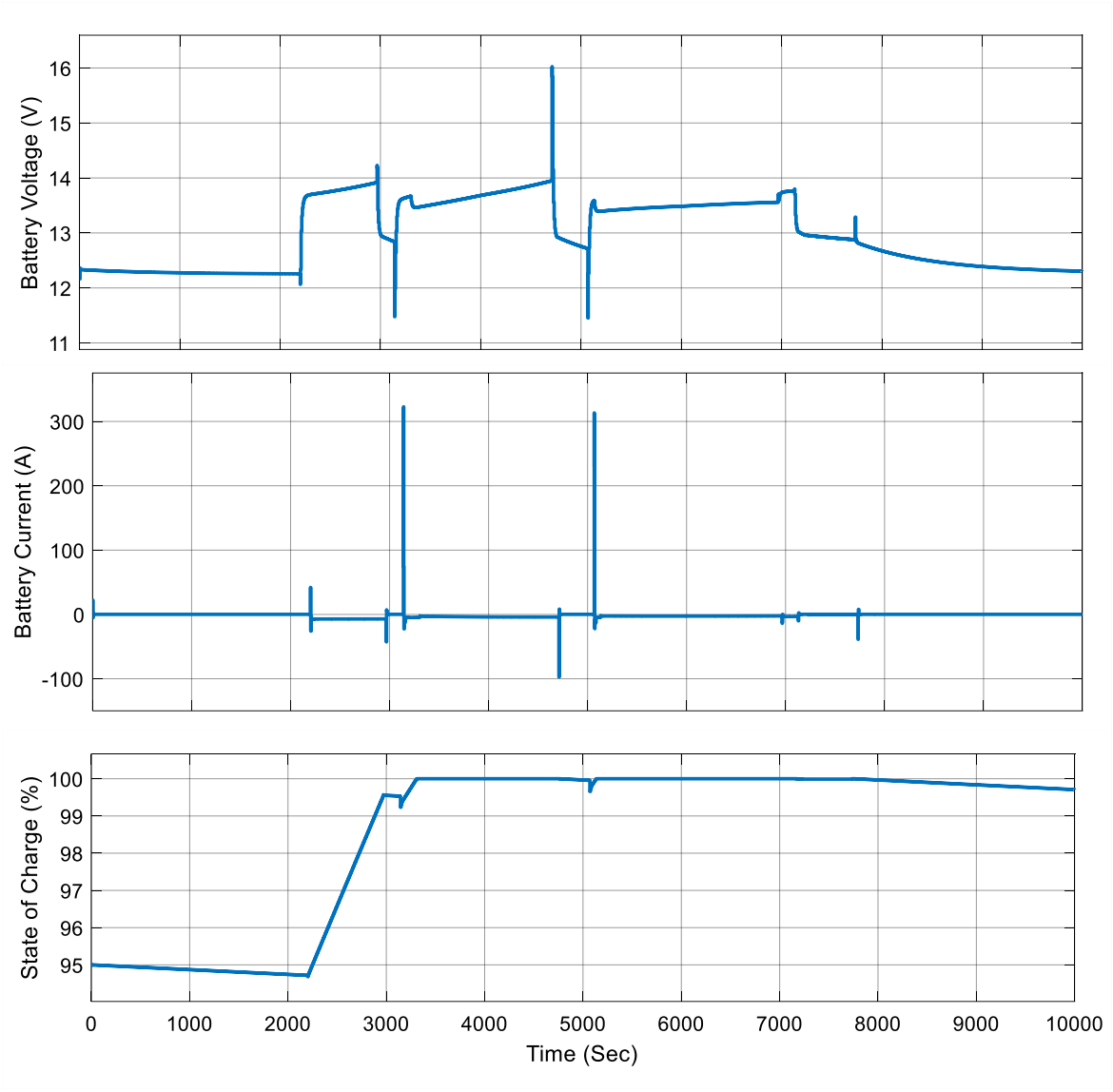


**Figure 3.30: Voltage, current, and SOC of the main battery for the idled engine case in HWFET driving cycle**

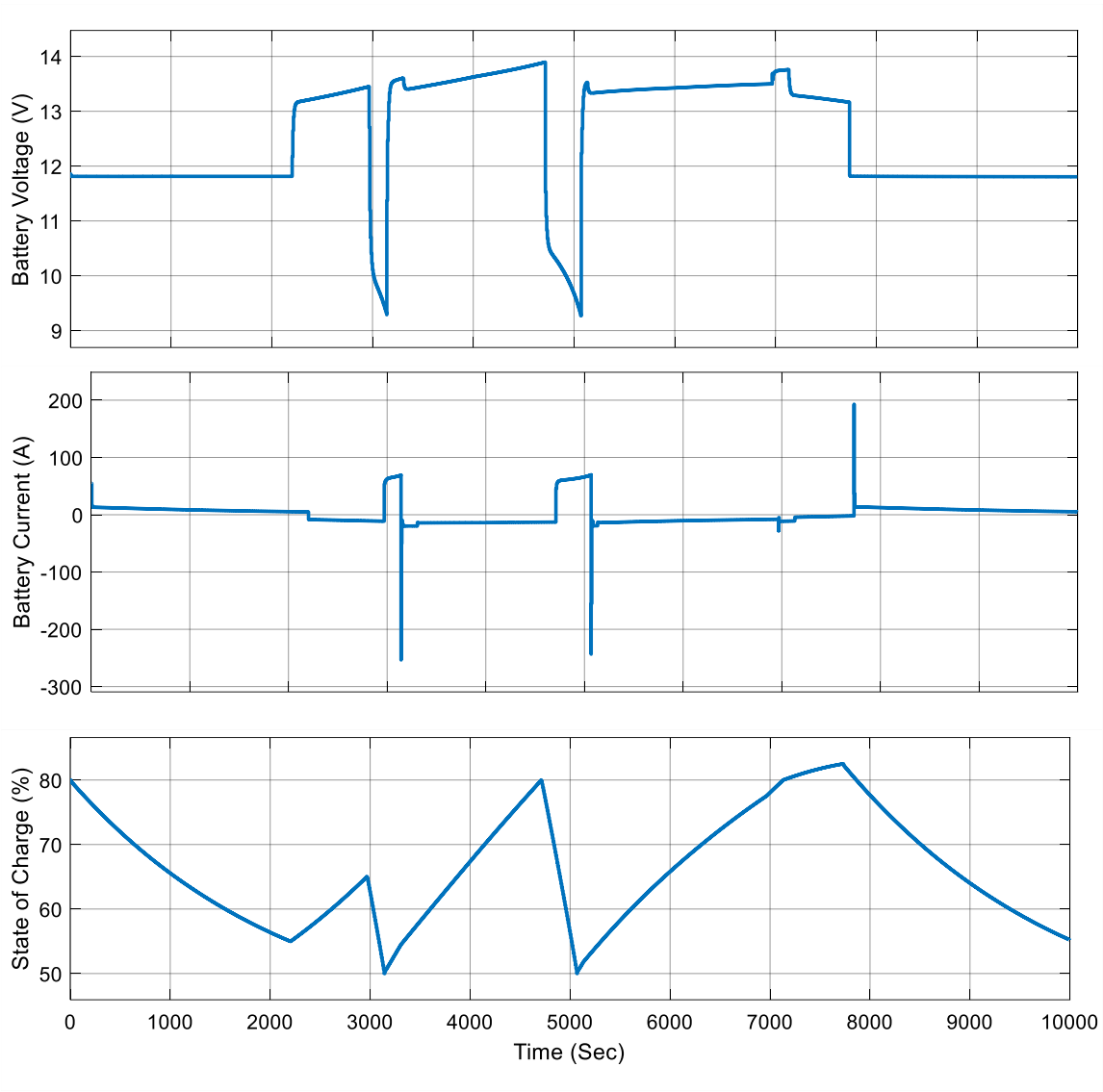


**Figure 3.31: Voltage, current, and SOC of the auxiliary battery for the idled engine case in HWFET driving cycle**

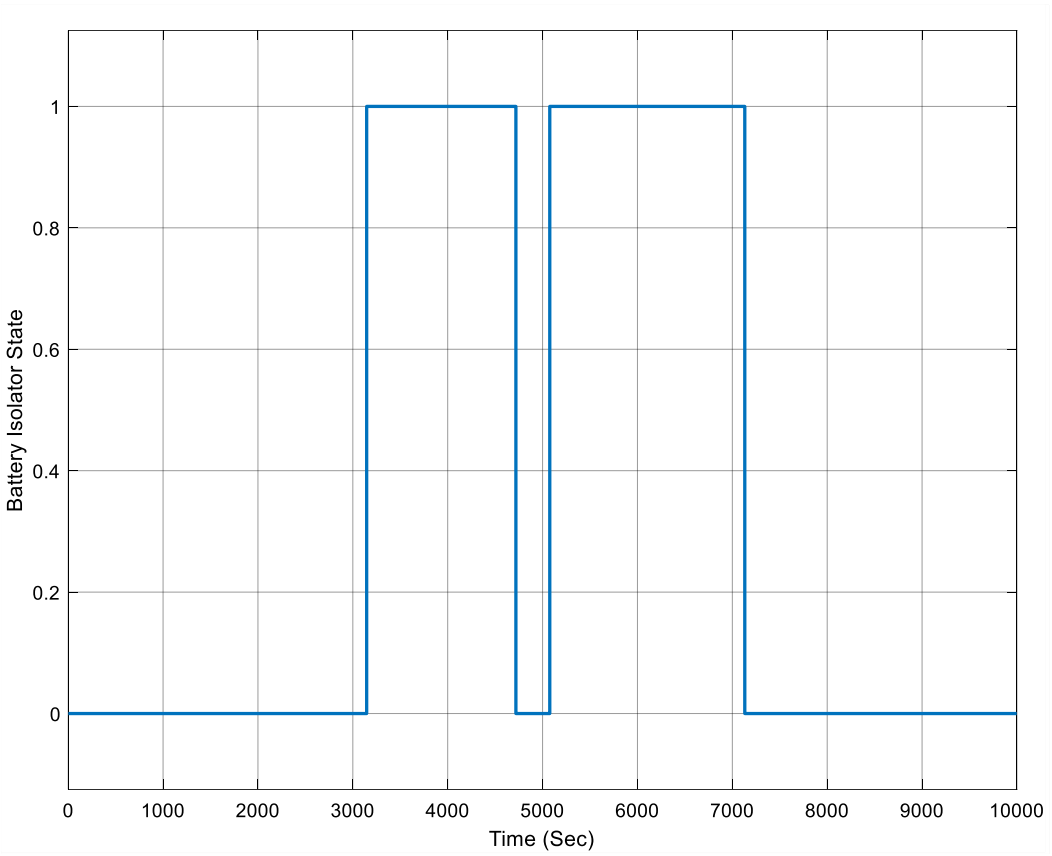


**Figure 3.32: State of the battery isolator switch for the idled engine case in HWFET driving cycle**

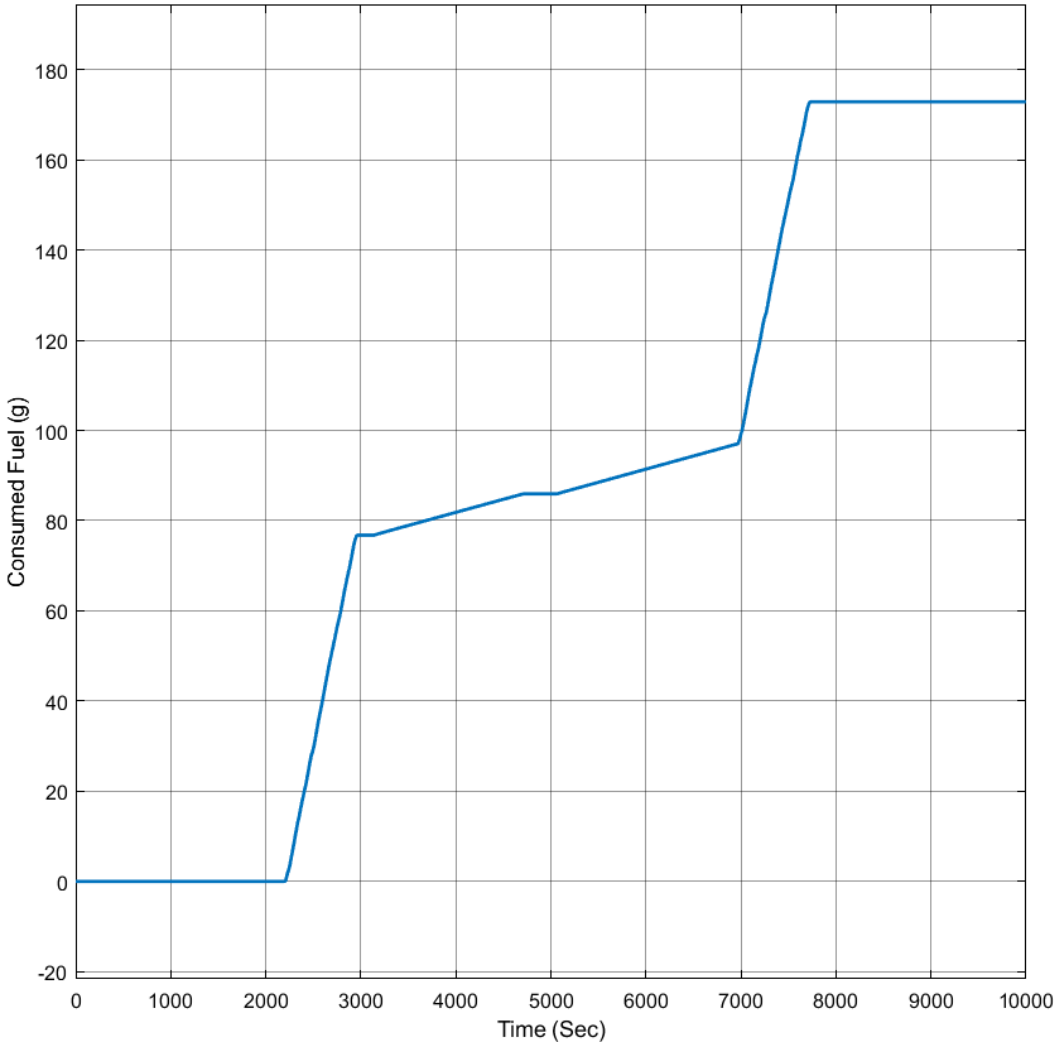


**Figure 3.33: Accumulated amount of the consumed fuel for the case of idled engine in HWFET driving cycle**

Now if we want to test the developed Deep RL-based EMS with a driving cycle specific to the high duty vehicles, the HD-UDDS can help us through. The resulting figures for the main battery and auxiliary battery are presented below in Figure 3.34 and Figure 3.35.

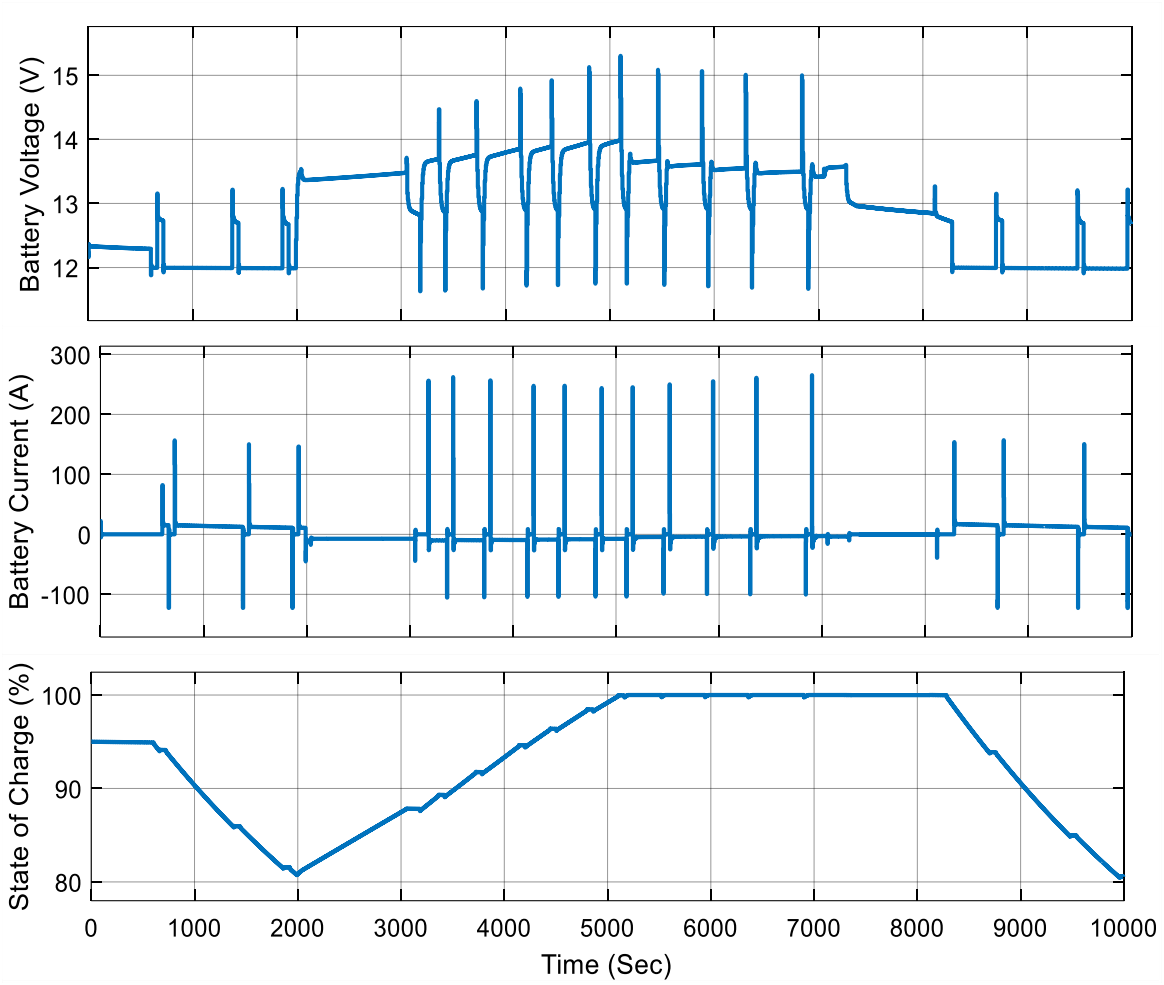


**Figure 3.34: Voltage, current, and SOC of the main battery for the Deep RL-based EMS in HD-UDDS driving cycle**

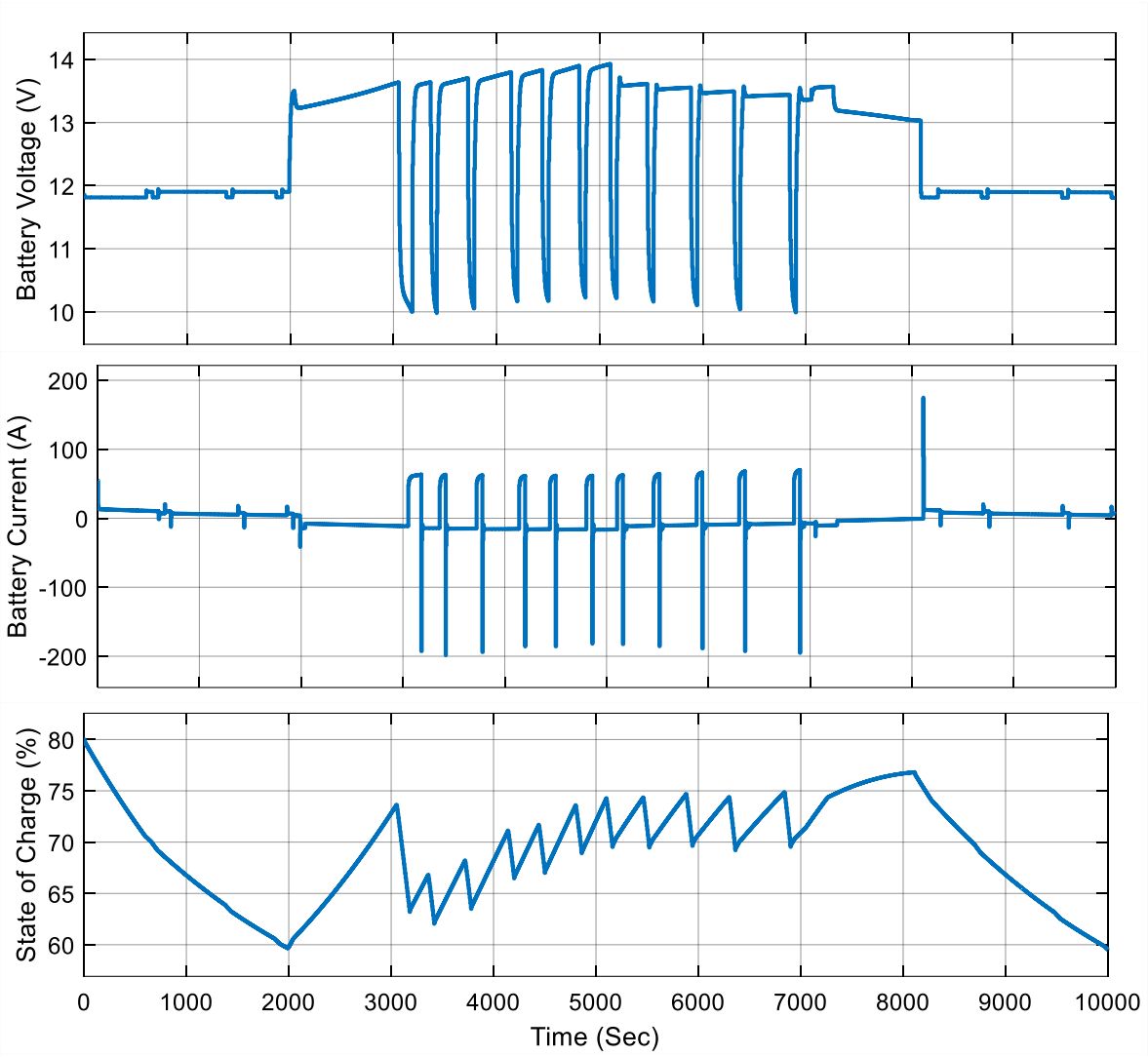


**Figure 3.35: Voltage, current, and SOC of the auxiliary battery for the Deep RL-based EMS in HD-UDDS driving cycle**

Also the battery isolator state and the consumed fuel in grams is depicted in Figure 3.36 and Figure 3.37 for the HD-UDDS driving cycle.

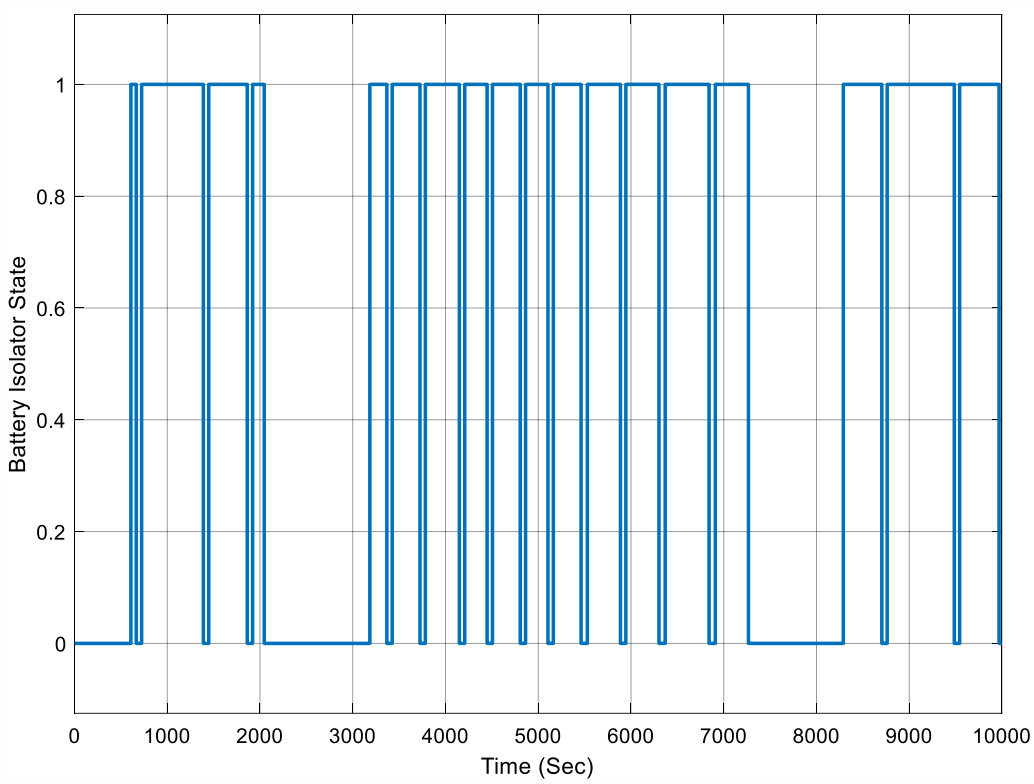


**Figure 3.36: State of the battery isolator switch for the Deep RL-based EMS in HD-UDDS driving cycle**

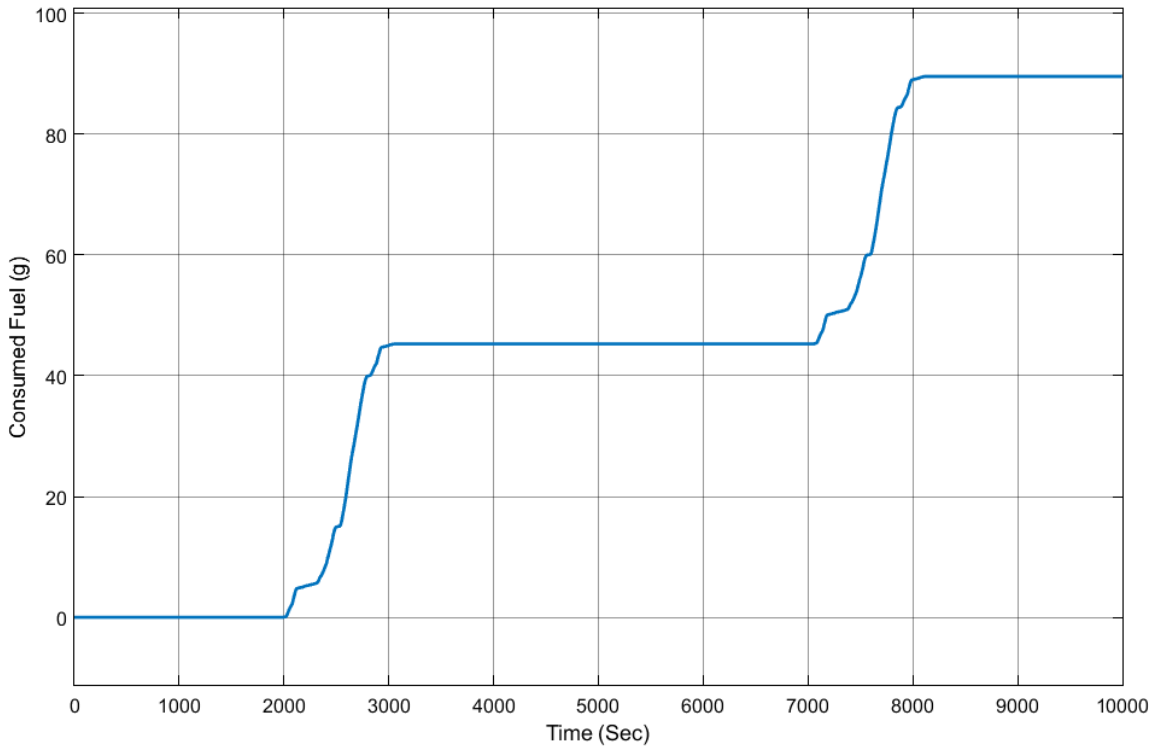


**Figure 3.37: Accumulated amount of the consumed fuel for Deep RL-based EMS in HD-UDDS driving cycle**

The figures showing the results for main battery, auxiliary battery, battery isolator and consumed fuel for the idled engine scenario are demonstrated in Figure 3.38, Figure 3.39, Figure 3.40, and Figure 3.41 for comparison. The fuel consumption has been improved by 18% by employing the Deep RL-based EMS which shows great savings.

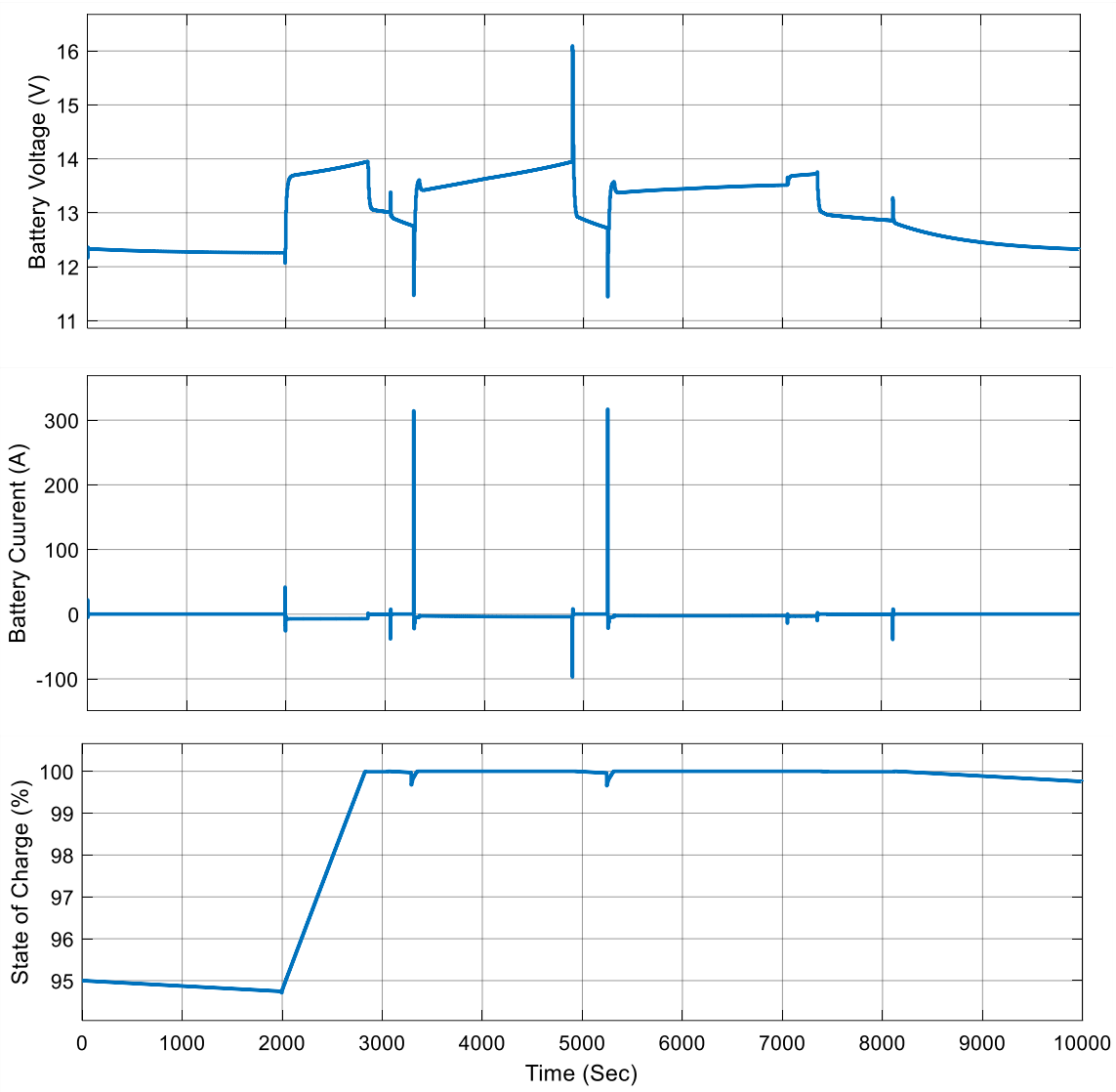


**Figure 3.38: Voltage, current, and SOC of the main battery for the idled engine case in HD-UDDS driving cycle**

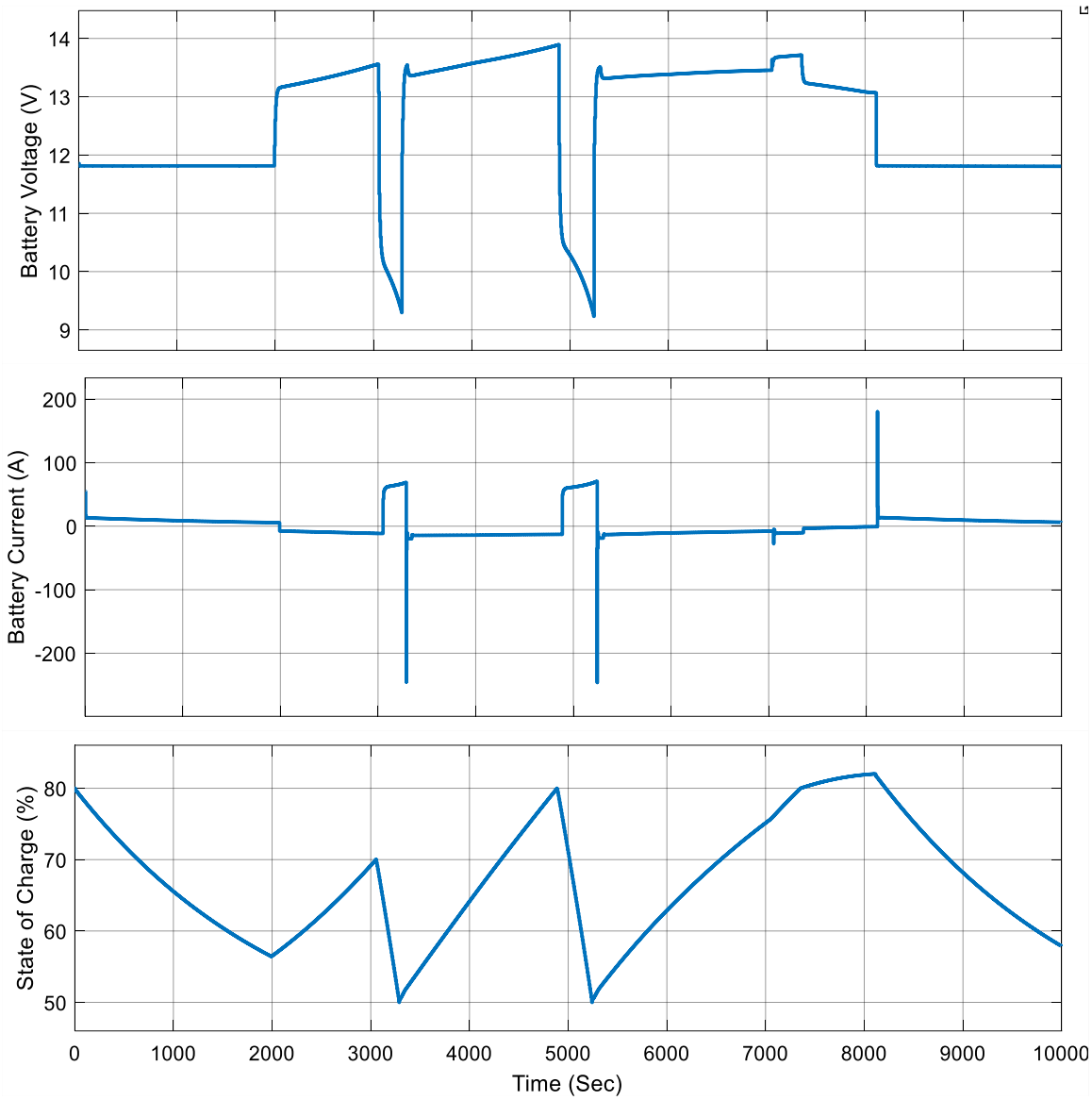


**Figure 3.39: Voltage, current, and SOC of the auxiliary battery for the idled engine case in HD-UDDS driving cycle**

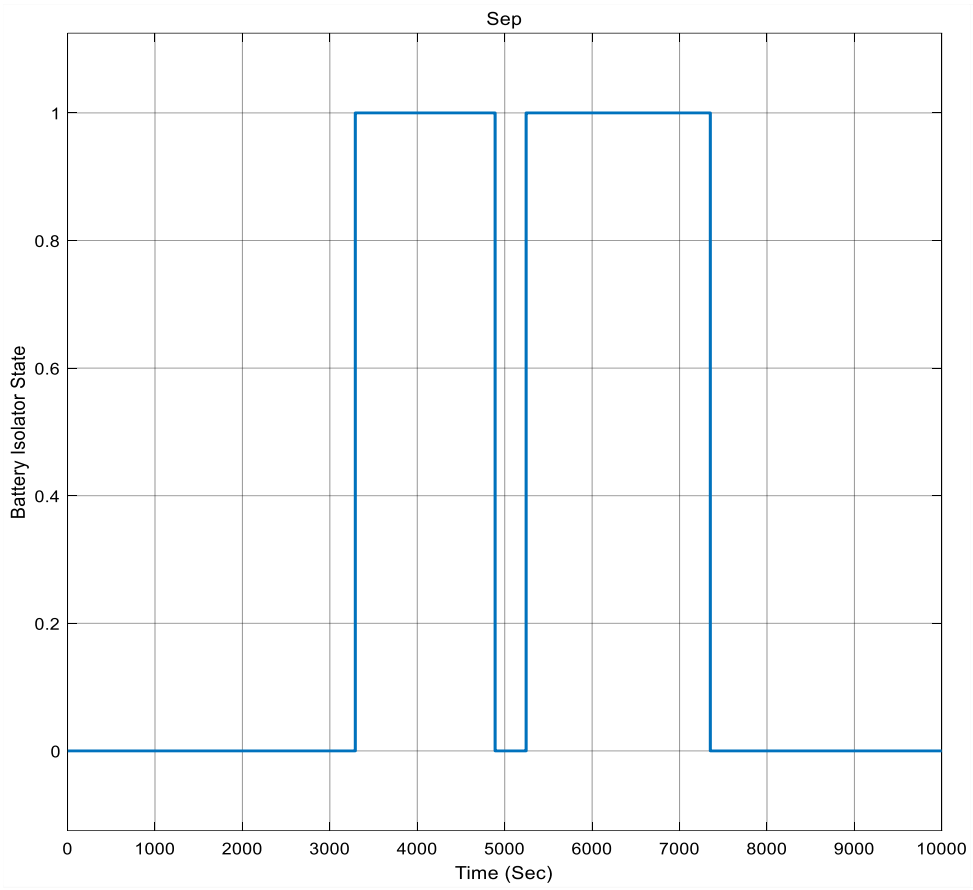


**Figure 3.40: State of the battery isolator switch for the idled engine case in HD-UDDS driving cycle**

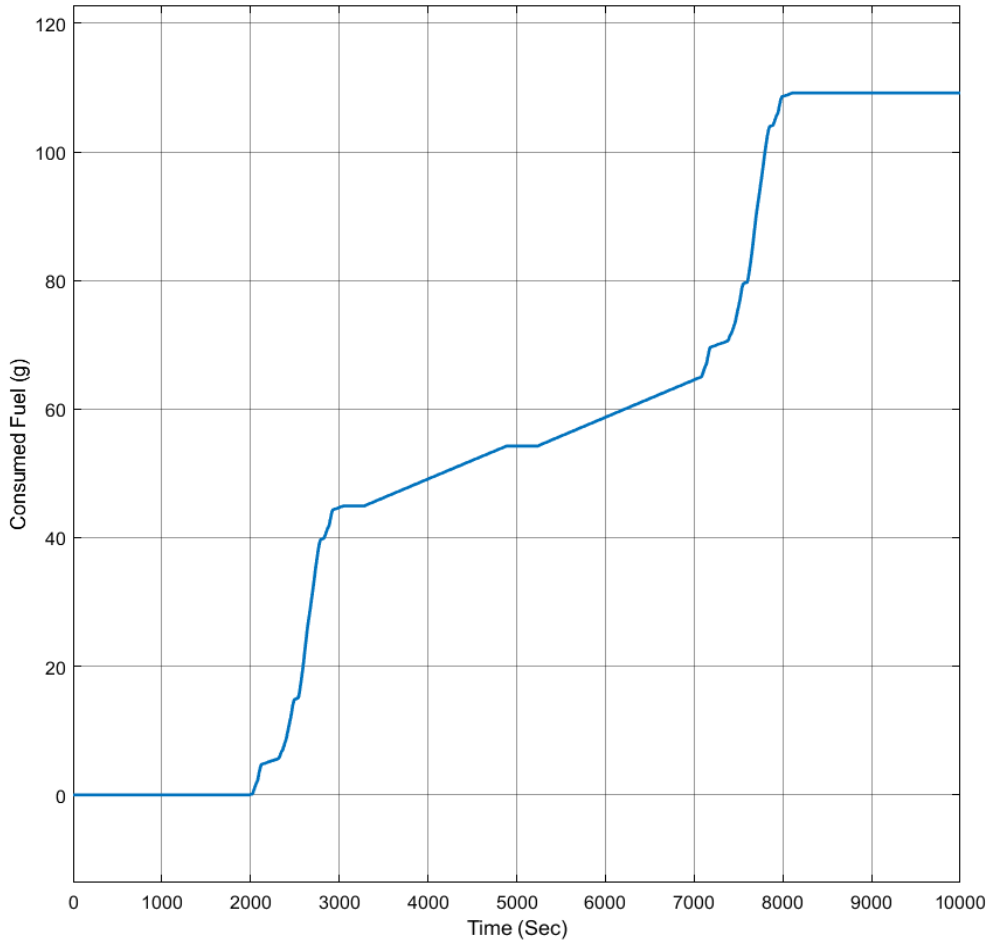


**Figure 3.41: Accumulated amount of the consumed fuel for the case of idled engine in HD-UDDS driving cycle**

## 3.8 Summary

A brief review of the implemented SIR system on two service vehicles in the City of Waterloo's fleet has been presented at the beginning of this chapter. The whole idea of using a battery isolator switch which controls whether the main and auxiliary batteries as well as their dependencies are connected or not based on their voltage level is explained. This research aimed at replacing this battery separator with a command triggered switch and developed three different EMS systems with different architectures and supporting concepts. All of these approaches were elaborated on in this chapter and the design process is also included for future efforts and developments. The intelligent Deep RL-based EMS strategy showed promising results and proved a 9.34% reduction in fuel consumption in the

simulations which is translated into GHG emissions reduction as well. Two other types of the EMS scenarios which were developed in this research, namely Rule-based and RL-based also show good results and can replace the voltage level controlled separator switch.

Putting all together clarifies that using the developed intelligent EMS strategy with the SIR system can maximize the fuel savings and GHG emissions reduction for the fleet service vehicles to a reasonable extent.

Also it is worth highlighting that the adapted and improved Q-Learning and Double DQN algorithms to the control framework paves the way for utilizing them in other control problems with just a little modifications regarding the problem parameters. This framework adaptation and improvement can be the theoretical contribution of this thesis to help the other researchers with their control problems.

# Chapter 4
# Data Acquisition

This chapter introduces the importance of data collection for this project and how the acquired data can lead to improving the performance of the EMS and even better predicting the behavior of the environment as well as the system. After that the chapter explains different kind of data which is used in this research and the required sensors for measuring them. Furthermore, the previously installed DAQ system is investigated and the reason for designing a new DAQ platform is stated. Finally, development process of the new DAQ system is presented and some enclosures are designed for different parts of it in order to get prepared for installation on a vehicle. The development of this system is originally adopted from [87].

## 4.1 Importance of Data Acquisition

This is a world which is governed by data nowadays. More and more researchers all over the world are attracted to the Deep Learning approaches to solve various problems involving optimization, regression or classification of some sort of data. Even in engineering problems which require experimental results and test scenarios having access to data for comparison or creating baselines seems necessary.

The intelligent EMS system designed for the vehicular SIR system has been trained based on the simulation data and the driving cycle which were obtained from real world scenarios and dynamics of the vehicle. The solar irradiance for this research is read from a lookup table which does not consider various situations and follows a default scenario. Although the developed system is working and the results are showing improvement compared to an uncontrolled system, further data from sensors could help improving different parts of the EMS system which were skipped because of lack of the data. If we can collect the data for different scenarios and various weather conditions which affect the performance of solar panels, definitely a more accurate controller can be developed.

## 4.2 Required Data and Sensors

The previous section elaborated on why we need to measure the data and collect it in an organized way so that we can make use of it for the purpose of understanding if the system is working well and

improving its logic and behavior. This section will focus on the data we need to obtain from the SIR system and the vehicle itself, as well as the required components which measure or collect these data for us.

One thing that we need to keep track of is the generated or consumed power which is represented in Watts. Power is obtained from multiplying the current flowing through and the voltage amount on each electrical component. As an instance the amount of generated power by solar panels can be calculated from measuring the voltage and current after the solar charge controller. We also need to measure the power coming out of the alternator to charge the main battery as well. This also requires some specific voltage and current sensors. Each of these sensors are used for different parts of the system which have their own level of voltage or current, along with other limitations. Hence, the sensors we choose should meet the required working constraints in terms of acceptable voltage, current, working temperature, size, and so on.

The following sections are dedicated to the design parameters and constraints we have for choosing the sensors required for types of data which are used in this research.

### 4.2.1 Voltage Sensors

Voltage sensor has to be used for measuring the voltage on terminals of the alternator, solar charge controller, and loads. It needs to measure voltages around 20V accurately based on the simulations. A voltage error of less than 1% (detecting at least 0.01V voltage difference) is sufficient for this research. Also because of the intensive weather conditions of Canada, the working temperature range needs to cover $-40°C$ to $80°C$.

### 4.2.2 Current Sensors

To calculate the consumed power at each component of the system, the data for the flowing current is required as well. We need to measure the current flowing through the alternator, batteries, and loads. It needs to measure currents between -50A and +50A with an accuracy of at least 0.01A. The working temperature should be above $-40°C$ as well.

### 4.2.3 Temperature Sensors

To measure the ambient temperature and also the surface temperature of the solar panels, we need to provide some specific temperature sensors which are able to correctly measure the temperature in different weather conditions in Canada which can have harsh temperatures. It needs to measure the temperatures between $-40°C$ and $80°C$ with at least $0.1°C$ accuracy.

### 4.2.4 Solar Irradiance Sensor

Solar irradiance sensor is required to keep track of the times the solar panels installed on the vehicle are getting the most out of the sun energy. It should measure the irradiance of the sun in watts per square meter ($\frac{W}{m^2}$) for the range of 0 ~ 1500 $\frac{W}{m^2}$ with at least 2.5% accuracy. It needs to have the ability of temperature compensation which ensures accuracy in different temperatures [88]. The working temperature for this sensor also needs to be above $-40°C$.

## 4.3 Previously Installed DAQ System for Research

Previous parts illustrated the importance of collecting the data from different parts of the vehicle and the SIR system for the purpose of enhancing the efficacy of developed intelligent EMS systems which clearly benefit from data. Also some of the required data and types of sensors to measure them are introduced and the operating conditions are briefly presented.

This project is launched on 2015, and its system design goes back to 2016. The main market for this system is supposed to be in service vehicles, school buses, ambulances, recreational vehicles, etc. The first generation of the SIR system is installed on two service vehicles in the City of Waterloo between January 2017 and June 2017. At that time an off-the-shelf DAQ system was chosen for this project which had lots of capabilities. Its tremendous features make it the best choice for research purposes. Later in the progress of the research the necessity of designing and development of a customized and cost-effective DAQ system was figured out. This approach comes from the fact that for commercial use of the SIR system which requires affordability as a key factor, the prices should stay reasonable. This part gives some facts and details about the previously installed data acquisition platform and its components as well as some data collected from the measurements and some analysis. A table of the prices is also presented in the next section as a reference for comparison with the newly developed DAQ system.

Figure 4.1 depicts National Instruments CompactRIO (NI cRIO-9031) with the required additional blocks which makes it a perfect DAQ system and controller platform. It collects and saves the data from the SIR system sensors, battery monitoring system, and solar charge controller. If we want to be more precise, this device is an embedded controller which features a field-programmable gate array (FPGA) and a real-time processor which works with Linux. It provides lots of fully functional connectivity ports, and has an extended operating temperature (-40 to 70 $°C$) which makes it suitable for the intense weather conditions of Canada. It can be programmed using LabVIEW FPGA [89].


**Figure 4.1: National Instruments cRIO-9031 DAQ+Controller system**

To have control on the vehicle and communicate with it, we need to employ a communication method with the car. Controller Area Network (CAN) bus communication is a protocol on which the CompactRIO communicates with the vehicle. cRIO-9031 requires a CAN Interface Module for the purpose of this communication. National Instruments also provides NI-9853 module which utilizes CAN interface for the system. It is capable of creating applications that require real-time and high speed CAN signals. Figure 4.2 shows NI-9853 separately which is featuring two CAN ports.

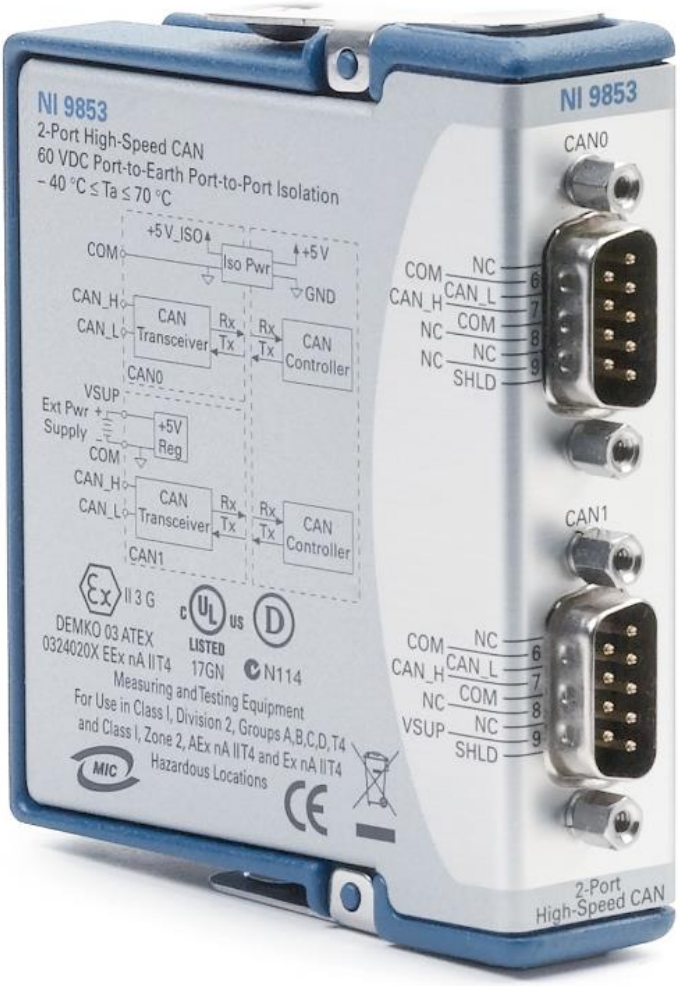


**Figure 4.2: National Instruments NI-9853 CAN Interface module**

Three other modules are also shown on Figure 4.1 attached to the cRIO-9031. NI-9203 is a current input module which has programmable input ranges and built-in noise cancellation. It measures input current signals between -24 mA to 24 mA. NI-9221 and NI-9201 are voltage input modules which measure input voltage signals between -60 V to 60 V, and -10 V to 10 V respectively. These modules also feature isolation and overcurrent protection.

The DAQ system has also another part which plays role when it comes to reading the vehicle related data such as vehicle speed, engine information, fuel consumption, etc. FleetCarma C2 is a device which is connected to the on-board diagnostics port (OBD) which is regularly located on left of the steering wheel and underneath the vehicle dashboard. This device has been used for the SIR project and is supposed to collect the aforementioned data from vehicle internal systems [90]. A picture of FleetCarma C2 and the OBD port of the vehicle is depicted in Figure 4.3.


**Figure 4.3: The OBD port of a vehicle at the left, and the FleetCarma C2 device at the right side**

Another component of the DAQ system is a high precision battery monitor which takes care of computing the state of charge on the auxiliary battery and consumed ampere hours by integrating the current flow of the battery. It also demonstrates the voltage of the battery on its small screen. For this system a Victron Energy – Blue Power BMV 700 Battery Monitor has been utilized which comes with a shunt current sensor investigating the current flowing into and out of the battery. Figure 4.4 shows this component [91]. Also for harvesting the energy of sun by the solar panels and directing it into the battery, a BlueSolar Charge Controller MPPT 100 | 30 by Victron Energy is used which is depicted in Figure 4.5 [92].


**Figure 4.4: Victron Energy Blue Power BMV 700 battery monitor system**


**Figure 4.5: Victron Energy BlueSolar Charge Controller MPPT 100 | 30**

To measure the required data there is an unassailable need to have the sensors which fit the applications and environmental limitations. Some of these sensors are voltage and current sensors which are mandated for different parts of the system. Also surface and ambient temperature sensors are

required for studying their effects on the performance of the system for the future steps. A solar irradiance sensor is installed besides solar panels to measure the available irradiance of the sun as well. Figure 4.6 is showing some of these sensors while a comprehensive plan of the DAQ system is demonstrated in Figure 4.7 that is provided by previous research group members collaborating in this project.


**Figure 4.6: Some sensors used in the previously installed SIR system. From top left to bottom right: Current sensor (Hall Effect), Irradiance sensor, Ambient Temperature sensor, Surface Temperature sensor**

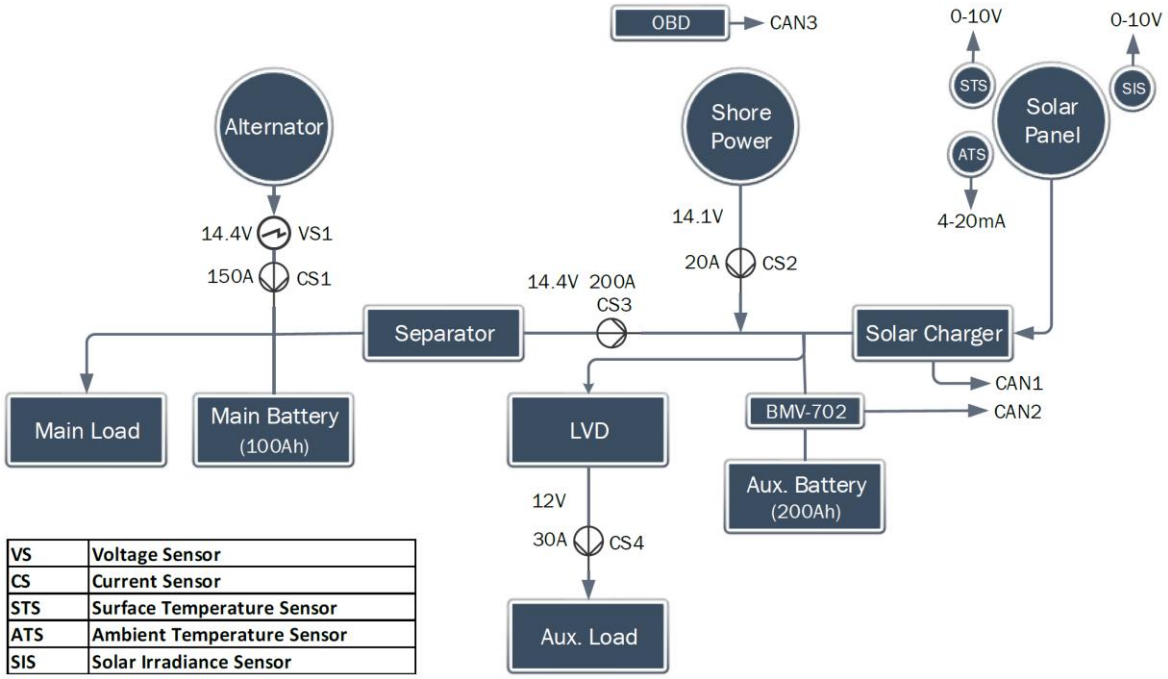


**Figure 4.7: A comprehensive plan of the previously installed DAQ system for the SIR system**

For the purpose of data collection and analysis the built-in LabVIEW FPGA program has been used which Figure 4.8 illustrates the graphical interface of it. This graphical interface and the programming of the system for collecting the data is done by the previous colleagues working on this project.

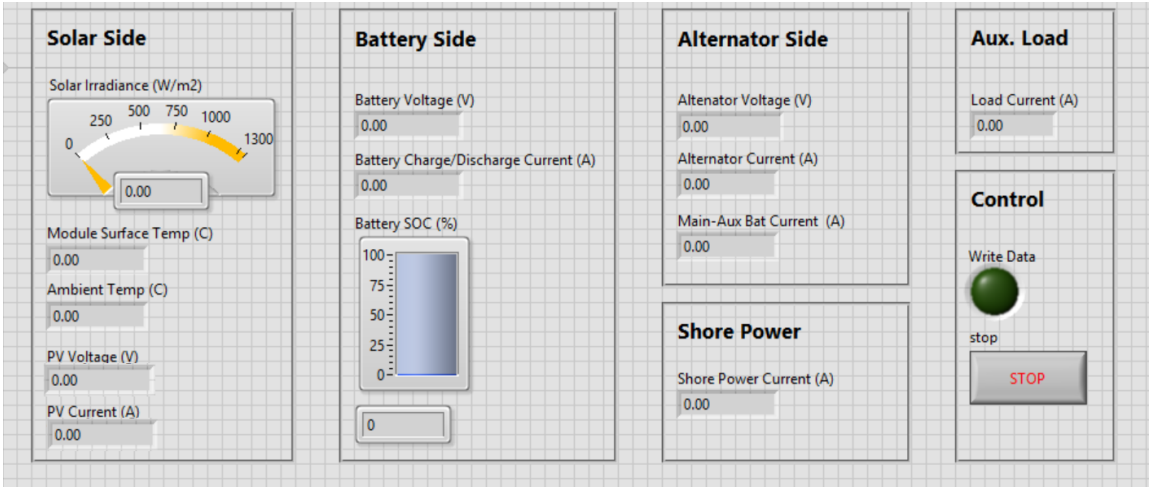


**Figure 4.8: Graphical interface of LabVIEW FPGA used for collecting data from NI cRIO-9031**

## 4.4 Development of New DAQ Platform

The previous section elaborated on the previously installed SIR system and its data acquisition strategy. It was designed and implemented mainly for the purpose of research which requires lots of data for future use. The problem with the existing DAQ system is that it is intensively expensive in a way that cannot be used commercially and does not introduce any benefits even in long term utilization. The best way to reduce the costs is by designing a new DAQ system which only provides the mandatory information. This section mainly focuses on the necessity of a new cost-effective DAQ system through comparison of the costs and the trade-off between having a perfect and expensive DAQ and control system, and a simpler goal-oriented cost-effective one.

Table 4.2 shows the costs for providing each vehicle with the previous DAQ system briefly. It is obviously adding up to more than $11,500 which is not reasonable at all for commercial installation. The main cause for such a high price is using the National Instruments DAQ system and its add-on blocks. This device works really well for research purposes but introduces a high price to the system. Consequently the main focus of the new design would be on replacing NI cRIO-9031 and its add-on components.

**Table 4.1: Estimated prices of the previously installed DAQ system**

| Component | Price | Component | Price |
|---|---|---|---|
| NI cRIO-9031 | $6,285 | BMV-700 | $246 |
| NI-9853 | $2,060 | Hall Effect current sensor | ~ $100 |
| NI-9203 | $865 | Irradiance sensor | ~ $300 |
| NI-9221 | $935 | Ambient temperature sensor | ~ $50 |
| NI-9201 | $665 | Surface temperature sensor | ~ $30 |

### 4.4.1 Central Processing Unit

As it is discussed earlier, NI cRIO-9031 features an FPGA along with an ARM processor which provide the ability of designing the required circuit using LabVIEW FPGA as well as a perfect calculation power. The process of reading data from sensors and creating a log file of them is not a sophisticated task which requires high calculation power and the capability of designing complicated electrical

circuits which FPGA can provide. The whole process can be done in a more compact and goal-oriented way using an application specific integrated circuit (ASIC) microcontroller.

Pursuing the above assumptions, and considering the fact that the processing unit should be able to support more I/O ports for possible future usage, a Raspberry Pi 3 B+ having a Broadcom BCM2873B0 processor chip is chosen [93]. This processor is designed based on the 64-bit ARM architecture, and its ARM cores are able to run at up to 1.4GHz. BCM2873B0 is a quad-core ARM Cortex-A53 which is depicted on Figure 4.9.

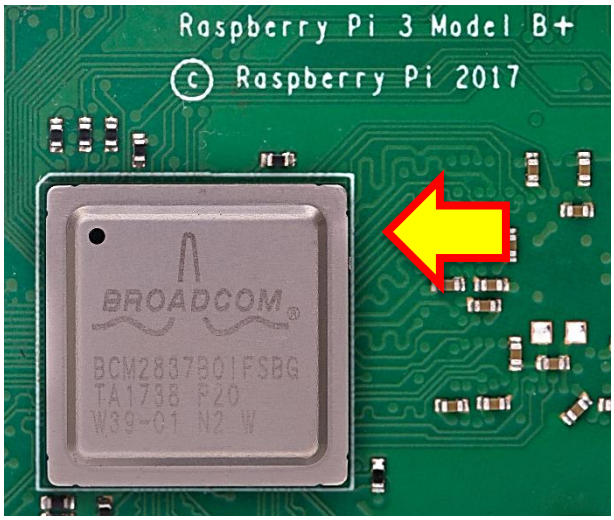


**Figure 4.9: The processor used for a Raspberry Pi 3 B+**

Raspberry Pi 3 model B+ has been used as the central board for connecting the sensors and storing the acquired data. Figure 4.10 shows a picture of it. As it is mentioned above the thinking brain of this embedded system is the BCM2873B0 microprocessor which enables the system to interact with lots of useful interfaces. It has 4 USB 2.0 ports connected through a hub to the microprocessor, HDMI port, SPI interface, and 40 general purpose input/output (GPIO) pins which is amazing. These pins can be used desirably and are capable of working with digital signals.

**Figure 4.10: A Raspberry Pi model 3 B+**

A Raspberry Pi 3 model B+ itself can be bought for a price as low as $46, and if comes along with a starter kit it is not more than $90. This amount is by far cheaper than the NI cRIO-9031, and we show that surprisingly this system does the required job. The only thing we need to consider here is that most of the available sensors provide an analog output. Therefore, in order to read the data from them in specified time frames, we need to convert their analog output to appropriate digital signals for working with GPIO pins. To achieve this goal, an analog to digital converter (ADC) system is required. There are multiple choices for an ADC system which is compatible with the Raspberry Pi GPIO pins. Table 4.2 lists some of the studied converter systems which are exclusively designed to be used with a Raspberry Pi.

**Table 4.2: Some off the shelf ADC boards studied for this research**

| ADC Name | Features | Price |
|---|---|---|
| Phidgets Sensor Interface Kit 8/8/8 | 8 analog inputs, 8 digital output, and USB output | $102.5 |
| 8 Channel 18-bit Analog to Digital Differential Converter for Raspberry Pi | Designed for Raspberry Pi; control via I2C port of Raspberry Pi | $30.5 |
| Expansion HAT for Raspberry Pi | Designed for Raspberry Pi | $43.9 |
| Raspberry Pi High-Precision AD/DA Expansion Board | Designed for Raspberry Pi | $35.8 |

Among all of the options mentioned above, a Raspberry Pi High-Precision AD/DA Expansion Board by Waveshare Design is chosen because of its great performance and precision, as well as simplicity of working with rich documentation. It is also capable of digital to analog signal conversion (DAC) which can be used for controlling the sensors or any type of analog actuators using the Raspberry Pi for future work and further development of the system. Figure 4.11 [94] graphs this $35 ADC expansion board. It has an onboard ADS1256 chip with 8 channels which works on converting analog signals to a digital form. It also features a DAC8532 with 2 channels which does the conversion the other way and changes digital signal into analog output when required.

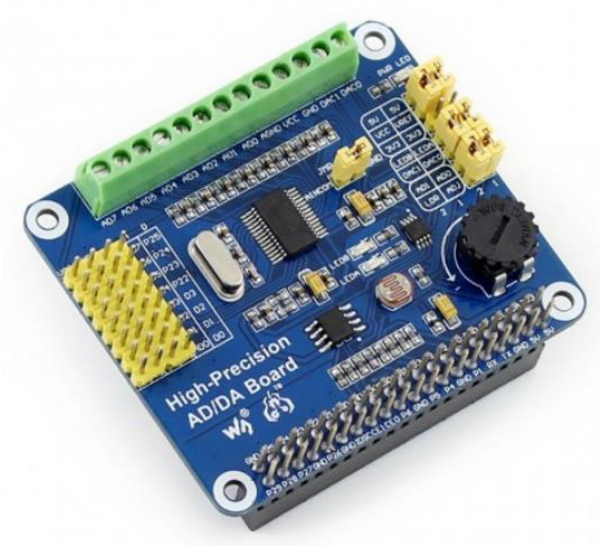


**Figure 4.11: A High-Precision AD/DA Expansion Board acting as an ADC for the Raspberry Pi**

An ADC expansion board attached to the Raspberry Pi board is what we have considered for the purpose of data reading and storage. Now if we know which type of sensors we need and where to put them in the vehicle, we can find some reasonable sensors with specifications that fit the data measurement and environmental requirements. A simple sketch of the data measurement and collection system is sketched in Figure 4.12. Based on this simple sketch we need to provide two sets of voltage and current sensors for measuring voltage and current from the PV and alternator sides, and also provide an ambient temperature sensor. The voltage and current in the load side will be determined via calculation such that the summation of the current in the battery node becomes zero according to the Kirchhoff's Current Law (KCL). Also using the summation of voltages in the loops we can find out the

amount of voltage for the load using Kirchhoff's Voltage Law (KVL). This data then can be used to calculate the amount of power at each part.

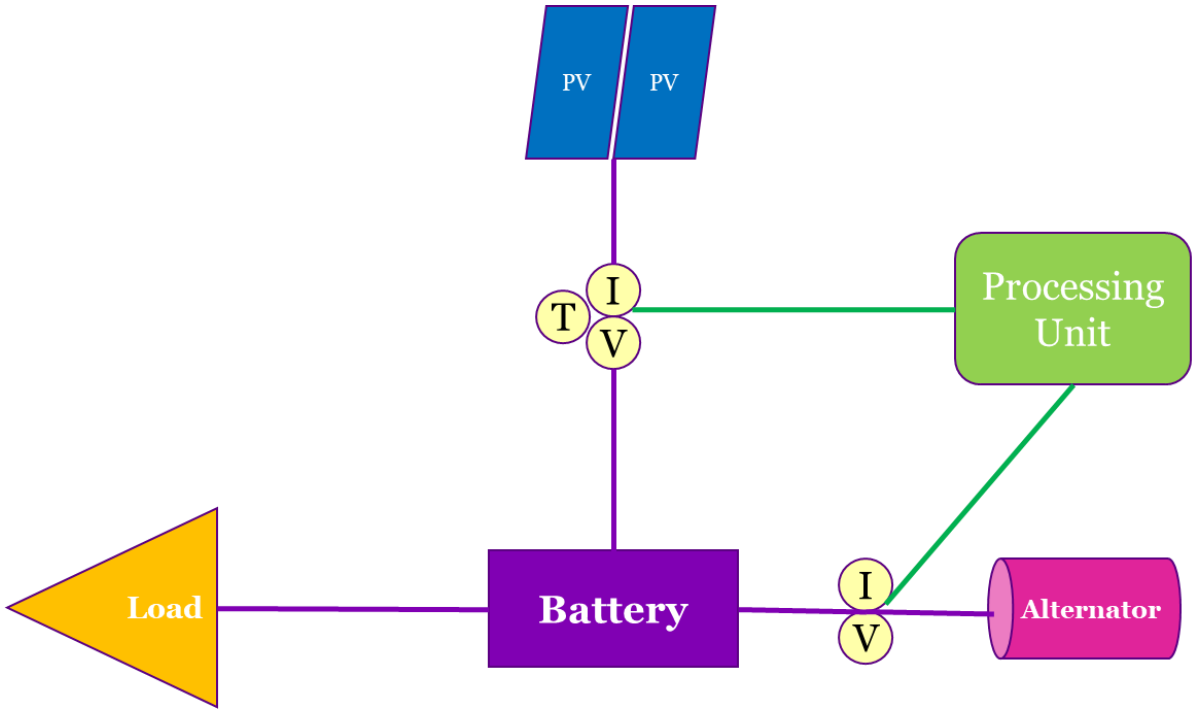


**Figure 4.12: A simple overview of the required data measurement points**

It was decided to remove the irradiance sensor and surface temperature sensors from the next generations for the commercial installation of the SIR system. It is because the data from them are of a higher level which are not required for the immediate data the driver or the fleet managers need. Omitting these sensors simplifies the system even more and reduces the cost of the whole system. Following subsections are included to address the sensor selection for the new DAQ system.

### 4.4.2 Voltage Sensor

The necessity of having at least two voltage sensors is explained earlier. For the new DAQ system a Phidgets 1135 Precision Voltage Sensor has been bought which meets the requirements for voltage measurement we discussed earlier in section 2 of the chapter. It connects to the ADC expansion board using 3 pins and is powered on by the Raspberry Pi and measures voltages between -30V to +30V and works between $-40°C$ and $+85°C$. It only costs $25 and can be seen in Figure 4.13 [95].

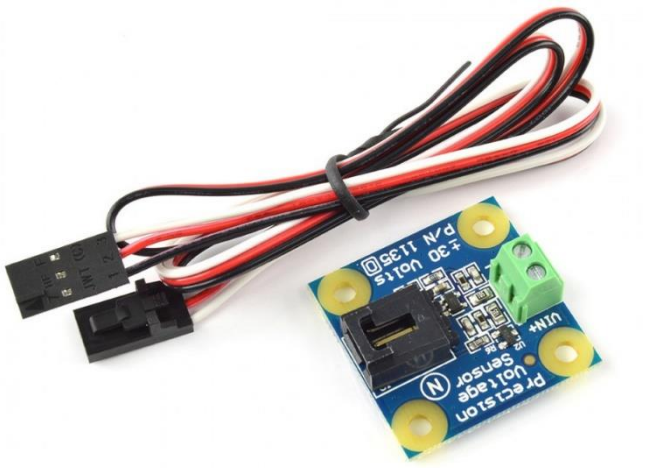


**Figure 4.13: A Phidgets 1135 Precision Voltage sensor**

### 4.4.3 Current Sensor

To calculate the power flow at each part of the system, we need to have at least two current sensors as well. Gravity 50A Current Sensor by DFRobot, featuring a Hall Effect based linear ACS758 chip is the chosen sensor for the new DAQ system which costs $21. It is capable of working in between $-40°C$ to $+150°C$ and its sensitivity is 40 mV/A while measuring 50A current. Figure 4.14 [96] depicts this current sensor.


**Figure 4.14: A Gravity 50A Current Sensor by DFRobot**

### 4.4.4 Ambient Temperature Sensor

Another sensor which is required for the purpose of data acquisition is the ambient temperature sensor. A DHT22 Temperature and Humidity Sensor by DFRobot which is capable of measuring $-30°C$ to $+80°C$ is selected. This sensor can be bought for just $9 and is shown in Figure 4.15 [97].

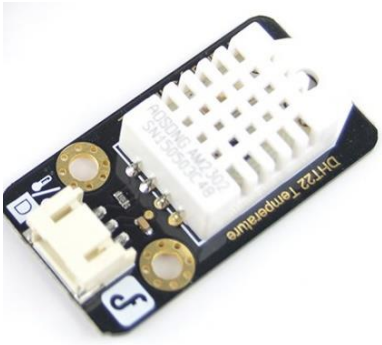


**Figure 4.15: DHT Temperature and Humidity sensor by DFRobot**

The Raspberry Pi with the ADC expansion board, and one current and voltage sensor mounted on it is depicted in Figure 4.16. The hardware configuration and the procedure of correctly mounting the sensors and connecting their wires to the ADC expansion board can be found in [94], [98]. Therefore, the configuration is not explained in detail here.

The complete system costs less than $230 which is way lower than the previously research-oriented DAQ system with an $11,500 price. It does the required job for the commercial use with a data logging code which is written in Python for the Raspberry Pi. Next subsection elaborates on the data logger and the calibration of sensors.

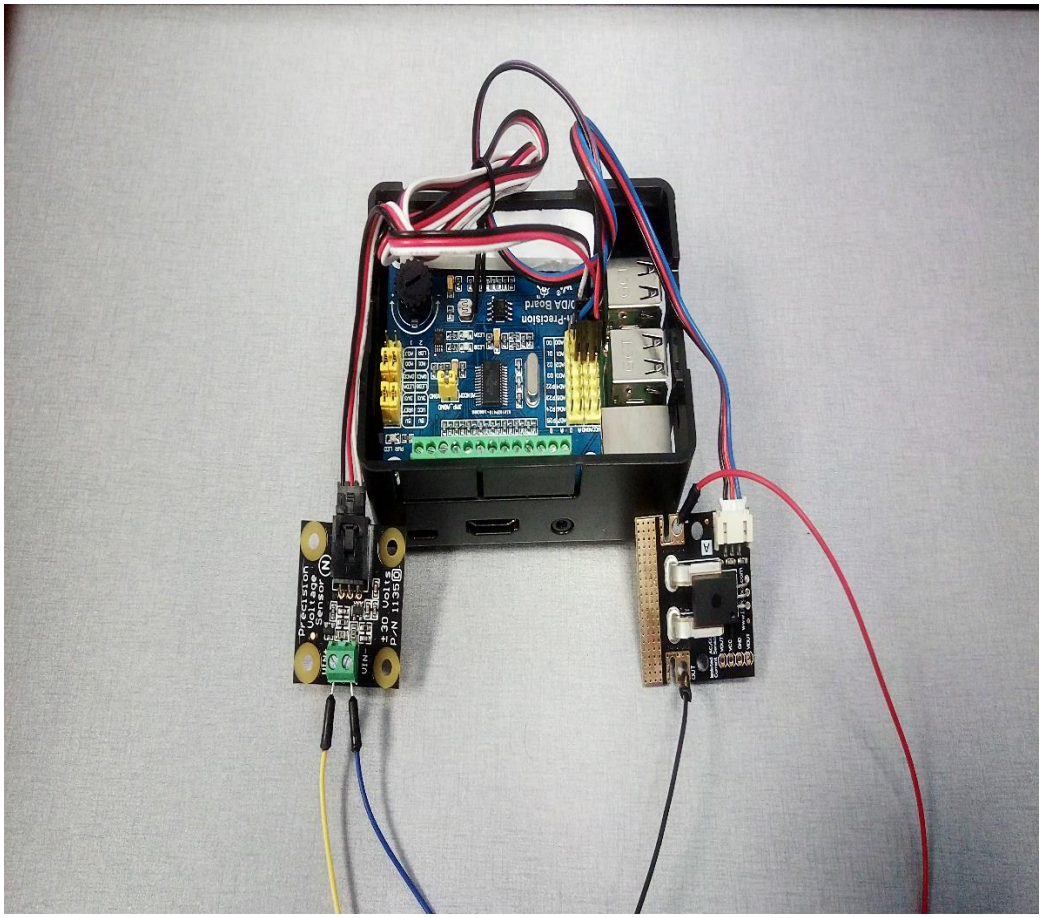

**Figure 4.16: The new DAQ system consisting of the Raspberry Pi board, ADC board, and two sensors (voltage and current sensors)**

### 4.4.5 Writing Data Logger and Calibration

To read the data from the sensors and store them in a log file we need to develop a program on the Raspberry Pi which reads data from GPIOs at a certain time. When the program asks for accessing the data on the GPIO pins, the digital data which is converted from the analog sensor outputs by the ADC expansion board, can be collected and stored in a log file. This can happen whenever the program asks for it and is limited by the sampling frequency of the sensors.

The data logger program uses a C library (ads1256) which is provided for the Texas Instruments ADS1256 chip and can be found in [87]. This library introduces the ADS1256 chip to the Raspbian OS

on the Raspberry Pi and make it usable. After installing the library and its requirements, it can be imported and then the data logger code is developed based on that using Python language.

At the first step, the "sample per second" (SPS) parameter is set and the gain can be varied (default gain value is 1). Then the value for each of the 8 channels is set to 0 before starting the data measurement. The ads1256 library has a "start" function which is called for initializing the ADC using the gain and SPS parameters. A comma-separated values (CSV) file is then created which is responsible for storing the data in different rows for each time the data logger is used. The frequency of reading the data is determined by the SPS parameter. After opening the log file (our .csv file) in the program, a forever loop ('while True') is defined in which the "read_all_channels()" function reads the data from sensors almost simultaneously. The read data is printed out and then written into the log file in a row, and the forever loop does the commands iteratively. The program continues reading the data unless the "stop" function is called. Figure 4.17 depicts the voltage measurement process and some samples of it while the code is running for an SPS of 25.

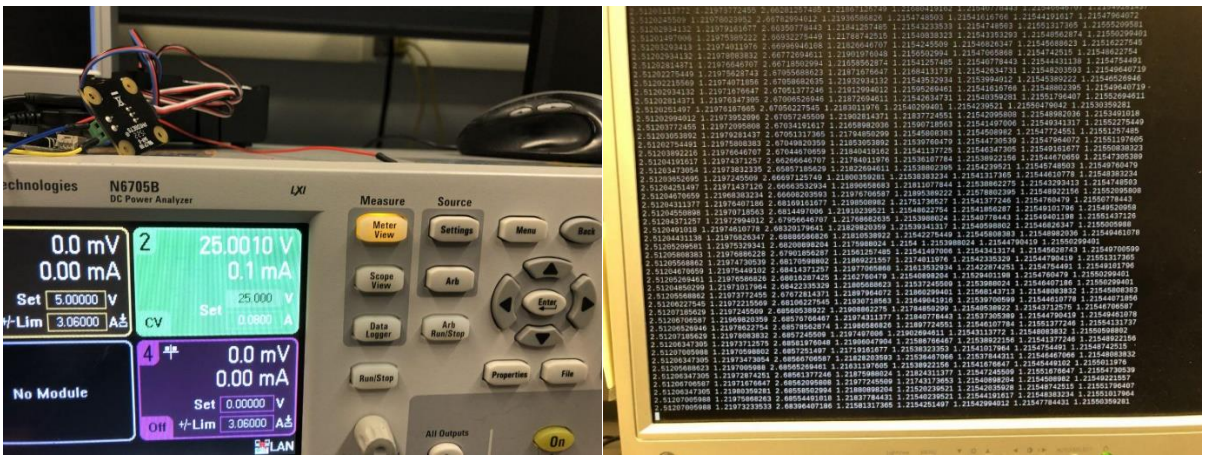


**Figure 4.17: Comparing the actual voltage with measured data from the voltage sensor for calibration**

To calibrate the sensors and get meaningful data from them which is assigned to a certain amount of voltage, current, or temperature, we need to use some precise devices which generate the reference voltage, current, and temperature for us. These calibrations are done in the lab environment. Just for including an example, Figure 4.18 shows the plot which maps the measured voltage to actual voltage values. This shows that:

$$\text{Actual Voltage} = 14.71 \times \text{Measured Voltage} - 36.936 \quad (4.1)$$

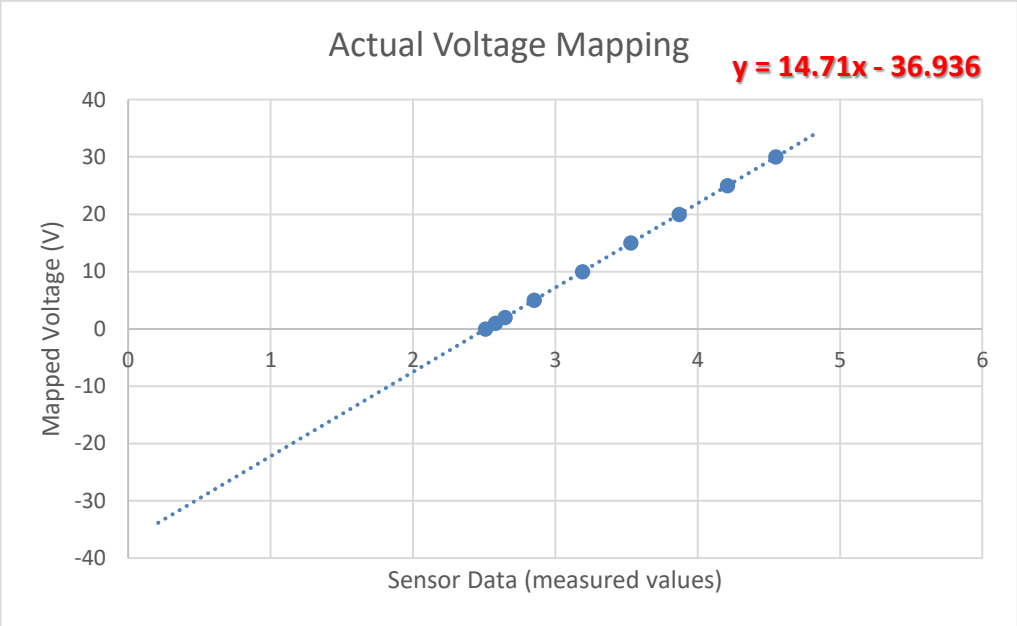


**Figure 4.18: Mapping the measured voltages to actual voltage values for calibration of the voltage sensor**

## 4.5 Designing Enclosures

After finalizing the design of the new DAQ system and implementation of it, the calibration part is also done and the system is ready to be utilized. To install different parts of the DAQ system into the designated areas of service vehicles, they need to have suitable enclosures minimizing the risk of electrical shock and protecting them against being hit by other parts.

This section is dedicated to the process of designing electrical enclosures using SolidWorks software. SolidWorks is a solid modeling computer-aided design and engineering program which helps developing 3D mechanical models from 2D designs and creates 3D-printing compatible design files. This software has the capability of handling sophisticated designs and modeling, however, is used in a simple way to provide enclosures for the developed DAQ system along with the required sensors. Enclosure designs for each component are as follows.

### 4.5.1 Raspberry Pi and ADC Expansion Board

A quite large enclosure is required for inserting the Raspberry Pi board with the ADC Expansion Board mounted on. This enclosure needs some trenches working as a ventilation system preventing the ICs from getting too hot. Also it requires some holes for wiring and cables. Figure 4.19 depicts a suggested design.

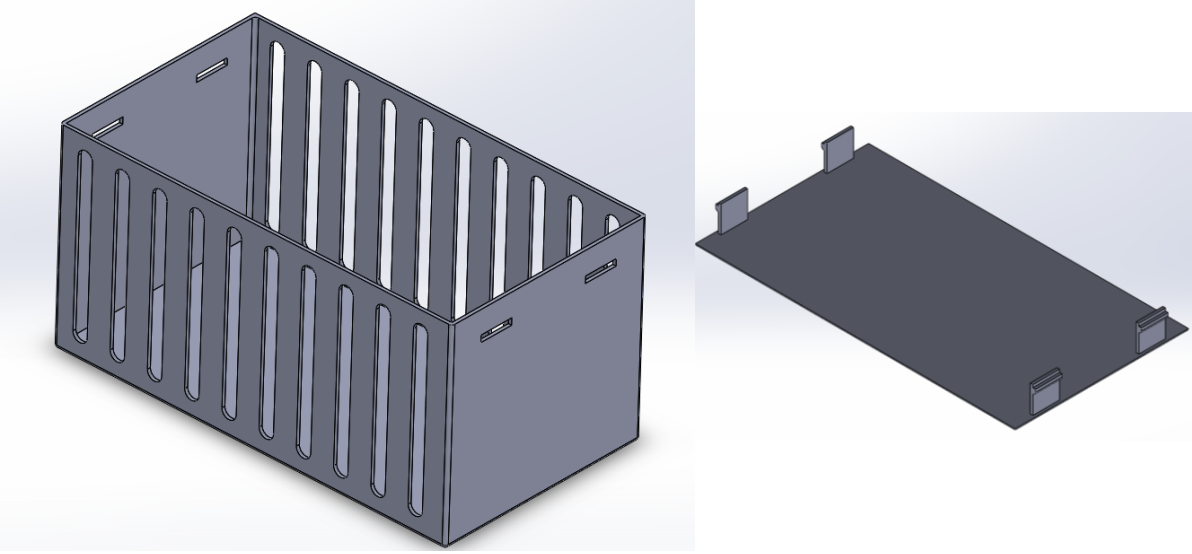

**Figure 4.19: Enclosure for Raspberry Pi and ADC expansion board**

### 4.5.2 Current Sensors

For the selected current sensor, we need to have a perfect fit enclosure and also consider the ventilation trenches as well. This is because the current flowing through the cables can make the IC of the sensor really hot. Also the enclosure needs to provide some holes for the screws and a rectangular hole for the data port. The designed enclosure is shown in Figure 4.20.

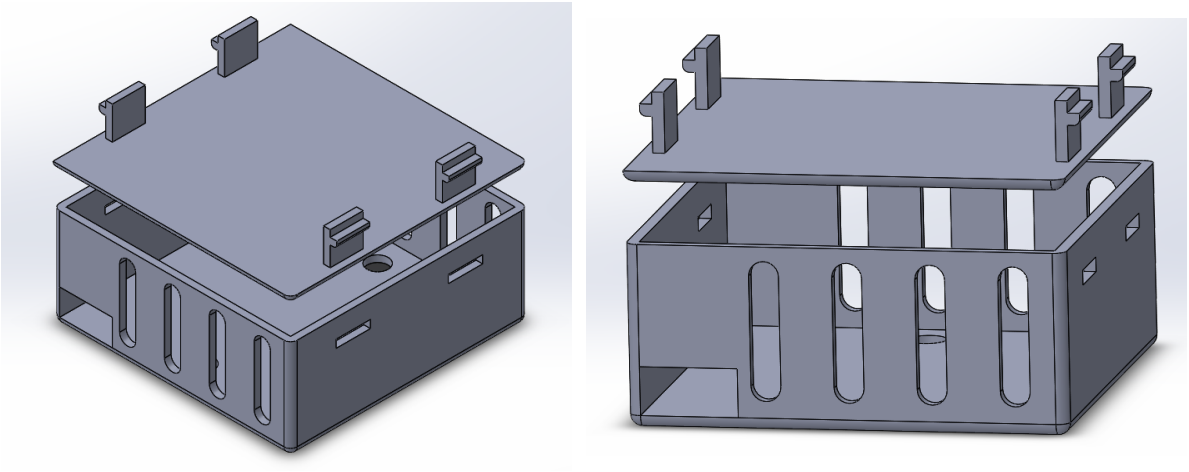

**Figure 4.20: Enclosure for Current Sensor**

### 4.5.3 Voltage Sensors

The voltage sensor selected for the DAQ system does not have any kind of ICs on it and therefore does not require ventilation trenches. The only thing this enclosure requires is the screw holes and some trenches for the data port and the cables connecting to the positive and negative terminals. Figure 4.21 demonstrates the designed enclosure.

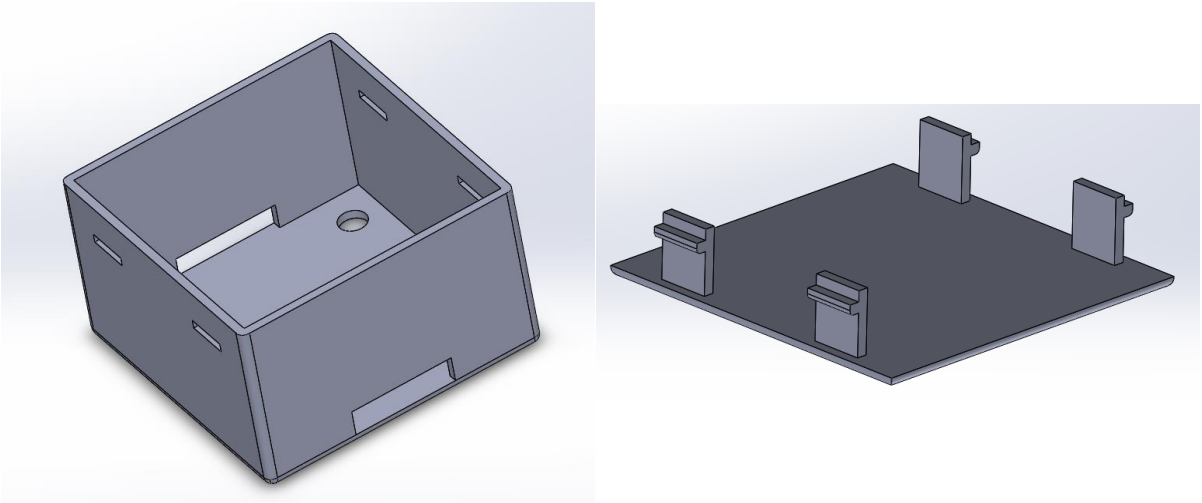

**Figure 4.21: Enclosure for Voltage Sensors**

### 4.5.4 Temperature Sensor

The ambient temperature sensor has to be directly exposed to the environment. Thus, a closing lid is not considered for that and also some trenches on the side wall are considered. Some holes for screws as well as a trench for the data port are considered in the design. Figure 4.22 depicts the designed enclosure.

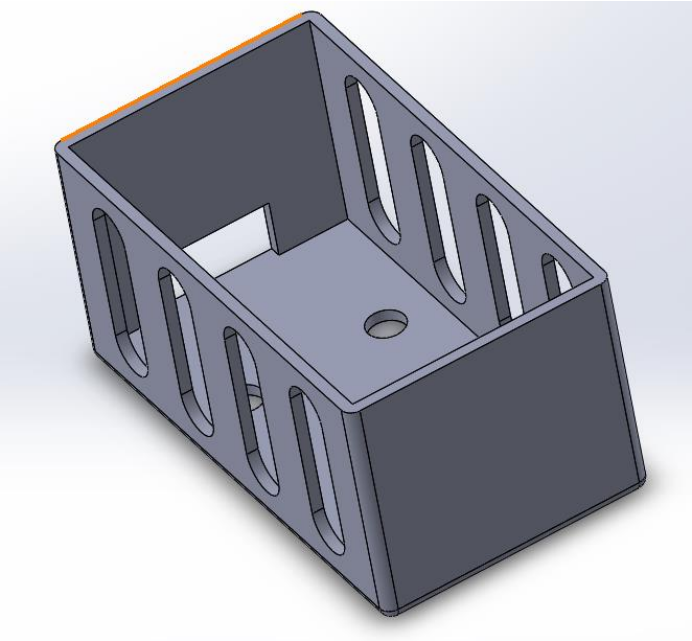

**Figure 4.22: Enclosure for Temperature Sensor**

## 4.6 Summary

This chapter revealed the importance of collecting the data and the required components for measurement and data acquisition to us in the beginning. Types of the measured data to be used for this goal as well as the required sensors are investigated. For the previously installed SIR systems on the service trucks of the City of Waterloo a combination of voltage, current, temperature, and irradiance sensors are utilized. The next section in this chapter attempted to briefly go over the previously used DAQ system and its various components which are used for the purpose of research and are really

sophisticated and expensive, although they provide lots of interesting data at a really high rate. All of these aroused the idea of a new cost-effective DAQ system development which has been addressed in the next parts among this chapter. This new system's development process is widely studied and presented in this chapter. The last section included design of some electrical enclosures for the developed system and its sensors which is mandatory when it comes to installing the new DAQ system for the next generation of the SIR-enabled vehicles.

# Chapter 5
# Conclusions and Future Work

This thesis has aimed at finding a solution to the problem of developing an intelligent energy management strategy for the previously implemented vehicular solar idle reduction system of the service vehicles which lacked a good controller. Another problem of the system which has been addressed in this research is related to its expensive and sophisticated DAQ system which made total price of the system too high for commercial installation. This research attempted to present solutions for the introduced problems and this thesis was then organized as follows to argue for that.

Chapter 1 presented a brief introduction on what the problem is about and drew an overview of the whole system. Chapter 2 was trying to cover the required background for coming up with the problem statement and reviewed the existing literature concerned with the GHG emissions reduction, idling reduction technologies, EMS systems for various usages, especially vehicles, different machine learning approaches with a focus on RL and Deep-RL methods which are mostly used for designing EMS systems, and the DAQ systems. Chapter 3 focused on the main part of this research which was developing a variety of EMS strategies to control the electrical power flow between the main and auxiliary batteries as well as the connected components to their busbars. Three different EMS strategies were developed and the results were presented to compare with a vehicle without SIR system. Chapter 4 attempted to address the second problem with the previously implemented SIR systems which involved a costly DAQ system. A new cost-effective system for measuring and storing the sensor data was developed and explained in this chapter to solve this issue.

## 5.1 Summary of Contributions

First of all the required background for providing the required mindset regarding performing the research was obtained and bolstered. Lots of time was dedicated to looking through the available literature about the GHG emissions, idling reduction technologies, EMS strategies, and DAQ systems. A huge portion of the time allocated for this research pertained to learning new ideas and how to work with different platforms. This research mandated working with Linux Ubuntu, Raspbian, and Windows operating systems. Also it was required to write the codes in MATLAB, Python and its different

dependencies, and also use the Linux Terminal and Windows Command Line. The process of connecting Simulink and Python through UDP communication was another challenging part of the research which needed lots of efforts. Also some efforts has been made to learn how to run the programs on super computers provided by Compute Canada and Sharcnet. The practical part of the project regarding designing and implementation of the cost-effective DAQ system also required lots of knowledge about the sensors, embedded systems, writing code for data logging, calibration, and using Solidworks software for designing the enclosures.

My contribution to this research in addition to the time spent on learning the skills which were not utilized by any of our research group members previously includes but is not limited to:

1. Providing a rich background review and studying the literature about the global warming issues, GHG emissions and the established policies pertaining to limit it, transportation emissions and idling state of vehicles and the existing technologies to reduce that, different energy management strategies and controllers used in vehicles, background on RL and Deep-RL methods, and DAQ systems and required sensors.

2. Developing three different EMS systems with various approaches in order to replace the battery isolator voltage-enabled switch with a command controlled switch. Rule-based controller is developed with some "if" conditions integrated into a Simulink function block. RL-based EMS is designed through writing a code for training a Q-Learning agent which produces a lookup table of Q-values. The state of the switch is controlled using the lookup table and discretized battery SOCs in this approach. The third and most intelligent EMS strategy pertains to Deep RL-based controller which resolved the issues of RL-based EMS system with size of the data and used continuous state space which covers the whole range of battery SOCs and not just some data points of it. In terms of theoretical contributions to this part, the Q-Learning and Double DQN frameworks which are widely used in games and trial and error tasks were adapted to the intelligent EMS development and controller designing with tremendous improvements. The developed frameworks can be utilized for various control tasks through changing the required parameters pertaining to each control problem. Also the results for this research proved that the intelligent EMS helps the vehicular SIR system lower the fuel consumption and reduce GHG emissions.

3. Designing and development of a new cost-effective goal-oriented DAQ system which measures and stores the sensor data at the desired sampling frequency. An extensive study about different

types of data and required sensors was done and the best choices for this part of the research were investigated. All of the components used for this purpose were bought off the shelf and had a reasonable price. A code for reading the data from connected sensors has been written which collects the data on the SD card memory mounted on the Raspberry Pi 3 B+. A variety of programming skills and some tricks for defining the hardware and software relationship were employed for this purpose. Also the developed system keeps the road open for further development and using Bluetooth data or data from CAN bus of the vehicle. The required electrical enclosures were also designed in order to keep the main board and sensors safe.

All of the aforementioned contributions make the vehicular SIR project a promising solution for the fleet service vehicles with reasonable roof top surface which can be used for flexible solar panels installation. This research helps them save money on the fuel consumption and also meet the GHG emissions reduction regulations.

## 5.2 Future Work

This research can be further extended in a variety of directions. The first step can be modification of the vehicle dynamics model and also improving the solar panels, and batteries models. A more accurate model can make the simulation results more analogous to experimental data which is collected from vehicles.

Another potential way is to work on the Deep RL-based EMS algorithm and trying to improve its network configuration based on the state of the art Deep-RL algorithms. A good scenario would be controlling the whole system together and not only the power flow between the batteries. This requires a continuous action space and a high fidelity model of the vehicle with sophisticated dynamics of the vehicle. Another approach to further improve this research is to compare its results with other types of optimal controllers such as model predictive controllers (MPC).

One alternative for the developed DAQ system can be replacing it with FPGA based embedded systems and developing the electrical board from scratch which helps minimizing the costs and improves the controllability and customizability of the board.

# Bibliography


[1] M. Meinshausen *et al.*, “Greenhouse-gas emission targets for limiting global warming to 2°C,” *Nature*, vol. 458, no. 7242, pp. 1158–1162, 2009.

[2] A. Bandivadekar *et al.*, “Reducing the fuel use and greenhouse gas emissions of the US vehicle fleet,” *Energy Policy*, vol. 36, no. 7, pp. 2754–2760, 2008.

[3] Natural Resources Canada, “Learn the facts: fuel consumption and $CO_2$,” [Online]. Available: https://www.nrcan.gc.ca/sites/www.nrcan.gc.ca/files/oee/pdf/transportation/fuel-efficient-technologies/autosmart_factsheet_6_e.pdf

[4] L. Gaines *et al.*, “Status and Issues for idling reduction in the United States,” *Argonne National Laboratory*, February, 2015.

[5] S. M. A. Rahman, H. H. Masjuki, M. A. Kalam, M. J. Abedin, A. Sanjid, and H. Sajjad, “Impact of idling on fuel consumption and exhaust emissions and available idle-reduction technologies for diesel vehicles - A review,” *Energy Convers. Manag.*, vol. 74, pp. 171–182, 2013.

[6] eNow, “Energy solutions for transportation reducing fleet operation costs using solar powered idle reduction technology,” *Energy Solution for Transportation*, White Paper, pp. 1–10.

[7] Y. Huang, H. Wang, A. Khajepour, H. He, and J. Ji, “Model predictive control power management strategies for HEVs: A review,” *J. Power Sources*, vol. 341, pp. 91–106, 2017.

[8] X. Qi, G. Wu, K. Boriboonsomsin, M. J. Barth, and J. Gonder, “Data-Driven Reinforcement Learning–Based Real-Time Energy Management System for Plug-In Hybrid Electric Vehicles,” *Transp. Res. Rec. J. Transp. Res. Board*, vol. 2572, no. 1, pp. 1–8, 2016.

[9] X. Qi, Y. Luo, G. Wu, K. Boriboonsomsin, and M. J. Barth, “Deep reinforcement learning-based vehicle energy efficiency autonomous learning system,” *IEEE Intell. Veh. Symp. Proc.*, no. Iv, pp. 1228–1233, 2017.

[10] X. Qi, Y. Luo, G. Wu, K. Boriboonsomsin, and M. Barth, “Deep reinforcement learning enabled self-learning control for energy efficient driving,” *Transp. Res. Part C Emerg. Technol.*, vol. 99, no. December 2018, pp. 67–81, 2019.

[11] Y. Hu, W. Li, K. Xu, T. Zahid, F. Qin, and C. Li, “Energy Management Strategy for a Hybrid Electric Vehicle Based on Deep Reinforcement Learning,” *Appl. Sci.*, vol. 8, no. 2, p. 187, 2018.

[12] H. Chaoui, H. Gualous, L. Boulon, and S. Kelouwani, “Deep reinforcement learning energy management system for multiple battery based electric vehicles,” *2018 IEEE Veh. Power Propuls. Conf. VPPC 2018 - Proc.*, pp. 1–6, 2019.

[13] V. Mnih *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[14] National Instruments, “Data acquisition,” [Online]. Available: http://www.ni.com/data-acquisition/

[15] Dyne Systems, “DynPro$_2$ Data Acquisition and Control System,” [Online]. Available: https://dynesystems.com/products/data-acquisition-and-control-systems/dynpro2/

[16] S. F. Tie and C. W. Tan, “A review of energy sources and energy management system in electric vehicles,” *Renew. Sustain. Energy Rev.*, vol. 20, pp. 82–102, 2013.

[17] J. T. Kiehl and K. E. Trenberth, “Earth’s Annual Global Mean Energy Budget,” *Bull. Am. Meteorol. Soc.*, vol. 78, no. 2, pp. 197–208, 1997.

[18] L. L. Gaines, C. J. Hartman, and M. Solomon, “Energy use and emissions of idling-reduction options for heavy-duty diesel trucks: A comparison,” *Transp. Res. Rec.*, vol. 2008, no. 2123, pp. 8–16, 2009.

[19] United States Department of Transportation, “Bureau of Transportation Statistics (BTS),” [Online]. Available: https://www.bts.gov/topics/national-transportation-statistics

[20] P. Karkatsoulis, P. Siskos, L. Paroussos, and P. Capros, “Simulating deep CO2 emission reduction in transport in a general equilibrium framework: The GEM-E3T model,” *Transp. Res. Part D Transp. Environ.*, vol. 55, pp. 343–358, 2017.

[21] C. Yang, D. McCollum, R. McCarthy, and W. Leighty, “Meeting an 80% reduction in greenhouse gas emissions from transportation by 2050: A case study in California,” *Transp. Res. Part D Transp. Environ.*, vol. 14, no. 3, pp. 147–156, 2009.

[22] G. Doluweera, H. Hosseini, and A. Sow, “Greenhouse gas emissions reductions in Canada through electrification of energy services,” *Can. Energy Res. Inst.*, no. 162, 2017.

[23] L. Gaines, E. Rask, and G. Keller, “Which is greener: idle, or stop and restart? Comparing fuel use and emissions for short passenger-car stops,” *Argonne National Laboratory*, 2012.

[24] Vehicle Technologies Office, “Idling Reduction for Emergency and Other Service Vehicles,” *Argonne National Laboratory*, 2015.

[25] United States EPA, “Learn about Idling Reduction Technologies (IRTs) for Trucks and School Buses,” [Online]. Available: https://www.epa.gov/verified-diesel-tech/learn-about-idling-reduction-technologies-irts-trucks-and-school-buses

[26] National Research Council, “Review of the 21st century truck partnership, second report,” Washington, DC: National Academies Press, 2012.

[27] B. Carlson, “12 Volt Auxiliary Load On-road Analysis,” *Idaho National Laboratory*, 2015.

[28] N. Matulić, G. Radica, F. Barbir, and S. Nižetić, “Commercial vehicle auxiliary loads powered by PEM fuel cell,” *Int. J. Hydrogen Energy*, vol. 44, no. 20, pp. 10082–10090, 2019.

[29] North American Council for Freight Efficiency, “Diesel APUs,” [Online]. Available: https://nacfe.org/technology/diesel-apus/

[30] North American Council for Freight Efficiency, ”Fuel operated diesel fired heaters,” [Online]. Available: https://nacfe.org/technology/fuel-operated-diesel-fired-heaters/

[31] North American Council for Freight Efficiency, ”Battery HVAC,” [Online]. Available: https://nacfe.org/technology/battery-hvac/

[32] M. Wang, E. Wolfe, T. Craig, O. Abdelaziz, and Z. Gao, “Design and Testing of a Thermal Storage System for Electric Vehicle Cabin Heating,” *SAE Tech. Pap.*, pp. 1–7, 2016.

[33] North American Council for Freight Efficiency, “Automatic engine start-stop,” [Online]. Available: https://nacfe.org/technology/automatic-engine-start-stop/

[34] North American Council for Freight Efficiency, “Solar Panels,” [Online]. Available: https://nacfe.org/technology/solar-panels-2/

[35] S. M. Hosseini, M. M. Majdabadi, N. L. Azad, J. Z. Wen, and A. K. Raghavan, “Intelligent energy management of vehicular solar idle reduction systems with reinforcement learning,” *2018 IEEE Veh. Power Propuls. Conf. VPPC 2018 - Proc.*, 2019.

[36] T. Hofman, M. Steinbuch, R. Van Druten, and A. Serrarens, “Rule-based energy management strategies for hybrid vehicles,” *Int. J. Electr. Hybrid Veh.*, vol. 1, no. 1, p. 71, 2007.

[37] C. Romaus, K. Gathmann, and J. Bocker, “Optimal energy management for a hybrid energy storage system for electric vehicles based on stochastic dynamic programming,” p. IEEE Power Electronics Society (PELS); IEEE Vehicu, 2010.

[38] F. Millo, L. Rolando, R. Fuso, E. Bergshoeff, and F. Shafiabady, “Analysis of different energy management strategies for complex hybrid electric vehicles,” *Comput. Aided. Des. Appl.*, vol. 11, 2014.

[39] F. Zhang, J. Xi, and R. Langari, “Real-time Energy Management Strategy Using V2V and V2I Communications,” *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 2, pp. 416–430, 2016.

[40] Z. Mokrani, D. Rekioua, and T. Rekioua, “Modeling, control and power management of hybrid photovoltaic fuel cells with battery bank supplying electric vehicle,” *Int. J. Hydrogen Energy*, vol. 39, no. 27, pp. 15178–15187, 2014.

[41] F. A. Bender, M. Kaszynski, and O. Sawodny, “Drive cycle prediction and energy management optimization for hybrid hydraulic vehicles,” *IEEE Trans. Veh. Technol.*, vol. 62, no. 8, pp. 3581–3592, 2013.

[42] H. Yin, C. Zhao, M. Li, and C. Ma, “Optimization based energy control for battery/super-capacitor hybrid energy storage systems,” *IECON Proc. (Industrial Electron. Conf.*, pp. 6764–6769, 2013.

[43] X. Qi, G. Wu, K. Boriboonsomsin, and M. J. Barth, “Development and Evaluation of an Evolutionary Algorithm-Based Online Energy Management System for Plug-In Hybrid Electric Vehicles,” *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2181–2191, 2017.

[44] M. A. Hannan, F. A. Azidin, and A. Mohamed, “Hybrid electric vehicles and their challenges: A review,” *Renew. Sustain. Energy Rev.*, vol. 29, pp. 135–150, 2014.

[45] M. Sorrentino, G. Rizzo, and I. Arsie, “Analysis of a rule-based control strategy for on-board energy management of series hybrid vehicles,” *Control Eng. Pract.*, vol. 19, no. 12, pp. 1433–1441, 2011.

[46] J. P. Trovão, P. G. Pereirinha, H. M. Jorge, and C. H. Antunes, “A multi-level energy management system for multi-source electric vehicles - An integrated rule-based meta-heuristic approach,” *Appl. Energy*, vol. 105, pp. 304–318, 2013.

[47] S. F. Tie and C. W. Tan, “A review of energy sources and energy management system in electric vehicles,” *Renewable and Sustainable Energy Reviews*, vol. 20. pp. 82–102, 2013.

[48] S. Ekhtiari, “A Trip Planning-Assisted Energy Management System for Connected PHEVs: Evaluation and Enhancement,” 2017.

[49] J. Peng, H. He, and R. Xiong, “Rule based energy management strategy for a series–parallel plug-in hybrid electric bus optimized by dynamic programming,” *Appl. Energy*, vol. 185, pp. 1633–1643, 2017.

[50] H. Banvait, S. Anwar, and Y. Chen, “A Rule-Based Energy Management Strategy for Plug- in Hybrid Electric Vehicle ( PHEV ),” pp. 3938–3943, 2009.

[51] G. Rizzo, M. Sorrentino, and I. Arsie, “Rule-Based Optimization of Intermittent ICE Scheduling on a Hybrid Solar Vehicle,” *SAE Int. J. Engines*, vol. 2, no. 2, pp. 521–529, 2009.

[52] A. Brahma, Y. Guezennec, and G. Rizzoni, “Optimal energy management in series hybrid electric vehicles,” no. June, pp. 60–64 vol.1, 2002.

[53] S. Q. Zhou, G. F. Guo, and Y. Y. Xiang, “Multi-Objective Optimal Sizing Hybrid Power System in a Solar Electric Vehicle Using Particle Swarm Optimization Algorithm,” *Adv. Mater. Res.*, vol. 694–697, pp. 2699–2703, 2013.

[54] L. V. Pérez, G. R. Bossio, D. Moitre, and G. O. García, “Optimization of power management in an hybrid electric vehicle using dynamic programming,” *Math. Comput. Simul.*, vol. 73, no. 1-4 SPEC. ISS., pp. 244–254, 2006.

[55] J. Liu and H. Peng, “Modeling and control of a power-split hybrid vehicle,” *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 6, pp. 1242–1251, 2008.

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

[57] M. Zhu, Y. Wang, J. Hu, X. Wang, and R. Ke, “Safe, Efficient, and Comfortable Velocity Control based on Reinforcement Learning for Autonomous Driving,” pp. 1–8, 2019.

[58] R. Xiong, J. Cao, and Q. Yu, “Reinforcement learning-based real-time power management for hybrid energy storage system in the plug-in hybrid electric vehicle,” *Appl. Energy*, vol. 211, no. 5, pp. 538–548, 2018.

[59] T. Liu, X. Hu, S. E. Li, and D. Cao, “Reinforcement Learning Optimized Look-Ahead Energy Management of a Parallel Hybrid Electric Vehicle,” *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 4, pp. 1497–1507, 2017.

[60] Y. Tan, W. Liu, and Q. Qiu, “Adaptive power management using reinforcement learning,” p. 461, 2010.

[61] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction. 2nd edition*, vol. 72, no. 10. 2018.

[62] J. Brownlee, “Supervised and Unsupervised Machine Learning Algorithms.” [Online]. Available: https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/. [Accessed: 28-Aug-2019].

[63] L. P. Kaelbling, M. L. Littman, and A. W. Moore, “Reinforcement learning - a survey,” *J. Artif. Intell.*, vol. 4, pp. 237–285, 1996.

[64] T. Shepeleva/Shutterstock, “Machine Learning Explained: Understanding Supervised, Unsupervised, and Reinforcement Learning.” [Online]. Available: https://datafloq.com/read/machine-learning-explained-understanding-learning/4478#.Wm0gAgjwqyI.twitter. [Accessed: 28-Aug-2019].

[65] D. Thrun, Sebastian; Burgard, Wolfram; Fox, “Probabilistic robotics,” *Commun. ACM*, vol. 45, no. 3, 2005.

[66] B. Gerald Tesauro and T. Keith, “Temporal Difference Learning and TD-Gammon,” *Commun. ACM*, vol. 38, no. 3, 1995.

[67] C. J. C. H. Watkins and P. Dayan, “Q-learning,” *Mach. Learn.*, vol. 8, no. 3–4, pp. 279–292, May 1992.

[68] M. A. Wiering and H. van Hasselt, “Ensemble algorithms in reinforcement learning,” *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 38, no. 4, pp. 930–936, Aug. 2008.

[69] V. François-Lavet, R. Fonteneau, and D. Ernst, “How to Discount Deep Reinforcement Learning: Towards New Dynamic Strategies,” pp. 1–9, 2015.

[70] M. Wiering and M. van Otterlo, *Reinforcement Learning: State-Of-the-Art*. Springer, 2012.

[71] T. P. Lillicrap *et al.*, “Continuous control with deep reinforcement learning,” in *ICLR*, 2015.

[72] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, “Trust region policy optimization,” *International conference on machine learning*, pp. 1889-1897, 2015.

[73] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal Policy Optimization Algorithms,” *Mach. Learn.*, 2017.

[74] H. V. Hasselt, “Double Q-learning,” *Mach. Learn.*, pp. 1–9, 2011.

[75] H. van Hasselt, A. Guez, and D. Silver, “Deep Reinforcement Learning with Double Q-learning,” *Mach. Learn.*, 2015.

[76] T. Kos, T. Kosar, and M. Mernik, “Development of data acquisition systems by using a domain-specific modeling language,” *Comput. Ind.*, vol. 63, no. 3, pp. 181–192, Apr. 2012.

[77] National Instruments, “What Is Data Acquisition? - National Instruments.” [Online]. Available: http://www.ni.com/data-acquisition/what-is/. [Accessed: 28-Aug-2019].

[78] Z. L. Enrique, “Digital DAQ v2.” [Online]. Available: https://en.wikipedia.org/wiki/File:DigitalDAQv2.pdf. [Accessed: 28-Aug-2019], Licensed under: CC BY-SA 4.0., Available: https://creativecommons.org/licenses/by-sa/4.0/deed.en., Copyright © 2016 Z.L. Enrique.

[79] National Instruments, “Data Acquisition (DAQ) - National Instruments.” [Online]. Available: http://www.ni.com/data-acquisition/?cid=Paid_Search-128994-US_Canada-Google_DAQ1_DAQ_Broad&s_kwcid=AL!6304!3!142158970983!b!!g!!%2Bdaq&gclid=Cj0KCQjwho7rBRDxARIsAJ5nhFqqwtg0wIPGJnz_auBIiv0bx7y4ENXxNn94Vk-NXalgYApnhrt4Qb8aArF8EALw_wcB. [Accessed: 28-Aug-2019].

[80] Auto Batteries, “How to Test Your Auto Battery | Battery Testing & Maintenance | Autobatteries.com.” [Online]. Available: https://www.autobatteries.com/en-us/battery-testing-and-maintenance/car-battery-voltage-and-testing. [Accessed: 28-Aug-2019].

[81] T. J. Barlow, S. Latham, I. S. Mccrae, and P. G. Boulter, “A reference book of driving cycles for use in the measurement of road vehicle emissions,” 2009.

[82] J. Postel, “User datagram protocol,” *ISI*, 1980.

[83] M. Abadi *et al.*, “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems,” *Mach. Learn.*, Mar. 2016.

[84] F. Chollet, “Keras,” 2015.

[85] G. Brockman *et al.*, “OpenAI Gym,” *Mach. Learn.*, Jun. 2016.

[86] P. J. Huber, “Robust Estimation of a Location Parameter,” *Ann. Math. Stat.*, vol. 35, pp. 73–101, 1964.

[87] F. Oliveira, “GitHub - fabiovix/py-ads1256: Python Library with C wrappers to read 8 channels from the Texas Instruments ADS1256 ADC chip,” *GitHub*, 2017. [Online]. Available: https://github.com/fabiovix/py-ads1256. [Accessed: 28-Aug-2019].

[88] IMT Solar, “Silicon Irradiance Sensor | Solar Irradiance Sensor – Reference Cell – Solar Measurement Equipment – IMT Solar.” [Online]. Available: https://imtsolar.com/products/silicon-irradiance-sensor/. [Accessed: 28-Aug-2019].

[89] National Instruments, “cRIO-9031 - National Instruments.” [Online]. Available: https://www.ni.com/en-ca/support/model.crio-9031.html. [Accessed: 28-Aug-2019].

[90] FleetCarma, “Charge the North | FleetCarma.” [Online]. Available: https://www.fleetcarma.com/chargethenorth/. [Accessed: 28-Aug-2019].

[91] Victron Energy, “BMV-700 - Victron Energy.” [Online]. Available: https://www.victronenergy.com/battery-monitors/bmv-700?_ga=2.108000331.835878279.1529571423-704845767.1501504286. [Accessed: 28-Aug-2019].

[92] Victron Energy, “BlueSolar MPPT 100/30 & 100/50 - Victron Energy.” [Online]. Available: https://www.victronenergy.com/solar-charge-controllers/mppt-100-30#pd-nav-image. [Accessed: 28-Aug-2019].

[93] Raspberry Pi, “Raspberry Pi hardware - Raspberry Pi Documentation.” [Online]. Available: https://www.raspberrypi.org/documentation/hardware/raspberrypi/. [Accessed: 28-Aug-2019].

[94] Waveshare Design, “Raspberry Pi AD/DA Expansion.” [Online]. Available: https://www.waveshare.com/High-Precision-AD-DA-Board.htm. [Accessed: 28-Aug-2019].

[95] Phidgets, “Precision Voltage Sensor - 1135_0 at Phidgets.” [Online]. Available: https://www.phidgets.com/?&prodid=108. [Accessed: 28-Aug-2019].

[96] Robotshop, “Gravity 50A Current Sensor (AC/DC) - RobotShop.” [Online]. Available: https://www.robotshop.com/ca/en/gravity-50a-current-sensor-ac-dc.html. [Accessed: 28-Aug-2019].

[97] Robotshop, “DHT22 Temperature and Humidity Sensor - RobotShop.” [Online]. Available: https://www.robotshop.com/ca/en/dht22-temperature-humidity-sensor.html. [Accessed: 28-Aug-2019].

[98] M. Larsson, “Raspberry Pi AD/DA Board Library for Window 10 IoT Core - Hackster.io.” [Online]. Available: https://www.hackster.io/laserbrain/raspberry-pi-ad-da-board-library-for-window-10-iot-core-c8cc34. [Accessed: 28-Aug-2019].