AN INTELLIGENT ENERGY MANAGEMENT STRATEGY FRAMEWORK FOR HYBRID ELECTRIC VEHICLES

AN INTELLIGENT ENERGY MANAGEMENT STRATEGY FRAMEWORK FOR HYBRID ELECTRIC VEHICLES

BY

REIHANEH OSTADIAN BIDGOLI, B.A.Sc.

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

© Copyright by Reihaneh Ostadian Bidgoli, January 2021

All Rights Reserved

Master of Applied Science (2021) McMaster University

(Electrical and Computer Engineering) Hamilton, Ontario, Canada

TITLE:	An Intelligent Energy Management Strategy Framework for Hybrid Electric Vehicles
AUTHOR:	Reihaneh Ostadian Bidgoli B.A.Sc. (Electrical Engineering) McMaster University, Hamilton, Ontario
SUPERVISOR:	Professor Ali Emadi
NUMBER OF PAGES:	xviii, 170

Abstract

This thesis proposes a novel framework for solving the energy management problem of Hybrid Electric Vehicles (HEVs). We aim to establish a practical and effective approach targeting an optimal Energy Management Strategy (EMS). A situation-specific Equivalent Consumption Minimization Strategy (ECMS) is developed to minimize fuel consumption and improve battery charge sustainability while maintaining an acceptable drive quality. The investigated methodology will be broadly applicable to all HEV applications; however, it will be well-suited for hybrid electric delivery applications.

First, the electrified powertrains and topologies of HEVs, along with an introduction of various energy sources, are assessed. Requirements of an intelligent EMS (iEMS) are explored, which propose a novel classification of iEMSs. Various iEMSs are discussed with a detailed description of their advantages, disadvantages, and principles. Furthermore, a comparison of vehicle communication protocols and controller chips that enable EMSs implementation is also provided.

Second, a detailed HEV model that best presents the vehicle behavior is established in MATLAB\Simulink. A baseline EMS is developed for the selected HEV using the publically available Toyota Prius MY10 data. The powertrain components and vehicle longitudinal modeling are explained in detail. The model demonstrates

a good agreement with real-world testing data with respect to energy-related aspects of powertrain and vehicle.

Third, a Dynamic Programming (DP) method as an offline global optimization EMS is developed. This method will serve as a benchmark to evaluate the situation-specific ECMS approach. The procedures for implementing DP to the HEV powertrain are explained in detail. A simplified backward vehicle model is established to be integrated into the DP algorithm. The DP results are compared with the baseline rule-based method, which shows significant improvement in fuel economy achieves by DP.

Fourth, the unsatisfactory performance of the rule-based method and the offline feature of DP contribute to developing an EMS with a trade-off between the simplicity of heuristic methods and optimal performance of global optimization algorithms. We suggest a situation-specific ECMS which can provide near-global optimal results while meeting the real-time application's criteria. This method employs a simple and effective adaption rule by utilizing three distinct Equivalence Factor (EF) values to keep the battery SOC in a small window. Finally, the performance of implemented EMSs is evaluated on both standard and real-world drive cycles.

Acknowledgements

This research work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

I would like to thank my supervisor, Professor Ali Emadi, for the immense support he provided throughout my graduate studies and for allowing me to continue my electrical and computer engineering education at McMaster University. I would not have had the opportunity to enter this field without his support.

Thanks to all my peers and fellow researchers at the McMaster Automotive Resource Center for sharing their professional expertise, guidance, and support. In particular, thanks to Atriya Biswas for taking the time, attention, and efforts toward improving this thesis. Thanks to John Ramoul for sharing his valuable knowledge in electrical and electronics vehicle architectures. Thanks to Saeed Amirfarhangi Bonab for providing guidance on dynamic programming.

I would like to thank my family, who has given me boundless love in my life. They have been tremendously supportive and encouraging to me throughout my studies.

Finally, of all people, I am most thankful for my husband, Amin, who has been by far the greatest blessing to me in my life. Thanks for helping me survive all the stress from this year and not letting me give up. This work would not have been possible without you.

Notation and Abbreviations

A-ECMS	Adaptive ECMS.
AE	Automotive Ethernet.
AI	Artificial Intelligence.
ANN	Artificial Neural Network.
ASIC	Application Specific Integrated Circuit.
CAN	Controller Area Network.
CD	Charge Depleting.
CDCS	Charge Depleting-Charge Sustaining.
CS	Charge Sustaining.
DCA	Domain Controller Architecture.
DNN	Deep Neural Network.
DP	Dynamic Programming.
DRL	Deep Reinforcement Learning.
DSP	Digital Signal Processor.
E/E	Electrical and Electronic.
EA	Evolutionary Algorithm.
ECMS	Equivalent Consumption Minimization Strategy.

ECUs Electronic Control Units.

EF Equivalence Factor.

EM Electric Machine.

EMS Energy Management Sterategy.

ESS Energy Storage System.

EV Electric Vehicle.

FC Fuel Cell.

FCHEV Fuel Cell Hybrid Electric Vehicle.

FPGA Field Programmable Gate Array.

GA Genetic Algorithm.

GOS Global Optimal Solution.

GPS Global Positioning System.

GPU Graphical Processing Unit.

HESS Hybrid Energy Storage System.

HEV Hybrid Electric Vehicle.

HIL Hardware-in-the-loop.

ICE Internal Combustion Engine.

LIN Local Interconnect Network.

LVDS Low Voltage Differential Signal.

MC Markov Chain.

MCM Markov Chain Model.

MCU Microcontroller Unit.

MDP Markov Decision Problem.

MOST Media Oriented Systems Transport.

MPC Model Predictive Control.

MPU Multi-core Microprocessor Unit.

PEMFCs Proton Exchange Membrane Fuel Cells.

PEV Pure Electric Vehicle.

PHEV Plug-in Hybrid Electric Vehicle.

PI Proportional-Integral.

PMP Pontryagin's Minimum Principle.

PSO Particle Swarm Optimization.

QP Quadratic Programming.

RL Reinforcement Learning.

SA Simulated Annealing.

SC Supercapacitor.

SDP Stochastic Dynamic Programming.

SHEV Series Hybrid Electric Vehicle.

SOC State of Charge.

SoC System-on-Chip.

TD Temporal Difference.

TEGs Thermoelectric Generators.

TPM Transition Probability Matrix.

UC Ultracapacitor.

UDDS Urban Dynamometer Driving Schedule.

V2I Vehicle to Infrastructure.

V2V Vehicle to Vehicle.

ZCA Zonal Controller Architecture.

Contents

Abstract **iii**

Acknowledgements **v**

1 Introduction **1**

1.1 Background and Motivation 1

1.2 Thesis Outline and Contributions 5

1.2.1 Energy Management Strategies (EMSs) 5

1.2.2 High-Fidelity Modeling of a Hybrid Electric Vehicle 6

1.2.3 Implementation of an Offline Optimal Energy Management Strategy 6

1.2.4 Development of a Real-time Energy Management Strategy . . 7

2 Energy Management Strategies (EMSs) **8**

2.1 Introduction 8

2.2 Electrified Powertrain Architectures 10

2.2.1 Energy Sources 11

2.2.2 Hybrid Powertrain Architectures 14

2.2.3 Electrical and Electronic Architectures 18

2.3 Novel Categorization of Intelligent EMSs 22
2.3.1 Principle-based Intelligent EMSs 24
2.3.2 Data-driven Intelligent EMSs 32
2.3.3 Composite Intelligent EMSs 38
2.4 Future Trends of Intelligent EMSs 41
2.5 Enabling Technologies 44
2.5.1 Controller Chips 44
2.5.2 In-Vehicle Communication Protocols 47
2.5.3 Connectivity 49
2.6 Summary 50

3 High-Fidelity Modeling of A Hybrid Electric Vehicle 52

3.1 Vehicle Model 53
3.1.1 Internal Combustion Engine 54
3.1.2 Electric Machines 56
3.1.3 Battery 58
3.1.4 Power Split Device 60
3.1.5 Final Drive 62
3.1.6 Wheels 62
3.1.7 Chassis 63
3.2 Driver 66
3.3 Energy Management Control Unit 67
3.3.1 Engine on-Propulsion 68
3.3.2 Engine off-Propulsion 69
3.3.3 Engine on-Braking 70

3.3.4 Engine off-Braking 71
3.4 Model Validation with ANL data 71
3.4.1 Fuel Consumption Model Evaluation 74
3.4.2 Battery State of Charge Model Evaluation 76
3.5 Summary 78

4 Implementation of an Offline Optimal Energy Management Strategy 79
4.1 Dynamic Programming 79
4.2 Mathematical Principle 81
4.3 Applying Dynamic Programming to HEV 84
4.3.1 State and Control Variables Selection 84
4.3.2 HEV Backward Model 87
4.4 Results 88
4.5 Summary 95

5 Development of a Real-time Energy Management Strategy 96
5.1 Introduction 96
5.2 Situation-specific Equivalent Consumption Minimization Strategy . . 99
5.2.1 Basic Equivalent Consumption Minimization Strategy 99
5.2.2 Proposed Framework of Equivalent Consumption Minimization Strategy 101
5.3 Simulink Model Overview 109
5.3.1 HEV Mode 110
5.3.2 EV Mode 112
5.4 Results and Discussion 114

5.4.1 Simulation Results 115
5.4.2 Real-world Drive Cycles 116
5.4.3 EPA Standard Drive Cycles 122
5.4.4 Fuel Consumption Comparison 127
5.4.5 Comparison of Proposed ECMS with PI-based ECMS 130
5.5 Summary 134

6 Conclusions and Future Work **136**
6.0.1 Conclusions 136
6.0.2 Future Work 139

References **142**

List of Figures

1.1 Breakdown of Canada's Emissions by Economic Sector (2018). 2

1.2 Estimation of market penetration of HEVs by 2030. 3

2.1 The series HEV architecture with plug-in capability. 15

2.2 The parallel HEV architecture with plug-in capability. 16

2.3 The series-parallel HEV architecture with plug-in capability. 17

2.4 Domain controller E/E architecture example. 21

2.5 Migration of E/E architecture adoptions for new centralized cloud computing. 22

2.6 A new classification of iEMSs. 24

2.7 The MPC control algorithm. 29

2.8 A simple concept of nonlinear function approximation. 33

2.9 DRL architecture. 37

2.10 One of the promising concepts of future trend on periodic update of EMS control strategy. 43

3.1 An overview of Toyota Prius model in Matlab Simulink. 53

3.2 The powertrain simulink model. 54

3.3 ICE fuel consumption map. 55

3.4 MGs efficiency maps. 57

3.5 The OCV-R battery model. 59
3.6 Selected HEV architecture 61
3.7 Free body diagram of a vehicle longitudinal dynamics. 64
3.8 Simulated tractive force versus the testing data of the tractive force for steady state driving schedule. 66
3.9 Driver schematic in simulink environment. 67
3.10 The rule based EMS flow chart. 68
3.11 Battery power vs. SOC and engine speed vs. power demand. 69
3.12 Simulated vehicle speed and engine speed versus testing data for steady-state driving schedule. 72
3.13 Simulated vehicle speed and engine speed versus testing data for UDDS driving schedule. 73
3.14 Fuel flow rate comparison for each driving schedule. 75
3.15 SOC trajectory comparison for each driving schedule. 77
4.1 DP backward computational procedure. 83
4.2 Backward-looking vehicle model scheme. 89
4.3 DP results of battery SOC, engine mode, engine, and MG2 power for UDDS drive cycle. 91
4.4 DP results of battery SOC, engine mode, engine, and MG2 power for Highway drive cycle. 92
4.5 DP results of battery SOC, engine mode, engine, and MG2 power for US06 drive cycle. 93
4.6 Fuel consumption comparison of Rule-based and DP over three drive cycles. 94

5.1 Simulation results of operation mode for US06 107
5.2 Flow chart for EF optimization. 108
5.3 The EF update strategy which is implemented in Stateflow. 109
5.4 An overview of the implemented ECMS controller in Simulink. 110
5.5 HEV mode subsystem. 112
5.6 EV mode subsystem. 114
5.7 Drive cycle profile, battery SOC trajectory, EF value, and vehicle mode are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively. 116
5.8 Real-world drive cycles. 117
5.9 Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively. 118
5.10 Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively. 119
5.11 Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively. 120
5.12 Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively. 123

5.13 Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively. 124
5.14 Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively. 125
5.15 Proposed algorithm to prevent the battery from working near battery SOC boundaries. 126
5.16 Results of fuel consumption, obtained by DP, rule-based, and proposed ECMS approaches for real-world and EPA drive cycles. 129
5.17 Battery SOC trajectory comparison of PI-based ECMS and Proposed ECMS. 133

List of Tables

2.1 Proof of "*curse of dimensionality*" 34

2.2 Summary of most popular controller chips on the market. 46

2.3 Summary of possible inter-domain and gateway communication protocols for domain control oriented E/E Architecture. 49

3.1 1.8 Liter Engine Parameters. 56

3.2 MGs Parameters. 58

3.3 Battery Parameters 60

3.4 Power Split Device Parameters. 62

3.5 Final Drive Parameters. 62

3.6 Wheel Parameters. 63

3.7 Chassis Parameters. 65

3.8 Total fuel consumption results. 76

3.9 Final SOC value comparison results for each driving schedule. 78

4.1 DP control and state variables. 86

4.2 DP variables. 90

4.3 Fuel consumption results. 94

5.1 Comparison of control strategies. 103

5.2 Fuel consumption results. 128

5.3 Comparison of fuel consumption results between PI-based ECMS and proposed ECMS. 131

Chapter 1

Introduction

1.1 Background and Motivation

Vehicle emissions lead to air pollution which is adversely impacting human health, environment, and economy. The United States Environmental Protection Agency (EPA) considers the transportation sector as a significant source of greenhouse gas emissions between 1990 and 2018 [1]. In Canada, for instance, according to Figure 1.1, the transportation sector is the second's greenhouse gas emitter. Climate change as a consequence of greenhouse gas emissions causes worldwide challenges. Conventional vehicles' dependency on fossil fuel sources drains the nonrenewable resources that cannot be easily replaced. To tackle the problems associated with conventional vehicles, the transportation industry is heading toward fuel-efficient and climate-safe vehicles.

The past decade has seen the rapid development of electrified vehicles. The Pure Electric Vehicle (PEV) has been attracting a lot of interest. However, there are specific issues with PEVs, including limited range, immature battery technology, high

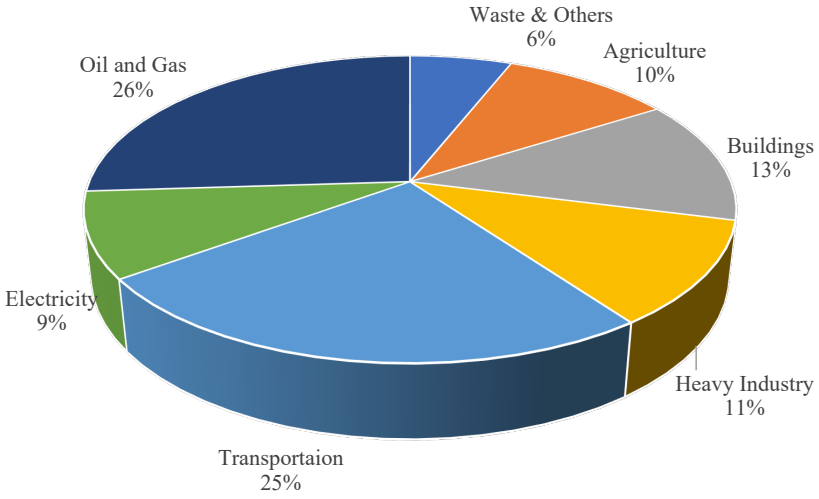


Figure 1.1: Breakdown of Canada's Emissions by Economic Sector (2018). (Adapted from [2])

cost, and limited charging infrastructure availability. Concerning current challenges with PEVs, a Hybrid Electric Vehicle (HEV) seems more practical and feasible for mass production. Although they still utilize Internal Combustion Engine (ICE), in comparison with conventional vehicles, HEVs are much fuel-efficient. Figure 1.2 shows a contious increase of HEV market until the year 2030. As it can be seen, the trend forecasts an 80% increase for HEVs until the year 2025.

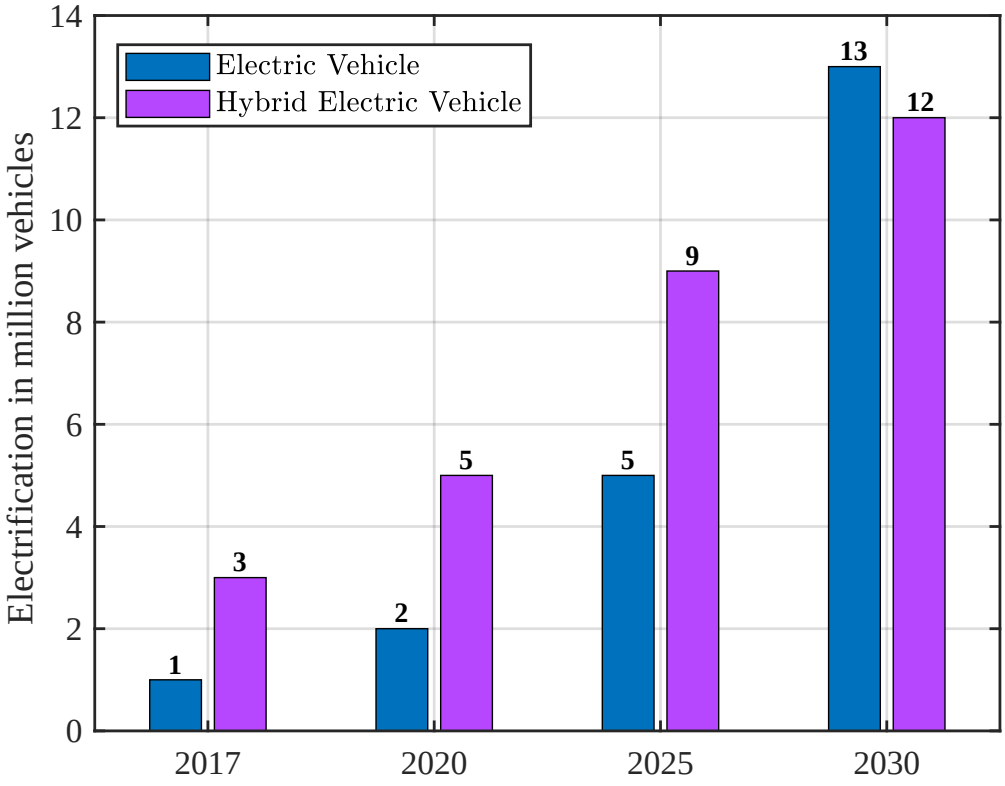


Figure 1.2: Estimation of market penetration of hybrid electric vehicles by 2030. (Adapted from [3])

Hybrid powertrain configuration comprises at least one supplemental energy source such as a battery in addition to ICE fuel tank. Hybridization of powertrain allows the vehicle owner to drive on multiple modes. Targeting the driving condition is one of the factors that dictates the driving mode to HEV. For instance, city driving involves frequent stop and go situations in which electric mode is much more efficient than the hybrid mode. To manage the use of energy sources in a hybrid electric drivetrain,

a supervisory controller is essential. An Energy Management Sterategy (EMS) distributes the power between energy sources in an electrified vehicle while meeting the driver power demand [4].

A great deal of studies have examined various EMSs to improve the fuel economy and the overall performance of hybrid powertrains. These methods either concentrate on optimal performance such as optimization-based algorithms or real-time capability such as heuristic methods. Rule-based EMSs are practical and well-suited for commercial applications; however, they cannot fully utilize the HEV powertrain capability. In this work, we focus on developing an EMS which can achieve near-optimal performance while ensuring the suitability for practical HEV application. Our developed algorithm does not require exact and predicted drive cycle information to obtain significant fuel economy improvement. This feature facilitates EMS utilization in commercial vehicles. The developed EMS adaptively modifies the control strategy to ensure minimizing the fuel consumption at each instant and maintain a battery charge sustaining mode, which is desirable for HEV applications.

This thesis finds a novel EMS framework that is broadly applicable to all HEVs. In particular, the proposed approach has felicitous performance for the situations when an HEV involves delivery or pick-up drive cycles that feature multiple stop locations. The results on real-world and standard drive cycles reveal the great potential of the situation-specific EMS to assist the industry in developing effective EMS control systems for commercial applications. United Parcel Service (UPS), as one of the large companies in package delivery, has already dedicated a number of hybrid electric delivery trucks in the U.K. and U.S. [5]. The company has also invested in bringing semi-electric trucks with a collaboration with Tesla [6]. The 18 months evaluation of

UPS second-generation diesel hybrid electric delivery van has shown a significant fuel economy improvement by 13% in comparison to its conventional vehicle counterpart [7]. This fuel-saving amount would be much more remarkable if an EMS controller with fuel optimal performance is embedded in the vehicle.

1.2 Thesis Outline and Contributions

In this section, we will outline the thesis by introducing the chapters. In each part, we will also explain our contributions.

1.2.1 Energy Management Strategies (EMSs)

This chapter provides a comprehensive review of the novel EMSs for electrified vehicles. We start the chapter by introducing the different electrified powertrain architectures following by briefly explaining energy sources integrated into electrified architectures, including battery, Supercapacitor (SC), Ultracapacitor (UC), and thermal or mechanical energy devices. Next, the requirements of intelligent EMSs and a new categorization of them into principle-based, data-driven, and composite methods are presented. The state of the art energy management methods, including Artificial Neural Network (ANN) and Reinforcement Learning (RL), are discussed in detail. We also elaborate on enabling technologies for implementing an energy management system to compare different controller chips and vehicle connectivity networks. Future trends and existing challenges are also presented, which generate fresh insight into EMSs. Finally, this chapter provides a thorough review of vehicle electrification and intelligent energy management systems for electrified vehicles.

1.2.2 High-Fidelity Modeling of a Hybrid Electric Vehicle

This chapter lays out the modeling of the Toyota Prius MY10, which is selected as our case study in this thesis. The first step is to build a vehicle model that shows similar behavior as the actual vehicle to establish an effective energy management controller. In this chapter, a system-level high-fidelity model of the vehicle using the MATLAB\Simulink environment is presented. The complete model is formulated in a forward-facing fashion, which allows it to track the desired velocity commands by the driver model. The model consists of three main components: the driver model, the supervisory EMS controller, and the vehicle plant model. The baseline EMS is simulated utilizing the online sources and available public data. Finally, the simulated model is validated against the testing data provided by Argonne National Laboratory (ANL). This chapter contributes to providing an HEV model that indicates similar behavior to the actual Toyota Prius MY10 for the most important energy-use metrics.

1.2.3 Implementation of an Offline Optimal Energy Management Strategy

This chapter discusses the implementation of Dynamic Programming (DP) as a global optimization tool for the energy management problem of HEVs. A brief introduction of DP theory is provided following by formulating the optimal energy management problem. The DP procedure for the selected HEV powertrain is explained in detail and the backward simulation model of Toyota Prius MY10 is built. The developed DP is able to minimize the fuel consumption over a pre-known drive cycle while preventing excessive engine on events. This chapter provides a benchmark solution for further evaluating the proposed real-time EMS in the next chapter.

1.2.4 Development of a Real-time Energy Management Strategy

We propose a novel framework to address an optimal EMS based on Equivalent Consumption Minimization Strategy (ECMS) focusing on long-term charge sustenance of battery. By the long-term charge sustenance, we impose a soft constraint on the battery SOC at the end of each drive cycle, which does not need the ending SOC to be exactly equal or very close to the reference SOC value, but the controller is capable of wobbling the battery SOC around the global reference value with a small amplitude in the long-term. To obtain a near-global optimal fuel economy performance, the method instantaneously minimizes the equivalent power consumed by the HEV. We also consider minimizing the excessive HEV operating mode change to avoid unpleasant driving experiences.

The EMS approach is established with low computational resources to be compatible with automotive industry requirements. It does not require any predicted or future driving information. Therefore, we believe it can facilitate the development of an optimal real-time EMS for real-world HEVs. The proposed method is tested on both standard and real-world drive cycles. To fully demonstrate the effectiveness of the proposed strategy, the simulation results are compared with rule-based, PI-based ECMS, and DP strategies. Results indicate that the proposed ECMS achieves the near-global optimal performance compared to the benchmark DP. The results also show that the proposed ECMS can achieve significant improvement in both fuel-economy and battery charge-sustaining compared to rule-based. Compared with PI-based ECMS, the proposed ECMS is more efficient with respect to fuel economy and ease of implementation.

Chapter 2

Energy Management Strategies (EMSs)

2.1 Introduction

Lives on earth have been drastically affected by the air pollution generated from vehicle emission, and global warming has caused drastic climate changes. One of the promising solutions is shifting toward fuel-efficient vehicle and electrification transportation. Combining different sources of energy such as a battery, an UC, a Fuel Cell (FC), and an ICE can help decrease fuel consumption and emissions.

PEV, which has zero emission, seems like a feasible solution, but its shorter operating range compared to a conventional vehicle and the insufficient infrastructure to accept this technology have limited its accessibility. The limited charging stations, long charging time, immature battery technology, high cost, and issues that happen to the power network should be addressed for going toward electrifying transportation [8]. Therefore, HEV which consists of an ICE with at least one Electric Machine (EM)

is more popular. The added degree of freedom from the EM in HEVs brings more complexity in powertrain, however, this results in better performance, high power, and low acoustic noise in comparison to conventional vehicles.

EMS tries to navigate energy between several energy sources considering one or multiple objectives while satisfying the driver's power demand. Energy consumption minimization, improving drivability, safety, increasing component lifetime, and emission reduction can be considered as an objective for an EMS problem. Drivability refers to the driver's comfort in terms of smooth gear shifting, low driveline vibrations and, reasonable engine on/off switches [9]. Safety focuses on tolerating possible faults that can occur in the vehicle components [10]. Carbon monoxide (CO), Hydrocarbons (HC), and Nitrogen Oxide (NO) are regarded as tailpipe emissions [11].

EMSs are generally categorized into rule-based and optimization-based methods. In rule-based methods, rules are achieved through engineers' knowledge, in addition to trial and error. Since the rules are driven without any prior knowledge of drive cycles, rule-based methods failed to address the optimal EMS [12]. In contrast, optimization-based methods derive optimal control inputs for powertrain components with the goal of mostly improving fuel economy. Optimization-based methods are generally classified into global optimization-based and real-time methods [13]. The global optimization-based method considers the energy management problem for a complete driving cycle. Regarding their high computational time, they cannot be implemented in real-time directly. Real-time methods implement instantaneous optimization problems that only consider the current state of the system.

Developing an approach with a tradeoff between simplicity of rule-based methods, optimality of global optimization approaches, and real-time capability of real-time

EMSs has been a concern for researchers. A vehicle compatible EMS with the ability to adapt to its environment with low computational resources would be considered as an intelligent energy management strategy (iEMS). Optimization-based approaches can be modified by combining with state of the art algorithms to satisfy iEMS requirements. With emerging data mining techniques and machine learning tools, EMSs are being driven towards data-based methods with the capability of adapting to real world driving situations. Computational burden, experimental implementation, and optimal performance are the remaining challenges that should be investigated. The detailed requirements, classification, and challenges of iEMSs are further addressed in this thesis.

2.2 Electrified Powertrain Architectures

The architecture that the EMS needs to control must be defined. Without a specified architecture, the EMS cannot be optimized or realized. In this section, electrified powertrain architectures, including its energy sources, propulsion devices and interfacing components are discussed. The two most popular Electrical and Electronic (E/E) architectures are presented to provide the readers with a proper background on why, where, and how the EMSs are introduced for the appropriate vehicle architectures. The future of mobility not only depends on the electrified, automated and connected vehicles but also the devices in the vehicles [14].

The electrification of the powertrain domain has garnered a lot of interest in the last couple of years due to the advancements made in electrical technology for energy sources, power conversion components, and different types of loads. Until recently,

ICE has been the main propulsion component of the powertrain in conventional vehicles, which is supplied directly by petroleum sources [15]. Other means of energy sources have been introduced and researched to ensure the automotive industry does not impact the environment negatively. As a way of introducing electrical energy sources, HEVs have been introduced in such a way where another energy source is used in conjunction with the ICE. The HEV has been a practical way of introducing sustainable and renewable energy sources into vehicles without disturbing the current infrastructure and causing too many risks in terms of safety and environmental impact.

Following the HEV, there are two other main electrified vehicle powertrain architectures that were studied, which are mentioned below [16, 17, 18]:

- Pure Electric Vehicle
- Hybrid Electric Vehicle
- Fuel Cell Hybrid Electric Vehicle

Each one of the above mentioned powertrain architectures is designated with different, but similar types of energy devices. Each architecture integrates the energy sources differently, but with similar technologies. Battery, SC, UC, and thermal or mechanical energy sources can be used as energy sources in the architectures.

2.2.1 Energy Sources

Each powertrain architecture integrates the energy sources differently but with similar technologies. Battery, SC, UC, and thermal or mechanical energy sources can be used as energy sources in the architectures.

Battery

Batteries have been widely used in all types of vehicle architectures, including the conventional ICE vehicle for the low voltage electronics [19]. Battery maturity has reached a certain level where its use in both household and commercial industries has been widely accepted [20]. Batteries also offer the ability to be recharged during regenerative braking periods of the vehicle, which can be very advantageous, since batteries' energy density is smaller compared to nonrenewable fuels. Three of the main types of battery cell technologies widely used today in xEVs have been the Lead-acid, Nickel-based, and Lithium-ion based batteries. Lead-acid batteries are common in household and commercial applications since they tend to be one of the cheaper solutions in battery cell technologies [21]. They also tend to have good enough efficiency and fast response to be incorporated in electrified vehicles. However, there are certain drawbacks associated with lead-acid batteries such as their negative impact on the environment, low specific energy, and their need to be replaced quite frequently. Nickel-based batteries have been used in HEVs for 14 years and manufactured mostly by Panasonic and Primearth EV Energy (PEVE) [21]. The power and energy density of the nickel-based battery, are significantly different from its counterpart of lead-acid. However, nickel-based suffer from a high self-discharge rate that would not benefit PEV with such technology since its range would be significantly affected. According to the detailed analysis of different battery chemistry in [19], lithium-ion batteries have unprecedented performance over the other technologies due to their higher energy density, no memory effect which increases its lifetime, and less environmental impact.

Supercapacitor and Ultracapacitor

SCs and UCs employ both electrostatic and electrochemical storage to be able to deliver electric power. Different than traditional capacitors, UCs have been used to enhance the traditional Energy Storage System (ESS) in terms of lifetime and power delivery [22]. UCs have been traditionally used whenever immediate spikes of power are demanded by the load since they have a high power density [23, 24]. This method has been used to increase the lifetime of lithium-ion battery or just to ensure the power demand is met within a small time frame. Another positive aspect of UC is its long lifetime, which is noticeable in comparison to batteries.

Fuel cell

FC energy sources have been widely used in Fuel Cell Hybrid Electric Vehicle (FCHEV) in both civilian vehicles as well as in-city transit buses [25]. FCs are electrochemical energy sources that produce electrical energy through a chemical reaction between the oxidant at the cathode and the fuel atoms at the anode of the device [25]. FCs, in contrast to batteries, need an unceasing source of fuel in addition to oxygen to operate [19]. The different types of FCs are categorized based on the electrolyte substance. Proton Exchange Membrane Fuel Cells (PEMFCs) are dominant in transportation due to high efficiency, high power density, and low-temperature operation. However, low performance at high current density, high cost, and durability are remaining problems of using PEMFCs in vehicles [26].

Mechanical and Thermal Devices

Other sources of ESS consists of thermal and mechanical devices. In mechanical ESS, the flywheel can be used in conjunction with ICE, or other rotating components since the flywheel is able to store kinetic energy, and the accumulated energy is proportional to its rotational velocity [27]. The flywheel tends to be an attractive solution whenever the vehicle exhibits high or medium power demands [28]. They are favorable in terms of their lifetime (>20 years) as they have a large number of charge/discharge cycles, which is independent of temperature [29]. A disadvantage of the flywheel is that it tends to be quite heavy and bulky since the energy storage capability is proportional to the speed but as well as the inertia of the flywheel which is determined by its mass and geometry. A type of thermal ESS is the Thermoelectric Generators (TEGs). Using the Seebeck effect, TEGs are solid-state devices that help to regenerate the energy lost through heat within vehicle components such as ICE, exhaust systems, power converters, and other heat-generating components [30].

2.2.2 Hybrid Powertrain Architectures

All of the aforementioned energy sources are used differently within different architectures of an electrified vehicle. The architecture of the vehicle highly dictates how the conversion and transfer of power are performed. Based on the three main architectures mentioned above, the most popular option of architectures for the electrified vehicle has been the HEV [31]. Many different HEV architectures imply many different possibilities to integrate the ICE with an electric battery or other energy sources. Some of the different HEV are listed below:

Series Hybrid Electric Vehicle

In a Series Hybrid Electric Vehicle (SHEV), only the electric motor drives the wheels, and ICE has no direct mechanical connection with wheels. This configuration permits the ICE to perform at its maximum efficiency, which is very useful for heavy commercial vehicles, military vehicles, and as well in buses [32, 33]. The engine provides electric energy through a generator for an ESS. An SHEV is similar to an Electric Vehicle (EV) but with higher electric driving range thanks to engine-generator unit [34]. An SHEV with Plug-in capabilities is shown in Figure 2.1.

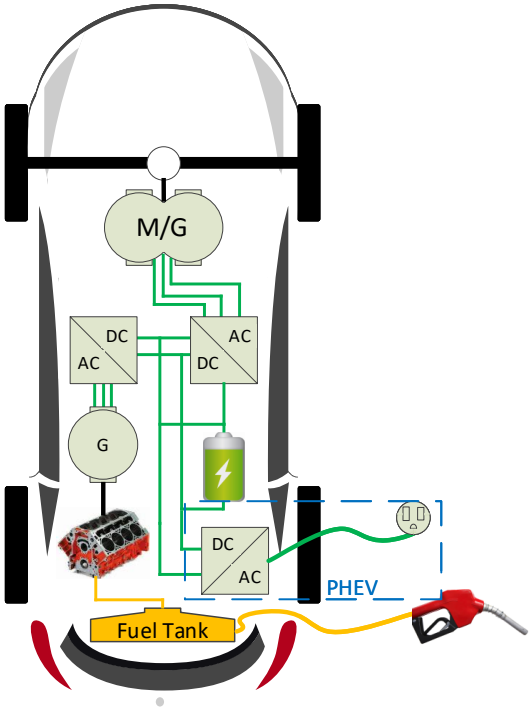


Figure 2.1: The series HEV architecture with plug-in capability.

Parallel Hybrid Electric Vehicle

Parallel HEV architectures employ electric motors and ICE in parallel to drive the wheels. Driving power is provided by both mechanical and electrical energy, which leads to higher powertrain efficiency in comparison to series configurations [35]. The parallel HEV architecture (with a plug-in) can be seen in Figure 2.2 where both ICE and electric motors are used in parallel.

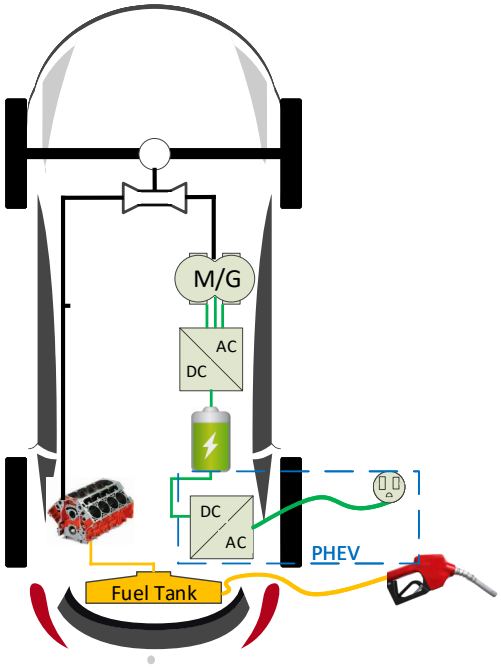


Figure 2.2: The parallel HEV architecture with plug-in capability.

Series-Parallel Hybrid Electric Vehicle

Series-Parallel, also called power-split configuration, provides paths for ICE to wheels as well as electric motors to wheels by using planetary gearsets [36, 33]. In other words, it includes both series and parallel paths to merge their advantages. Engine

power can be used to generate electricity to either charge the battery or drive the vehicle through the electrical path as well as provide mechanical power to drive the vehicle by itself or in parallel to electric motor [34]. Toyota, Ford, and Lexus have been using this configuration for quite a while [33].

Lastly, for all of these architectures, a plug-in capability can be introduced which lets the owner of the vehicle charge the vehicle whenever not used. This ensures that there is a maximum amount of electrical energy stored whenever the vehicle would pursue a journey. This can maximize range and can be helpful in reducing fuel consumption.

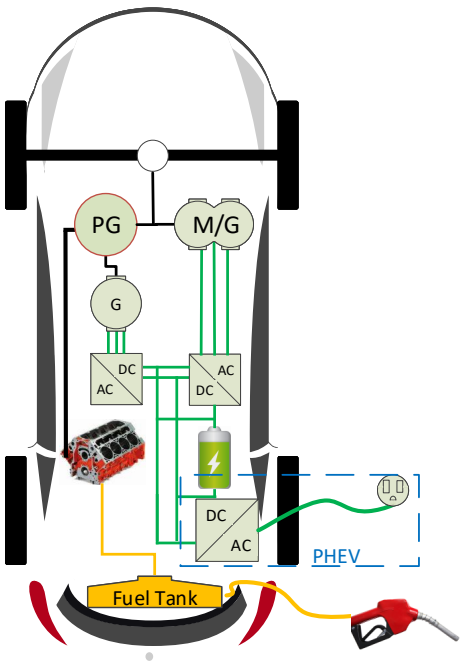


Figure 2.3: The series-parallel HEV architecture with plug-in capability.

2.2.3 Electrical and Electronic Architectures

The E/E architecture of such electrified vehicles described in Section 2.2 has come to a paradigm shift where the in-vehicle E/E architecture Electronic Control Units (ECUs) that were connected in a central "Gateway" or "Plug-in" type of control architecture is moving towards a more distributed or centralized control architecture [37]. ECUs consist of embedded controllers that perform multiple functions in the vehicle to minimize the driver's effort of driving the car, ensuring safe control and monitoring of the vehicle components while also ensuring the entertainment systems in the vehicle are functioning properly. Most of the ECUs in conventional vehicles are mainly tasked to do a single vehicle function including but not limited to:

- Adaptive Cruise Control (ACC)
- Anti-Lock Braking System (ABS)
- Battery Management System (BMS)
- Engine Control Module (ECM)
- Light Switch Module (LSM)
- Park Distance Control (PDC)

In the conventional E/E architecture of a vehicle, there can be 70 to 100 ECUs with their own functions. The communication is performed on a dedicated "gateway" bus depending on the safety level of the ECU. The trend of electrification is pushing for a redesign of the E/E architecture of an electrified vehicle. This redesign step is coming from the bottlenecks found in the migration of xEV technologies where requirements such as flexibility, scalability, external communication, computing power,

communication bandwidth and functional complexity are all increasing [38]. This increase has pushed the limits of current technology used in conventional vehicles and different types of communication architecture are needed along with different types of ECU distribution. Furthermore, having so many ECUs in the vehicle requires high qualification costs that can have a huge impact on the manufacturer of the vehicle [39]. Requiring many ECUs increases energy consumption which is non-ideal for xEVs where range is crucial and limited.

The move to centralized E/E architecture is inevitable but this has to come gradually as not to disturb the vehicle infrastructure. Some requirements for new E/E architectures consist of acquiring large amounts of environmental data along with many parallel computations to ensure proper processing of the vast amount of data [40]. The move from a distributed E/E architecture to a more centralized one has been adopted by two similar but different types:

- Domain Controller Architecture (DCA): Domain specific ECUs with possible domain overlaps through dedicated gateway.
- Zonal Controller Architecture (ZCA): Domain independent with a central in-vehicle/external computer with possible zone ECUs.

The DCA, a more centralized type of control architecture is taken at the vehicle level. Figure 2.4 shows how such a domain controller architecture can look like based on a combination of [41, 42, 43]. Functions of traditional ECUs are merged together to make one powerful domain controller communicating through a dedicated inter-domain bus. To ensure proper domain configuration and functional safety of the vehicle, the domains must be grouped by subsystems that are classified in terms of

physics and non-physics but also must have a high amount of synergy [14]. The main criterion is functional safety in terms of function combination in domains. If a failure occurs in one of the domains, the failure must be dealt with appropriately and not affect other domains while bringing the vehicle at a fail-safe state. The DCA can be separated into three different types of components where there are inter-domains, or better known as smart sensor/actuators, power domain controllers that combine multiple ECU functions together to ensure functional safety and performance and finally a central gateway to interconnect the domains. Some examples of the domains of a DCA based on Figure 2.4 are listed below:

- Human Machine Interface (HMI)
- Autonomous Driving Assisted Systems (ADAS)
- Connectivity
- Body
- Powertrain
- Chassis

Each domain communicates through a central gateway to ensure data is transferred properly between inter-domains and domain controllers. Each inter-domain is comprised of smart actuators or smart sensors that communicate necessary data. An inter-domain could comprise electric motors, pumps, on-board chargers, x-by-wire for example. By using a DCA, each domain controller could use the same hardware, operating system, and software with just different software application layers. This would be extremely beneficial in terms of the costs of the manufacturers. The DCA

has been a preferred choice for manufacturer's in today's volume production and premium vehicles but not in low price vehicles [14, 41].

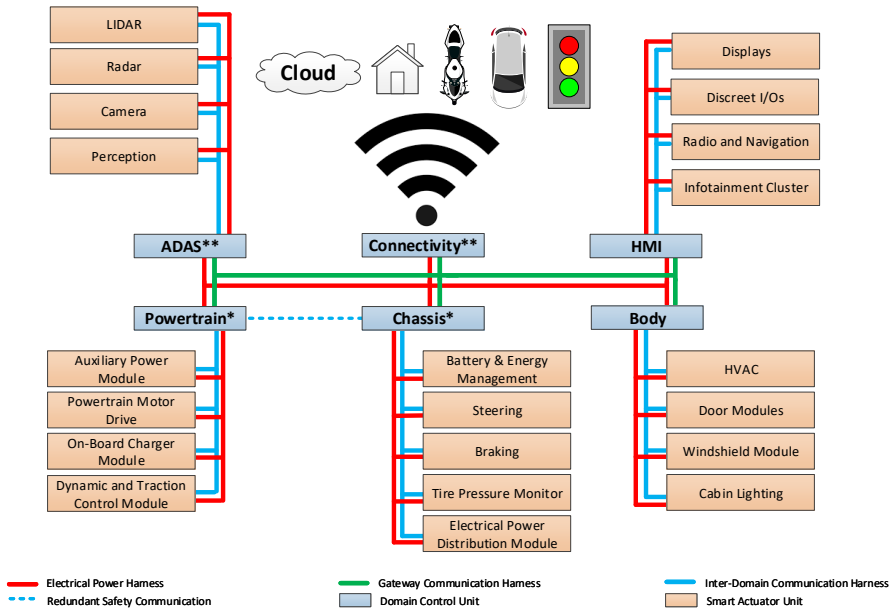


Figure 2.4: Domain controller E/E architecture example.

As in the DCA, the ZCA must have the same requirements in terms of fail-safe, secure, upgradeable (software and hardware), connected, and self-aware/learning [44]. In the ZCA, a more centralized approach is being taken with the same goal of the DCA to minimize the ECU count in vehicles. Sensors and actuators communicate individually through their own dedicated communication bus to a very powerful centralized supervisory controller taking the decisions for each actuator/sensor function [45]. Furthermore, this architecture enables connectivity to external servers along with cloud computing and control of vehicles. This large step is complicated right now but a gradual step to that mindset is through the adoption of the ZCA by incorporating the DCA. Gradual migration of ECU functionalities must be performed

to adapt to the infrastructure [46]. This gradual migration is shown in Figure 2.5.

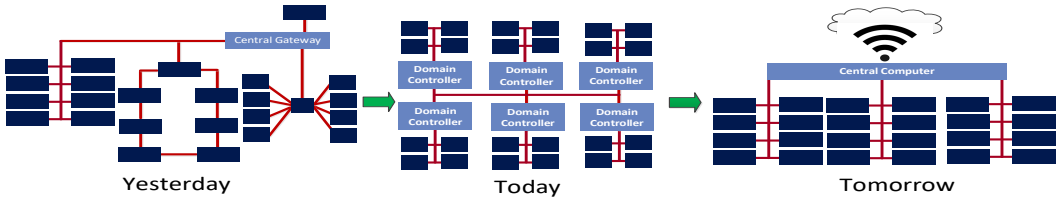


Figure 2.5: Migration of E/E architecture adoptions for new centralized cloud computing.

2.3 Novel Categorization of Intelligent EMSs

Bestowed with the ongoing researches since the last couple of decades, competencies of EMS for an electrified powertrain have grown significantly. They seek to obtain methods with low computational load and compatible with a real-world situation along with optimal performance. There are relatively few studies describing the requirements of an iEMS and providing a categorization. Authors in [47] have highlighted real-time EMSs with an emphasis on the optimality of the control strategy. In this study, different approaches that can be integrated into EMSs are examined to enable energy management systems for real-time implementation. Authors in [48] have categorized the existing EMSs by considering data-based approaches such as machine learning-based EMSs. Overall, these aforementioned review articles indicate the need for a separate study that will enumerate the requirements of an iEMS. This section intends to provide a comprehensive categorization of the existing iEMSs followed by a brief list of criteria, which should be satisfied by an EMS before being called an iEMS. It's noteworthy to mention that it is not intended here to furnish a comparison between ordinary EMSs and iEMSs but to rearrange the existing EMSs as per the

following criteria.

A number of criteria are considered when an EMS features an intelligent controller. Several classical EMSs are excluded from being intelligent by introducing these eligibility criteria. For instance, DP, which is a well-known global optimal approach, can not be called an iEMS since it does not satisfy the real-time capability requirement. Based on the authors knowledge and experience, the requirements for an iEMS are listed below:

1 The iEMS controller should be real-time implementable.

2 The iEMS controller can learn from its past external environment scenarios during real-world deployment.

3 The iEMS controller is adaptive to new environmental scenarios or conditions for what the EMS controller has not been modeled within the simulation stage.

4 The iEMS controller adapted solution needs to converge. Ideally, this is for any EMS controller. No controller will be deployed if it is not converged to the assumptions of the designer.

5 The iEMS controller should be able to predict the future. This requirement is considered a possible soft requirement. As an example, Model Predictive Control (MPC) approach has the ability to predict the future over the prediction horizon.

While studies are exploring advanced EMSs to satisfy all the aforementioned iEMS requirements, there is always some compromise on several requirements to make a trade-off between them. For instance, enabling the prediction and learning feature

of an iEMS would increase the computational burden, which leads to difficulty for real-time implementation.

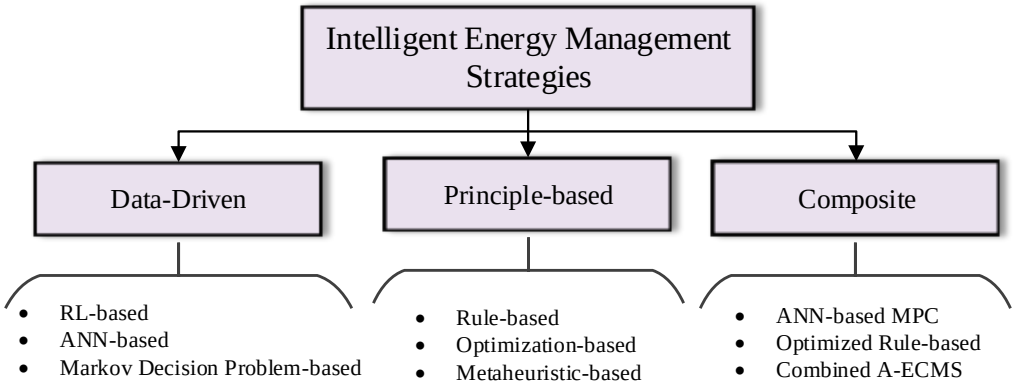


Figure 2.6: A new classification of iEMSs.

2.3.1 Principle-based Intelligent EMSs

There are two categories for the principle-based iEMSs which are rule-based and optimization-based algorithms. Rule-based approaches are frequently used in commercial vehicles such as the Toyota Prius and the Honda Insight [49]. Optimization-based methods include instantaneous optimization algorithms, that are employed for an EMS with the goal of improving fuel economy in most cases.

Rule-based

Rule-based methods are defined by a set of rules extracted from engineers' experience and knowledge. Rules also can be extracted for a specific driving cycle from the global optimal results that are achieved by global optimization algorithms [50].

Rule-based methods offer several attractive features, which include simplicity, real-time capability, and easy implementation. Rule-based methods can be categorized into deterministic and fuzzy-based. Thermostat strategy [26], power follower [51], modified power follower [52], and state machine [53] are methods used for deterministic rule-based. Low efficiency regarding the high number of on and off power sources, and ignoring driver's power demand in defining rules are the main weaknesses of thermostat strategy [26]. Power follower provides a solution for drawbacks of thermostat strategy by considering engine as the main power source in the vehicle along with the battery State of Charge (SOC) and driver's power demand as constraints [51]. The power follower method fails to consider fuel emissions and consumption; therefore, a modified power follower is proposed in [54].

Charge Depleting-Charge Sustaining (CDCS) and blended strategy are two main deterministic rule-based methods that are used for a Plug-in Hybrid Electric Vehicle (PHEV). In contrast to HEVs, PHEVs benefit from high capacity batteries. Therefore, the battery is in the Charge Depleting (CD) mode during most of the trip time. Rule-based methods that are defined for HEVs can be used for Charge Sustaining (CS) mode to avoid the battery depletion. In the CDCS strategy, the vehicle goes in electric mode until the battery reaches the specified SOC value (CD mode), and then the control strategy tries to keep SOC at this level (CS mode) until the end of the trip [55]. In a blended approach, the control strategy seeks to reduce the battery discharge rate by assisting the engine in CD mode.

Fuzzy rule-based is suitable for EMS of HEVs regarding the inherent feature of fuzzy logic, which allows a degree of uncertainty to inputs. One of the main positive aspects of fuzzy rule-based approaches is their robustness to input variations. Fuzzy

based methods are divided into three main categories; conventional, adaptive, and predictive fuzzy EMSs [56]. Adaptive fuzzy tries to consider driving environment factors and make the method more robust to the environment. Predictive fuzzy, by employing driving history, can predict the future state and then decide to split the power. Authors in [57] have implemented a predictive fuzzy-based method that benefits from a Global Positioning System (GPS) to inform of future traffic flow.

Despite the advantages that rule-based methods offer for an EMS, it does not directly consider fuel consumption or emission, as well as, it is not robust to imprecise measurements and component variations. Besides, they are based on engineers expertise, and there is not any specific methodology for extracting rules. On the other hand, fuzzy rule-based methods fail to guarantee the optimal power split in an EMS. Also, definition for a set of fuzzy rules can be time-consuming and might not be ideal for real-world driving cycles.

Optimization-based

Most of the review papers divide optimization-based methods into global optimization and real-time approaches. Global optimization methods can not be considered as iEMSs since they cannot be implemented in real-time. DP is a numerical backward global-optimization method based on Bellman's principle of optimality [58]. DP is computationally intensive and depends on the complete knowledge of the driving cycle. To evaluate iEMSs, DP results are mostly considered as a benchmark.

Real-time methods convert the global optimization problem to an instantaneous optimization. These methods minimize the cost function in the current state of vehicle performance instead of considering the entire trip. ECMS and MPC are the

most popular methods which belong to this category. A brief introduction for each method is provided below:

ECMS: ECMS is established to facilitate the real-time implementation of EMSs [59]. This method considers the objective function as a summation of engine fuel consumption and battery equivalent fuel consumption. The additional term extends the ability of ECMS to consider the energy consumption of a powertrain rather than only fuel consumption of an ICE. Equivalence Factor (EF) is the key issue in the performance of ECMS, which scales electric energy to equivalent fuel consumption. EF optimal value depends on the driving cycle. According to [60], inappropriate EF selection can lead to battery depletion or overcharging. Several methods are established to tune the EF real-time during vehicle operation. Adaptive ECMS (A-ECMS) has been developed to mitigate the EF dependency on driving cycle information. Authors in [60] have classified the A-ECMS to three methods that utilize different tools to adjust an EF value online. The tools are:

- Future driving cycle information predictor [61, 62]
- Pattern recognition algorithm [63]
- Battery SOC feedback [64]

In fact, the last method which is battery SOC feedback can be integrated into other A-ECMSs [64]. Even though ECMS is a sensitive approach to driving cycle information, a significant benefit of this approach is real-time implementation. Adaptive methods can be further investigated to provide an approach with results close to global optimal solutions by DP.

MPC: MPC strategy has several attractive features which make MPC effective for nonlinear, and multiple-input and multiple-outputs (MIMO) systems with constraints such as electrified vehicle powertrain [65]. Figure 2.7 shows MPC algorithm steps. In contrast to DP where the optimization conducts over the whole driving cycle, in MPC, an optimization algorithm is implemented in a short time horizon in each time step that enables MPC for real-time implementation. The optimization part is employed to minimize the error between the predicted and the desired plant output along with fuel minimization or battery SOC sustaining goals. DP [66], Quadratic Programming (QP) [67], Particle Swarm Optimization (PSO) [68], and other optimization algorithms can be employed for a short time optimization step. Once the control variables are calculated, the first control input applies to the electrified powertrain, and the prediction horizon shifts to the next time step. This process is repeated until the end of the trip. MPC requires high accuracy prediction information, which makes it computationally expensive for real-world implementation. Artificial Intelligence (AI) and Markov Chain (MC) predictors are widely implemented for the MPC prediction part that are explained in detail in Sections 2.3.2 and 2.3.3.

Metaheuristic-based

In mathematics, especially in optimization, metaheuristics are a category of decision making procedures, which can reach to close neighborhood of the Global Optimal Solution (GOS) with far less computation efforts and with a limited amount of indispensable information. Metaheuristics may not yield exact GOS, but the proximity of the solution yielded by meta-heuristics to GOS is praiseworthy, especially with

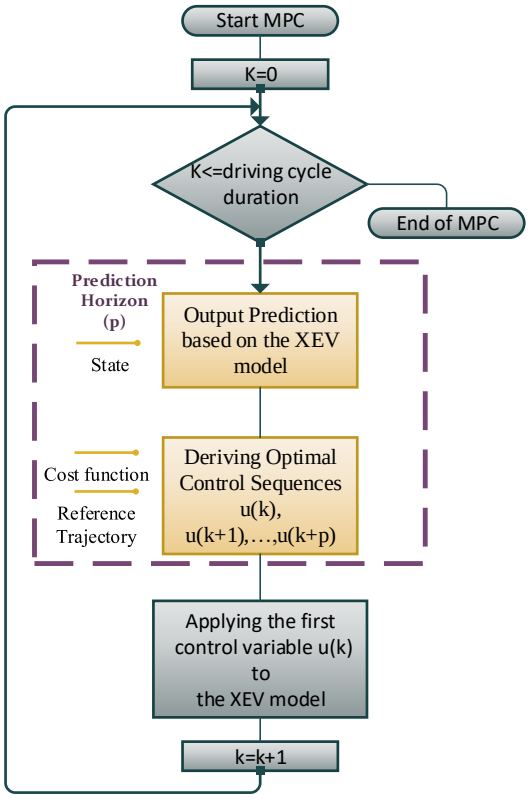


Figure 2.7: The MPC control algorithm.

limited computational effort and information. That is why metaheuristics have attracted major attention from application-oriented research community and industry because they can afford to compromise little deviation from GOS at the expense of convenience in real-time implementation. Metaheuristics reach the near-GOS with a perfect balance between exploration and exploitation [69, 70]. Metaheuristics reduce the computational effort of searching near-GOS by avoiding a significant portion of the futile control space. Authors in [69] have made a notable contribution by presenting a comprehensive review of the application of different metaheuristics in solving

multitudinous problems associated with PHEV. Such problems include the articulation of EMS, optimum component sizing, smart charging strategies, etc.

There are a lot of metaheuristics available such as PSO, a few varieties of PSO, Genetic Algorithm (GA), Simulated Annealing (SA), Ant Colony Optimization (ACO), etc. PSO is a stochastic online optimization technique that employs different particles to randomly search for the suboptimal or optimal solution within the whole solution space [71, 72]. PSO reduces the computational time by not sweeping through all possible solutions, but randomly culminating in the suboptimal or optimal solution [73]. Quite a handful of literature implemented PSO offline for finding the optimal threshold parameters of a rule-based control which can be implemented in real-time [74, 75, 76]. Authors in [74] have optimized threshold parameters of a simple CDCS control strategy with PSO. Whereas, authors in [76] have articulated a rule-based EMS strategy whose threshold parameters are updated periodically with the help of PSO. The periodic parameters update process is triggered by a fuzzy drive cycle recognition system to make the rule-based control apposite for different types of drive cycles. Apparently, it seems that PSO might not be suitable for real-time implementation, but a few literatures [73, 77, 78, 79] made the online implementation feasible with reduced computational time. As far as the online performance is concerned, PSO can outperform not only the genetic algorithm, but other evolutionary algorithms [79]. Authors in [79] have presented an online and real-time PSO implementation for optimizing two control variables such as power-split ratio and gear number to assist a rule-based online control for the EMS. Although the PSO was not solely responsible for the EMS in this study, the implementation of PSO was at every time-step of online simulation. Authors in [78] have improvised the search method of

the PSO to accelerate the search process and consequently managed to obtain better performance with less computational effort. Compared to traditional PSO, Improved PSO (IPSO) accounts for the position of the worst particle while updating the velocities of every particle at each iteration. Authors in [79] implemented Dynamic PSO (DPSO) and proved its superiority over traditional PSO. Real-time Hardware-in-the-loop (HIL) simulation results of an optimal torque distribution strategy for an EV with three electric motors corroborate that instantaneous optimization through PSO can be achieved decent proximity with the global optimal result obtained by DP [77].

GA is generally not implemented online due to its computational burden and incumbency of prior knowledge of the drive cycle. However, GA can be appointed as a local optimizer using a sliding backward time window, and the local optimization can be executed in real-time [80]. Authors have proposed a GA-based online optimization for the EMS of an EV with a Hybrid Energy Storage System (HESS).

Several researchers have marked SA as a remarkable metaheuristic to be appointed in the EMS for electrified powertrains in recent years. Although none of the applications were iEMS for HEVs, SA has been employed as a real-time implementable local optimizer, to optimally distribute the power between battery and UC for an EV [81, 82]. SA culminates to its best performance when the search space is restricted by certain rules [81, 82]. Similar behavior is also exhibited by PSO in real-time HIL simulation, when its search space is dynamically constricted by a set of rules [83]. In [82], SA has been appointed as a local optimizer, finding instantaneous optimal power-sharing between UC and the battery in real-time at an interval of 10 milliseconds. In a nutshell, metaheuristics carry great potential in the form of real-time implementation in elevating the iEMSs to a new level.

2.3.2 Data-driven Intelligent EMSs

The system dynamics of the powertrain are incumbent on both analytical and numerical methods. Data-driven approaches can be used to replace any kind of incumbency of system dynamics, such as the dynamics of a physical system, prediction system, and classification method. Whenever there is a difficulty in mathematical modeling of an implicit system dynamics, data-driven approaches assist as a savior to model it. Data-driven approaches primarily focus on mimicking the system dynamics through the mapping of input to output relationship.

Artificial Neural Network-based

Multi-layer perceptron, widely known as ANN, is the most abundantly used nonlinear function approximator in various fields of data science. ANNs are appropriate for deciphering the dubious input-output system dynamics, which is highly nonlinear and difficult to model with an analytical approach.

Bestowed with effective learning algorithms, ANNs are highly competent in deciphering the inherent input-output characteristics of any physical system if sufficient data are available. Training methods can be broadly categorized into supervised learning and unsupervised learning.

The architecture of a generalized nonlinear approximator can be expressed with the following relation as given in [84]:

$$\tilde{Q}(s;\psi) = g(\psi(\lambda)\Phi_d(s)) \tag{2.3.1}$$

where $g(.)$ is a nonlinear function representing the architecture of the approximator,

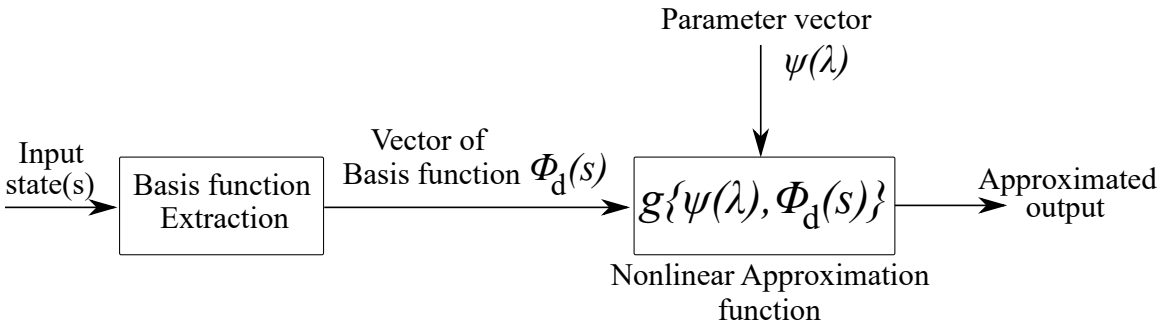


Figure 2.8: A simple concept of nonlinear function approximation.

$\Phi_d(s)$ is the vector of feature or basis functions of the states, and ψ is the parameter vector as shown in Figure 2.8.

For EMS application, ANN can represent the iEMS controller, velocity predictor [85], model of vehicle [86], and driving trends predictor. Quality of predictive EMSs highly depends on the accuracy of the predicted variables. Authors in [87] have implemented three ANNs for predicting road types, driving trends, battery power, and engine speed. The performance of iEMS with the implementation of ANNs is directly governed by the quality of the training data-set. For the ANN-based iEMSs, DP is generally chosen as the source of the training data since DP can yield exact global optimal results in comparison with other global optimization techniques such as Stochastic Dynamic Programming (SDP), GA, PSO, and Evolutionary Algorithm (EA) [88].

Both performance and computation time of DP depend on the discretization of state and control variables along with the dimension of state and control space. The computational time shoots up exponentially as the discretization becomes finer. Needless to say, that curse of dimensionality will be inevitable if either dimension or discretization of state or control variables increases beyond a certain limit, as depicted by table 2.1. τ_{ice}, ω_{ice}, and gear mode are typical control variables.

Table 2.1: Proof of "*curse of dimensionality*" as the number of discrete variables increases.

Case	State variables (discretization)				Control variables (discretization)			Required cells for Dynamic programming
2 states, 1 control	Speed (20)	Acceleration (20)			τ_{ice} (20)			$20 \times 20 \times 20 = 8000$
3 states, 2 controls	Speed (20)	Acceleration(20)	Roadgrade(5)		τ_{ice} (20)	ω_{ice} (20)		$20 \times 20 \times 5 \times 20 \times 20 = 400000$
3 states, 2 controls	Speed (40)	Acceleration(20)	Roadgrade(10)		τ_{ice} (40)	ω_{ice} (40)		$40 \times 20 \times 10 \times 40 \times 40 = 1.28 \times 10^7$
4 states, 3 controls	Speed (40)	Acceleration(40)	Roadgrade(10)	Powerdemnad (40)	τ_{ice} (40)	ω_{ice} (40)	Gear Mode(5)	$40 \times 40 \times 10 \times 40 \times 40 \times 40 \times 5 = 5.12 \times 10^8$

Consequently, gathering data for ANNs in an iEMS is a time-intensive process. In order to expedite the process of data collection, researchers are focusing on finding new strategies yielding near-optimal offline control with far less computational time [89].

Reinforcement Learning-based

RL is one of the machine learning methods that has gained a lot of attention nowadays and is applied in many different fields such as robotic control, traffic management, space exploration rovers, and autonomous vehicles. RL-based agents are especially tailored for sequential decision making, where the long-term return is more prioritized rather than short-term rewards. The agent and environment are the two cardinal parts of RL. The agent's capability of yielding better control decisions improves through

reinforcement learning as the agent accumulates more experience [90].

In RL, the agent interacts with the environment which can be mathematically modeled through the states $(S_t) \in \mathcal{S}$, actions $(A_t) \in \mathcal{A}$, and reward function $(r_t) \in \mathcal{R}$. The sequential decision making along with the sequence of environment states is widely known as Markov Decision Problem (MDP). The noteworthy characteristic of MDP is the fact that the agent does not need to look through the history of the environment's states in order to make a decision at the present time-step. The underlying concept behind such a fact is the property of state variables that probability of landing upon S_{t+1} depends only on the S_t and not on any other states in the past. The dynamics of MDP is defined by the probability of moving to state S' at time $t+1$ if action (A_t) is applied on S_t at time t as given in [91]:

$$\mathcal{P}_{S_t S'}^{A_t} = Pr(S'|S_t, A_t) \tag{2.3.2}$$

This probability of transitioning from the current state to the next state can be stored for all time-steps in a matrix format, known as Transition Probability Matrix (TPM). Technically, the RL agent should find an optimal policy function $(\pi(S))$ which dictates the rule of finding an optimal action at a given state. The RL agent wields two types of goodness functions, i.e., state value functions $(V(S_t))$, and action value functions $(Q(S_t, A_t))$ for finding the optimal policy function.

$\mathcal{P}_{S_t S'}^{A_t}$ is the cardinal characteristics governing the model of any given MDP. If the model of a given MDP is available to the RL agent for the entire MDP, DP-based algorithms, which are also known as model-based algorithms, can be employed to find the global optimal policy for the MDP. However, in real-world situations, where prior information of the entire MDP model is not available to the RL agent,

model-free algorithms such as Temporal Difference (TD) learning algorithms are most appropriate for finding near-optimal policy [90].

Authors in [92] have implemented an iEMS based on RL algorithm and the TPM of states, which are driver power demand, vehicle speed, and battery SOC, is calculated offline by different driving cycles for the proposed strategy. On the other hand, authors in [93] have proposed an iEMS which updates the TPM of driver's power demand in real-time to find the optimal policy for the RL agent. As far as the real-time implementation is concerned, only a few papers have presented the real-time implementation of RL in their literature. However, there are a handful of papers that have described the prospect of real-time implementation. The proposed RL algorithm for a hybrid electric tracked vehicle (HETV) in [94] is implemented through HIL test and it is compared with DP to validate its optimality and adaptability.

Similar to DP, as mentioned in 2.3.2, implementation of RL-based algorithms can be impeded with the curse of dimensionality if they are implemented through the tabular method. Even if we use a coarse discretization, the number of states and of feasible actions can be as high as for instance 4000 and 2500, respectively [95].

A balance between exploration and exploitation is another important factor governing subtle characteristics of the RL algorithm. A higher value of the exploration-exploitation ratio is highly recommended at the beginning of the agent training to encourage exploration throughout the entire action space. Authors in [96] have given detailed analysis of selecting random exploration rate value between 1% to 20% and its impact on vehicle performance and fuel consumption for an HEV.

Lately, the application of Deep Reinforcement Learning (DRL) for implementing EMSs has been soaring from last five years [97]. DRL employs Deep Neural Network

(DNN) in order to express state value function $V(S_t)$, action value function $Q(S_t, A_t)$, and policy function $\pi(S)$ with function approximation instead of tabular approach and hence, eradicates the hindrances posed by large number of quantized state and action variables. Authors in [98] have employed a DRL-based approach for Q-learning where the agent leverages both offline and online learning for better performance. Performance of the proposed iEMS is compared to that of rule-based EMS and it is shown that the iEMS leads to a reduction of fuel consumption by 10.09% under the Urban Dynamometer Driving Schedule (UDDS). Figure 2.9 depicts an overview of DRL.

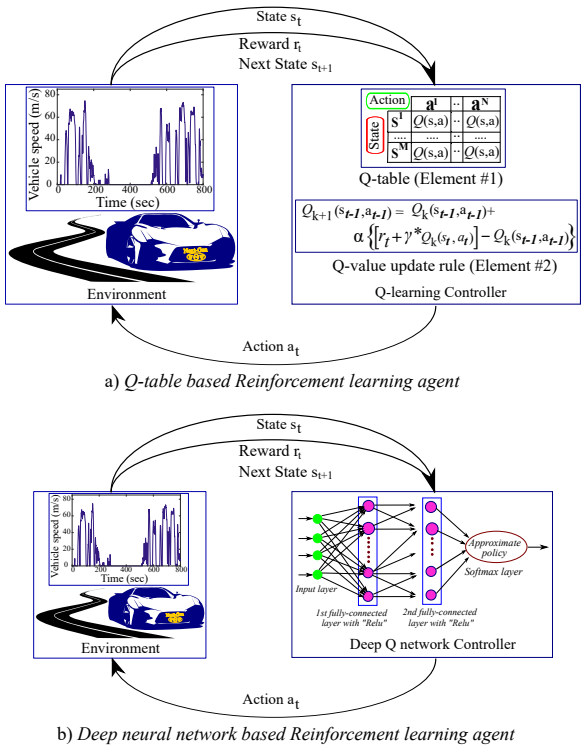


Figure 2.9: DRL architecture. Left box in both a) and b) shows the environment comprising vehicle and drive cycle. The right hand side box in a) represents a Q-table based RL agent whereas, it is replaced with a deep-Q-network in DRL as shown in b).

Markov Decision Problem-based

Markov Chain Model (MCM) is a popular method to predict different aspects of future driving scenario. According to MCM, the probability of getting a certain state in the next time-step only depends on the state of the current time-step. This probability is numerically referred to as TPM in probability terminology. Authors in [99] have presented an iEMS framework, which is comprised of an MCM predicting road grade, speed of vehicle, vehicle start-stop or acceleration-deceleration, and SDP acting as an optimization tool over prediction horizon, for a parallel HEV.

TPM is the major characteristic element of overall MDP and hence, governs the overall execution process of MDP. If enough historical data of the driving scenario available, a static TPM can be constructed and can be used for predicting N-step values of future driving scenarios in real-time applications [99, 100]. However, enough historical data is often not available and an updating TPM can be articulated in such a case [101]. The N-step MDP can be solved through SDP in offline to obtain optimal control over N-step future horizon if the static TPM is available. But, it is convenient to use RL [102] or adaptive dynamic programming [101] to solve the MDP engendered from an updating TPM. QP has also become a lucrative option as an optimization strategy for the MDP, associated with online updating TPM since QP is feasible for real-time implementation [103].

2.3.3 Composite Intelligent EMSs

Composite iEMS is a growing topic which has been getting attention recently. For instance, composite methods combine global optimization tools into principle-based methods to mitigate their imperfections.

Artificial Neural-Network based MPC

The predicted output in an MPC algorithm should follow the reference trajectory, and the MPC performance highly depends on the prediction accuracy. The prediction task can be done in different ways. Authors in [104] have classified MPC approaches based on the method of prediction to frozen-time, prescient, AI, exponential varying, telematic, and stochastic MPC.

There are several publications on ANN- based predictor for MPC-based iEMS [105, 106, 107]. A Radial Basis Function (RBF) ANN velocity predictor is employed in [106], which is trained with four different driving cycles to cover both highway and urban city driving conditions. The simulation results show that the predictive EMS consumes 659.1 g fuel over the same trip as DP consumes 628.5 g. An MPC based iEMSs is proposed to increase the battery life of an EV in [107]. ANN-based short term velocity predictor is applied in combination with the MPC algorithm that leads to a 17.8% improvement in battery life in comparison to the three different methods, which includes a rule-based, instantaneous approach, and an SC voltage based strategy.

Optimized Rule-based

Rule-based methods can be implemented in real-time and they are easy to understand. However, they do not offer optimal performances in EMSs. Rule-based EMS includes different operation mode, which is switched by threshold parameters of battery SOC, vehicle speed, and torque capability of energy sources. Section 2.3.1 provides some studies which implement PSO in combination with rule-based methods. In addition, there are more studies that have attempted to implement an optimization algorithm

for optimizing threshold parameters [108, 109, 110]. Authors in [111] have employed GA optimization to optimize different threshold parameters such as the maximum and minimum value of SOC and electric launch speed of a rule-based EMS in a parallel HEV.

The integration of optimization methods is not limited to deterministic rule-based approaches. Authors in [112] have optimized the rule set and membership functions of a fuzzy logic method by the PSO algorithm to yield a better fuel economy. Authors in [112] have also compared the performance of the fuzzy optimized method with that of a deterministic rule-based approach. The optimized method with the PSO algorithm leads to 10.26% reduction in fuel consumption.

Combined A-ECMS

A-ECMS seeks to update EF to ensure better fuel economy and charge sustenance of the battery in real-world driving. Authors in [62] have suggested implementing a velocity predictor to ensure the adaptation of EF. Authors in [61] have conducted an A-ECMS which updates EF periodically by means of an ANN short term velocity predictor. The results show 3% improvement in comparison to a traditional ECMS. A Convolutional Neural Network (CNN) is employed in [113] to address velocity prediction by considering Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I) communication technology. Three different scenarios are considered for city traffic modeling. A-ECMS block tunes the EF by accessing to predicted velocity and battery SOC feedback. Predicted A-ECMS proves 0.2% to 5% fuel economy improvement for three different scenarios. Driving pattern recognition algorithms such as fuzzy and machine learning methods can be used to improve an A-ECMS performance. Authors

in [114] have used K Nearest Neighbor (KNN) to classify different driving styles. A driving simulator is used for gathering the driver's driving style in order to feed the KNN module. The results confirmed an 8.28% average improvement for different driving styles over traditional ECMS.

2.4 Future Trends of Intelligent EMSs

So far, studies on iEMSs of electrified powertrain vehicles have been carried out by many different methods. A considerable amount of work will need to be done to validate the reliability of these methods. A systematic blueprint of experimental validation should be developed for each of the iEMSs, reviewed here, to corroborate their reliability and feasibility in actual deployment. This review provides the following insights for future research in the field of iEMS:

1. Additional work is required to ameliorate the existing demerits of implemented RL algorithms. The agent should be able to take more dimensions of input state variables in order to discriminate multiple real-world driving scenarios with subtle differences. Q-table based agents become incompetent to handle more dimensions of the state variable. Consequently, DNN-based agents are becoming an enticing option to researchers. Various dual ANN-based agent structures such as Policy Gradient (PG), Deep Deterministic Policy Gradient (DDPG), Advanced Asynchronous Actor-Critic (A3C) should be explored for faster convergence. Such advanced agent structures should be validated in online or real-time emulation along with their offline design and simulation. It would be an exciting and obviously challenging task to design and develop any of the RL agents, reviewed here, for a couple of real-world driving missions and, test its performance in another unfamiliar driving mission. There is ample

room for further progress in DRL to make the results of iEMS controller adaptive and much closer to the global optimal solution.

2. Most of the work in the field of EMS is limited to the simulation level. It is needed to go beyond papers and implement the suggested methods experimentally to see the real-world challenges.

3. Multi-objective EMS development should get more focus in future investigations because only the minimization of fuel consumption and tailpipe emission will certainly operate ICE around its best operating points but it might overexploit electric motors, battery and other cardinal components apart from ICE. Objectives such as minimization of battery health degradation, drivability optimization, electric motor longevity should be included in the overall cost structure.

4. With the rapid advancement of intelligent transportation systems (ITS), data-driven iEMS are escalating, becoming lucrative options to explore. Accessibility to traffic data, geographic road map, and road geometry makes the prediction based iEMSs more reliable and adaptive. Therefore, the future trend would be developing advanced data-driven iEMSs.

5. Cloud-based EMS is a significant progression that needs to be more investigated. An example of these kinds of EMSs can be the determination of the best possible vehicle trip information with the objective of minimum fuel consumption through a cloud-interaction system. In fact, cloud computing would generate the optimal route/velocity trajectory and the results would go back to the driver through a visual interface [115].

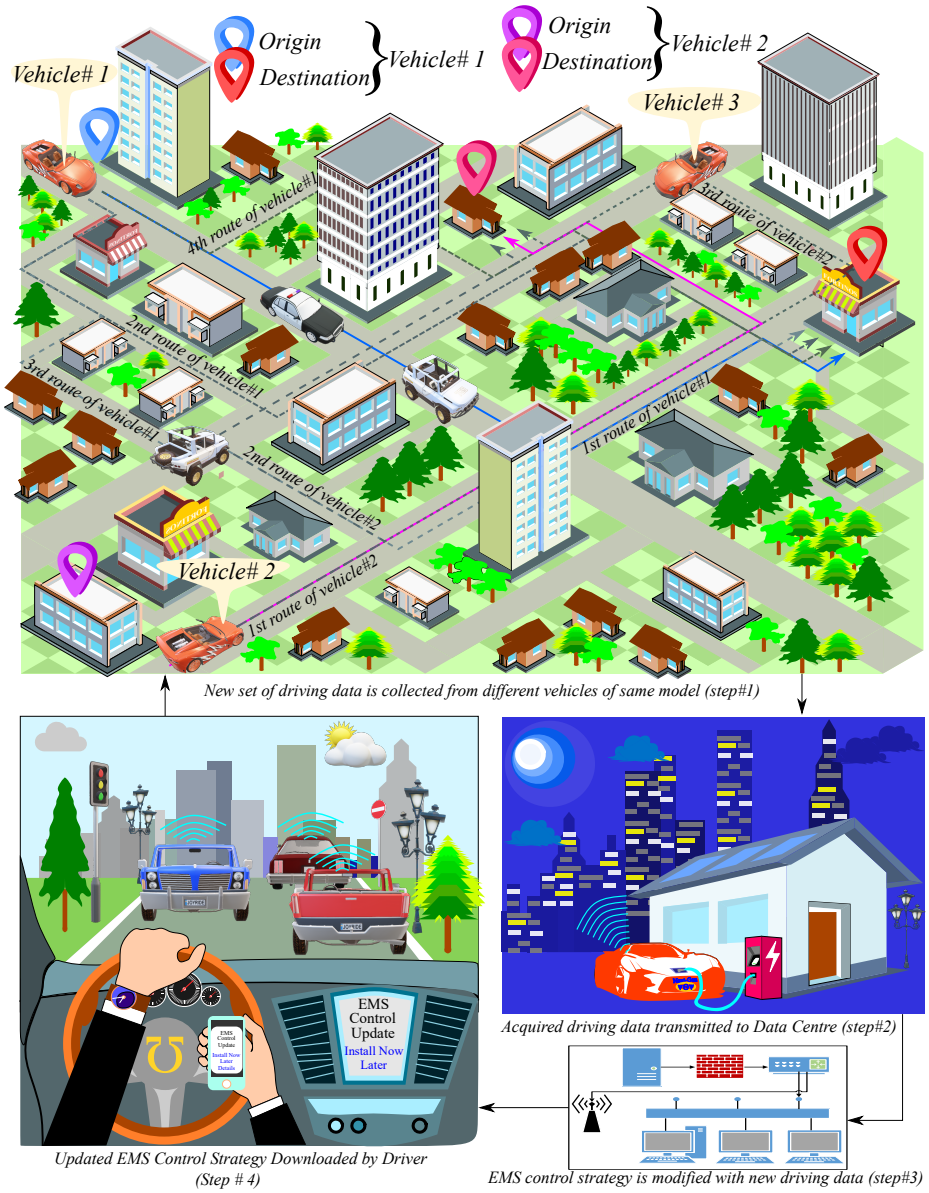


Figure 2.10: One of the promising concepts of the future trend on a periodic update of EMS control strategy. A set of vehicles with same electrified powertrain will acquire driving data for a predefined period (step#1), the acquired data will be uploaded to cloud-computing servers while the vehicle is externally recharged (step#2), new EMS control will be generated at the server based on the recent driving pattern of the vehicles (step#3). The availability of the most updated EMS control will be notified to each of those vehicles through both smartphone and vehicle infotainment system (step#4).

2.5 Enabling Technologies

Intelligent methods introduced in Section 2.3 have been widely researched due to the advancements in technologies. With the cost and size of high computational platforms decreasing and the rise of high bandwidth communication increasing due to the availability of cheap computation platforms, the deployment of intelligent methods into xEVs is becoming more and more attractive. The aspect of hardware and software need to be mixed and both need to have a fail-safe requirement [116]. The primary enabler of technologies for the deployment of intelligent methods in commercial vehicles includes embedded controller chips, communication protocols, and connectivity functions, which are summarized in this section.

2.5.1 Controller Chips

The ECUs are used in conventional vehicle architectures utilize embedded 8, 16 and 32-bit processors with a clock frequency of 40MHz. A 2MB code is flashed and encrypted onto the ECU memory, usually a non-volatile memory (NVM), by the manufacturer [117]. Compared to personalized computers, these ECUs have very little computational power. This is the reason why there are a limited number of vehicle operations that can be contained inside an ECU. The number of ECUs have been increasing in vehicles due to the utilization of the same type of controller chip in the distributed architecture. Some specific ECUs have been implemented using a Digital Signal Processor (DSP) for actuating or sensing a vehicle component signal.

By utilizing the technological advances in controller chips, many functions can be combined into a single but powerful ECU which is the trend of the DCA described in Section 2.2.3. With the DCA and the ability of using intelligent algorithms, much

more powerful ECUs need to be implemented as the domain controller such that a vehicle fail-safe operation is achieved. Furthermore, instead of having complex ECUs actuating or sensing signals from vehicle components, smart actuators/sensors will be used that can comprise of a simple but fast ECU. With these requirements and the writers' experience, the domain controllers and smart actuators/sensors ECU could be implemented using the following technologies that have garnered much attention in aerospace, data centers or servers, and machine learning or artificial intelligence industries [40]. Multi-core Microprocessor Unit (MPU), Field Programmable Gate Array (FPGA), and Application Specific Integrated Circuit (ASIC) can be used as domain controllers. Smart actuators/sensors include Microcontroller Unit (MCU), Graphical Processing Unit (GPU), System-on-Chip (SoC), ASIC, and digital signal processor (DSP).

A summary of the ECU main controller chips mentioned in this section can be found in Table 2.2. As described previously, MCUs, MPUs and mainly DSPs are used in today's vehicles. Some exceptions do include using SoCs for Local Interconnect Network (LIN) communication-based slave nodes or ASICs for specific types of communication protocols that would alleviate the burden of computational resources on ECUs [116]. To move to a future centralized computational platform that includes all domain functionalities or even all vehicle functionality, safety will be paramount to the integration of these technologies. The ISO26262-part 5 has been an integral part of the development and introduction of complex electronic hardware into these new E/E architectures. The reason for this is these new electronics must perform safety critical functions such as steering, accelerating and braking so if a component fails, it has to do so in a fail-safe manner. Functional safety must remain a top priority while

introducing these new technologies.

Table 2.2: Summary of most popular controller chips on the market.

	MCU/MPU	**FPGA or SoC**	**GPU**	**DSP**
Overview	Single or Multi-Core architectures with many peripherals already coded.	A collection of logical gate arrays that can be configured in the field.	Originally designed for graphics - Includes many parallel ALU cores.	Optimized processor for the specific application
Strengths	• Ease of programming • Versatility • Debug is simple	• Configured for a specific application • Can be re-programmed in field • Low power consumption • Parallelism of functions • Design can become ASIC design for high volumes • Determinism	• Very high computational performance • Fast video processing algorithms • IPs for DSP, ML and image processing	• Low price • Application specifically optimized
Weaknesses	• High power consumption • Sequential operations within core • Limited parallelism	• Difficult to program • Difficult to debug • Sequential operations are challenging	• High power consumption • Cannot perform certain algorithms • Application specific	• High power consumption • Cannot perform certain algorithms • Application specific
Suitable Domain Application in Electrified Vehicles	• Inter-domain HMI Interface • Inter-domain ADAS • Inter-domain Connectivity • Energy management systems • Supervisory/Domain Monitoring and Control	• Inter-domain ADAS • Motor control • Power electronics control • Sensor fusion • Battery management system • Inter-domain HMI Interface	• Sensor fusion • Inter-domain ADAS • Inter-domain HMI Interface • Camera vision interface	• Motor control • Power electronics control • Battery Management Systems • Connectivity

2.5.2 In-Vehicle Communication Protocols

The requirement of increased bandwidth has put in-vehicle communication protocols at the front of the new E/E architecture design to ensure reliability and safety are taken into consideration when adding new sensors and combining ECU functions together. This is due to the increasing number of electrical components that need to communicate with each other to ensure appropriate functionality [37]. Furthermore, the complication of including legacy devices and code has put challenges to the introduction of new E/E architectures since many communication protocols are currently included in conventional distributed architectures.

Up until recent years, the main communication protocol inside of vehicles has been the Controller Area Network (CAN) developed by Robert Bosch GmbH in 1986 [118]. The CAN protocol has been attractive for manufacturers due to its ability to be flexible, low cost, and scalability. Newer CAN protocols, such as the CAN flexible data rate (CAN-FD), have tried to increase the bandwidth from 1Mb/s all the way up to 10Mb/s for some instances [40].

FlexRay has been introduced by the consortium of BMW, Daimler Chrysler, Philips Semiconductors, Motorola and Bosch [118]. It is a network communication that has been introduced to enable the safety critical "X-By-wire" applications such as steering, throttle, braking and many more [37]. Flexray offers time-triggered communications, synchronized global time-frame and a real-time data transmission by using a time division multiple access (TDMA) technique which is a requirement for safety critical functions such as "X-By-wire" technologies where no mechanical link is present in case of failure [119].

Other communication protocols include Low Voltage Differential Signal (LVDS),

LIN and Media Oriented Systems Transport (MOST). LVDS communication has been widely used in many applications such as industrial, aerospace, telecom and as well in Automotive. LVDS comprises of a serial communication working in complimentary pairs to ensure the common mode noise is removed from the lines. This makes this communication robust but complex to implement as the length of the wires are the bottle neck of this application due to having a minimum of two wires for differential signals. LIN was introduced in 1998 to use in applications to supplement CAN protocol where cost is critical and the bandwidth is low [120]. It consists of a single wire and a low cost solution for simple actuation and sensing [118]. LIN is typically used for vehicle door, seat and temperature control devices. MOST has been introduced by MOST cooperation in 1998 [118]. MOST works well with any type of media such as video, audio, radio and more. The communication medium has been through optical fibers and can support up to 64 nodes in a ring topology [119].

Automotive Ethernet (AE) is a promising solution for solving the bandwidth problems related to data and being able to perform sensor fusion easily. AE has been a promising solution due to its similarity to normal Ethernet that has been used as the local area network (LAN) for most computers and day-to-day lives [40]. The wide acceptance of the Ethernet protocol has been the main driver in using AE in transportation vehicles [40]. In terms of fail-safe criteria, the Ethernet protocol includes many standards up-to-date that ensure the safe communication between Ethernet nodes. One disadvantage of using AE is that a switch node Ethernet device would need to be used. With the advancements in controller technologies described in Section 2.5.1, this would not be a problem as many functions can be integrated into one ECU along with an AE input decoder. A summary of the communication protocols

mentioned can be found in Table 2.3.

Table 2.3: Summary of possible inter-domain and gateway communication protocols for domain control oriented E/E Architecture.

	CAN-FD	**FlexRay**	**LIN**	**MOST**	**LVDS**	**Automotive Ethernet**
Bandwidth	1-10Mb/s	20Mb/s	20kb/s	150Mb/s	655Mb/s	1Gb/s
Strengths	• Robust • Low Cost • Supports distributed controls	• Fault Tolerant • Time Triggered (Similar to TTCAN) • Supports distributed controls	• Robust • Low Cost • Simplistic	• Robust • Optimized for images and networks • MOST25/50 can be optically driven	• General • Internal IP's can be generated.	• TCP/IP integration • High Bandwidth • Robust • Supports distributed controls
Weaknesses	• Obsolescence	• Obsolescence • Medium Cost	• Low bandwidth • Single failure point • Obsolescence	• High Cost • Complex • Extra hardware needed	• Extra hardware needed • Complex • Signal integrity is needed	• Complex • High cost • Extra hardware needed
Suitable application in Electrified Vehicles	Inter-domain safety critical applications and/or redundancy	Inter-domain ADAS and safety critical applications	Inter-domain HMI Interfaces and Body	Sensor fusion, Infotainment and redundancy	Inter-domain ADAS and safety critical applications	Gateway communication, ADAS and Inter-domain Connectivity

2.5.3 Connectivity

The concept and contents of vehicle connectivity have significantly expanded and have enabled a whole new world for the electric car to be designed in. Some of the key applied technologies involved in the connectivity impact of the E/E architecture are listed below [121]:

- WiFi

- Cellular Network
- Global Navigation Satellite System (GNSS)
- V2X (Vehicle-to-Vehicle, Vehicle-to-Grid, etc.)
- Dedicated Short-Range Communications (DSRC)
- Over-The-Air (OTA) updates

With these types of technologies, external communication of the vehicle can be manifested much easier than before. This brings in added safety with the added "nodes" that can communicate with the vehicle. Such nodes can be smart traffic lights, buildings, houses, cellphones, power generation units that can communicate with each other to ensure all information is passed between each vehicle [40]. This will add additional constraints to the vehicle EMS such that they can become more efficient in real-life and can adapt to its environment stimulus.

2.6 Summary

The present chapter has aimed to introduce a novel categorization followed by a detailed discussion of iEMSs. The features which enable an EMS to be intelligent are listed, which is rarely seen in other reviews. Detailed explanations, advantages, disadvantages, and future research directions are presented for each method. The analysis of RL-based, ANNs-based, and markov decision problem-based EMSs are undertaken here, which leads to more profound insight into iEMSs. Composite iEMSs are studied as a new category, and they can be explored more in future research. More

broadly, the review establishes an introduction to enabling technologies of implementing iEMSs.

Chapter 3

High-Fidelity Modeling of A Hybrid Electric Vehicle

To evaluate a proposed EMS, the vehicle model needs to be developed before the implementation stage. This allows us to consider different aspects of EMS and possible improvements. In this chapter, we present the detailed modeling of the selected HEV. The HEV model is built utilizing the Matlab simulink software.The challenge here is to build a realistic vehicle model that delivers the right accuracy and is acceptable for the application of EMS. To achieve these objectives, we validated the HEV model against experimental data provided by ANL.

Toyota Prius MY10 has one of the most commercialized transmissions that is considered as our case study. However, the vehicle model and proposed EMS can be extended for other HEV powertrains as well. The vehicle simulink model consists of three main parts: driver, controller, and vehicle powertrain. In order to simulate vehicle behavior, a reference driving cycle is fed to the model as the input, and the driver by utilizing a proportional-integral controller calculates the desired torque for

the actuators of the powertrain [122]. The controller block is where the EMS is implemented and it generates the control signals for the powertrain components. The required traction force at the wheel is computed in the powertrain block. Integration of the calculated force which is the simulated vehicle speed goes back to the driver block and creates a feedback loop. This method of implementation is called a forward modeling approach.

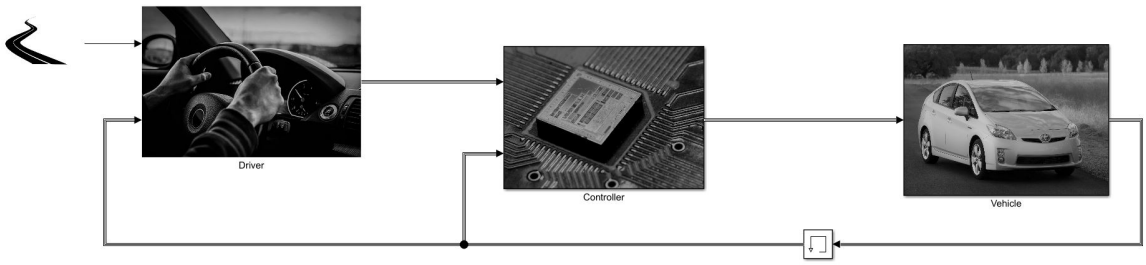


Figure 3.1: An overview of Toyota Prius model in Matlab Simulink.

3.1 Vehicle Model

Power sources, power split drive train, and vehicle dynamics are modeled in the vehicle subsystem. Figure 3.2 depicts the powertrain components in simulink. The quasi-static model is used to model the powertrain components, which has shown enough accuracy for control applications. Since the development of EMSs includes frequent model evaluation over a drive cycle, neglecting the transit behavior of vehicle components results in a fast vehicle model [123].

We used different articles and data sources to choose the vehicle parameters and lookup tables' data [124, 125, 126, 127, 128, 129]. For instance, the Oak Ridge National Laboratory provides a report on the Toyota Pius hybrid synergy drive system

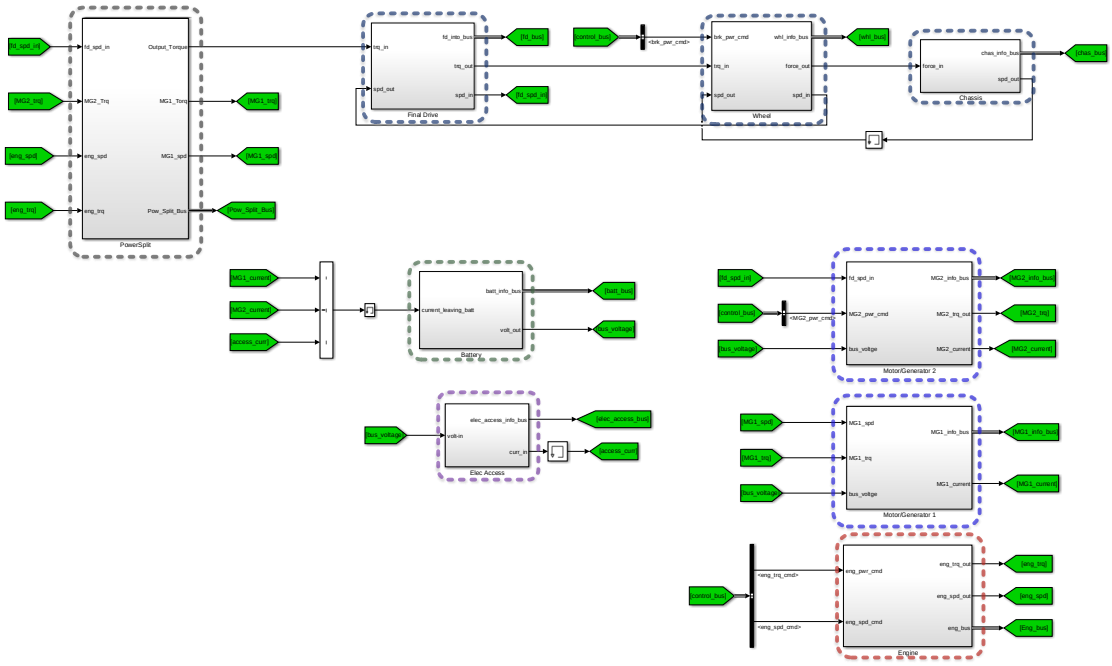


Figure 3.2: The powertrain simulink model.

[128]. Authors in [125, 126, 127] have examined the Toyota Prius MY10 testing data and validated their proposed model.

3.1.1 Internal Combustion Engine

A quasi-static ICE model is used to simulate ICE power and fuel consumption. Although the ICE quick dynamics and thermal effects are neglected here, the quasi-static ICE model has sufficient accuracy in simulating the EMS requirements. Engine torque and speed are the inputs of ICE block which determine the amount of fuel consumed through a fuel consumption lookup table. The fuel mass flow rate is calculated by equation 3.1.2. The maximum engine torque contour is also considered as a separate look-up table to guarantee that engine torque remains in its acceptable range.

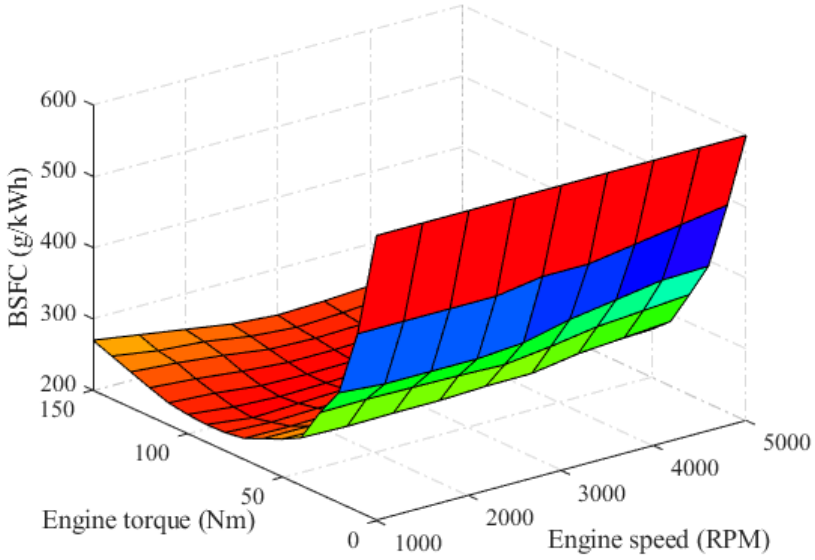


Figure 3.3: ICE fuel consumption map [130].

$$P_e = T_e \times w_e \tag{3.1.1}$$

$$fuel(k) = \frac{1}{\rho_{fuel} \times 3600000} \cdot P_e(k) \cdot f_e(n_e, T_e) \cdot t_{step} \tag{3.1.2}$$

where the P_e is the engine mechanical (output) power, $fuel(k)$ is the engine fuel mass flow rate at the kth step, T_e is the engine torque, w_e and n_e are the engine speeds in rad/s and rpm, respectively. ρ_{fuel} is the gasoline fuel density, and $f_e(n_e, T_e)$ represents the Brake Specific Fuel Consumption (BSFC) in g/kWh.

Table 3.1: 1.8 Liter Engine Parameters.

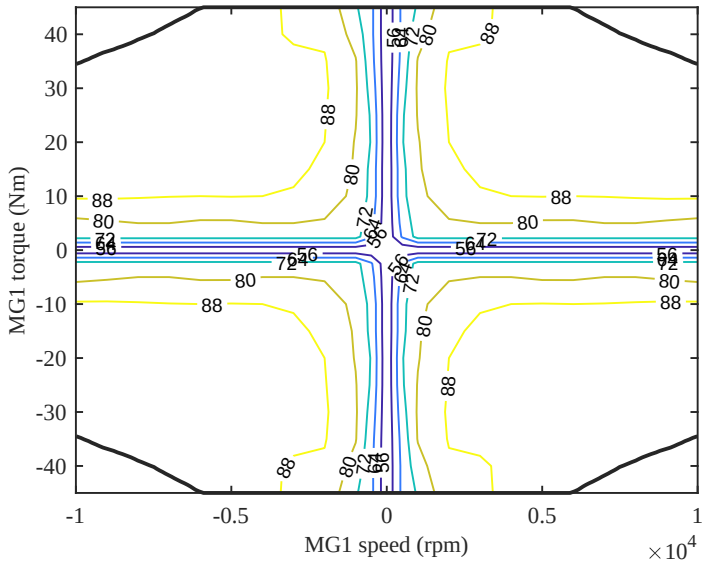
Parameter	Symbol	Value
Max. Power	$P_{e,max}$	73 kW
Max. Torque	$T_{e.max}$	142 Nm
Moment of inertia	J_e	0.18 kgm^2

3.1.2 Electric Machines

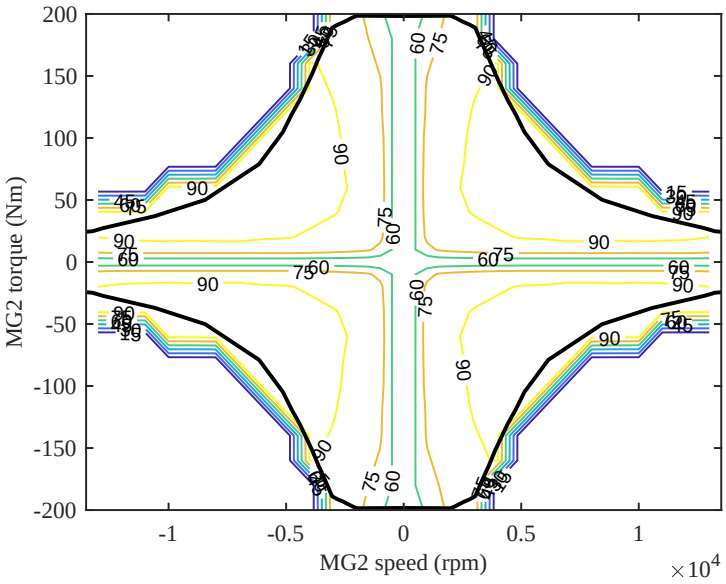
Permanent Magnet Synchronous Motors (PMSMs) have been used widely in vehicular propulsion systems due to their high efficiency, high power density, and high torque which permits higher speeds of these motors. Two PMSMs referred to as Motor/Generator 1 (MG1), the smaller one, and Motor/Generator 2 (MG2), the one usually operates in motor mode, are employed in Toyota Prius MY10. The static map model is used to simulate both electric machines and lookup tables to ensure the MG1 and MG2 torques not exceed their maximum capability. Also, MG1 and MG2 efficiency maps which are a function of torque and speed, are utilized to compute the electrical power utilizing equation 3.1.4. It should noted that the efficiency of an inverter has been emerged into the efficiency of the electric machines.

$$P_{MG,mech} = T_{MG} \times w_{MG} \tag{3.1.3}$$

$$P_{MG,elec} = \begin{cases} \eta_{MG} \cdot P_{MG,mech} & if P_{MG,mech} \geq 0 \\ \dfrac{1}{\eta_{MG}} \cdot P_{MG,mech} & if P_{MG,mech} < 0 \end{cases} \tag{3.1.4}$$



(a) MG1 efficiency map.



(b) MG2 efficiency map.

Figure 3.4: MGs efficiency maps. Contours show the MGs efficiency and the black solid line are the torque maximum versus MGs speed.

where η_{MG} is the MG efficiency and it maps the torque and speed to the efficiency value of MG1 or MG2. The equation 3.1.4 is valid for both MG1 and MG2 in the HEV model.

Table 3.2: MGs Parameters.

Component	Parameter	Symbol	Value
MG1	Max. Power	$P_{MG1,max}$	42 kW
	Max. Torque	$T_{MG1,max}$	45 Nm
	Moment of inertia	J_{mg1}	0.02 kgm^2
MG2	Max. Power	$P_{MG2,max}$	60 kW
	Max. Torque	$T_{MG2,max}$	207 Nm
	Moment of inertia	J_{mg2}	0.05 kgm^2

3.1.3 Battery

Considering three main battery chemistries, which are lead-acid, Nickel-Metal Hydride (NiMH), and Lithium-ion (Li-ion), NiMH is best suited for an HEV application due to its high energy density and reasonable cost. The Toyota Prius battery pack consists of 28 Panasonic prismatic NiMH modules-each containing 6 numbers of 1.2 V cells-connected in series to produce a nominal voltage of 201.6 volts [128]. The battery specifications are summarized in Table 3.3. The battery is modeled as a simple electrical circuit with a voltage source and a resistance in series that depicted in Figure 3.5. More complex model could be used to present battery behavior, however validation process with testing data shows that the accuracy of utilized battery model is sufficient for the purpose of this thesis. A lookup table is used to obtain

the open-circuit voltage V_{oc} which is a function of SOC. Battery terminal voltage and power are calculated by equations 3.1.5 and 3.1.6, respectively.

$$V_b = Voc - R_{int} \times I_b \tag{3.1.5}$$

$$P_b = V_b \times I_b = Voc \times I_b - R_{int} \times I_b^2 \tag{3.1.6}$$

Therefore, battery current can be obtained by below equation:

$$I_b = \frac{V_{oc} - \sqrt{V_{oc}^2 - 4R_{int}P_b}}{2R_{int}} \tag{3.1.7}$$

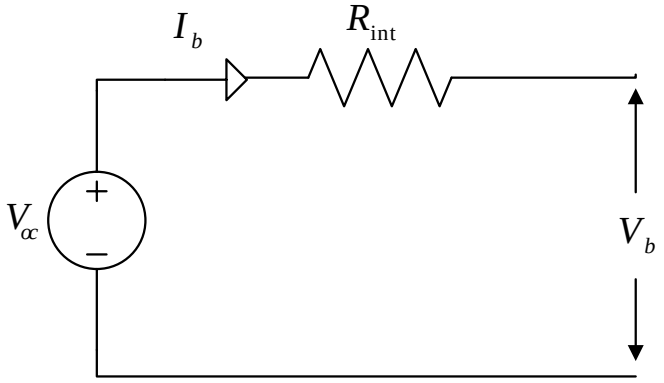


Figure 3.5: The OCV-R battery model.

and battery SOC is estimated utilizing coulomb-counting method:

$$SOC(k+1) = SOC(k) - \frac{I_b(k)}{3600 \times C_b} \cdot t_{step} \tag{3.1.8}$$

SOC, is the battery state of charge, I_b is the battery current, t_{step} is the simulation period and C_b is the battery nominal capacity in ampere hours. Here, positive sign

Table 3.3: Battery Parameters

Parameter	Symbol	Value
Internal Resistance	R_{int}	0.32 Ω
Nominal Voltage	V_{oc}	201.6 V
Capacity	C_b	6.5 Ah

is considered for battery discharging and negative sign for battery charging. Modeling the boost converter which is located between the battery and the two electrical machines is not considered in this study.

3.1.4 Power Split Device

The power split device consists of two Planetary Gear Sets (PGSs). A PGS comprises a ring, carrier, and sun gear that connects through planet gears. As is shown in Figure 3.6, MG1 and ICE are connected to the sun and carrier gear of the first PGS, respectively. The MG2 is coupled to the sun gear of the second set which its carrier is grounded. The first PGS is responsible for splitting the ICE power and the second PGS acts as a fixed reduction gear for MG2 [125]. To model a power split device in Matlab simulink, we used quasi-static equations. It should be noted that the power split losses are not included here.

$$r_1 \cdot w_{R1} + w_{mg1} = (1 + r_1) \cdot w_e \tag{3.1.9}$$

$$T_e = -\frac{r_1}{1 + r_1} \cdot T_{mg1} = -(r_1 + 1) \cdot T_{R1} \tag{3.1.10}$$

$$w_{mg2} = r_2 \cdot w_{R2} \tag{3.1.11}$$

$$w_{R1} = w_{R2} = r_d \cdot w_{wheel} \tag{3.1.12}$$

$$T_{R2} = r_2 \cdot T_{mg2} \tag{3.1.13}$$

$r_i = N_{ri}/N_{si}, i = 1, 2$ represents the planatory gear ratio in which N_{ri} is the number of teeth in the ring gear and N_{ri} is the number of teeth in the sun gear.

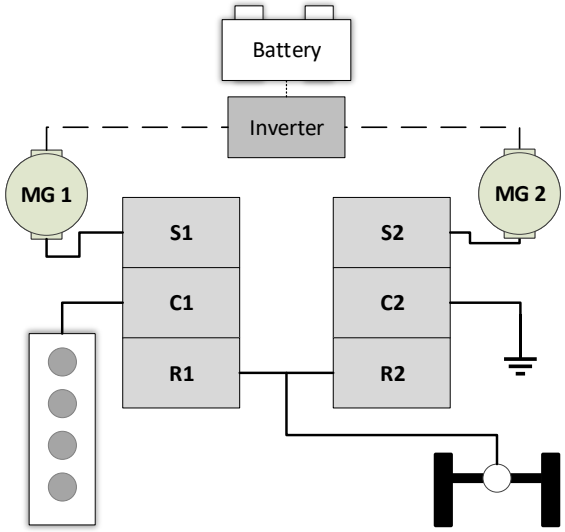


Figure 3.6: Selected HEV architecture .

Table 3.4: Power Split Device Parameters.

Component	Parameter	Symbol	Value
PGS1	Number of teeth in ring gear	N_r	78
	Number of teeth in sun gear	N_s	30
PGS2	Number of teeth in ring gear	N_r	58
	Number of teeth in sun gear	N_s	22

3.1.5 Final Drive

The final drive block task is to transfer the power from the power split device to the wheels. To consider final drive power loss, final drive efficiency is assumed as a constant number $\eta_d = 0.95$, and the total final drive ratio is considered as $r_d = 3.268$.

Table 3.5: Final Drive Parameters.

Parameter	Symbol	Value
Final drive efficiency	η_d	0.95
Final drive ratio	r_d	3.268
Moment of inertia	J_d	0.1 kgm^2

3.1.6 Wheels

The wheels are modeled to accounts for rolling resistance power losses. The wheel radius value is considered as $r_{wh} = 0.31\ m$. Wheel output force and speed are as

follows:

$$w_{wh} = \frac{v_{veh}}{r_{wh}} \tag{3.1.14}$$

$$T_{wh} = \frac{T_{in} + T_{brake} - T_{roll}}{r_{wh}} \tag{3.1.15}$$

where the v_{veh} is the vehicle longitudinal speed, T_{brake} is the brake torque which comes from the controller block. T_{roll} represents the rolling resistance losses.

Table 3.6: Wheel Parameters.

Parameter	Symbol	Value
Wheel radius	r_{wh}	0.31 m
Moment of inertia	J_{wh}	0.8 kgm^2

3.1.7 Chassis

The vehicle free body diagram is shown in Figure 3.7. Newton's second low derives the longitudinal vehicle dynamics by considering all the forces which are acting on the vehicle. The road is considered as a flat plane with a grade of θ. The road load forces include gravitational force, aerodynamic resistance, and rolling resistance. Therefore, the vehicle dynamic equations are as follows:

$$F_{grade} = m_v g sin(\theta) \tag{3.1.16}$$

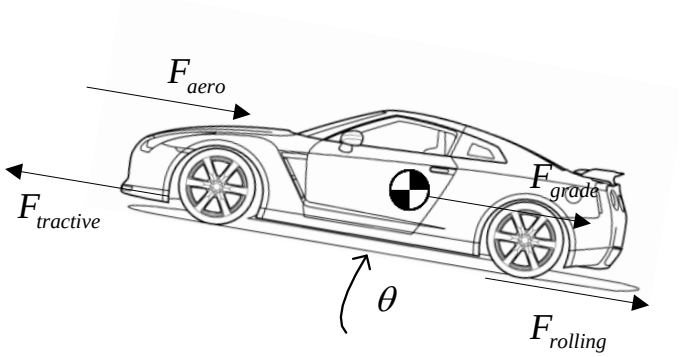


Figure 3.7: Free body diagram of a vehicle longitudinal dynamics.

F_{grade} is the gravitational force, where m_v stands for the vehicle mass and g is the gravitational acceleration equals to 9.81 m/s^2.

$$F_{roll} = m_v g \left(C_o + C_1 v\right) cos(\theta) \tag{3.1.17}$$

F_{roll} is the rolling resistance force, C_o and C_1 are the rolling resistance coefficients.

$$F_{aero} = \frac{1}{2} \rho_{air} C_d A v_{veh}^2 \tag{3.1.18}$$

F_{aero} is the aerodynamic force, ρ_{air} is the air mass density, C_d is the aerodynamic drag coefficient of the vehicle and A is the vehicle frontal area of. Therefore, the tractive force can be calculated as follows:

$$F_{tractive} = m_{equiv}\, a_{veh} + F_{aero} + F_{roll} + F_{grade} \tag{3.1.19}$$

where $F_{tractive}$ is the tractive force required at the wheel, a is the acceleration of the vehicle. By considering components' inertia, the vehicle equivalent mass can be yielded as:

$$\begin{aligned} J_{total} &= (r_1^2 J_{mg1} + (\frac{r_1}{1+r_1})^2 J_e + r_2^2 J_{mg2}) r_d^2 + J_d + 4J_{wh} \\ m_{inertia} &= \frac{J_{total}}{r_{wh}^2} \\ m_{equiv} &= m_v + m_{inertia} \end{aligned} \tag{3.1.20}$$

The vehicle longitudinal velocity is determined by implementing equation 3.1.19 in the chassis block.

In order to validate the model of longitudinal vehicle dynamics, the simulated tractive force is compared with the testing data of the tractive force for steady-state drive cycle testing data.

Table 3.7: Chassis Parameters.

Parameter	Symbol	Value
Vehicle mass	m_v	1480 kg
Gravitational constant	g	9.81 m/s^2
Constant term of the rolling resistance polynomial	C_0	0.002
Linear term of the rolling resistance polynomial	C_1	0.0002 s/m
Drag coefficient	C_d	0.25
Frontal Area	A	2.5 m^2
Air density	ρ_{air}	1.225 kg/m^3

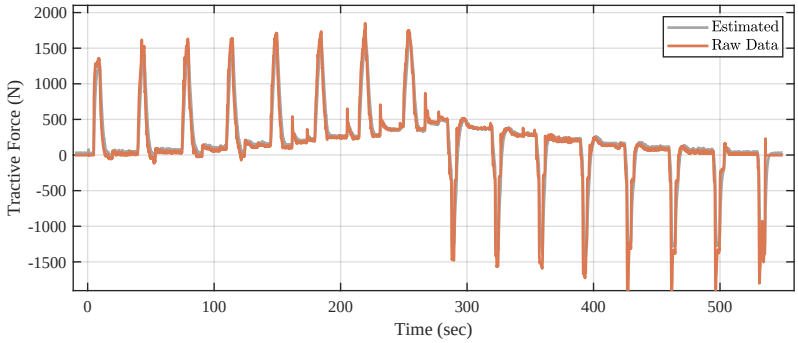


Figure 3.8: Simulated tractive force versus the testing data of the tractive force for steady state driving schedule. The RMS error is equal to 73.9 Nm.

3.2 Driver

The main driver task is to track the desired driving cycle as its input. The driver model consists of two primary parts: 1) feedforward block 2) feedback block. The feedforward block is responsible for generating the requested power and torque by utilizing the equations which are provided in the Section 3.1.7. The feedback block, which is a PID controller, aims to control the speed. Powertrain model is prone to physical limitations such as motor speed, torque, and other components criteria. When an actuator signal is saturated, the PID controller can compensate for the speed or torque deviation. The PID controller coefficients are tuned to match the vehicle speed with the reference driving cycle. Figure 3.9 shows an overview of the driver model, which receives the reference and vehicle simulated speed and outputs the desired propulsion and braking power.

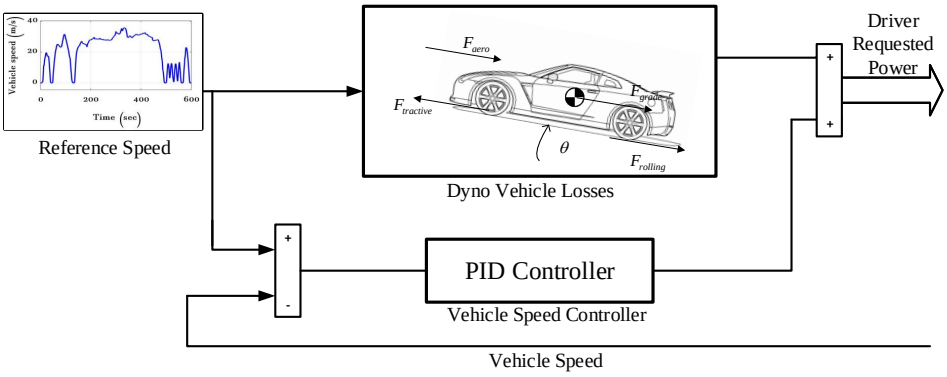


Figure 3.9: Driver schematic in simulink environment.

3.3 Energy Management Control Unit

In order to validate our model, a rule-based EMS is chosen that imitates the rule-based method, which is implemented in real Toyota Prius MY10. There are several studies that analyze the testing data provided by ANL to extract the baseline rule-based strategy [125, 131]. A flow chart is provided in Figure 3.10 to give an overview of the implemented EMS.

P_{th}, V_{th}, and SOC_{th} are the threshold values which determine the engine on or off mode. When the vehicle is accelerating (propulsion mode), there is only one way that the ICE turns off in which the SOC is greater than the SOC_{th} while the vehicle power demand and speed are lower than their thresholds to prevent the battery from over depletion. When the vehicle is decelerating (braking mode), the engine goes off. However, comparing engine speed with vehicle speed dataset leads to the fact that in most cases, when the vehicle speed is greater than V_{th}, the engine speed remains at idle speed. This would benefit the engine to start again from idling mode.

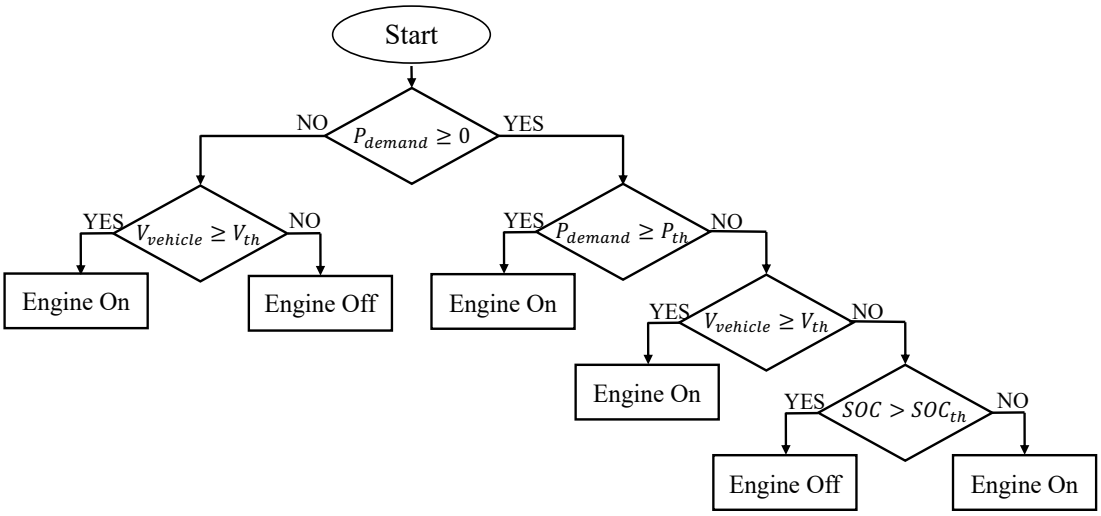


Figure 3.10: The rule based EMS flow chart.

3.3.1 Engine on-Propulsion

In this mode, engine power is calculated by subtracting power demand and battery power. The relation between battery SOC and battery power is shown in Figure 3.11. Having the engine power and using the data provided in Figure 3.11 helps us to find the engine speed. The rest of the components' torque and speed along with braking command are determined as follows:

$$\begin{aligned} P_{batt} &= f_1(SOC) \\ P_{eng} &= P_{dmd} - P_{batt} \\ w_{eng} &= f_2(P_{dmd}) \\ T_{eng} &= P_{eng}/w_e \end{aligned} \tag{3.3.1}$$

w_{mg1} is calculated by using equations 3.1.9 and 3.1.12. T_{mg1} is computed by equation 3.1.10. w_{mg2} can be obtained using equation 3.1.12. Now, MG2 torque can be determined as:

$$T_{mg2} = (P_{batt} - P_{mg1})/w_{mg2} \tag{3.3.2}$$

Since the vehicle is in propulsion mode, the braking power is equal to zero.

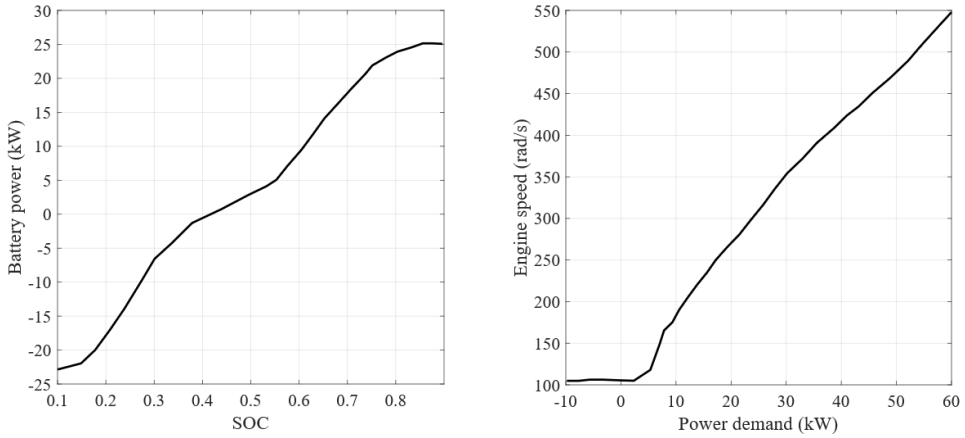


Figure 3.11: The left box shows f_1 and the right box represents f_2.

3.3.2 Engine off-Propulsion

In the acceleration mode, if the power demand and vehicle speed are lower than their threshold and the battery soc is above its minimum value, the engine can be turned off. Therefore, the vehicle switches to pure electric mode and the engine torque and speed are zero. Since the MG1 is coupled to the engine, the MG1 torque and speed value are zero as well and the vehicle is running solely by MG2. w_{mg2} is obtained by

equation 3.1.12 and T_{mg2} can be written as :

$$\begin{aligned} w_{eng} &= 0 \\ T_e &= 0 \\ w_{mg1} &= 0 \\ T_{mg1} &= 0 \\ T_{mg2} &= P_{dmd}/w_{mg2} \end{aligned} \tag{3.3.3}$$

The vehicle does not provide braking power command in this mode. Therefore:

$$P_{brk} = 0 \tag{3.3.4}$$

3.3.3 Engine on-Braking

During deceleration, if the vehicle speed is above the speed threshold, the engine does not turn off and it keeps idling. This idling state will help engine to be ready for providing power again.

$$\begin{aligned} w_{eng} &= 1000\ rpm \\ P_{eng} &= 1000\ W \\ T_{eng} &= P_{eng}/w_e \end{aligned} \tag{3.3.5}$$

w_{mg1} is calculated by using equations 3.1.9 and 3.1.12. T_{mg1} is computed by equation 3.1.10. w_{mg2} can be obtained using equation 3.1.12. MG2 torque can be determined as:

$$T_{mg2} = f_3(v) \cdot P_{dmd}/w_{eng} \tag{3.3.6}$$

where the $f_3(v)$ stands for regenerative braking fraction which is function of vehicle linear speed and it is modeled by utilizing a look-up table. The braking power formula is as follows:

$$P_{brk} = P_{dmd} - P_{mg2} \tag{3.3.7}$$

3.3.4 Engine off-Braking

When the driver takes his foot off the pedal and the vehicle speed is decreasing to below the vehicle speed threshold, the engine fully turns off. The engine torque and speed equal to zero. MG1 speed and torque, MG2 speed and torque, and braking power command can be calculated same as engine on -braking in Section 3.3.3.

3.4 Model Validation with ANL data

To compare the future proposed energy management strategy with the baseline Toyota Prius MY10 control strategy, it is crucial to establish a vehicle model that imitates the real HEV behavior. In this section, simulink's developed model is validated using the Toyota Prius MY10 chassis dynamometer test data. Simulated vehicle speed and engine speed are compared with measured data in Figures 3.12 and 3.13.

Several reasons lead to some slight differences between the simulated and raw data. Our model does not consider the effect of temperature on powertrain components. For instance, the engine thermal model is not developed in our model that leads to a discrepancy between the simulated engine speed and testing data for the first 100 seconds of the UDDS drive cycle where the vehicle is warming up [125]. The discrepancy between simulated and testing data can also happen due to the simplifications

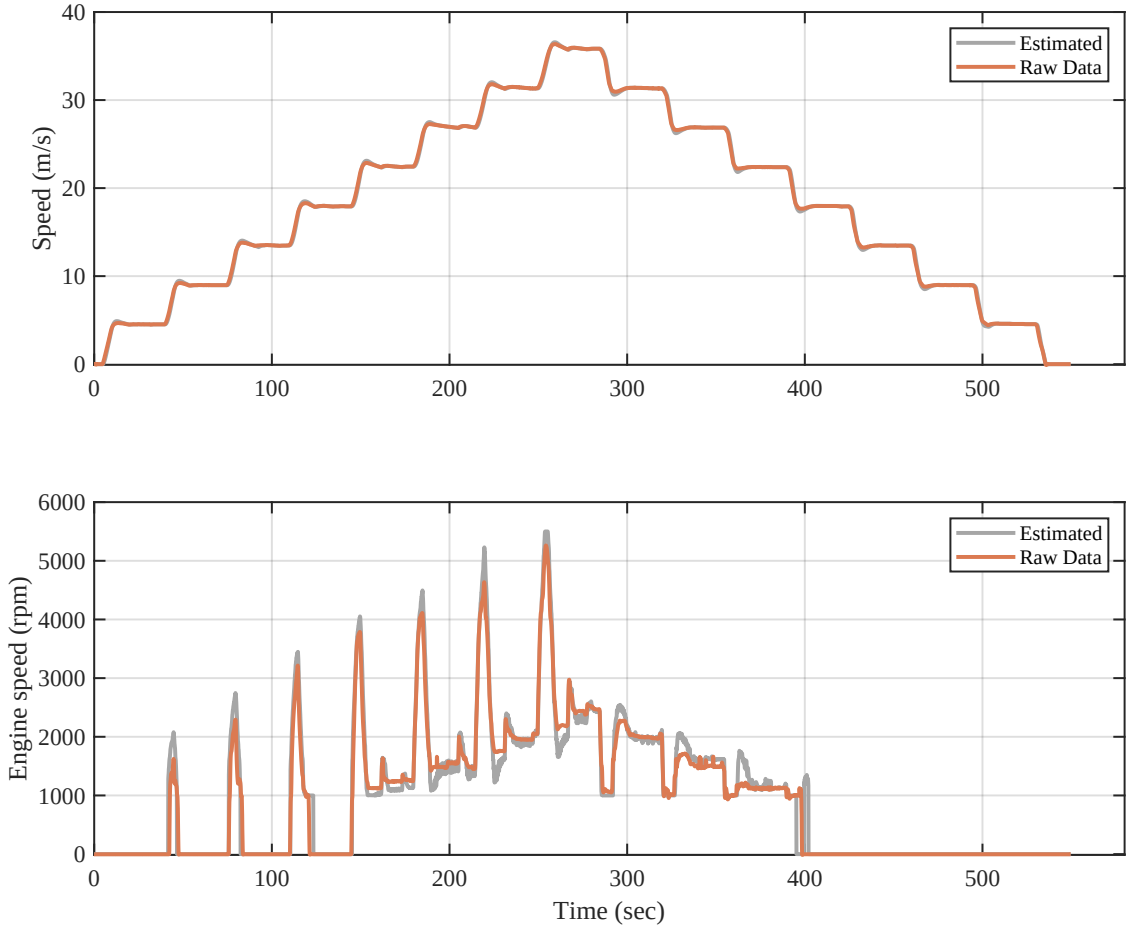


Figure 3.12: Simulated vehicle speed and engine speed versus testing data for steady-state driving schedule. The RMS error is equal to 0.16 m/s, and 156.34 rpm, respectively.

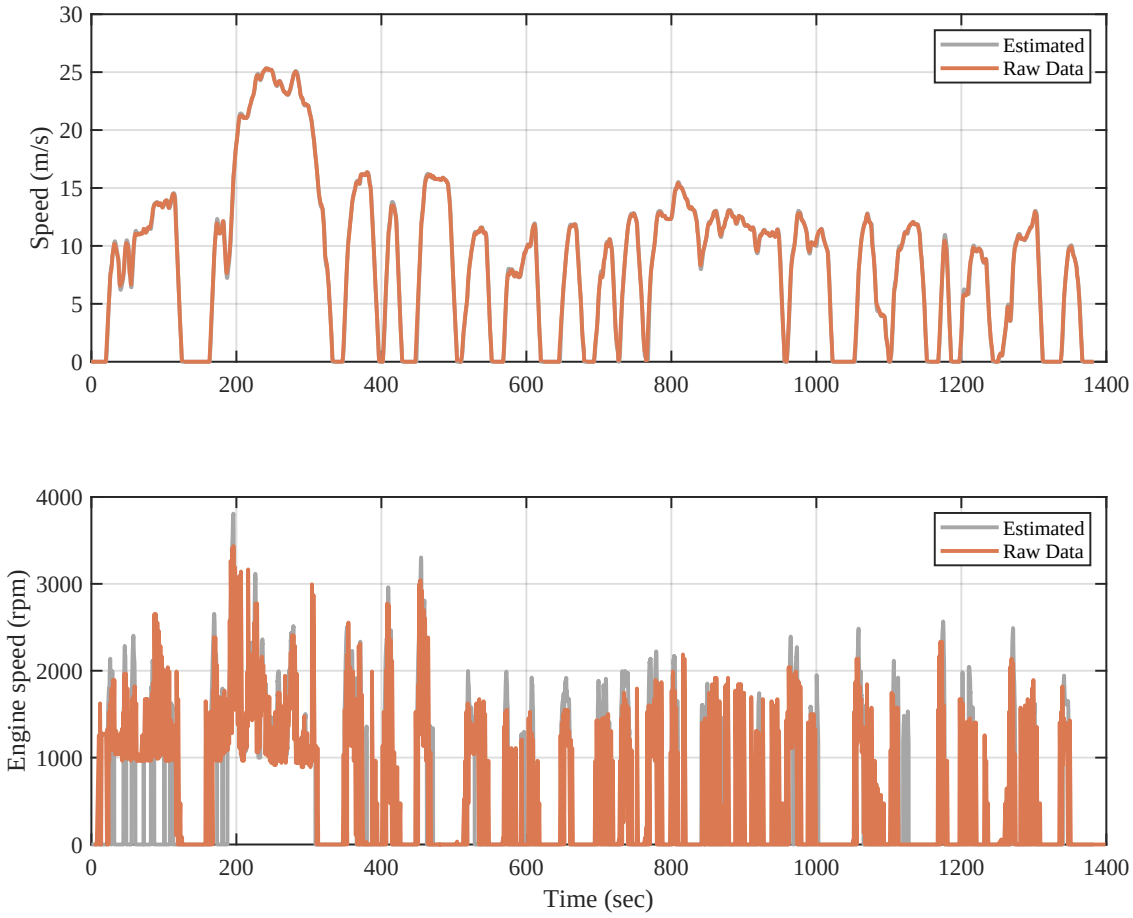
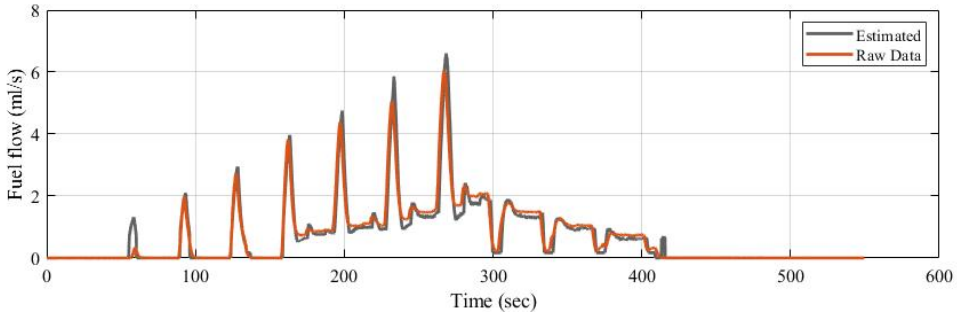


Figure 3.13: Simulated vehicle speed and engine speed versus testing data for UDDS driving schedule. The RMS error is equal to 0.24 m/s, and 615.35 rpm, respectively.

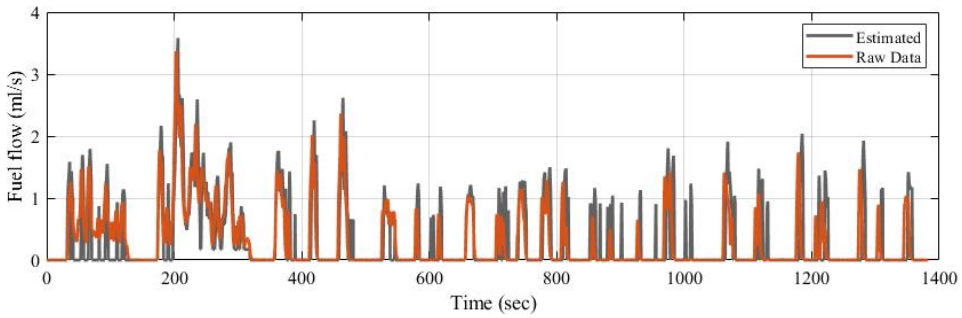
that have been made on the simulink model. For instance, the model does not account for the dynamic of the vehicle components, including power split dynamics, tire dynamics, engine dynamics, etc. Even though modeling the components' dynamics improves model accuracy, it will enhance the model complexity and computational time. Therefore, the developed simulink model sufficiently models the real HEV for controls development. Our study focuses on improving fuel economy and battery charge sustainability; hence, in the following sections, fuel consumption and battery SOC trajectory are validated against testing data.

3.4.1 Fuel Consumption Model Evaluation

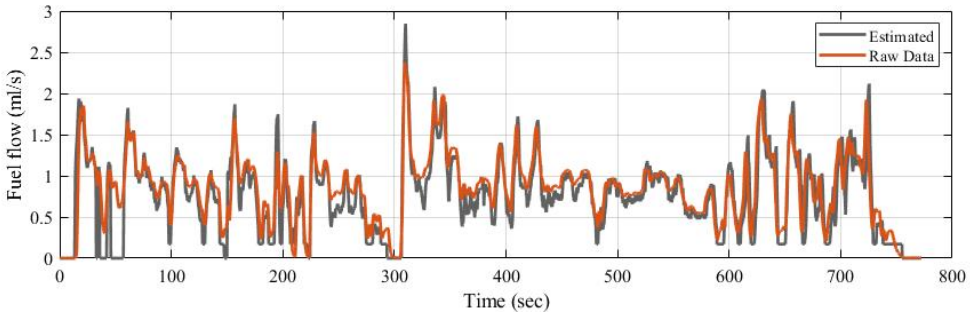
Engine fuel consumption is calculated using equation 3.1.2 in our simulink model. The fuel flow rate is shown here over UDDS, steady state, and Highway drive cycles. In the first approximately 100 seconds of the UDDS drive cycle, the estimated fuel does not follow the raw data. The reason is that we did not consider the engine thermal model and fuel consumption is affected by engine warm up process at the beginning of cycle. Following the 100 seconds, the fuel flow rates are in a good agreement.



(a) Fuel flow rate raw data compared to simulated fuel flow rate in steady-state driving cycle.



(b) Fuel flow rate raw data compared to simulated fuel flow rate in UDDS driving cycle.



(c) Fuel flow rate raw data compared to simulated fuel flow rate in Highway driving cycle.

Figure 3.14: Fuel flow rate comparison for each driving schedule.

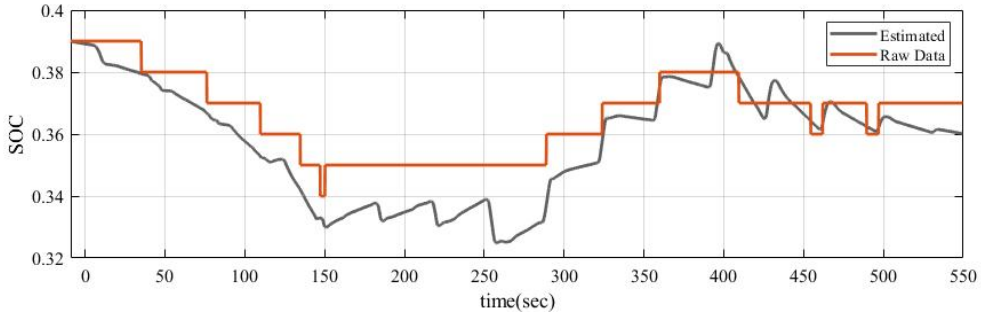
Table 3.8: Total fuel consumption results.

Driving cycle	Raw data (ml)	Estimated fuel (ml)	Error (%)
UDDS	403.5	414.7	+2.7
Highway	576.8	590.4	+2.3
US06	729.5	700.5	-3.9
Steady-state	388.8	376.1	-3.2

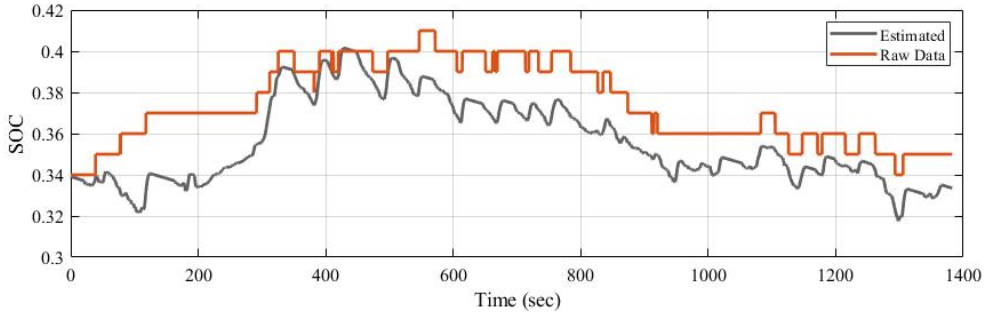
Generally, the estimated fuel consumption follows the testing data. Results are provided in Table 3.8 for four drive cycles. The estimated fuel consumption results are within 5% of actual measured data and that is an enough accuracy for analyzing the energy management of our model.

3.4.2 Battery State of Charge Model Evaluation

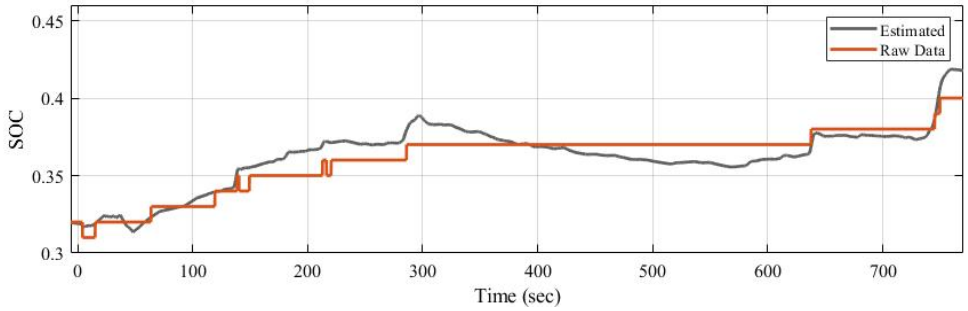
In this section, the battery SOC trajectories of three driving cycles are presented and compared with testing data. It can be seen from figures that estimated SOC trajectories follow the testing data perfectly. However, some gaps can be found between raw and estimated data. For instance, at the beginning of UDDS drive cycle, in the testing data, engine turns on and the battery is charging. But in the simulated SOC trajectory, the battery is discharging due to engine off condition. Following by 100 second, the simulated SOC trajectory keeps the pattern of testing data set.



(a) Comparison of SOC trajectories for Steady State driving schedule.



(b) Comparison of SOC trajectories for UDDS driving schedule.



(c) Comparison of SOC trajectories for Highway driving schedule.

Figure 3.15: SOC trajectory comparison for each driving schedule.

Table 3.9: Final SOC value comparison results for each driving schedule.

Driving cycle	Raw data	Estimated SOC	Diff.
UDDS	0.35	0.333	-0.017
Highway	0.40	0.417	0.017
US06	0.44	0.441	0.001
Steady-state	0.37	0.36	-0.01

3.5 Summary

The detailed model of powertrain components, baseline energy management strategy, and the driver for Toyota Prius MY10 HEV have been presented in this chapter. This chapter aimed to develop a model with acceptable accuracy for further analyzing fuel consumption and battery performance. Validation results showed that the model's accuracy is sufficient to assess the effectiveness of future EMS strategies.

Chapter 4

Implementation of an Offline Optimal Energy Management Strategy

4.1 Dynamic Programming

Global optimization EMSs have a pivotal role in the development of an iEMS. The goal of these methods is mostly minimization of fuel consumption, which has been a significant concern for the electrified vehicle. These methods provide benchmark solutions in order to evaluate and improve iEMSs. For instance, rules in rule-based methods can be extracted or modified by comparing with DP results [132]. Control parameters in real-time EMSs are usually optimized offline utilizing global optimization algorithms. Authors in [133] have performed DP several times to have an estimate of the optimal parameters to determine the equivalence factor. Recent advances in AI has made data-driven approaches such as ANN-based EMSs more popular. The

accuracy of an ANN-based EMS highly depends on the quality and precision of the ANN training dataset. To ensure the optimality of training data sets, several series of global optimization algorithms on different driving schedules are conducted that prepares the training dataset [88]. Optimal sizing of powertrain components has been widely researched employing global optimization EMSs. The component sizing, along with managing the power split among different energy sources, contributes to efficient operation in terms of components' lifetime, total powertrain cost, and energy consumption. Authors in [134] have considered a multi-objective optimization problem, including optimal component sizing, gear shifting, and engine on-off command utilizing the DP and convex optimization approaches. Authors in [135] have set out three optimization targets, including reducing the battery and fuel cell degradation, minimizing the energy consumption of supercapacitors, and the optimal design of those energy sources for a hybrid electric city bus.

Considering the above-mentioned advantages of global optimization EMSs, it is worth studying DP for our selected power split HEV. We have formulated DP for the Toyota Prius 2010 that provides valuable results that can be used to improve the performance of the real-time EMS which will be introduced in the next chapter.

There is a large volume of published studies implementing DP on different powertrain architectures. Authors in [136] have conducted a study on a parallel HEV to compare two EMSs including DP and a combined DP-Pontryagin's Minimum Principle (PMP) to optimize the gear shift command, power split ratio, start-stop functionality, and fuel consumption. The results in [136] confirm the optimality of the DP-PMP method, which has a close performance to DP. Authors in [137] have examined two power-split configurations in terms of fuel economy. They used DP

as a global optimization method to evaluate the effectiveness of adding clutches in the power split configuration that improves the fuel economy. Authors in [132] have performed DP to optimize a baseline rule-based method on a series PHEV in order to mimic the DP performance. Authors in [138] have utilized DP to assess the fuel economy performance of SDP and ECMS methods on a Toyota Prius 2010.

DP is computationally intensive. A growing body of literature has investigated reducing the DP computational time [132, 139, 140]. However, these studies mostly have assumed some simplifications, which affect the optimal results or they suggested using prepared data set to eliminate the number of control variables, which might deteriorate DP results. In our approach, we attempted to eliminate dependent control variables repeatedly used in DP employing for Toyota Prius 2010 in other studies. Besides, the goal of DP in our studies is not only fuel minimization but also improving drivability which prevents excessive engine on events.

4.2 Mathematical Principle

DP is a numerical optimization method based on Bellman's principle of optimality [58]. It is widely used to provide benchmark solutions as a global optimization algorithm for EMS problems. To explain the DP concept, consider a discrete-time system:

$$x_{k+1} = f(x_k, u_k), \qquad k = 0, 1, ..., N-1 \tag{4.2.1}$$

Where the x_k is the state vector within state space $X(k)$ and u_k are the control variables within input space $U_k(x_k)$. The DP approach as a global optimization algorithm is responsible for finding the control sequences which minimize the cumulative cost

while respecting the control and state constraints. Considering $\pi = [u_0, u_1, ..., u_{N-1}]$ as the control policy, the discrete-time format of total cost function is expressed as follows:

$$J_\pi = g(x_N) + \sum_{k=0}^{N-1} L(x_k, u_k) \tag{4.2.2}$$

where J_π is the aggregated cost, $g(x_N)$ is the terminal cost, and $L(x_k, u_k)$ is the instantaneous cost. The optimal control policy $\pi^* = [u_0^*, u_1^*, ..., u_{N-1}^*]$ is the policy that minimize the total cost (equation 4.2.2):

$$J_{\pi^*} = \min_{\pi} J_\pi(x_0) \tag{4.2.3}$$

DP as a backward optimization algorithm, starts from the final step N and it proceeds backward by obtaining the optimal control sequences through minimizing the below expression:

$$J_k^*(x_k) = \min_{u_k}[L(x_k, u_k) + J_{k+1}^*(x_{k+1})] \qquad k = N-1, N-2, ..., 1, 0 \tag{4.2.4}$$

with intial value

$$J_N^*(x_N) = g(x_N)$$

where the $J_k^*(x_k)$ expresses the optimal cost-to-go function at state x_k including the kth time step to the final step N. At the step (k=0), the algorithm achieves the minimum total cost $J^*(x_0)$.

The result of backward DP process is the optimal control policy $u_k^*(x_k)$ for $k = 0, .., N-1$. In fact, DP evaluates all admissible control values at each amissible state value to minimize the equation at each grid point. Figure 4.1 shows the DP backward

process. During forward simulation, when the initial state value is determined, the optimal control signals map along with equation 4.2.1 will be utilized to calculate the state trajectory and obtain the optimal control sequence.

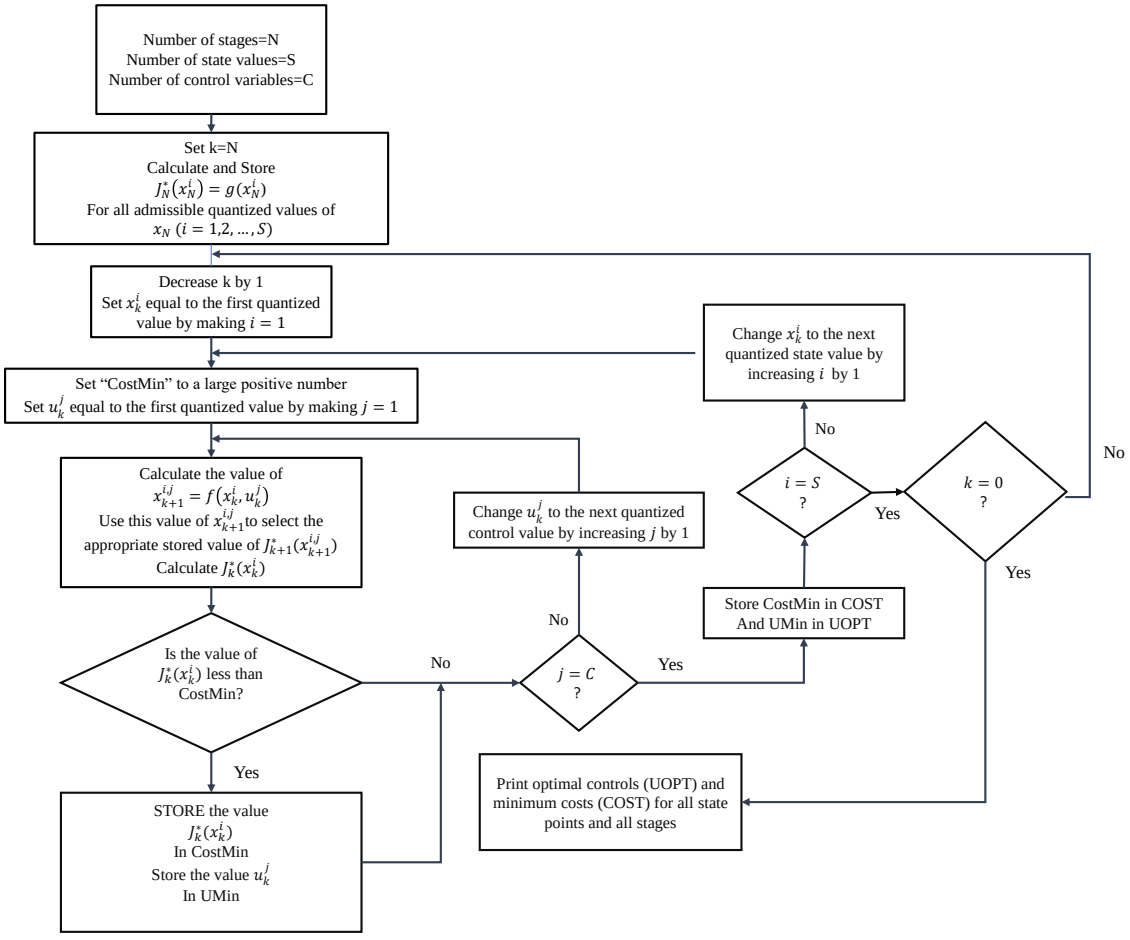


Figure 4.1: DP backward computational procedure. (Adapted from [141])

4.3 Applying Dynamic Programming to HEV

4.3.1 State and Control Variables Selection

To start implementing DP, we need to choose the appropriate state and control variables. As it is explained in section 2.3.2, with adding the number of state and control variables, the computational time would increase exponentially. Therefore, we attempted to pick independent control and state variables which are adequate to model the selected power-split HEV for the DP purposes.

The ultimate goal here is to minimize the total fuel consumption, which is the summation of fuel mass rates over the whole drive cycle. Engine fuel mass rate at each time step is calculated as follows:

$$\begin{gathered} P_e = T_e \times w_e \\ fuel(k) = \frac{1}{\rho_{fuel} \times 3600000} \cdot P_e(k) \cdot f_e(n_e, T_e) \cdot t_{step} \end{gathered} \tag{4.3.1}$$

Looking at the above equation, at first glance, it seems that the engine torque and speed can be chosen as the control variables since they determine the cost function value at each instant. However, we need to make sure that these two variables are sufficient to determine the operating conditions of all the powertrain components. Let's consider the PGs equations which are shown below:

$$r_1 \cdot w_{R1} + w_{mg1} = (1 + r_1) \cdot w_e \tag{4.3.2}$$

$$\begin{aligned} w_{R1} = w_{R2} = r_d \cdot w_{wheel} \\ w_{mg2} = r_2 \cdot w_{R2} \end{aligned} \tag{4.3.3}$$

$$T_e = -\frac{r_1}{1+r_1} \cdot T_{mg1} \tag{4.3.4}$$

$$T_{mg2} = \frac{T_{in}}{r_2} - \frac{r_1}{r_1+1} \cdot \frac{1}{r_2} \cdot T_e \tag{4.3.5}$$

During DP implementation, the vehicle speed is known in advance, and the MG2 speed is calculated by equation 4.3.3. Having the engine torque as the control variable, MG2 torque is calculated by equation 4.3.5 . Assuming the engine speed and torque as the control variables, the MG1 speed and torque can be obtained by equation 4.3.2 and 4.3.4, respectively. Hence, the engine torque and speed are adequate to determine the operations of all the powertrain components.

So far we have determined the control variables, but the state variables are still not defined. For the state variable, battery SOC is chosen since it represents the battery's dynamic and electrical energy path

$$SOC(k+1) = SOC(k) - \frac{I_b(k)}{3600 \times C_b} \cdot t_{step} \tag{4.3.6}$$

The selected powertrain architecture allows two operating modes, which are HEV and EV modes depending on the engine status. To avoid unpleasant deriving experience due to excessive engine start events, we can penalize the engine start events by considering the engine mode as the second state variable. Engine mode can take only two values where 1 shows the on status, and 0 is for the off status. The control

Table 4.1: DP control and state variables.

State Variables	Control Variables
SOC	T_e
$mode$	w_e

and state variables selection are summarized in the Table 4.1.

Now the global optimization EMS problem is summarized as follows:

$$\begin{aligned}
&\underset{\pi}{minimize}\ J(u_k, x_k) = \sum_{k=0}^{N-1} L(x_k, u_k) \\
&s.t. \\
&L(x_k, u_k) = [\dot{m}_f(T_e(k), w_e(k)) + \mu \cdot (engine_s(k)) > 0] \\
&x(k) = [SOC(k), mode(k)] \\
&u(k) = [T_e(k), w_e(k)] \\
&SOC(0) = SOC(N) \\
&SOC \in [SOC_{min}, SOC_{max}] \\
&mode = [0, 1] \\
&T_e \in [T_{e_{min}}, T_{e_{max}}] \\
&T_{mg2} \in [T_{mg2_{min}}, T_{mg2_{max}}] \\
&T_{mg1} \in [T_{mg1_{min}}, T_{mg1_{max}}] \\
&w_e \in [w_{e_{min}}, w_{e_{max}}] \\
&w_{mg2} \in [w_{mg2_{min}}, w_{mg2_{max}}] \\
&w_{mg1} \in [w_{mg1_{min}}, w_{mg1_{max}}]
\end{aligned} \tag{4.3.7}$$

where $\dot{m}_f(T_e(k), w_e(k))$ stands for fuel mass rate at each moment which is calculated by equation 4.3.1, μ is the engine start events penalty factor, and $engine_s(k)$ is defiend as the current engine status minus that of last step.

4.3.2 HEV Backward Model

The DP Matlab function introduced in [142] is utilized along with the backward model of Toyota Prius MY10, which is built in the Matlab script, to implement DP as an EMS in this thesis. There is no driver model in backward modeling, and the vehicle speed is known beforehand, which determines the torque required at the wheels. The torque demand propagates to powertrain components via drivetrain. In other words, the imposed drive cycle's power determines how much each powertrain component should supply to meet the demanded power [143]. MGs, battery, engine, and power split device are modeled using the look-up tables and equations similar to the forward-facing model in chapter 3. An overview of the backward vehicle model is shown in Figure 4.2.

To build the backward vehicle model, since the drive cycle information, including velocity and acceleration, are known, the torque and power demand at every time step is obtained by:

$$\begin{aligned} F_{dmd} &= m_{equiv}\, a_{veh} + F_{aero} + F_{roll} + F_{grade} \\ T_{dmd} &= F_{dmd} \cdot r_{wh} \\ P_{dmd} &= F_{dmd} \cdot V_{veh} \end{aligned} \tag{4.3.8}$$

Following by above formula, the torque required at the final drive are given by:

$$T_{in} = \frac{T_{dmd}}{r_d} \cdot \eta_d^{-sgn(P_{dmd})} \tag{4.3.9}$$

The engine torque and speed are determined by control variables discretization. Therefore, the MG2 torque and speed can be calculated by equations 4.3.5 and 4.3.3, respectively. The MG1 torque and speed are specified at each time step through equations 4.3.4 and 4.3.2. Having speed and torque of ICE, MG2, and MG1 calculated, allows us to determine the state variables as well. Battery power is obtained using:

$$P_{batt} = P_{MG1,elec} + P_{MG2,elec} + P_{access} \tag{4.3.10}$$

where the $P_{MG,elec}$ for both MG1 and MG2 can be driven from equation 3.1.4. P_{access} is the constant power of electrical accessories. By having the battery power, the battery SOC can be evaluated by equation 4.3.6. Also, engine mode as the second state variable is specified by the engine speed sign as follows:

$$mode(k) = sgn(w_e(k)) \tag{4.3.11}$$

Finally, the cost function at each time step is calculated according to equation 4.3.7.

4.4 Results

DP is first run backward which outputs the optimal control signals at every grid point. During the forward DP, when the initial state variables are specified, the optimal control path will be obtained utilizing the control map governed by backward DP. In

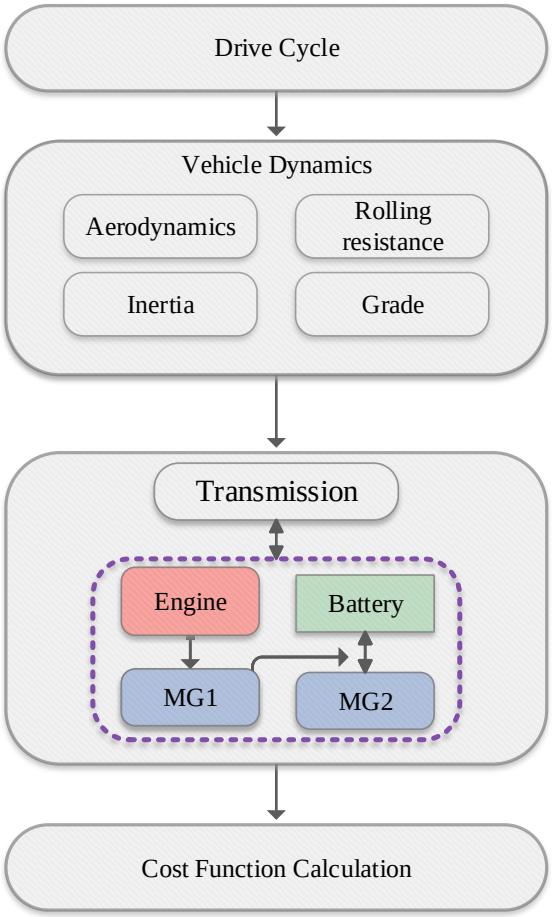


Figure 4.2: Backward-looking vehicle model scheme.

this section, the SOC and engine mode as the state variables are considered as 0.5, and 0, respectively. The state and control variables are discretized as is shown in Table 4.2 for the DP simulation. The simulation time step is is 1 (sec). It should be noted that the higher number of discretized points leads to a higher quality of DP optimal solution while increasing the DP processing time. In this simulation, the number of discrete points for each variable is tuned to make a trade-off between the DP execution time and the quality of optimal results.

The battery SOC, engine mode, engine and MG2 power are presented for UDDS,

Table 4.2: DP variables.

Variable	Min	Max	N
SOC	0.3	0.8	501
Mode	0	1	2
$T_e\ (Nm)$	0	142	31
$w_e\ (rad/s)$	0,104.71	575.96	31

Highway, and US06 drive cycles in Figures 4.3, 4.4, and 4.5. Charge-sustaining can be achieved for all driving cycles which represent city, highway, and aggressive driving conditions.

Table 4.3 compares the fuel consumption results obtained by rule-based and DP. The Decrease value is calculated as:

$$Decrease = \frac{FC_{DP} - FC_{RB}}{FC_{RB}} \times 100 \tag{4.4.1}$$

According to [144], when comparing the fuel consumption of a charge-sustaining HEV, the difference between the initial and final battery SOC value affects the actual fuel consumption. Whether there is an increase in the final SOC value or decrease, the fuel can be saved, or it should be compensated by running the engine-generator set. The fuel consumptions that are reported in the Table 4.3, are the corrected fuel consumption considering the net change in the battery SOC.

As can be seen from Table 4.3, DP outperforms the rule-based strategy for all driving schedules by an average of 26.67 % reduction in fuel consumption. Although the DP can not be implemented for real-time application, the comparison of the DP and rule-based results reveals the need for a real-time implementable EMS with near-global optimal performance.

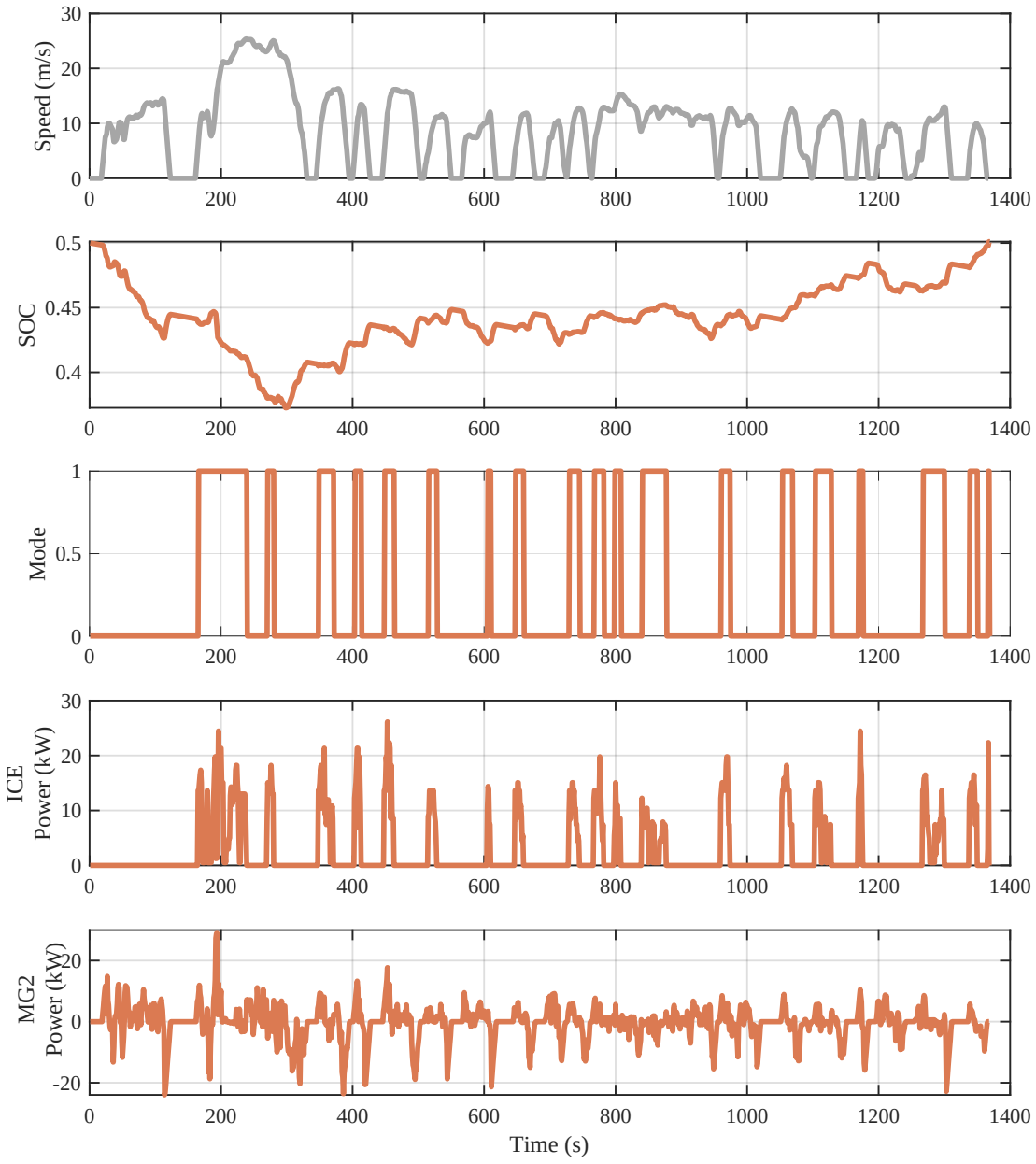


Figure 4.3: DP results of battery SOC, engine mode, engine, and MG2 power for UDDS drive cycle.

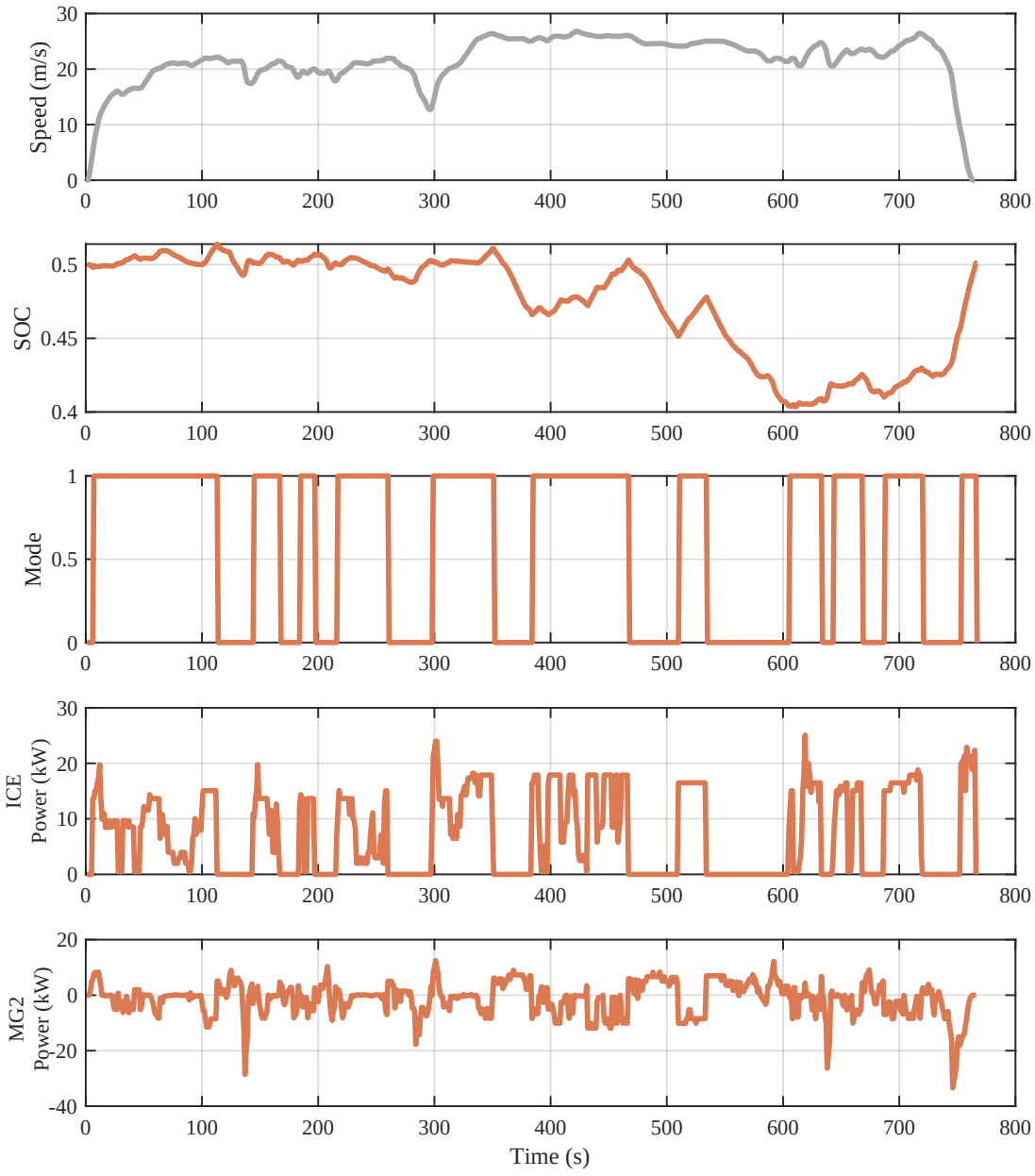


Figure 4.4: DP results of battery SOC, engine mode, engine, and MG2 power for Highway drive cycle.

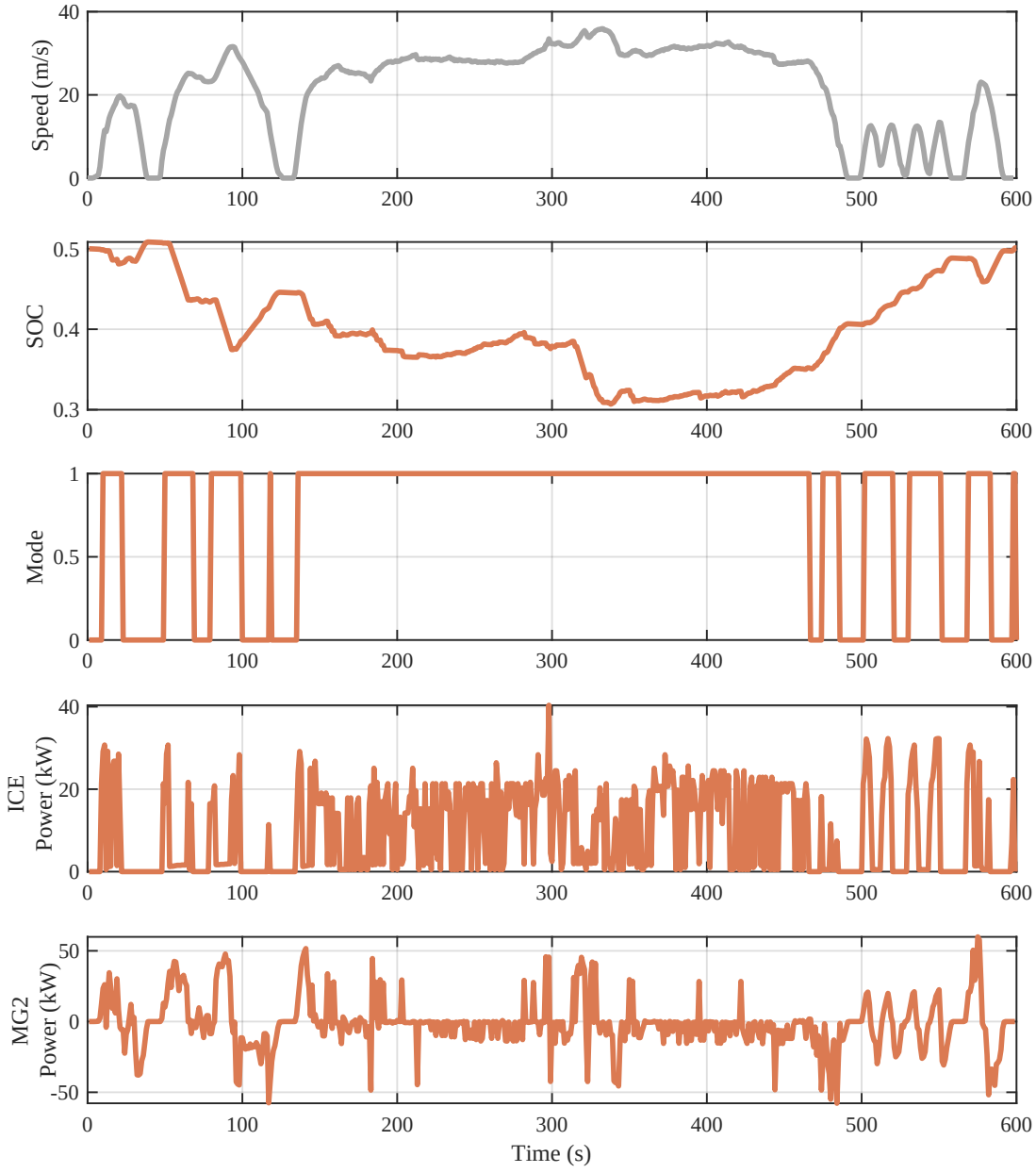


Figure 4.5: DP results of battery SOC, engine mode, engine, and MG2 power for US06 drive cycle.

Table 4.3: Fuel consumption results.

		UDDS	Highway	US06
FC (ml)	Rule-based	440.97	613.81	687.48
	DP	314.4	460.1	506.9
	Decrease (%)	-28.70	-25.05	-26.26

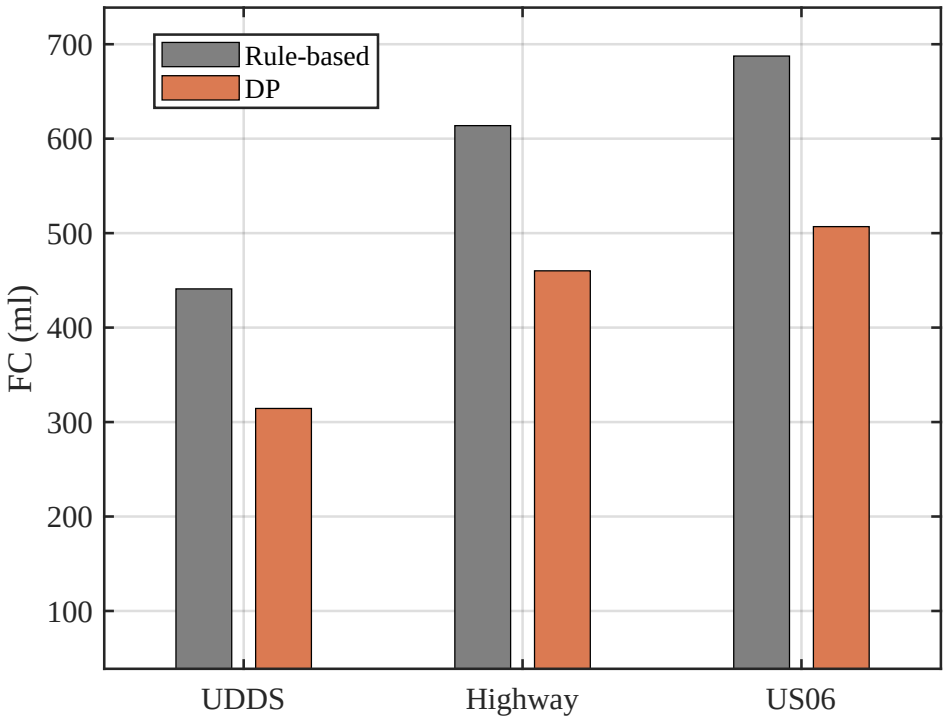


Figure 4.6: Fuel consumption comparison of Rule-based and DP over three drive cycles.

4.5 Summary

This chapter set out to provide an offline global optimization algorithm to solve the EMS problem in our selected HEV. First, the theory of DP as a chosen global optimal method was introduced. Second, we discussed the state and control variables selection, and the EMS optimal problem is formulated. Third, the implementation of DP for our model and the backward vehicle simulation were investigated. Finally, the DP simulation results were shown for three EPA standard drive cycles, including UDDS, Highway, and US06. The results indicate the room for fuel consumption and battery performance improvement in real-time EMSs that are further investigated in the next chapter.

Chapter 5

Development of a Real-time Energy Management Strategy

5.1 Introduction

Considerable number of methods have been utilized to address EMSs. Rule-based methods [145, 146], DP [147, 148], PMP [149], MPC [150, 151], and ECMS [152, 153] are some of established EMSs. Rule-based EMSs have been successful because of their simplicity, real-time capability, and ease of implementation for industrial purposes. Vehicle controllers' data is part of automotive companies confidential documents. Several studies attempt to derive and analyze the EMS supervisory controller of real-world HEVs [154, 155, 156, 157]. Authors in [157] have extracted the rule-based strategy implemented in Toyota Prius 2010, utilizing the testing data provided by ANL. Authors in [155] have conducted a similar study to obtain engine on\off control, which is embedded into Chevrolet volt 2016. Although the rules-based methods are practical and well-suited for commercial applications, they are not able to fully

utilize the HEV powertrain capability. The considerable limitation with heuristic methods is that they do not guarantee to be optimal. Several studies have attempted to improve rule-based results in terms of optimality [158, 159, 160]. Authors in [159] have modified a rule-based strategy with DP optimal control signals that leads to almost 50% improvement in fuel reduction. Authors in [160] have employed GA to optimize the threshold parameters of rule-based EMS. These studies show the potential improvement in the fuel economy of rule-based EMSs. However, integrating offline optimization algorithms into rule-based methods is not practical in real-world applications due to the computational burden and the complexity of methods for HEVs. Besides, even considering the global optimization method results to adjust heuristic rules, the updated strategy does not necessarily guarantee optimal performance for all driving cycles. In addition to optimality concerns of rule-based EMSs, intensive work needs to be done to calibrate and adjust the rules for all driving situations, which is a time-consuming and complex task.

The issues mentioned above with rule-based methods and the computational burden of global optimization algorithms contribute to the use of instantaneous optimization methods for EMSs. Commonly established instantaneous EMSs are ECMS, MPC, extremum seeking [161], and etc. ECMS indicates excellent performance among these methods because it can provide near-global optimal results while meeting the real-time application criteria. ECMS considers minimizing the summation of fuel consumed by ICE and the equivalent fuel of battery electrical energy usage at each instant. The vital parameter in ECMS is the EF, which converts the electrical energy to equivalent fuel consumption. EF value varies for each driving cycle. Authors in [162] have shown the use of optimal EF value for a selected drive cycle leads to the

same performance as DP. Without relying on the drive cycle predictors, it is most likely that the EMS will not have the information about future driving situations in real-world driving [163, 164]. The EF case-sensitivity issue has encouraged studies to conduct EF online adjustment that leads to adaptive ECMS (A-ECMS). The real-time EF adjustment can be done in three common approach including predicted driving information, pattern recognition, and battery SOC feedback [165]. Authors in [166] have used a clustering method to classify the driver's driving style, which is integrated into the ECMS algorithm to adjust the EF value according to each driving style. Authors in [167] have developed a velocity predictor combined with A-ECMS to update EF value using the predicted velocity. Authors in [168] have used an ANN to predict the drive cycle's power demand which is used to predict the optimal SOC reference of a PHEV. In their proposed method, the EF value is updated using a PI controller to minimize the battery SOC difference with its reference trajectory. Several studies have attempted to implement driving pattern recognition methods to categorize drive cycles in order to select the appropriate EF for the A-ECMSs [169, 170]. Battery SOC feedback is mostly used in A-ECMSs to ensure the SOC lies in the pre-defined range. Authors in [171] have formed the EF update function in terms of battery SOC difference with the SOC reference value multiplied by a correction factor which is calculated offline.

Although these aforementioned articles have presented the state of the art EMSs, they still lack in the following aspects from an industrial perspective:

- The vehicle industry has not welcomed the prediction-based methods due to the reliability issue of predicted parameters. If prediction methods are utilized for real-world vehicles, the EMSs should be equipped with fault detection and

diagnostics control strategies to avoid abrupt driving situations.

- A-ECMSs based on pattern recognition and prediction methods are not viable since they require intensive calculation, which restricts their dominant use in the automotive industry.

- PI controllers have good performance at giving the charge sustaining battery SOC trajectory; however, the controller's gain might differ for each drive cycle. Proper tuning of PI controller gain is a time-consuming task and requires manual work. Hence it can not be completely said that PI-based EF update rules are completely independent of human intervention during real-world driving.

An ECMS method is proposed in this study that does not require parameters tuning compared with the rule-based EMS. The method provides near-global optimal fuel consumption results without utilizing predicted driving cycle information. The method is capable of keeping the battery SOC in a predefined interval while respecting drivability concerns to bring comfort for the passengers and driver.

5.2 Situation-specific Equivalent Consumption Minimization Strategy

5.2.1 Basic Equivalent Consumption Minimization Strategy

ECMS is an online instantaneous optimization technique that minimizes a cost function representing an equivalent fuel consumption [172]. In [173], the authors introduced this concept in 1999 as a tool to scale down the global optimization problem

into an instantaneous problem which can be solved online. The cost function is defined as a summation of the fuel amount consumed by the engine and the equivalent fuel consumption of electrical energy consumption. The electrical power is converted to equivalent fuel consumption using the EF such that the total fuel consumption can be minimized [174].

The EF value plays a vital role in determining the near-optimality of the EMS and maintaining the battery's charge sustenance. An EF's value greater than its optimal value tends to recharge the battery more than depleting. In contrast, an EF's value smaller than the optimal value prioritizes depletion of the battery over recharging it.

There are three distinct paths available for achieving charge sustenance in ECMS online simulation or real-time applications. First, an optimal constant value of the EF can be found by the trial-and-error method if the EMS controller knows the complete drive cycle before real-time driving. Second, a PI controller can be employed to update the EF in real-time so that the battery SOC, at any time during the drive cycle, does not deviate much from the reference SOC value. Although the PI controller does not guarantee to finish the drive cycle exactly with the same SOC value as reference SOC, PI controller can finish the drive cycle within a reasonable tolerance of $\pm 2\%$ of battery SOC when the EMS controller does not know anything about the future driving situations. Third, the EF's value can be optimized for small future time-windows based on the predicted driving situations for those time-windows. However, prediction-based ECMS would not be practical due to the reliability and implementation issues, at least for the foreseeable future. Among the methods mentioned above, the PI-based EF update method is the most accredited and easy to implement in the real-time HEV application where the future drive cycle information is unavailable.

5.2.2 Proposed Framework of Equivalent Consumption Minimization Strategy

We have found a framework comprised of simple ECMS and three distinct values of EF for HEV applications and the charge-sustaining operation of PHEVs. As far as the HEV is concerned, due to limited charging opportunities, it is preferred to maintain the SOC trajectory around a predefined SOC value, which may not be the case for every drive cycle but should be achieved at least in the long-term. Moreover, Li-ion batteries offer maximum efficiency within the bracket of 40% to 60% SOC. Therefore, an EMS should be articulated in a way so that it can operate the powertrain of the HEV within the efficient operating zone of the battery. As far as the charge-sustaining mode of a PHEV is concerned, the EMS has much less room to deviate from a typical strict tolerance of $\pm 3 - 4\%$ since the charge-sustaining mode activates typically around $20 - 25\%$.

Working Principle of Situation-specific ECMS

It is assumed that the HEV is supposed to operate around a global reference value of battery SOC. With this assumption, the proposed framework has to ensure the battery's charge sustenance around the global reference value throughout the long-term. In other words, we make the sustaining constraint less strict by a relaxation of the final SOC value, which is not restricted to be equal or very close to the reference SOC value by the end of each drive cycle. However, the controller can retain the battery SOC around the reference value with a small amplitude of about $\pm 15\%$ in the long-term and prevents immoderate depletion or overcharging.

A legitimate question can be raised about the justification of articulating a separate framework of ECMS since similar wobbling characteristics of battery SOC in long-term driving can be easily obtained through PI controller-assisted ECMS or even a simple rule-based control.

It is the perfect time to reveal the last objective of the proposed framework. The framework must yield a fuel-economy performance that is as close as to the global optimal performance yielded by an offline numerical method such as DP. It is noteworthy to mention that neither rule-based control nor PI-based ECMS is articulated targeting global-optimality. The PI-based ECMS is primarily articulated as an instantaneous optimal control strategy with the capability of satisfying reasonable charge sustenance for every drive cycles. The PI-based ECMS exerts secondary emphasis on near-global optimality of fuel-economy. Table 5.1 presents a consolidated and brief comparison among three EMS control strategies, i.e., rule-based, PI-based ECMS, and our proposed ECMS.

Table 5.1: Comparison of control strategies.

Control Strategy	Primary Objectives	Advantages	Disadvantages
Rule-based	a) Long-term charge sustenance. b) Ease of real-time computation.	Computationally simple and swift.	a) Articulation of rules is difficult. b) Does not guarantee on global-optimality. c) Too many tuning parameters.
PI-based ECMS	a) Charge sustenance of every drive cycle. b) Instantaneous optimization.	Reasonable charge sustenance is achieved for every drive cycle.	a) At least three parameters need to be tuned for different drive cycles. b) Does not focus on global-optimality.
Proposed Situation-specific ECMS	a) Long-term charge sustenance. b) Instantaneous optimization. c) Achieving near-global optimality of fuel-economy.	a) Long-term charge sustenance. b) Focus on global optimality for each drive cycle. c) No tuning parameters. d) Simple and effective.	Three generic EF values are chosen for city, highway, and aggressive type drive cycles.

In real-world driving scenarios, there is a possibility of interacting with a countless number of drive cycles. EMS control strategies embedded in real-world HEVs are generally articulated only based on a few federal drive cycles and possibly for a handful of real-world drive cycles. The EMS cannot deploy global optimal methods during real-time driving due to the current limitation of the computational power of microcontroller and insufficient future driving situation's prediction capability. Therefore, most of the Original Equipment Manufacturers (OEMs) concentrate on articulating EMS control strategy utilizing the global optimal results for the aforementioned drive cycles (federal plus a limited number of real-world drive cycles) and deploy the vehicle

on the road with that EMS. There is less chance of obtaining near-global optimal fuel economy performance with that EMS for all real-world drive cycles. The proposed situation-specific ECMS caters to an ingenious and reliable solution to the problem mentioned above. The proposed controller achieves near-global optimal fuel economy while maintaining the battery SOC around a reference value without requiring the drive cycle knowledge in advance.

Our proposed situation-specific ECMS method develops an EF update strategy comprising of three distinct EF values. The EF update strategy will select either of these EF based on comparing the initial battery state and the global reference SOC value. For the given vehicle specification and characteristic maps of the prime-movers, an optimal value of EF can be selected based on the trial-and-error method in a model-in-the-loop simulation for several pre-known standard federal drive cycles. For the proposed framework, also another two values of EF, i.e.,charging -EF and discharging-EF, are computed for which the HEV will finish the drive cycle with higher battery SOC and lower SOC, respectively.

Now, it is time to elucidate the situation-specific ECMS. Suppose, an HEV starts its first real-world drive cycle from an initial SOC same as the global reference SOC value. Once again, it should be reiterated that the objective of the proposed ECMS framework is to maintain charge sustenance in long-term driving. Since the EMS started the drive cycle exactly from global SOC reference, the constant optimal EF value will be used for the entire drive cycle. Once the drive cycle ends, the ending SOC might be higher or lower than the reference SOC value. If the final SOC is lower than the reference value, the ECMS controller will switch to charge mode (charging-EF) until the SOC trajectory hits the reference value. Once the reference SOC value

is obtained, the controller shifts to sustain mode till the end of the trip. If the ending SOC is higher than the reference SOC value, the discharge scenario (discharging-EF) will first happen, and second, the sustaining mode. The controller is defined to compensate for the intense battery charge or discharge and maintain the battery SOC around a reference value. Since the controller updates the EF by comparing the initial SOC with the reference SOC value, the strategy is best suited for delivery trucks or taxi applications. A delivery vehicle driver makes many stops throughout the day between the distribution centers and delivery locations, which allows the controller to update the strategy frequently. Similarly, taxi drivers take multiple destinations to pick up and drop off passengers, so the situation-specific ECMS routinely switches between different EF modes to maintain the battery charge sustenance.

ECMS Optimization Problem Formulation

The ECMS achieves the optimal control operating points of ICE and MGs by minimizing the equivalent fuel power consumption at each instant that is defined as follows:

$$P_{eq}(t) = P_{fuel}(t) + \lambda P_{batt}(t) \tag{5.2.1}$$

where the P_{batt} stands for battery power, P_{fuel} is the ICE fuel power, and λ is the EF. The P_{fuel} is calculated by the following equation:

$$P_{fuel} = \frac{1}{3.6 \times 10^6} \cdot P_e \cdot f_e(n_e, T_e) \cdot LHV_{fuel} \tag{5.2.2}$$

The LHV_{fuel} stands for the fuel lower heating value which is a constant number equals to 34478 Jg^{-1}. P_{batt} is calculated utilizing electric machines' electrical power such as:

$$P_{batt}(t) = P_{MG1,elec}(t) + P_{MG2,elec}(t) + P_{access} \tag{5.2.3}$$

$P_{MG,elec}$ for both MG1 and MG2 can be driven from equation 3.1.4. P_{access} is the constant power of electrical accessories.

To prevent excessive switching between HEV and EV mode, a negative reward will be applied to equation 5.3.1 to reduce the switching frequency. This strategy improves drivability and brings comfort for driver and passengers. Therefore the final cost function at each time instance is defined as :

$$P_{eq}(k) = P_{fuel}(k) + \lambda P_{batt}(k) + R_m(k) \tag{5.2.4}$$

$$R_m(k) = \begin{cases} P & \text{if} \quad mode_{k-1} = mode_k \\ 0 & \text{otherwise} \end{cases} \tag{5.2.5}$$

where the P is a negative constant number.

Figure 5.1 shows the vehicle mode's simulation results with or without applying the penalty to the cost function. As shown in Figure 5.1, the problem of frequent switches of operation modes was solved by the penalty function. Moreover, the switching frequency of the operation modes can be decreased as the value of P increases.

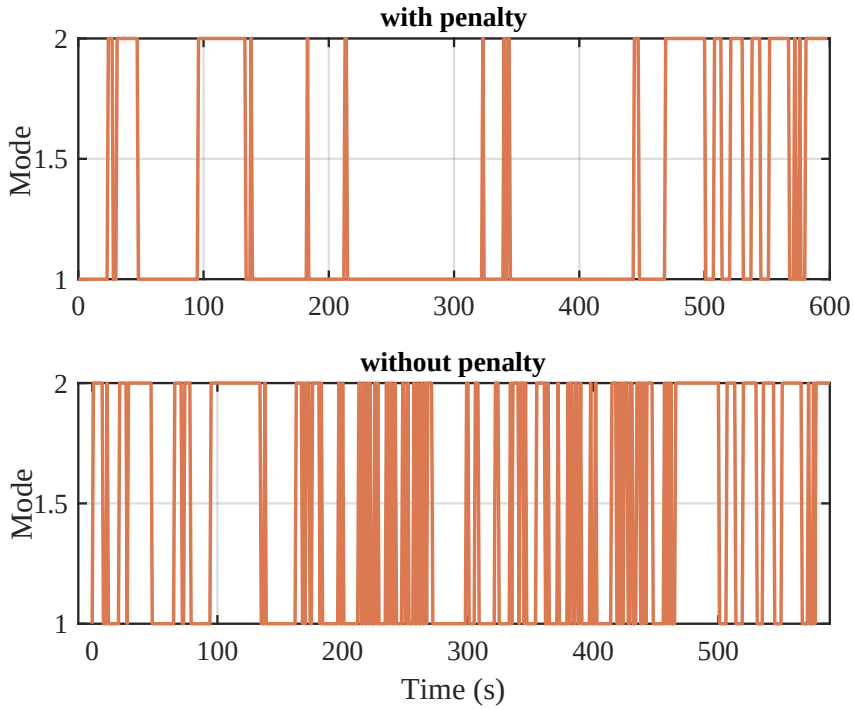


Figure 5.1: Simulation results of operation mode with and without penalizing the cost function for US06. 1 represents the HEV mode, and 2 is for EV mode.

Equivalence Factor Update Strategy

To implement the situation-specific ECMS, three different EF values have been considered that are called $\lambda_{sustain}$, λ_{charge}, and $\lambda_{discharge}$. To find the values for three selected EFs, we run the model with different drive cycles, including UDDS, Highway, US06 City, and US06 Highway. For each drive cycle, the EF optimal value that ensures a charge-sustaining battery SOC trajectory is determined. The iterative process to find the EF optimal value for each cycle is shown in Figure 5.2. While we have prioritized city driving, $\lambda_{sustain}$ is selected among EF optimal values which have been calculated for each cycle. λ_{charge} and $\lambda_{discharge}$ were selected to ensure battery

charging and discharging profiles for all cycles, respectively.

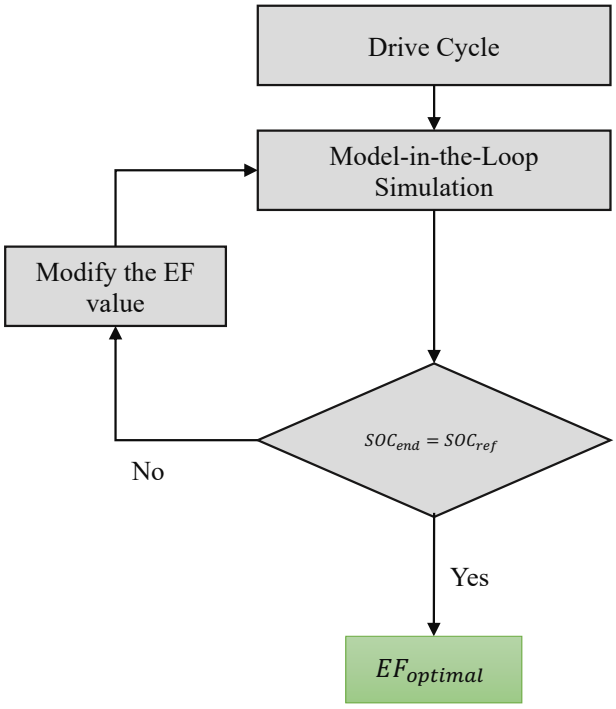


Figure 5.2: Flow chart for EF optimization of each drive cycle.

According to the battery initial SOC the EF update strategy will to switch to one of the pre-determined EFs. There can be three different scenarios:

1. $SOC_{ini} > SOC_{ref}$: To avoid overcharging the battery, the $\lambda_{discharge}$ will be selected to increase the use of electrical energy. The EF value will switch to $\lambda_{sustain}$ when the battery SOC trajectory hits the reference value.

2. $SOC_{ini} < SOC_{ref}$: To prevent battery depletion, the λ_{charge} will be selected to force the battery to be recharged. The EF value will switch to $\lambda_{sustain}$ when the battery SOC trajectory hits the reference value.

3. $SOC_{ini} = SOC_{ref}$: The $\lambda_{sustain}$ will be selected for the entire trip. The EF updating strategy is shown in Figure 5.3.

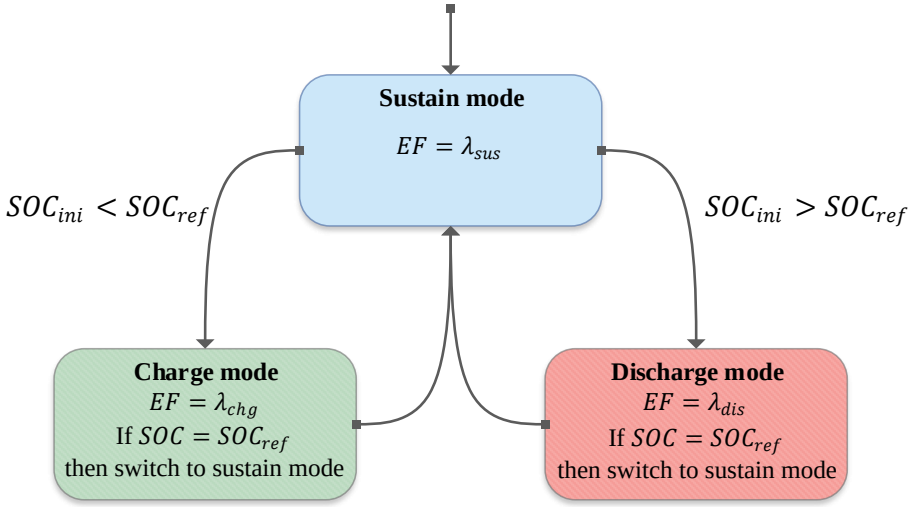


Figure 5.3: The EF update strategy which is implemented in Stateflow.

5.3 Simulink Model Overview

The proposed controller is developed using Simulink. The top layer of the simulated ECMS control system is presented as shown in Figure 5.4, which consists of the HEV Mode Block (HMB), EV Mode Block (EMB), EF update block, and Cost function Minimization Block (CMB). At each time instant, the EF is first updated by the above mentioned proposed ECMS technique. In each mode, a cost function utilizing equation 5.3.1 is calculated. CMB compares those two cost functions provided by EMB and HMB, and it selects the minimum value that determines the vehicle's

operating mode and its associated control signals. A brief working principle of both modes and their subsystems are explained below:

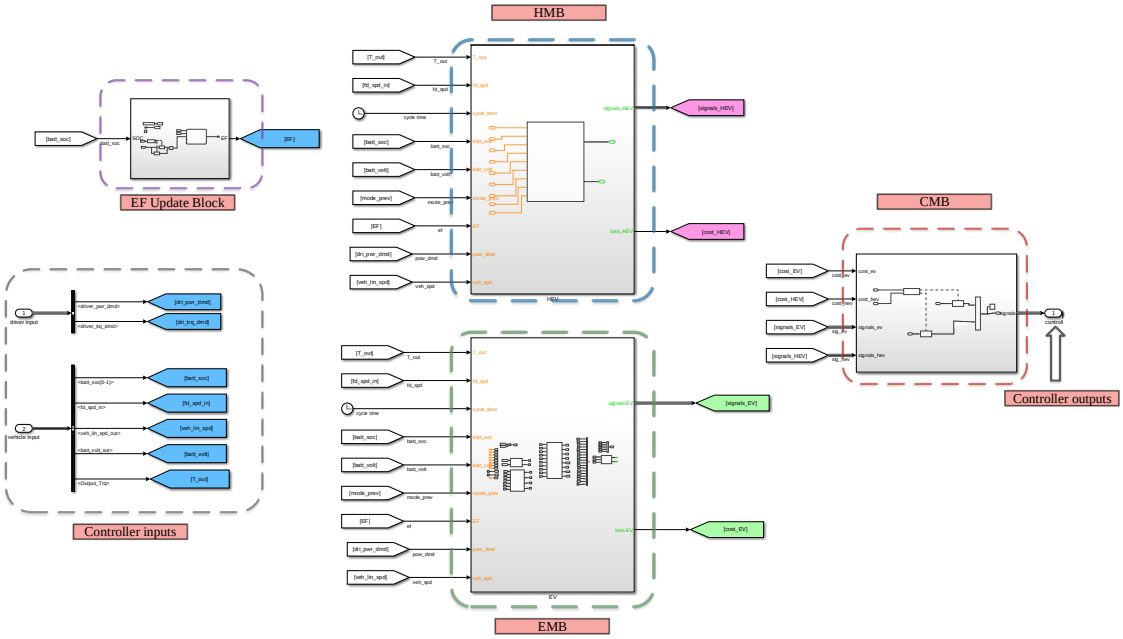


Figure 5.4: An overview of the implemented ECMS controller in Simulink.

5.3.1 HEV Mode

In the HMB, there are two degrees of freedom available in the system: engine torque and speed, which can completely define the system so the rest of the powertrain component parameters can be evaluated. Both engine torque and speed are discretized in their acceptable range. Suppose, at each time instant, the admissible engine speed interval is discretized to N_w number of values. The maximum engine torque is specified for each discretized engine speed value, determining the engine torque admissible

range. The engine speed and torque discretized value are calculated as follows:

$$w_e^i(t) = w_{e,min} + \frac{w_{e,max} - w_{e,min}}{N_w - 1}\,(i-1),\ i = 1, ..., N_w \tag{5.3.1}$$

for each $w_e^i(t)$, there is a maximum allowable engine torque $T_{e,max}$ which specifies the upper value in the admissible engine torque range such as:

$$T_e^i(t) = T_{e,min} + \frac{T_{e,max} - T_{e,min}}{N_\tau - 1}\,(i-1),\ i = 1, ..., N_\tau \tag{5.3.2}$$

where the N_w and N_τ represents the finite number of engine speed and torque, respectively.

Therefore, in the HEV mode, the cost function needs to be measured for the $N_w \times N_t$ combination of operating points in each instant. For each pair of $(w_e^i(t), T_e^i(t))$, the cost function is calculated using equation 5.2.4. During the cost calculation for each of the combinations, individual component limits are considered for penalizing infeasible combinations with a high-cost function value. HMB compares all of these combinations and chooses the minimum cost associated with the current time step as the $cost_{hev}(t)$. This value of cost is stored as the minimum for the HEV mode along with its corresponding control signals.

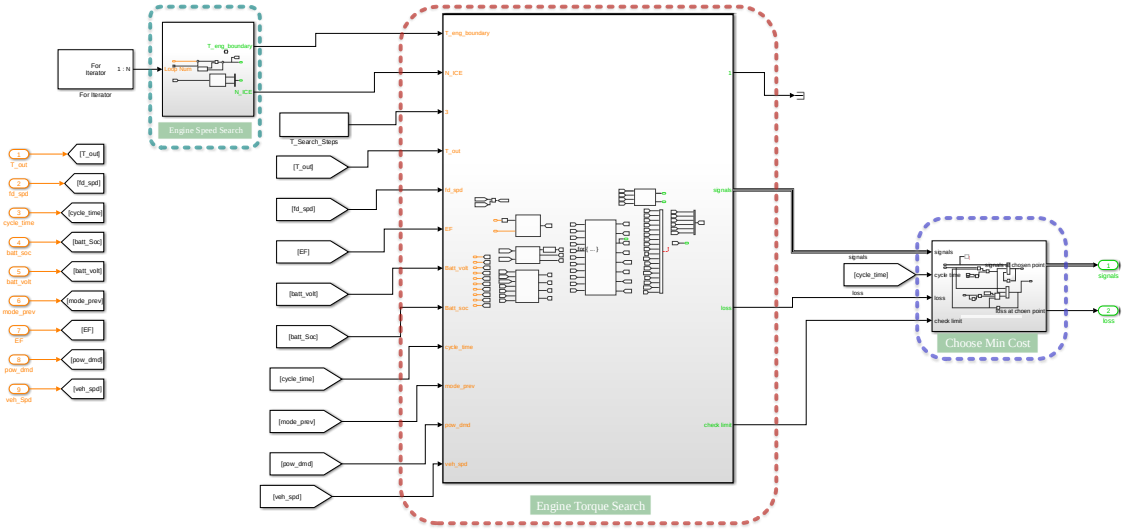


Figure 5.5: HEV mode subsystem.

5.3.2 EV Mode

In the EMB, since the engine is assumed to be off, the engine torque and speed are equal to zero and, MG2 is responsible for running the vehicle solely. There are zero degrees of freedom in the EV mode, and hence all the components need to operate at a certain desired value. Knowing the power demand at each time step, the MG2 operating point is calculated, and the electrical energy consumption is evaluated along with the associated EMB cost ($cost_{ev}(t)$) which is derived by equation 5.2.4.

After the $cost_{ev}(t)$ and $cost_{hev}(t)$ are calculated at instant t, the CMB chooses the minimum value which determines the final operating mode. CMB passes the associated control signals to the powertrain block.

The control variables that lead to an infeasible working condition are penalized by assigning a high cost. Whenever the solutions do not respect the physical limitations of powertrain components or they result in states lie out of their admissible region,,

the cost function would increase by a large cost. There is a check limit block in each mode intended to impose the components' constraints, which are summarized as follows:

$$\begin{aligned} P_{batt} &\in [P_{batt,min}, P_{batt,max}] \\ SOC &\in [SOC_{min}, SOC_{max}] \\ T_e &\in [T_{e_{min}}, T_{e_{max}}] \\ T_{mg2} &\in [T_{mg2_{min}}, T_{mg2_{max}}] \\ T_{mg1} &\in [T_{mg1_{min}}, T_{mg1_{max}}] \\ w_e &\in [w_{e_{min}}, w_{e_{max}}] \\ w_{mg2} &\in [w_{mg2_{min}}, w_{mg2_{max}}] \\ w_{mg1} &\in [w_{mg1_{min}}, w_{mg1_{max}}] \end{aligned} \tag{5.3.3}$$

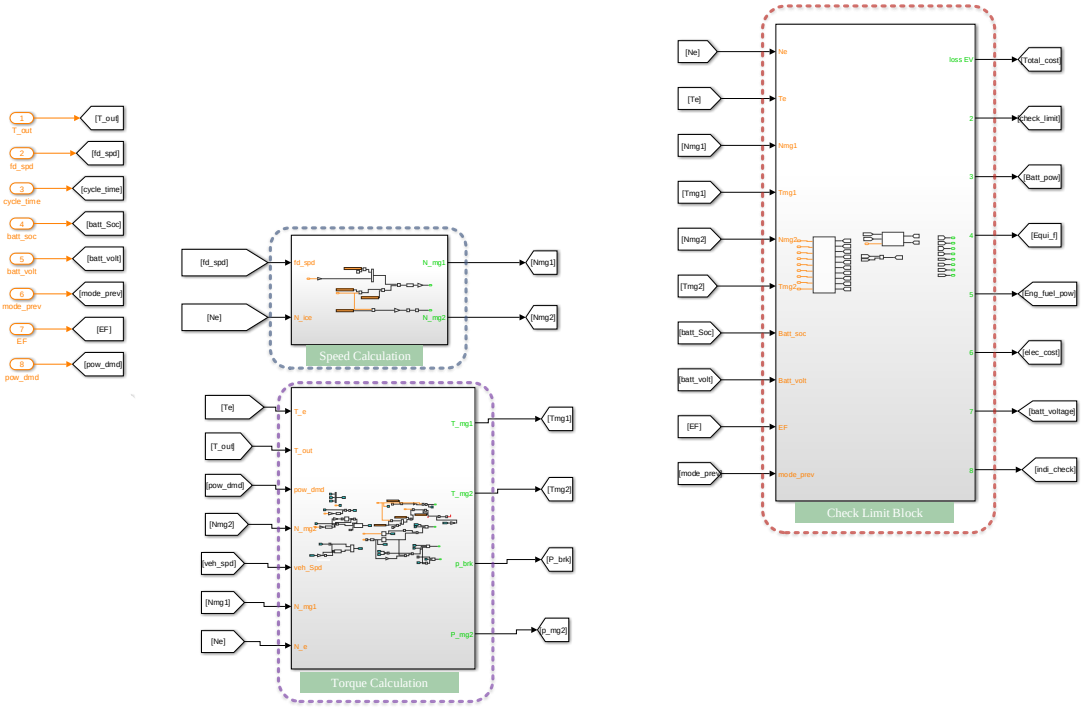


Figure 5.6: EV mode subsystem.

5.4 Results and Discussion

This section analyzes the performance of the proposed situation-specific ECMS on both EPA standard and real-world drive cycles. It is worth mentioning that in the proposed ECMS, the EF is updated whenever the vehicle starts from a fully-stop condition. A fully-stop condition is defined as whenever the vehicle is completely shut off. Having zero speed during the driving due to traffic lights, traffic jams or other possible reasons would not consider as a fully-stop condition. To investigate the effectiveness of the proposed EF update strategy, various combinations of drive cycles are considered in this study. Throughout this section, the reference SOC value

is assumed as 60%.

5.4.1 Simulation Results

To demonstrate the working principle of the situation-specific ECMS, we start by testing the approach on a scenario that consists of two micro trips. Figure 5.7 shows a journey which starts with a city driving cycle and after a fully-stop condition, starts the second trip which is a highway driving cycle. These two drive cycles' data are real-world recorded data provided by the National Renewable Energy Laboratory (NREL). Since the trip starts with an initial SOC equals to the reference SOC value $SOC_{ini} = SOC_{ref} = 60\,\%$, the EF update strategy switches to sustain mode until the end of the trip. Once the city trip ends at t=2690 s, the SOC_{ini} is updated for the next trip, which is highway driving. When the highway portion begins, since the $SOC_{ini} < SOC_{ref}$, the EF update strategy switches to charging mode until the SOC trajectory hits the $SOC = 60\,\%$. From that moment, as it can be seen from Figure 5.7, the EF value updates to $EF = \lambda_{sustain}$. This example perfectly demonstrates the long-term concept in our proposed method. As can be seen, the ending SOC for the first trip is lower than the reference SOC value, however, the strategy charges the battery to the reference SOC value during the next trip. The SOC trajectory which is obtained during the scenario, maintains around $SOC = 60\,\%$. The net change in SOC is 4.04% for this scenario.

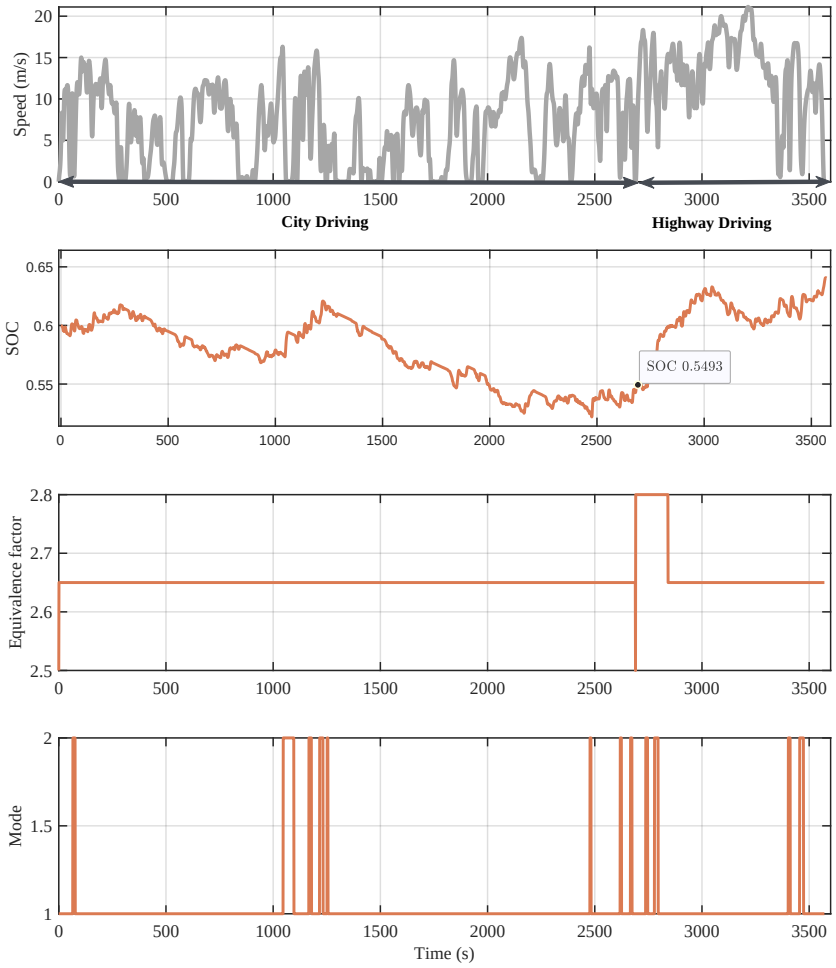


Figure 5.7: Drive cycle profile, battery SOC trajectory, EF value, and vehicle mode are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively.

5.4.2 Real-world Drive Cycles

To illustrate the performance of our proposed ECMS, three real-world drive cycles that are shown in Figure 5.8 are selected to test the charge sustainability and fuel-saving capability of our proposed ECMS. These drive cycles are generated by MCM at McMaster Automotive Resource Center (MARC). The strategy is tested over various

combinations of drive cycles. For instance, CYC 132 consists of real-world cycle 1, real-world cycle 2, and real-world cycle 3 in order, including two fully-stop conditions between cycles 1 and 2, cycles 2 and 3. Three different sequences, including CYC 132, CYC 312, and CYC 321, are considered for simulation. Different orders of these three drive cycles have been tested to simulate the different levels of charge at the beginning of each drive cycle, which further demonstrates our proposed method's robustness. DP and rule-based strategies are also simulated to compare their SOC trajectory and fuel consumption with our introduced method.

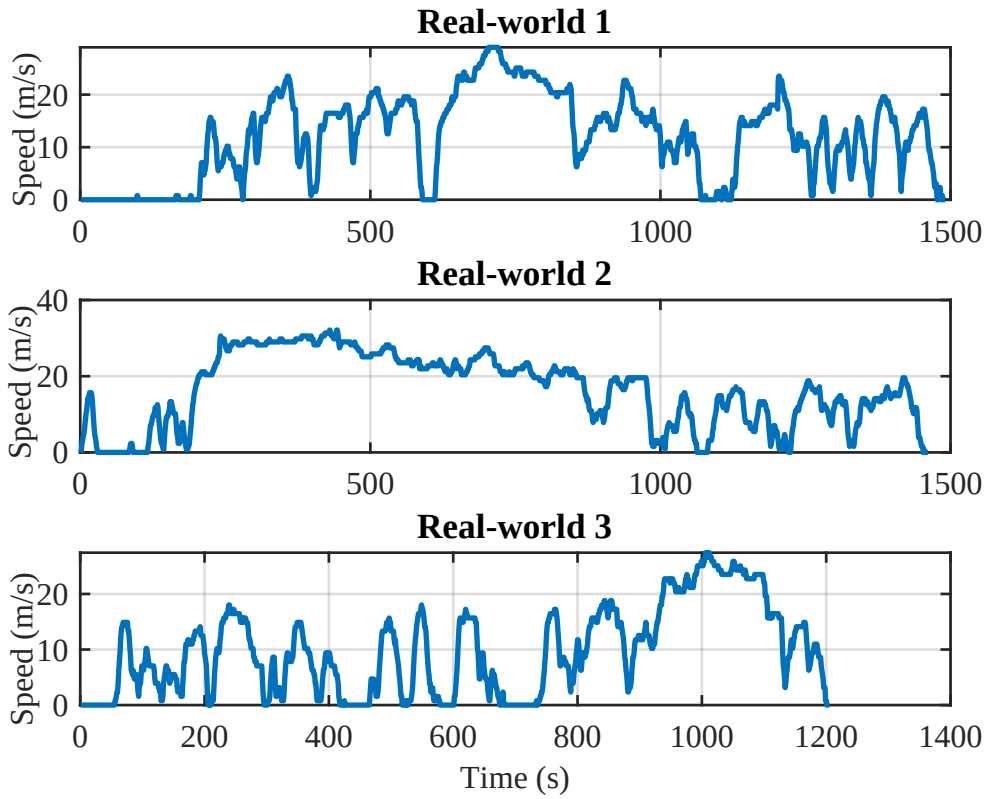


Figure 5.8: Real-world drive cycles.

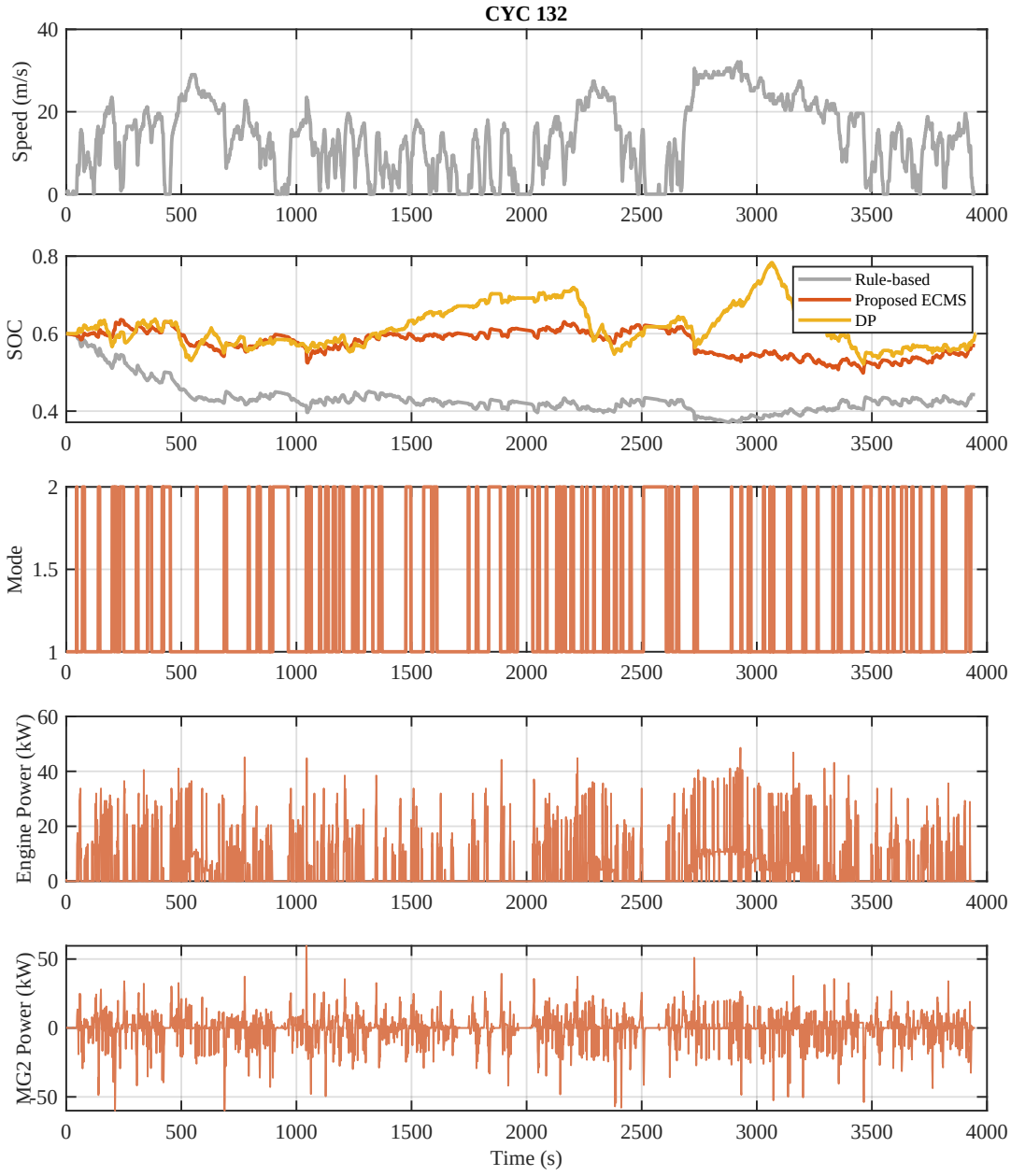


Figure 5.9: Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively.

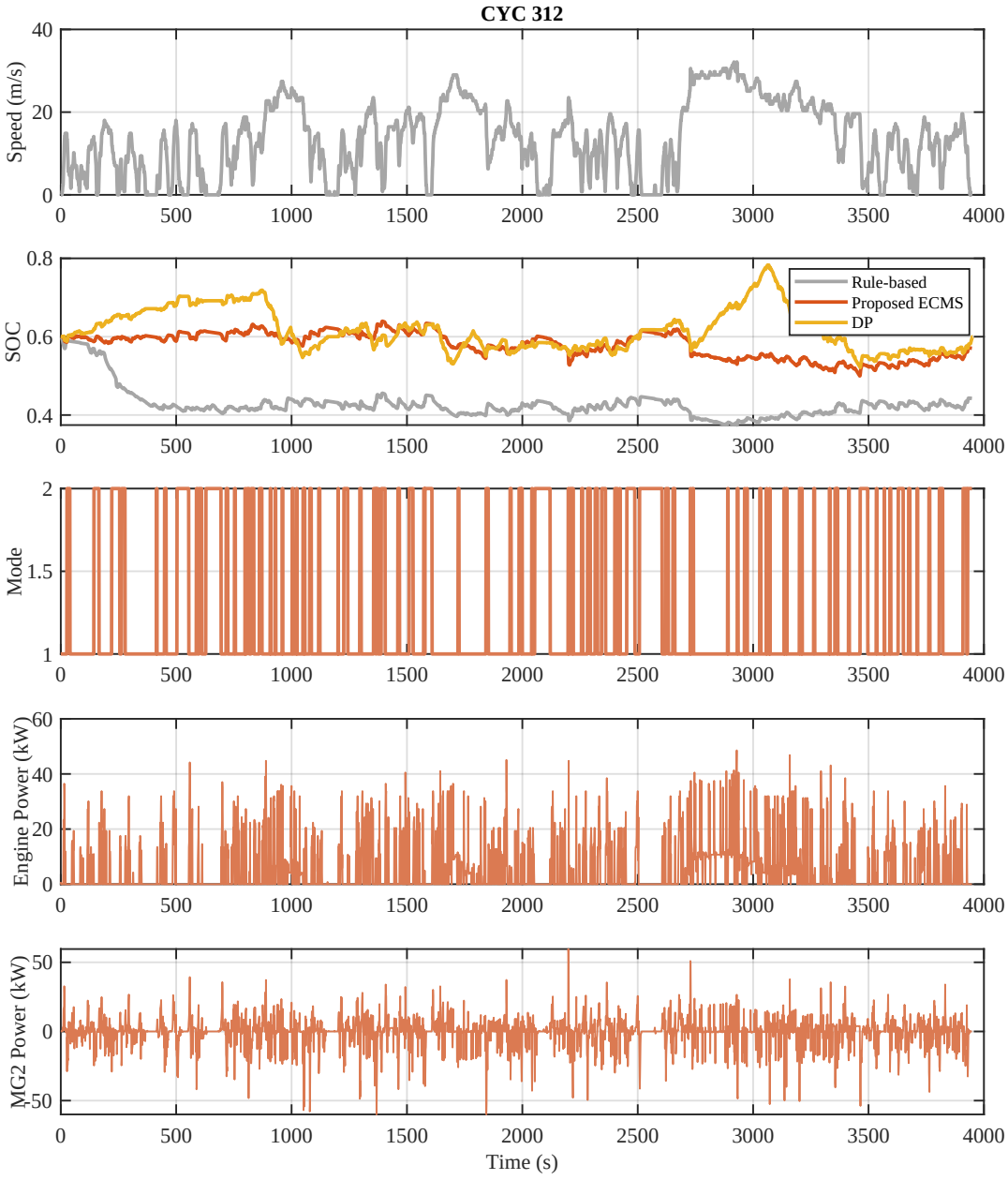


Figure 5.10: Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively.

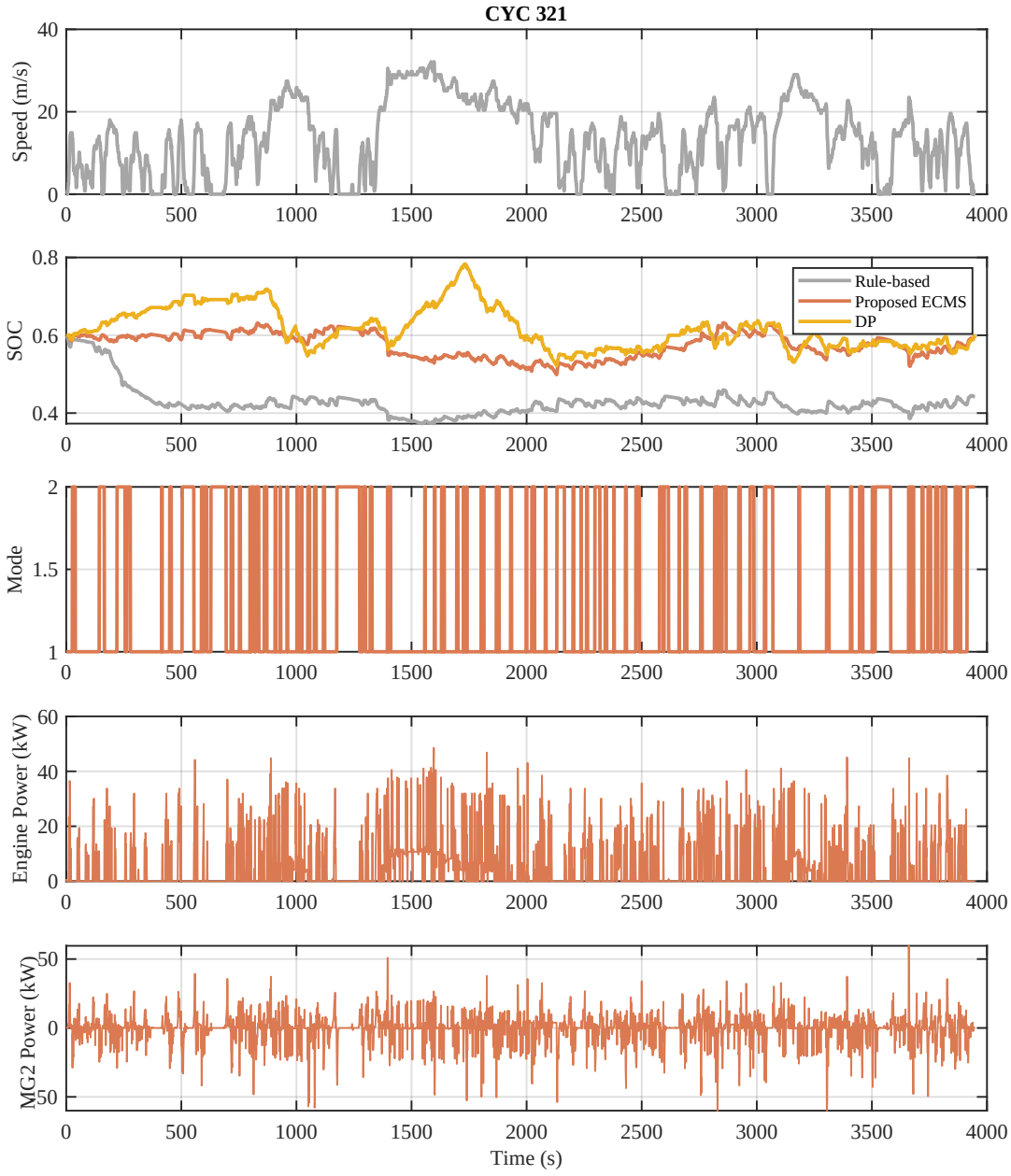


Figure 5.11: Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively.

Figures 5.9, 5.10, and 5.11 depict the simulation results on CYC 132, CYC 312, and CYC 321. Comparing the SOC trajectories, the proposed ECMS tries to bring the SOC value closer to the reference value of 60 % during each journey. Although the SOC trajectory deviates from the reference value during the real-world 2 drive cycle, it tends to recover till the end of the journey. The proposed method also showed a similar increase or decrease as the shape of the DP SOC trajectory except for the beginning of the real-world 2 drive cycle where DP charges the battery while the ECMS discharges the battery. This is because the ECMS is an instantaneous optimization algorithm and does not account for the global drive cycle behavior. The DP tends to charge the battery at the beginning of drive cycle 2 and use the battery when the power is lower by the rest of drive cycle 2. Therefore, this difference in the appearance of SOC profiles can be explained as the local optimization approach's behavior, in which the future drive cycle information is not available.

Regardless of the type of drive cycle, the rule-based strategy depletes the battery to $SOC = 40\,\%$, and following that, it attempted to restrict the battery usage to a tiny window. This behavior is preferred for PHEV applications where the off-board charging opportunity is available. Making the battery to work near the SOC boundary is not desirable for efficient and safe HEV operation. During the beginning of the real-world 2 cycle, the proposed strategy depletes the battery to 50 % to meet the drive cycle's high power. However, the rule-based strategy tries very strongly to maintain around $SOC = 40\,\%$ which limits the HEV performance.

5.4.3 EPA Standard Drive Cycles

We also compared the situation-specific ECMS with DP and rule-based methods over EPA standard drive cycles, including UDDS, Highway (HWFET), and US06. To assess the EF updated strategy, the different sequences of EPA cycles are explored. Figures 5.12, 5.14, and 5.13 depict the speed profile, SOC trajectories, vehicle mode, engine, and MG2 power over three different journeys.

Results on SOC comparison for all three cases have shown that the proposed ECMS strategy keeps the battery SOC trajectory in the expected interval and close to the reference SOC value. Consider the ECMS performance for the US06 drive cycle in "UDDS-Highway-US06" and "US06-Highway-UDDS". In the "UDDS-Highway-US06" case, the SOC starts with a value around 46 % and it ends at 66 %, therefore the controller was able to charge the battery gradually to keep the SOC close to the reference value. In contrast, in the "US06-Highway-UDDS" case, since the $SOC_{ini} = 60\,\%$, the controller tried to keep the battery SOC close to the reference value and the SOC ends at 60.59 %. Therefore, the controller was able to efficiently update the EF value. The proposed method also showed a similar increase or decrease as the shape of the DP SOC trajectory with a small difference in the SOC amplitude. The proposed ECMS has no a-priori knowledge of the driving cycle. Despite this, it maintains the SOC in the acceptable range and remains close to the optimal solution. In comparison to the real-world drive cycles case, the rule-based strategy uses the battery more, even in the UDDS drive cycle it depletes the battery to near 30%. Therefore, rule-based strategy leads to a large net change in SOC which is not preferable for HEV application.

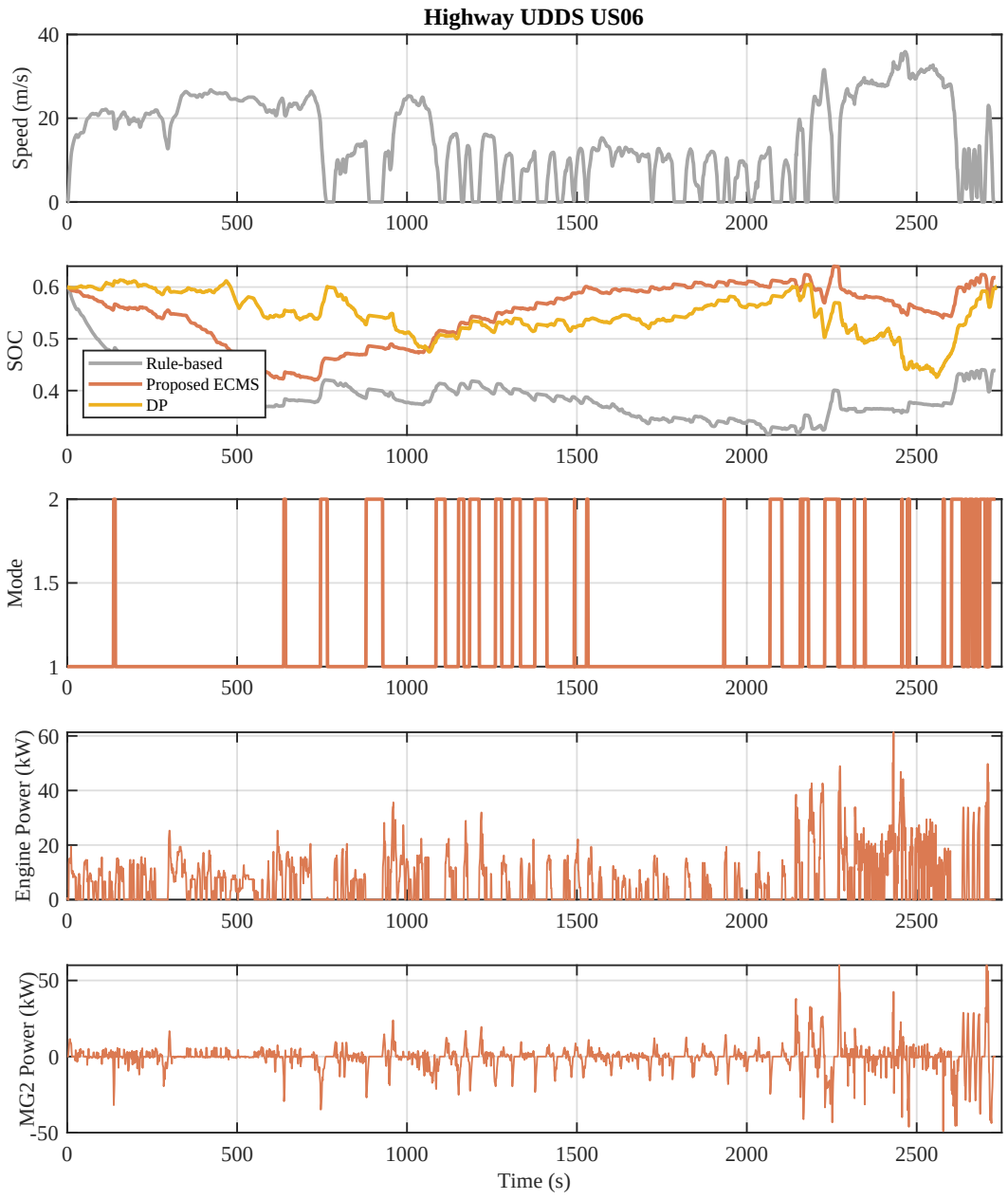


Figure 5.12: Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively.

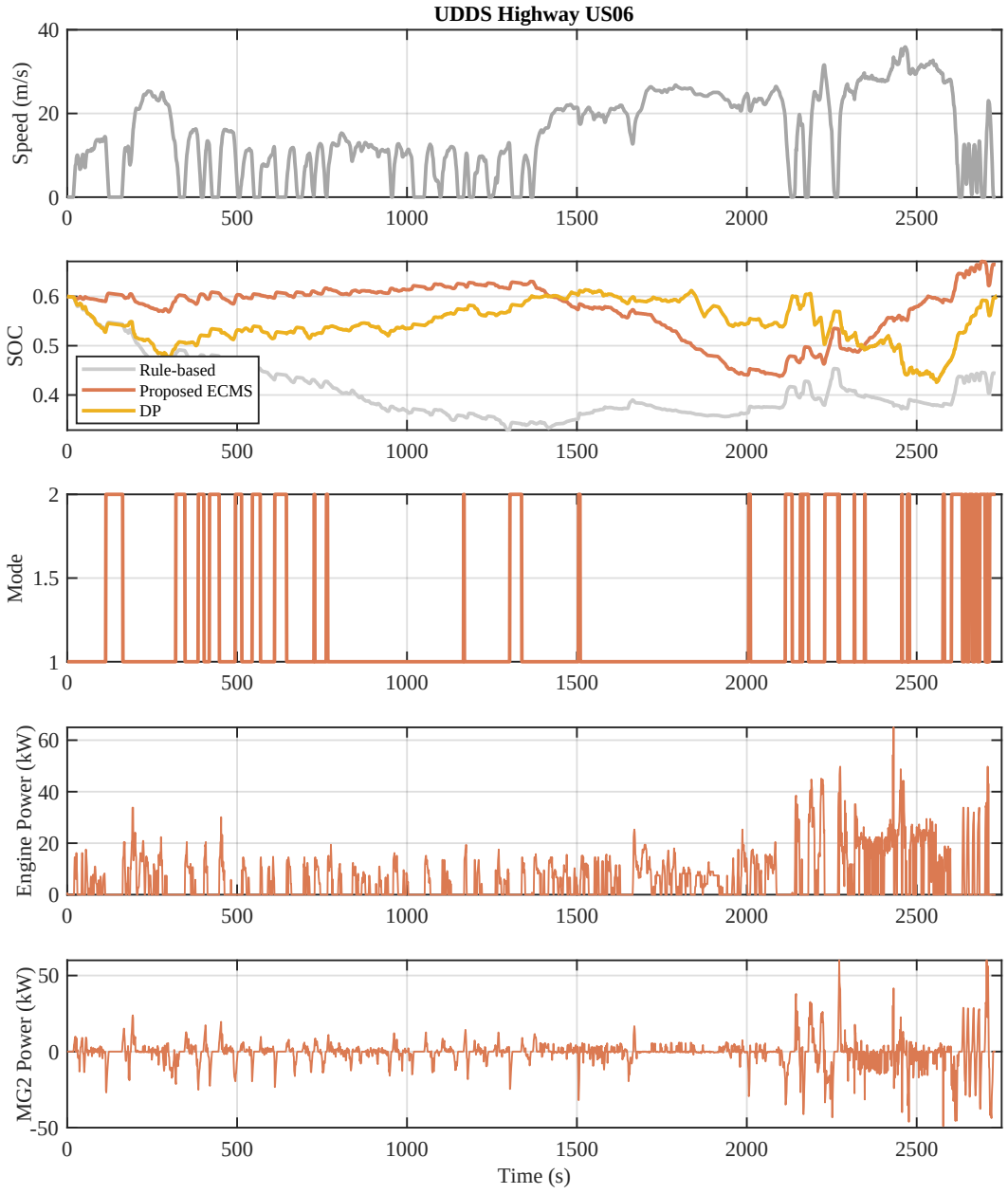


Figure 5.13: Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively.

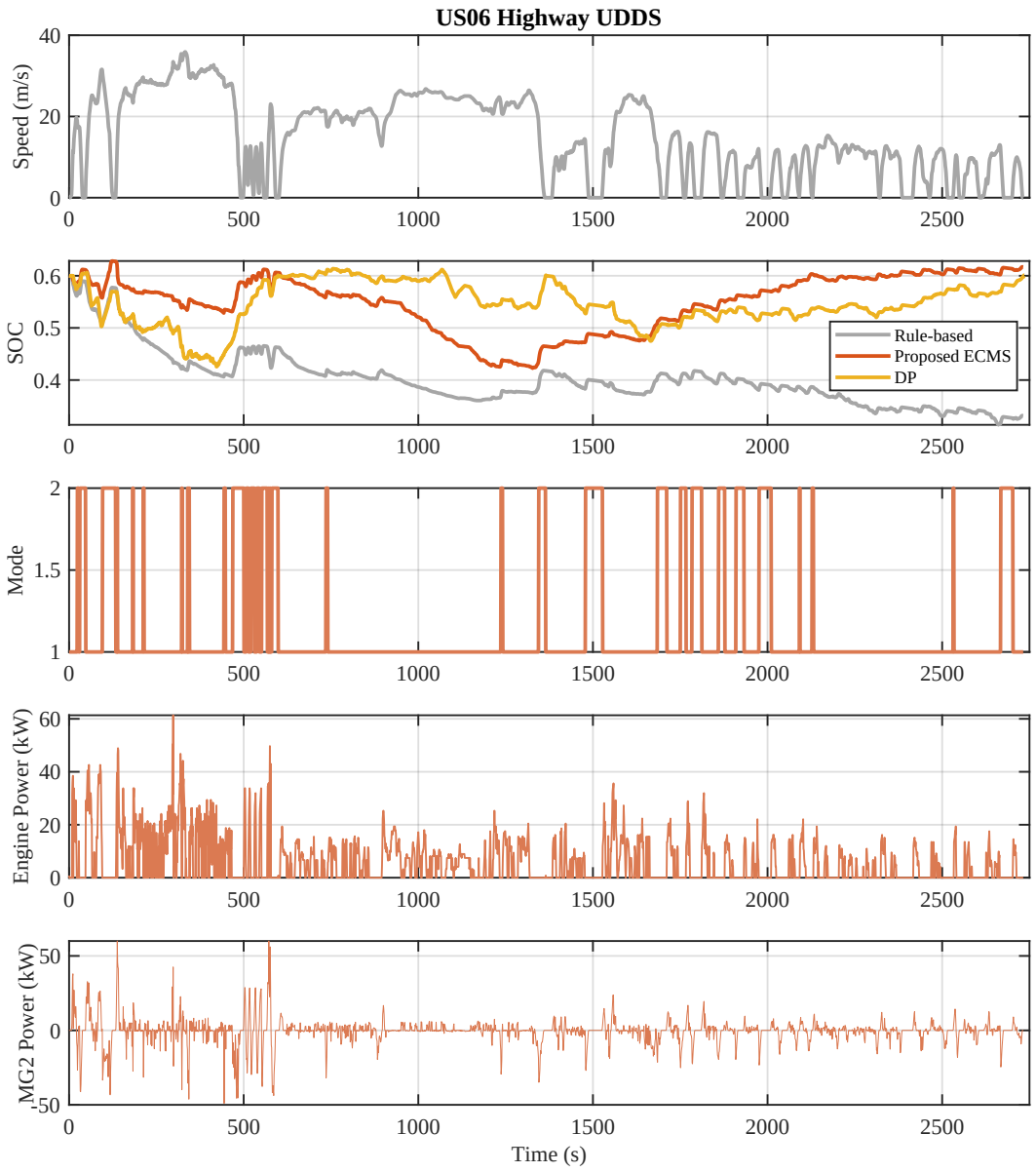


Figure 5.14: Drive cycle profile, battery SOC trajectory, vehicle mode, engine, and MG2 power are depicted. In the mode plot, 2 and 1 stand for EV and HEV mode, respectively.

Long-Duration Drive Cycles

The performance of situation-specific ECMS on both real-world and EPA standard drive cycles have been evaluated. The method is able to maintain the battery SOC trajectory in the neighborhood of global reference SOC value while ensuring a near-global optimal performance. As a future work, one possible extension of this method is to prevent the battery operation near the SOC boundaries in a long-duration drive cycle. It can be done in several ways. One feasible solution is to monitor the battery SOC variation during the sustain mode and if it hits the SOC boundaries, we can switch the EF mode to either charge or discharge mode. The proposed switching algorithm in sustain mode is shown below:

```
if        SOC_min < SOC < SOC_max
        Remain in Sustain Mode
elseif    SOC_max ≤ SOC
        Switch to Discharge Mode
else
        Switch to Charge Mode
end
```

Figure 5.15: Proposed algorithm to prevent the battery from working near battery SOC boundaries.

5.4.4 Fuel Consumption Comparison

Fuel consumption results are presented in Table 5.2 over the six journeys that have been shown in previous parts. All values are corrected for final SOC variations. Diff. 1 and Diff. 2 are calculated using the following equations:

$$\begin{aligned} Diff.\,1 &= \frac{FC_{ECMS} - FC_{DP}}{FC_{DP}} \times 100 \\ Diff.\,2 &= \frac{FC_{ECMS} - FC_{RB}}{FC_{RB}} \times 100 \end{aligned} \tag{5.4.1}$$

It can be seen from the Table that the proposed strategy outperforms the rule-based method by improving the fuel economy by approximately 20%. This is because the rule-based methods are developed based on heuristics and the concept of efficient ICE operating regions. In contrast, in our proposed method, fuel consumption is minimized utilizing the mathematical model for fuel consumption. The proposed ECMS method achieves a reasonably good performance in comparison with DP results. The fuel consumption of the ECMS method is higher than DP by an average of 8.10 %. The near-optimal performance for the ECMS method is achieved while the EMS does not have access to the full knowledge of drive cycle information, and the method is real-time implementable.

Table 5.2: Fuel consumption results.

		CYC 312	CYC 321	CYC 132	UDDS-Highway-US06	US06-Highway-UDDS	Highway-UDDS-US06
FC (ml)	DP	1674.40	1674.40	1674.40	1316.19	1316.19	1316.19
	Rule-based	2530.46	2530.92	2184.68	1673.58	1682.92	1669.91
	Proposed ECMS	1861.63	1853.2	1861.52	1373.04	1387.4	1387.8
	Diff. 1 (%)	11.18	10.6	11.17	4.31	5.41	5.42
	Diff. 2 (%)	-26.43	-26.77	-14.79	-17.95	-17.55	-16.8
Final SOC (%)	DP	60	60	60	60	60	60
	Rule-based	44.23	44.22	44.21	44.41	33.15	43.87
	Proposed ECMS	57.04	59	56.89	66.47	61.69	61.8

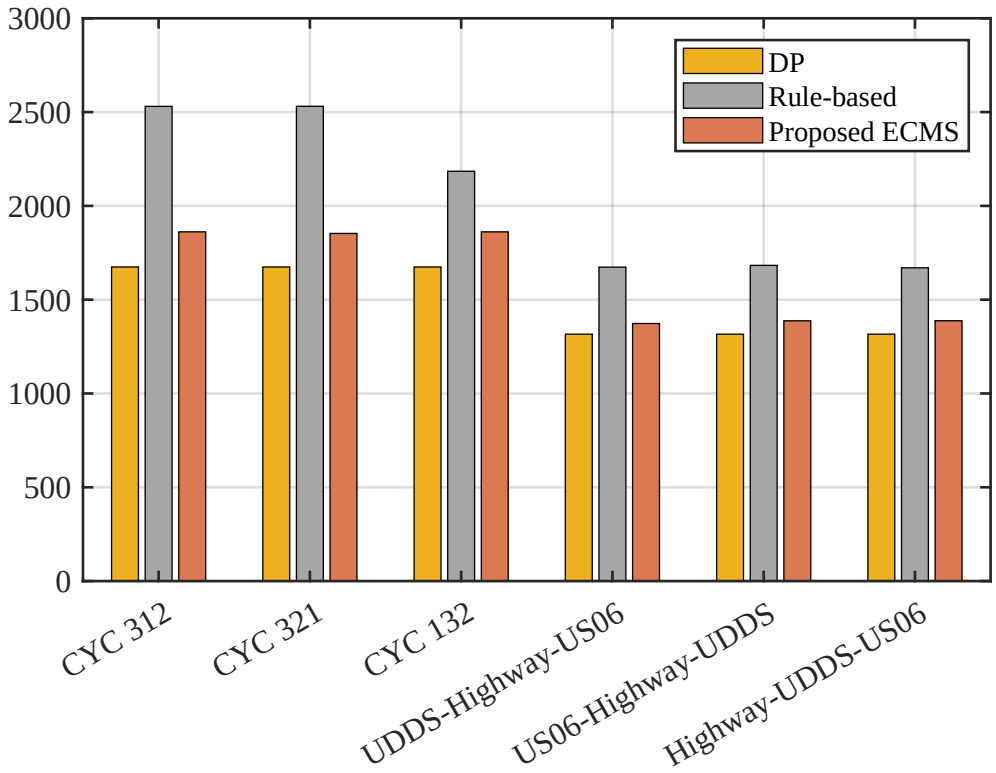


Figure 5.16: Results of fuel consumption, obtained by DP, rule-based, and proposed ECMS approaches for real-world and EPA drive cycles.

5.4.5 Comparison of Proposed ECMS with PI-based ECMS

To show the advantages of our proposed situation-specific ECMS, PI-based ECMS is introduced here. This is a classical method for the EF adaptation law in which the EF is adjusted to ensure that the actual SOC remains around the reference value. PI-based ECMS can be a suitable approach to evaluate the balance between the battery charge sustenance and fuel economy. The EF is updated as follows:

$$\lambda(t) = \lambda_0 + K_p \left(SOC_{ref} - SOC(t)\right) + K_i \int_0^t (SOC_{ref} - SOC(t))dt \tag{5.4.2}$$

where $\lambda(t)$ is the EF, λ_0 is the initial EF value, K_p, and K_i are the proportional and integral gains.

Figure 5.17 compares the SOC trajectories which are obtained by situation-specific ECMS and PI-based ECMS. As it can be seen, the SOC trajectory in PI-based ECMS follows the reference SOC perfectly while the situation-specific ECMS keeps the battery SOC around the reference value in a larger SOC range. However, given that the PI-based ECMS uses a narrower SOC range than does the situation-specific ECMS, it does not necessarily mean a superior solution because, in the absence of future information, the narrow usage of SOC range could limit hybrid performance. On the other hand, PI-based ECMS increases fuel consumption because it imposes an additional constraint, which leads to instantaneously compensate the SOC deviations at the price of the near-global minimization of the fuel consumption. Table 5.3 compares the fuel consumption results of PI-based and situation-specific ECMSs. The PI-based ECMS consumes 13% more fuel on average comparing to DP results. In contrast, this

number is reduced to 10.7% for situation-specific ECMS. Take the CYC 123 scenario for example, the situation-specific ECMS consumes 35 ml (25.93 g) fuel lesser than PI-based ECMS.

Table 5.3: Comparison of fuel consumption results between PI-based ECMS and proposed ECMS.

Method	CYC 213	CYC 231	CYC 123
PI-based ECMS	1891.3	1895.4	1890.5
Proposed ECMS	1854.89	1858.7	1855.5
Diff. (ml)	36.41	36.7	35

The implemented PI-based ECMS is a representative example of the ECMS method in which EF value is updated through a PI controller formulation. By properly tuning the proportional and integral gain along with a near-optimal selection of EF value for λ_0, fuel consumption can be minimized close to the global optimization results. Authors in [133] have discussed the effect of proportional gain variation on the battery SOC trajectory and fuel consumption. Decreasing the K_p value would result in slightly better fuel consumption while it increases battery SOC excursion. Although the choice of K_p and K_i affects the battery SOC window size, the situation-specific ECMS allows more ample SOC variation than the PI-based due to the less frequent correction of EF.

At least three parameters need to be tuned for a PI-based controller, which includes an open-loop adjustment of λ_0, which is drive cycle-dependent, and the PI

controller's gain, which should be tuned to manage the SOC excursions for all driving cycles. Authors in [175] have shown how the performance of PI-based changes by any slight modifications in the value of λ_0. In contrast, to determine the EF values ($\lambda_{sus}, \lambda_{chg}, \lambda_{dis}$) in situation-specific ECMS, first we need to select several pre-defined drive cycles and for each drive cycle one value that is EF optimal, will be obtained. Among those EF optimal values, one can ensure a near charge-sustaining operation for all pre-defined drive cycles will be selected as λ_{sus}. λ_{chg} and λ_{dis} will be selected based on the fact that any values above the EF optimal value will result in a charging SOC profile and any value below the EF optimal will deplete the battery. Therefore, the implementation of situation-specific ECMS requires significantly fewer tuning efforts.

Overall, it appears that PI-based ECMS can achieve near-global fuel consumption by properly tuning the gains for each dive cycle, and it can manage the battery SOC range in a proper window. On the other hand, the situation-specific ECMS allows the battery to be used more freely while ensuring a near-global optimal performance with lesser tuning effort than PI-based ECMS.

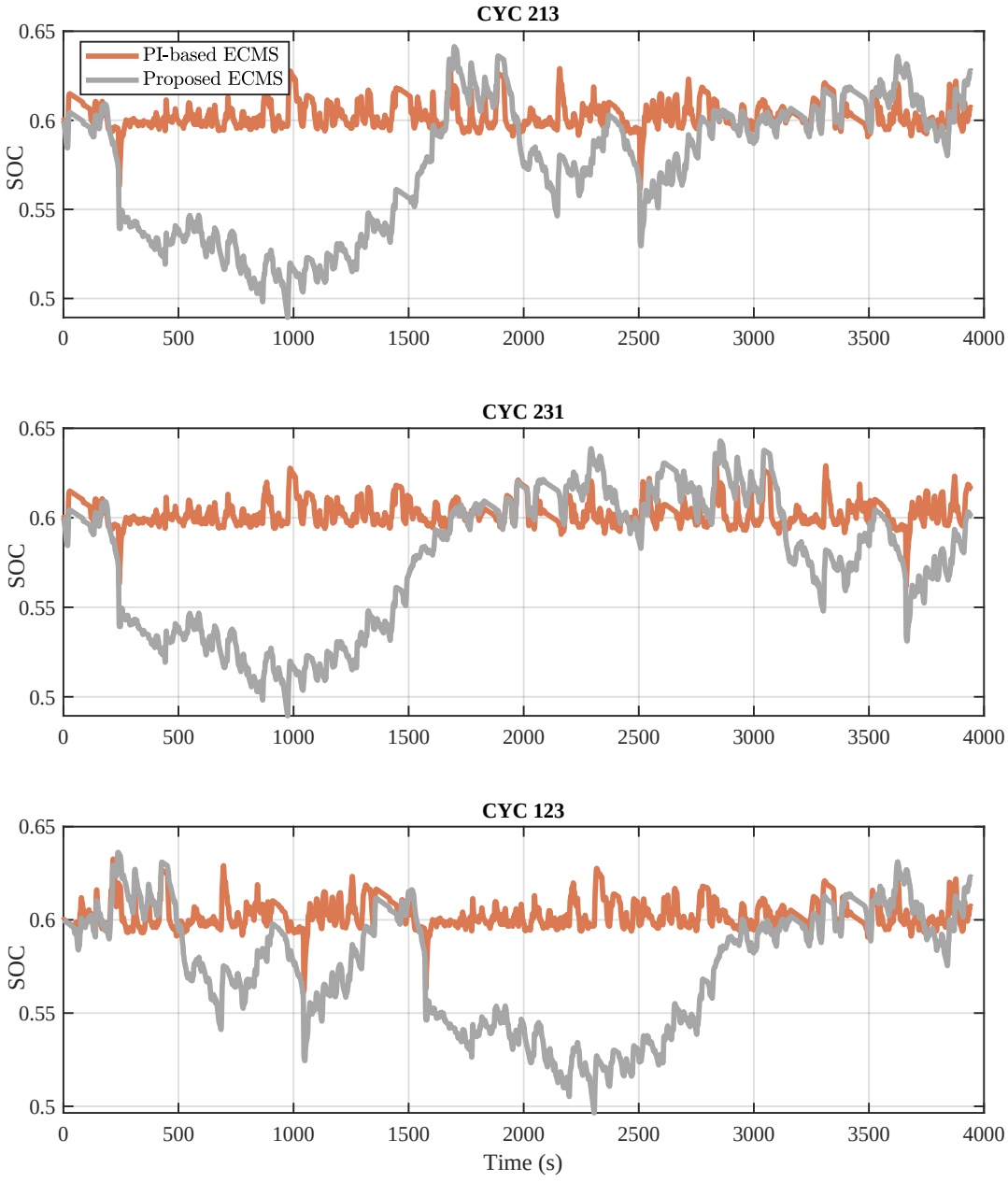


Figure 5.17: Battery SOC trajectory comparison of PI-based ECMS and Proposed ECMS.

5.5 Summary

We proposed a novel framework to address a near-optimal EMS for a power-split HEV. The suggested methodology employed a simple and effective ECMS algorithm to address the following important features:

1. The strategy can satisfy the battery's charge sustenance in the long-term.

2. The strategy does not require any predicted or future driving information, which enables its real-time implementation.

3. The strategy provides close performance to a global optimal EMS (DP).

First, we discussed the existing EMSs, and we elaborated on their limitations, which motivates us to provide the suggested ECMS algorithm. An overview of the ECMS algorithm and its working principle is presented. Next, the proposed framework, along with a mathematical formulation is provided. The situation-specific ECMS is designed to update the EF value utilizing the difference between the battery initial SOC and the reference SOC value. Therefore, the strategy is able to frequently update the EF value throughout the situations consisting of multiple stops where the battery's initial SOC is changed after each stop. The proposed controller is well-suited for the delivery and taxi applications that feature routes with multiple stops throughout the day, so the controller regularly updates the EF value. Finally, the results on both EPA standard drive cycles and real-world driving schedules are presented. Results show that the situation-specific ECMS outperforms the rule-based strategy developed in real Toyota Prius by an average of 20% in fuel consumption reduction. Although, the situation-specific ECMS generates slight suboptimal results by 8.10 % on average

compared to the DP, the controller does not have any knowledge about the drive cycles before the simulation stage and the method is real-time implementable.

Chapter 6

Conclusions and Future Work

Hybrid propulsion systems have been successful in addressing low tailpipe emissions and high fuel economy. EMSs are essential to fully exploit hybrid car's capabilities. This thesis develops an implementable solution in real-world HEVs platform. We considered the automotive industry's acceptance of integrating an effective EMS in addressing battery sustainability and fuel consumption reduction.

In this chapter, we discuss the main conclusions of our work. We also present the potential improvements in the form of future work.

6.0.1 Conclusions

First, we provided a comprehensive review of EMSs by proposing a new categorization of existing EMSs. The review highlights the need for an effective EMS to be embedded in electrified vehicles and how different methods can be employed to solve the EMS problem. The E/E architecture of electrified vehicles is also described, along with the brief introduction of ECUs and their communication functions within

the E/E architecture. The primary enablers of the technologies for the deployment of EMSs in electrified vehicles, including embedded controller chips, communication protocols, and connectivity functions, are discussed. Finally, the review reflects both theoretical and experimental challenges of EMSs for electrified vehicles. The material contained in chapter 2 is published: R. Ostadian, J. Ramoul, A. Biswas and A. Emadi, "Intelligent Energy Management Systems for Electrified Vehicles: Current Status, Challenges, and Emerging Trends," in IEEE Open Journal of Vehicular Technology, vol. 1, pp. 279-295, 2020, doi: 10.1109/OJVT.2020.3018146.

Second, a Simulink model of Toyota Prius MY10, as our case study, was developed. The established vehicle model allows us to study the effect of different control strategies. The model includes three primary parts: driver, EMS supervisory controller, and vehicle powertrain. We provided detailed explanations for modeling different sections such as power split, engine, MGs, and battery. The baseline control strategy implemented in real Toyota Prius MY10 is simulated utilizing the testing data provided by ANL and online sources. Finally, the vehicle model is validated using a comparison of the simulation results with experimental data, showing a good agreement. For instance, all simulated fuel consumption are within 4% of the physically measured data.

Third, DP, as a global optimization EMS algorithm, was implemented for our case study. A detailed explanation for applying DP to the vehicle model was presented. A backward-facing vehicle model was built in MATLAB script to be integrated into the DP optimization algorithm. We focused on control and state variables' selection to reduce the computational effort of DP. Finally, the results achieved by DP were

compared to the rule-based strategy previously implemented in chapter 3. The results revealed that DP outperforms the rule-based strategy by an average of 26.57% reduction in fuel consumption. Although DP is an offline EMS method that needs complete knowledge of the drive cycle beforehand, it is a necessary step to develop real-time implementable EMSs.

Fourth, we have established a novel framework to address battery charge sustainability in HEVs while maximizing the fuel economy. The proposed method is based on the ECMS algorithm, which provides near-global optimal results. In contrast to DP, the ECMS algorithm does not require a-priori knowledge of the drive cycle and is real-time implementable. The proposed ECMS controller can maintain the battery SOC in the neighborhood of reference SOC value in the long-term. In other words, the framework ensures that the battery SOC oscillates around the reference value with a small amplitude throughout the long-term usage of an HEV. To have a credible comparison, the proposed ECMS performance was compared with rule-based and DP. The results indicate the effectiveness of the proposed ECMS by improving the fuel economy by 20% compare to the rule-based method. The results also exhibit a close performance to DP. The fuel consumption of the ECMS method is higher than DP by an average of 8.10 %, which is a reasonably good performance achieved by the ECMS method without access to the full knowledge of drive cycle information. Consequently, we achieved a simple and effective EMS that is tractable for real-time implementation while ensuring battery charge sustenance in the long-term and secures near-global optimal performance.

6.0.2 Future Work

This thesis demonstrates an energy management controller's performance for improving total energy cost and battery charge sustainability. However, further research is needed to expand this study. This section provides some recommended future work for each part of this thesis:

- **High-fidelity Model**

  To enhance the developed HEV model's accuracy, the model should account for a thermal battery and ICE model as the two main components of the powertrain. Considering the thermal effects of these components helps to monitor the EMS controller's behavior under diffrent thermal conditions in real-world situations that can further improve EMS performance. A dynamic model of power split device can also be explored that accounts for the power losses due to friction between gears and the acceleration and inertia effect.

- **Multi-objective EMS**

  Multi-objective energy management optimization can be addressed in future work. This thesis mainly focuses on fuel economy improvement. Consideration of tailpipe emissions control, components' durability such as battery health degradation minimization, drivability, and comfort can make a huge performance improvement in commercialized HEVs.

- **Drive Cycle Development Using Real-world Scenarios**

  The developed adaptive ECMS is well-suited for hybrid electric delivery applications. For the future work of this study, we can target a real-world case of a hybrid electric delivery vehicle. The next step is to collect the vehicle's

daily driving routes' data. In this way, we can construct the typical drive cycles for the driver. Finally, having real-world drive cycles ready, we are able to tune and optimize the ECMS control parameters effectively. This work would include building the model-in-the-loop vehicle simulation and validation, gathering daily trip information, drive cycle development, and testing the proposed ECMS in the vehicle model.

- **PI-based ECMS**

  We have found that a PI-based ECMS controller can achieve a reasonable battery charge-sustaining mode. However, this method prioritizes the compensation of SOC trajectory deviation and does not provide near-optimal fuel economy. In future work, we will explore simple and effective adaptive control strategies to dynamically tune the PI controller's gain to improve fuel consumption minimization.

- **ECMS Application for an Autonomous HEV**

  With the rapid advancement of Intelligent Transportation Systems (ITS) and autonomous vehicle's progress, accessibility to traffic data, road conditions, and vehicle information are more feasible. An autonomous vehicle observes environmental information through the fusion of cameras, sensors, GPS, and LIDAR data. This information, combined with the road network data and vehicle dynamics, can determine the future drive cycle information. In this way, offline optimization-based results on diffrent drive cycle patterns achieved by optimization-based methods such as DP and ECMS can be used online in the

power control unit to perform an optimal energy distribution. One possible extension of this thesis can be exploring the ECMS application on an Autonomous HEV (A-HEV).

- **Experimental Implementation**

  To effectively investigate the real-time performance and applicability of the proposed method in this thesis, an experimental examination through HIL test needs to be done. In this way, reliability and feasibility challenges can be effectively addressed.

References

[1] Transportation and climate change, . URL https://www.epa.gov/transportation-air-pollution-and-climate-change/carbon-pollution-transportation.

[2] Greenhouse gas sources and sinks: executive summary 2020, 2020. URL https://www.canada.ca/en/environment-climate-change/services/climate-change/greenhouse-gas-emissions/sources-sinks-executive-summary-2020.html.

[3] Mild hybrid electric vehicle (mhev) introduction, 2020. URL https://x-engineer.org/automotive-engineering/vehicle/hybrid/mild-hybrid-electric-vehicle-mhev-introduction/.

[4] Stephan Uebel, Nikolce Murgovski, Conny Tempelhahn, and Bernard Bäker. Optimal energy management and velocity control of hybrid electric vehicles. *IEEE Transactions on Vehicular Technology*, 67(1):327–337, 2017.

[5] James Billington. Ups introduces groundbreaking hybrid electric delivery trucks, 2019. URL https://www.electrichybridvehicletechnology.com/

news/buses-commercial-vehicles/ups-introduces-groundbreaking-hybrid-electric-delivery-trucks.html.

[6] Ups has new hybrid trucks with a neat trick, 2019. URL https://www.cnn.com/2019/09/04/tech/ups-hybrid-truck-trnd/index.html.

[7] Eighteen-month final evaluation of ups second generation diesel hybrid-electric delivery vans. Technical report, National Renewable Energy Laboratory, U.S. Department of Energy, 2012. URL https://www.mostcooperation.com/.

[8] Salman Habib, Muhammad Mansoor Khan, Farukh Abbas, Lei Sang, Muhammad Umair Shahid, and Houjun Tang. A comprehensive study of implemented international standards, technical challenges, impacts and prospects for electric vehicles. *IEEE Access*, 6:13866–13890, 2018.

[9] Daniel F Opila, Deepak Aswani, Ryan McGee, Jeffrey A Cook, and Jessy W Grizzle. Incorporating drivability metrics into optimal energy management strategies for hybrid vehicles. In *Proceedings of the IEEE Conference on Decision and Control*, pages 4382–4389, 2008. ISBN 9781424431243. doi: 10.1109/CDC.2008.4738731.

[10] Pei Zhang, Fuwu Yan, and Changqing Du. A comprehensive analysis of energy management strategies for hybrid electric vehicles based on bibliometrics. *Renewable and Sustainable Energy Reviews*, 48:88–104, 2015.

[11] Light duty vehicle emissions, . URL https://www.epa.gov/greenvehicles/light-duty-vehicle-emissions.

[12] Maximilian Helbing, Stephan Uebel, Conny Tempelhahn, and Bernard Bäker. An evaluated review of powertrain control strategies for hybrid electrical vehicles. *ATZelektronik worldwide*, 10(4):46–51, 2015.

[13] Ali Emadi. *Advanced electric drive vehicles*. CRC Press, Boca Raton, FL, Oct. 2014. ISBN 978-1-4665-9769-3.

[14] Wolfgang Stolz, Robert Kornhaas, Ralph Krause, and Tino Sommer. Domain control units-the solution for future e/e architectures? In *SAE Technical Paper*. SAE Technical Paper, Apr. 2010.

[15] Arunkumar Jayakumar, Andrew Chalmers, and Tek Tjing Lie. Review of prospects for adoption of fuel cell electric vehicles in new zealand. *IET Electrical Systems in Transportation*, 7(4):259–266, 2017.

[16] Y. Yang, K. Arshad-Ali, J. Roeleveld, and A. Emadi. State-of-the-art electrified powertrains-hybrid, plug-in, and electric vehicles. *International Journal of Powertrains*, 5(1):1–29, 2016.

[17] Krishna Veer, Singh Hari, Om Bansal, and Dheerendra Singh. A comprehensive review on hybrid electric vehicles : architectures and components. *Journal of Modern Transportation*, 27(2):77–107, 2019. ISSN 2196-0577. doi: 10.1007/s40534-019-0184-3.

[18] Fuad Un-Noor, Sanjeevikumar Padmanaban, Lucian Mihet-Popa, Mohammad Nurunnabi Mollah, and Eklas Hossain. A comprehensive study of key electric vehicle (ev) components, technologies, challenges, impacts, and future direction of development. *Energies*, 10(8), Aug. 2017.

[19] Alireza Khaligh and Zhihao Li. Battery, ultracapacitor, fuel cell, and hybrid energy storage systems for electric, hybrid electric, fuel cell, and plug-in hybrid electric vehicles: State of the art. *IEEE transactions on Vehicular Technology*, 59(6):2806–2814, Jul. 2010.

[20] B. Bilgin, P. Magne, P. Malysz, Y. Yang, V. Pantelic, M. Preindl, A. Korobkine, W. Jiang, M. Lawford, and A. Emadi. Making the case for electrified transportation. *IEEE Transactions on Transportation Electrification*, 1(1):4–17, June 2015. ISSN 2372-2088. doi: 10.1109/TTE.2015.2437338.

[21] E. Chemali, M. Preindl, P. Malysz, and A. Emadi. Electrochemical and electrostatic energy storage and management systems for electric drive vehicles: State-of-the-art review and future trends. *IEEE Trans. Emerg. Sel. Topics Power Electron.*, 4(3):1117–1134, Sep. 2016. ISSN 2168-6777. doi: 10.1109/JESTPE.2016.2566583.

[22] Andrew Burke and Hengbing Zhao. Applications of supercapacitors in electric and hybrid vehicles. Technical report, Institute of transportation studies, Davis, California, USA, April 2015.

[23] John Ramoul, Ephrem Chemali, Lea Dorn-Gomba, and Ali Emadi. A neural network energy management controller applied to a hybrid energy storage system using multi-source inverter. In *2018 IEEE Energy Conversion Congress and Exposition (ECCE)*, pages 2741–2747. IEEE, Sep. 2018.

[24] J. Shen and A. Khaligh. A supervisory energy management control strategy in a battery/ultracapacitor hybrid energy storage system. *IEEE Transactions*

on Transportation Electrification, 1(3):223–231, Oct. 2015. doi: 10.1109/TTE.2015.2464690.

[25] Gergana Vacheva and Nikolay Hinov. An overview of the state of art of fuel cells in electric vehicles. In *2019 International Conference on Creative Business for Smart and Sustainable Growth (CREBUS)*, pages 1–4, 2019.

[26] Siang Fui Tie and Chee Wei Tan. A review of energy sources and energy management system in electric vehicles. *Renewable and sustainable energy reviews*, 20:82–102, Apr. 2013.

[27] Björn Bolund, Hans Bernhoff, and Mats Leijon. Flywheel energy and power storage systems. *Renewable and Sustainable Energy Reviews*, 11(2):235–258, 2007.

[28] R Pena-Alzola, Rafael Sebastián, Jerónimo Quesada, and Antonio Colmenar. Review of flywheel based energy storage systems. In *2011 International Conference on Power Engineering, Energy and Electrical Drives*, pages 1–6, May 2011.

[29] Mustafa E Amiryar and Keith R Pullen. A review of flywheel energy storage system technologies and their applications. *Applied Sciences*, 7(3):286, 2017.

[30] Romina Rodriguez, Matthias Preindl, James S Cotton, and Ali Emadi. Review and trends of thermoelectric generator heat recovery in automotive applications. *IEEE Transactions on Vehicular Technology*, 68(6):5366–5378, Jun. 2019.

[31] A. Emadi, K. Rajashekara, S. S. Williamson, and S. M. Lukic. Topological overview of hybrid electric and fuel cell vehicular power system architectures

and configurations. *IEEE Trans. Veh. Technol.*, 54(3):763–770, May 2005. ISSN 0018-9545. doi: 10.1109/TVT.2005.847445.

[32] M. Ehsani, Y. Gao, and J. M. Miller. Hybrid electric vehicles: Architecture and motor drives. *Proceedings of the IEEE*, 95(4):719–728, Apr. 2007. ISSN 0018-9219. doi: 10.1109/JPROC.2007.892492.

[33] Wisdom Enang and Chris Bannister. Modelling and control of hybrid electric vehicles (a comprehensive review). *Renewable and Sustainable Energy Reviews*, 74:1210–1239, 2017.

[34] Wei Liu. *Hybrid electric vehicle system modeling and control*. John Wiley & Sons, 2017.

[35] United States Department of Energy. Hybrid electric vehicles. URL `https://www.autonomie.net/references/hev_26.html`.

[36] Ali Emadi. *Handbook of automotive power electronics and motor drives*. CRC press, 2017. ISBN 9781420028157. doi: 10.1201/9781420028157.

[37] Varun M Navale, Kyle Williams, Athanassios Lagospiris, Michael Schaffert, and Markus-Alexander Schweiker. Revolution of e/e architectures. *SAE International Journal of Passenger Cars-Electronic and Electrical Systems*, 8(2015-01-0196):282–288, Mar. 2015.

[38] Detlef Zerfowski and Andreas Lock. Functional architecture and e/e-architecture – a challenge for the automotive industry. In Michael Bargende, Hans-Christian Reuss, Andreas Wagner, and Jochen Wiedemann, editors, *19. Internationales Stuttgarter Symposium*, pages 909–920, Wiesbaden,

2019. Springer Fachmedien Wiesbaden. ISBN 978-3-658-25939-6.

[39] M Krueger, Stefan Straube, A Middendorf, Daniel Hahn, T Dobs, and K-D Lang. Requirements for the application of ecus in e-mobility originally qualified for gasoline cars. *Microelectronics Reliability*, 64:140–144, 2016.

[40] DianGe Yang, Kun Jiang, Ding Zhao, ChunLei Yu, Zhong Cao, ShiChao Xie, ZhongYang Xiao, XinYu Jiao, SiJia Wang, and Kai Zhang. Intelligent and connected vehicles: Current status and future perspectives. *Science China Technological Sciences*, 61(10):1446–1471, 2018.

[41] System architecture advisory services. Technical report, Continental Automotive GmbH, 2015. URL `https://www.continental-automotive.com/`.

[42] Electronic vehicle management: Forward-looking vehicle management solutions. Technical report, Continental Automotive GmbH, 2014. URL `https://www.continental-automotive.com/`.

[43] Wired for performance: New electronics architecture for commercial vehicles. Technical report, Continental Automotive GmbH, 2015. URL `https://www.continental-automotive.com/`.

[44] Martin Buechel, Jelena Frtunikj, Klaus Becker, Stephan Sommer, Christian Buckl, Michael Armbruster, Andre Marek, Andreas Zirkler, Cornel Klein, and Alois Knoll. An automated electric vehicle prototype showing new trends in automotive architectures. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 1274–1279, 2015.

[45] S. Brunner, J. Roder, M. Kucera, and T. Waas. Automotive e/e-architecture enhancements by usage of ethernet tsn. In *2017 13th Workshop on Intelligent Solutions in Embedded Systems (WISES)*, pages 9–13, 2017.

[46] Stefan Kugele, Vadim Cebotari, Mario Gleirscher, Morteza Farzaneh, Christoph Segler, Sina Shafaei, Hans-Jrg Vgel, Fridolin Bauer, Alois Knoll, Diego Marmsoler, and Hans-Ulrich Michel. Research challenges for a future-proof e/e architecture - a project statement. Sep. 2017.

[47] Ahmed M Ali and Dirk Söffker. Towards optimal power management of hybrid electric vehicles in real-time: A review on methods, challenges, and state-of-the-art solutions. *Energies*, 11(3):476–500, Feb. 2018.

[48] Nan Xu, Yan Kong, Liang Chu, Hao Ju, Zhihua Yang, Zhe Xu, and Zhuoqi Xu. Towards a smarter energy management system for hybrid vehicles: A comprehensive review of control strategies. *Applied Sciences*, 9(10):2026, May 2019.

[49] Sanjaka G Wirasingha and Ali Emadi. Classification and review of control strategies for plug-in hybrid electric vehicles. *IEEE Transactions on vehicular technology*, 60(1):111–122, 2010.

[50] Domenico Bianchi, Luciano Rolando, Lorenzo Serrao, Simona Onori, Giorgio Rizzoni, Nazar Al-Khayat, Tung-Ming Hsieh, and Pengju Kang. A rule-based strategy for a series/parallel hybrid electric vehicle: an approach based on dynamic programming. In *ASME 2010 Dynamic Systems and Control Conference*, pages 507–514, Sep. 2010.

[51] Y. Cheng, K. Chen, C. C. Chan, A. Bouscayrol, and S. Cui. Global modeling and control strategy simulation for a hybrid electric vehicle using electrical variable transmission. In *2008 IEEE Vehicle Power and Propulsion Conference*, pages 1–5, 2008.

[52] Valerie H Johnson, Keith B Wipke, and David J Rausen. Hev control strategy for real-time optimization of fuel economy and emissions. Technical report, SAE Technical Paper, Apr. 2000.

[53] Yujie Wang, Zhendong Sun, and Zonghai Chen. Energy management strategy for battery/supercapacitor/fuel cell hybrid source vehicles based on finite state machine. *Applied Energy*, 254:113707, Nov. 2019. ISSN 03062619.

[54] João Pedro Fernandes Trovão and Paulo José Gameiro Pereirinha. Control scheme for hybridised electric vehicles with an online power follower management strategy. *IET Electrical Systems in Transportation*, 5(1):12–23, Sep. 2014.

[55] Mahyar Vajedi, Amir Taghavipour, and Nasser L Azad. Traction-motor power ratio and speed trajectory optimization for power split phevs using route information. In *ASME 2012 International Mechanical Engineering Congress and Exposition*, pages 301–308. American Society of Mechanical Engineers Digital Collection, Nov. 2012.

[56] Kamil Çağatay Bayindir, Mehmet Ali Gözüküçük, and Ahmet Teke. A comprehensive overview of hybrid electric vehicle: Powertrain configurations, powertrain control techniques and electronic control units. *Energy conversion and Management*, 52(2):1305–1313, Feb. 2011.

[57] MH Hajimiri and Farzad Rajaei Salmasi. A fuzzy energy management strategy for series hybrid electric vehicle with predictive control and durability extension of the battery. In *2006 IEEE Conference on Electric and Hybrid Vehicles*, pages 1–5, Pune, India, 2006.

[58] Moshe Sniedovich. Dynamic programming and principles of optimality. *Journal of Mathematical Analysis and Applications*, 65(3):586–606, Oct. 1978. ISSN 10960813. doi: 10.1016/0022-247X(78)90166-X.

[59] Gino Paganelli, Gabriele Ercole, Avra Brahma, Yann Guezennec, and Giorgio Rizzoni. General supervisory control policy for the energy optimization of charge-sustaining hybrid electric vehicles. *JSAE review*, 22(4):511–518, Oct. 2001.

[60] Simona Onori and Lorenzo Serrao. On adaptive-ecms strategies for hybrid electric vehicles. In *Proceedings of the international scientific conference on hybrid and electric vehicles, Malmaison, France*, volume 67, pages 1–7, Dec. 6-7, 2011.

[61] Chao Sun, Fengchun Sun, and Hongwen He. Investigating adaptive-ecms with velocity forecast ability for hybrid electric vehicles. *Applied Energy*, 185:1644–1653, Jan. 2017.

[62] Cristian Musardo, Giorgio Rizzoni, Yann Guezennec, and Benedetto Staccia. A-ecms: An adaptive algorithm for hybrid electric vehicle energy management. *European Journal of Control*, 11(4-5):509–524, Nov. 2005.

[63] Bo Gu and Giorgio Rizzoni. An adaptive algorithm for hybrid electric vehicle

energy management based on driving pattern recognition. In *ASME 2006 International Mechanical Engineering Congress and Exposition*, pages 249–258. American Society of Mechanical Engineers Digital Collection, Jan. 2006.

[64] Simona Onori, Lorenzo Serrao, and Giorgio Rizzoni. Adaptive equivalent consumption minimization strategy for hybrid electric vehicles. In *ASME 2010 dynamic systems and control conference*, pages 499–505. American Society of Mechanical Engineers Digital Collection, Sep. 12-15, 2010.

[65] E. F. Camacho and C. Bordons. Model predictive control. In *Advanced Textbooks in Control and Signal Processing*. 2007. doi: 10.1201/9781351170802-9.

[66] H. A. Borhan, A. Vahidi, A. M. Phillips, M. L. Kuang, and I. V. Kolmanovsky. Predictive energy management of a power-split hybrid electric vehicle. In *2009 American Control Conference*, pages 3970–3976, 2009.

[67] Shuo Zhang, Rui Xiong, and Fengchun Sun. Model predictive control for power management in a plug-in hybrid electric vehicle with a hybrid energy storage system. *Applied Energy*, 185:1654–1662, Jan. 2017.

[68] Yafei Wang, Xiangyu Wang, Yong Sun, and Sixiong You. Model predictive control strategy for energy optimization of series-parallel hybrid electric vehicle. *Journal of cleaner production*, 199:348–358, 2018.

[69] Imran Rahman, Pandian M Vasant, Balbir Singh Mahinder Singh, and M Abdullah-Al-Wadud. Novel metaheuristic optimization strategies for plug-in hybrid electric vehicles: A holistic review. *Intelligent Decision Technologies*, 10 (2):149–163, 2016.

[70] Ozlem Senvar, Ebru Turanoglu, and Cengiz Kahraman. Usage of metaheuristics in engineering: A literature review. In *Meta-heuristics optimization algorithms in engineering, business, economics, and finance*, pages 484–528. IGI Global, 2013.

[71] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 4, pages 1942–1948, Nov. 1995. doi: 10.1109/ICNN.1995.488968.

[72] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43, Oct. 1995. doi: 10.1109/MHS.1995.494215.

[73] S. Caux, D. Wanderley-Honda, D. Hissel, and M. Fadel. On-line energy management for hev based on particle swarm optimization. In *2010 IEEE Vehicle Power and Propulsion Conference*, pages 1–7, Sep. 2010. doi: 10.1109/VPPC.2010.5729197.

[74] Xiaolan Wu, Binggang Cao, Jianping Wen, and Yansheng Bian. Particle swarm optimization for plug-in hybrid electric vehicle control strategy parameter. In *2008 IEEE Vehicle Power and Propulsion Conference*, pages 1–5, Sep. 2008. doi: 10.1109/VPPC.2008.4677635.

[75] Zeyu Chen, Rui Xiong, Kunyu Wang, and Bin Jiao. Optimal energy management strategy of a plug-in hybrid electric vehicle based on a particle swarm optimization algorithm. *Energies*, 8(5):3661–3678, 2015. ISSN 1996-1073. doi: 10.3390/en8053661.

[76] Zeyu Chen, Rui Xiong, and Jiayi Cao. Particle swarm optimization-based optimal power management of plug-in hybrid electric vehicles considering uncertain driving conditions. *Energy*, 96:197 – 208, 2016. ISSN 0360-5442. doi: https://doi.org/10.1016/j.energy.2015.12.071.

[77] Y. Yang, Y. Shih, and J. Chen. Real-time torque distribution strategy for an electric vehicle with multiple traction motors by particle swarm optimization. In *2013 CACS International Automatic Control Conference (CACS)*, pages 233–238, Dec. 2013. doi: 10.1109/CACS.2013.6734138.

[78] Syuan-Yi Chen, Yi-Hsuan Hung, Chien-Hsun Wu, and Siang-Ting Huang. Optimal energy management of a hybrid electric powertrain system using improved particle swarm optimization. *Applied Energy*, 160:132 – 145, 2015. ISSN 0306-2619. doi: https://doi.org/10.1016/j.apenergy.2015.09.047.

[79] Syuan-Yi Chen, Chien-Hsun Wu, Yi-Hsuan Hung, and Cheng-Ta Chung. Optimal strategies of energy management integrated with transmission control for a hybrid electric vehicle using dynamic particle swarm optimization. *Energy*, 160:154 – 170, 2018. ISSN 0360-5442. doi: https://doi.org/10.1016/j.energy.2018.06.023.

[80] Maciej Wieczorek and Mirosaw Lewandowski. A mathematical representation of an energy management strategy for hybrid energy storage system in electric vehicle and real time optimization using a genetic algorithm. *Applied Energy*, 192:222 – 233, Apr. 2017. ISSN 0306-2619. doi: https://doi.org/10.1016/j.apenergy.2017.02.022.

[81] J. P. F. Trovo, V. D. N. Santos, P. G. Pereirinha, H. M. Jorge, and C. H. Antunes. A simulated annealing approach for optimal power source management in a small ev. *IEEE Transactions on Sustainable Energy*, 4(4):867–876, Oct. 2013. ISSN 1949-3037. doi: 10.1109/TSTE.2013.2253139.

[82] Joo P. Trovo, Paulo G. Pereirinha, Humberto M. Jorge, and Carlos Henggeler Antunes. A multi-level energy management system for multi-source electric vehicles an integrated rule-based meta-heuristic approach. *Applied Energy*, 105:304 – 318, 2013. ISSN 0306-2619. doi: https://doi.org/10.1016/j.apenergy.2012.12.081.

[83] J. P. F. Trovo, V. D. N. Santos, C. H. Antunes, P. G. Pereirinha, and H. M. Jorge. A real-time energy management architecture for multisource electric vehicles. *IEEE Transactions on Industrial Electronics*, 62(5):3223–3233, May 2015. ISSN 1557-9948. doi: 10.1109/TIE.2014.2376883.

[84] John N. Tsitsiklis and Benjamin van Roy. Feature-based methods for large scale dynamic programming. *Machine Learning*, 22(1):59–94, Mar. 1996. ISSN 1573-0565. doi: 10.1007/BF00114724.

[85] Chao Sun, Xiaosong Hu, Scott J Moura, and Fengchun Sun. Velocity predictors for predictive energy management in hybrid electric vehicles. *IEEE Transactions on Control Systems Technology*, 23(3):1197–1204, May 2015.

[86] Jichao Liu, Yangzhou Chen, Jingyuan Zhan, and Fei Shang. Heuristic dynamic programming based online energy management strategy for plug-in hybrid electric vehicles. *IEEE Transactions on Vehicular Technology*, 68(5):4479–4493, May 2019.

[87] Jungme Park, ZhiHang Chen, and Yi L Murphey. Intelligent vehicle power management through neural learning. In *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7, Jul. 2010.

[88] Zheng Chen, Chunting Chris Mi, Jun Xu, Xianzhi Gong, and Chenwen You. Energy management for a power-split plug-in hybrid electric vehicle based on dynamic programming and neural networks. *IEEE Transactions on Vehicular Technology*, 63(4):1567–1580, May 2014.

[89] P. G. Anselma, Y. Huo, J. Roeleveld, G. Belingardi, and A. Emadi. Slope-weighted energy-based rapid control analysis for hybrid electric vehicles. *IEEE Transactions on Vehicular Technology*, 68(5):4458–4466, May 2019. ISSN 1939-9359. doi: 10.1109/TVT.2019.2899360.

[90] R. S. Sutton and A. G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition, 1998. ISBN 0262193981.

[91] Richard S. Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Proceedings of the 12th International Conference on Neural Information Processing Systems*, NIPS99, page 10571063, Cambridge, MA, USA, Nov. 1999. MIT Press.

[92] Yanli Yin, Yan Ran, Liufeng Zhang, Xiaoliang Pan, and Yong Luo. An energy management strategy for a super-mild hybrid electric vehicle based on a known model of reinforcement learning. *Journal of Control Science and Engineering*, 2019, Apr. 2019.

[93] Teng Liu, Bo Wang, and Chenglang Yang. Online markov chain-based energy management for a hybrid tracked vehicle with speedy q-learning. *Energy*, 160: 544–555, Oct. 2018.

[94] Guodong Du, Yuan Zou, Xudong Zhang, Zehui Kong, Jinlong Wu, and Dingbo He. Intelligent energy management for hybrid electric tracked vehicles using online reinforcement learning. *Applied Energy*, 251:113388, Oct. 2019.

[95] Atriya Biswas, Pier G Anselma, and Ali Emadi. Real-time optimal energy management of electrified powertrains with reinforcement learning. In *2019 IEEE Transportation Electrification Conference and Expo (ITEC)*, pages 1–6, Jun. 2019.

[96] Bin Xu, Dhruvang Rathod, Darui Zhang, Adamu Yebi, Xueyu Zhang, Xiaoya Li, and Zoran Filipi. Parametric study on reinforcement learning optimized energy management strategy for a hybrid electric vehicle. *Applied Energy*, 259: 114200, Feb. 2020.

[97] Pu Zhao, Yanzhi Wang, Naehyuck Chang, Qi Zhu, and Xue Lin. A deep reinforcement learning framework for optimizing fuel economy of hybrid electric vehicles. In *2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 196–202, Jan. 2225, 2018.

[98] Yue Hu, Weimin Li, Kun Xu, Taimoor Zahid, Feiyan Qin, and Chenming Li. Energy management strategy for a hybrid electric vehicle based on deep reinforcement learning. *Applied Sciences*, 8(2):187, 2018.

[99] X. Zeng and J. Wang. A parallel hybrid electric vehicle energy management strategy using stochastic model predictive control with road grade preview. *IEEE Transactions on Control Systems Technology*, 23(6):2416–2423, Nov. 2015. ISSN 2374-0159. doi: 10.1109/TCST.2015.2409235.

[100] Yongchang Du, Yue Zhao, Qinpu Wang, Yuanbo Zhang, and Huaicheng Xia. Trip-oriented stochastic optimal energy management strategy for plug-in hybrid electric bus. *Energy*, 115:1259 – 1271, 2016. ISSN 0360-5442. doi: https://doi.org/10.1016/j.energy.2016.09.056.

[101] Weimin Li, Guoqing Xu, Zhancheng Wang, and Yangsheng Xu. Dynamic energy management for hybrid electric vehicle based on adaptive dynamic programming. In *2008 IEEE International Conference on Industrial Technology*, pages 1–6, Apr. 2008. doi: 10.1109/ICIT.2008.4608440.

[102] Zheng Chen, Hengjie Hu, Yitao Wu, Renxin Xiao, Jiangwei Shen, and Yonggang Liu. Energy management for a power-split plug-in hybrid electric vehicle based on reinforcement learning. *Applied Sciences*, 8(12), 2018. ISSN 2076-3417. doi: 10.3390/app8122494.

[103] G. Ripaccioli, D. Bernardini, S. Di Cairano, A. Bemporad, and I. V. Kolmanovsky. A stochastic model predictive control approach for series hybrid electric vehicle power management. In *Proceedings of the 2010 American Control Conference*, pages 5844–5849, Jun. 2010. doi: 10.1109/ACC.2010.5530504.

[104] Yanjun Huang, Hong Wang, Amir Khajepour, Hongwen He, and Jie Ji. Model predictive control power management strategies for hevs: A review. *Journal of Power Sources*, 341:91–106, Feb. 2017.

[105] Changle Xiang, Feng Ding, Weida Wang, and Wei He. Energy management of a dual-mode power-split hybrid electric vehicle based on velocity prediction and nonlinear model predictive control. *Applied energy*, 189:640–653, Mar. 2017.

[106] Chao Sun, Scott Jason Moura, Xiaosong Hu, J Karl Hedrick, and Fengchun Sun. Dynamic traffic feedback data enabled energy management in plug-in hybrid electric vehicles. *IEEE Transactions on Control Systems Technology*, 23 (3):1075–1086, May 2015.

[107] Y. Hu, C. Chen, T. He, J. He, X. Guan, and B. Yang. Proactive power management scheme for hybrid electric storage system in evs: An mpc method. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–12, 2019.

[108] Zheng Chen, Chris Chunting Mi, Rui Xiong, Jun Xu, and Chenwen You. Energy management of a power-split plug-in hybrid electric vehicle based on genetic algorithm and quadratic programming. *Journal of Power Sources*, 248:416–426, Feb. 2014.

[109] Liang Li, Yahui Zhang, Chao Yang, Xiaohong Jiao, Lipeng Zhang, and Jian Song. Hybrid genetic algorithm-based optimization of powertrain and control parameters of plug-in hybrid electric bus. *Journal of the franklin institute*, 352 (3):776–801, Mar. 2015.

[110] Jingxian Hao, Zhuoping Yu, Zhiguo Zhao, Peihong Shen, and Xiaowen Zhan. Optimization of key parameters of energy management strategy for hybrid electric vehicle using direct algorithm. *Energies*, 9(12):997, Nov. 2016.

[111] Morteza Montazeri-Gh, Amir Poursamad, and Babak Ghalichi. Application of

genetic algorithm for optimization of control strategy in parallel hybrid electric vehicles. *Journal of the Franklin Institute*, 343(4-5):420–435, 2006.

[112] Chunfang Yin, Shaohua Wang, Chengquan Yu, Jiaxin Li, and Sheng Zhang. Fuzzy optimization of energy management for power split hybrid electric vehicle based on particle swarm optimization algorithm. *Advances in Mechanical Engineering*, 11(2):1687814019830797, 2019.

[113] Fengqi Zhang, Junqiang Xi, and Reza Langari. Real-time energy management strategy based on velocity forecasts using v2v and v2i communications. *IEEE Transactions on Intelligent Transportation Systems*, 18(2):416–430, 2016.

[114] Xiang Tian, Yingfeng Cai, Xiaodong Sun, Zhen Zhu, and Yiqiang Xu. An adaptive ecms with driving style recognition for energy optimization of parallel hybrid electric buses. *Energies*, 189:116151, 2019.

[115] E. Ozatay, S. Onori, J. Wollaeger, U. Ozguner, G. Rizzoni, D. Filev, J. Michelini, and S. Di Cairano. Cloud-based velocity profile optimization for everyday driving: A dynamic-programming-based solution. *IEEE Transactions on Intelligent Transportation Systems*, 15(6):2491–2505, 2014.

[116] Luc Van Dijk. Future vehicle networks and ecus architecture and technology considerations, 2017. URL `https://www.nxp.com/`.

[117] S. Jadhav and D. Kshirsagar. A survey on security in automotive networks. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–6, 2018.

[118] Shane Tuohy, Martin Glavin, Ciaran Hughes, Edward Jones, Mohan Trivedi, and Liam Kilmartin. Intra-vehicle networks: A review. *IEEE Transactions on Intelligent Transportation Systems*, 16:1–12, Jan. 2014. doi: 10.1109/TITS.2014.2320605.

[119] Fabienne Nouvel, Wilfried Gouret, P. Mazrio, and Ghas Zein. *Automotive Network Architecture for ECUs Communications*, pages 69–90. IGI Global, Apr. 2009. ISBN 1-60566-338-7. doi: 10.4018/978-1-60566-338-8.ch004.

[120] B Kumar and J Ramesh. Automotive in vehicle network protocols. In *2014 International Conference on Computer Communication and Informatics*, pages 1–5, Jan. 2014. doi: 10.1109/ICCCI.2014.6921836.

[121] Shugang Jiang. Vehicle e/e architecture and its adaptation to new technical trends. In *SAE Technical Paper*, Apr. 2019.

[122] Dai-Duong Tran, Majid Vafaeipour, Mohamed El Baghdadi, Ricardo Barrero, Joeri Van Mierlo, and Omar Hegazy. Thorough state-of-the-art analysis of electric and hybrid vehicle powertrains: Topologies and integrated energy management strategies. *Renewable and Sustainable Energy Reviews*, 119:109596, 2020.

[123] Amir Taghavipour, Ramin Masoudi, Nasser L. Azad, and John McPhee. High-fidelity modeling of a power-split plug-in hybrid electric powertrain for control performance evaluation. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 55843, page V001T01A008. American Society of Mechanical Engineers, 2013.

[124] Namwook Kim and Aymeric Rousseau. Thermal impact on the control and the efficiency of the 2010 toyota prius hybrid electric vehicle. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 230(1):82–92, 2016.

[125] Namwook Kim, Aymeric Rousseau, and Eric Rask. Autonomie model validation with test data for 2010 toyota prius. Technical report, SAE Technical Paper, 2012.

[126] Kyoungho Ahn and Hesham Rakha. A simple hybrid electric vehicle fuel consumption model for transportation applications. Technical report, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2019.

[127] Eric Rask, Michael Duoba, Henning Lohse-Busch, Daniel Bocci, et al. Model year 2010 (gen 3) toyota prius level 1 testing report. Technical report, Argonne National Lab.(ANL), Argonne, IL (United States), 2010.

[128] Timothy A Burress, Steven L Campbell, Chester Coomer, Curtis William Ayers, Andrew A Wereszczak, Joseph Philip Cunningham, Laura D Marlino, Larry Eugene Seiber, and Hua-Tay Lin. Evaluation of the 2010 toyota prius hybrid synergy drive system. Technical report, Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States)., 2011.

[129] Saeed Amirfarhangi Bonab. *ASEMS: Autonomous Specific Energy Management Strategy*. MASc dissertation, McMaster University, 2019.

[130] Jinglai Wu, Jiageng Ruan, Nong Zhang, and Paul D Walker. An optimized real-time energy management strategy for the power-split hybrid electric vehicles. *IEEE Transactions on Control Systems Technology*, 27(3):1194–1202, May 2019.

[131] Hybrid electric vehicle testing. URL `https://www.anl.gov/es/energy-systems-d3-2010-toyota-prius`.

[132] Hussein Basma, Charbel Mansour, Houssam Halaby, and Anis Baz Radwan. Methodology to design an optimal rule-based energy management strategy using energetic macroscopic representation: Case of plug-in series hybrid electric vehicle. 7(3):188–198, 2018.

[133] Alexandre Chasse, Antonio Sciarretta, and Jonathan Chauvin. Online optimal control of a parallel hybrid with costate adaptation rule. *IFAC proceedings volumes*, 43(7):99–104, 2010.

[134] Mitra Pourabdollah, Nikolce Murgovski, Anders Grauers, and Bo Egardt. An iterative dynamic programming/convex optimization procedure for optimal sizing and energy management of phevs. *IFAC Proceedings Volumes*, 47(3):6606–6611, 2014.

[135] Hongliang Jiang, Liangfei Xu, Jianqiu Li, Zunyan Hu, and Minggao Ouyang. Energy management and component sizing for a fuel cell/battery/supercapacitor hybrid powertrain based on two-dimensional optimization algorithms. *Energy*, 177:386–396, 2019.

[136] Viet Ngo, Theo Hofman, Maarten Steinbuch, and Alex Serrarens. Optimal

control of the gearshift command for hybrid electric vehicles. *IEEE Transactions on Vehicular Technology*, 61(8):3531–3543, 2012.

[137] Xiaowu Zhang, Huei Peng, Jing Sun, and Shengbo Li. Automated modeling and mode screening for exhaustive search of double-planetary-gear power split hybrid powertrains. In *Dynamic Systems and Control Conference*, volume 46186, page V001T15A002. American Society of Mechanical Engineers, 2014.

[138] Jinming Liu and Huei Peng. Modeling and control of a power-split hybrid vehicle. *IEEE transactions on control systems technology*, 16(6):1242–1251, 2008.

[139] Zachary D Asher, David A Baker, and Thomas H Bradley. Prediction error applied to hybrid electric vehicle optimal fuel economy. *IEEE Transactions on Control Systems Technology*, 26(6):2121–2134, 2017.

[140] Yalian Yang, Huanxin Pei, Xiaosong Hu, Yonggang Liu, Cong Hou, and Dongpu Cao. Fuel economy optimization of power split hybrid vehicles: A rapid dynamic programming approach. *Energy*, 166:929–938, 2019.

[141] Donald E Kirk. *Optimal control theory: an introduction*. Courier Corporation, 2004.

[142] Olle Sundstrom and Lino Guzzella. A generic dynamic programming matlab function. In *2009 IEEE control applications,(CCA) & intelligent control,(ISIC)*, pages 1625–1630. IEEE, 2009.

[143] Ganesh Mohan, Francis Assadian, and Stefano Longo. Comparative analysis of forward-facing models vs backward-facing models in powertrain component sizing. 2013.

[144] Hybrid-EV Committee et al. Recommended practice for measuring the exhaust emissions and fuel economy of hybrid-electric vehicles, including plug-in hybrid vehicles. In *Proceedings of the SAE Std. J1711_SAE World Congress, Detroit, MI, USA*, pages 13–15, 2010.

[145] Kukhyun Ahn and P Y Papalambros. Engine optimal operation lines for power-split hybrid electric vehicles. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 223(9):1149–1162, 2009. doi: 10.1243/09544070JAUTO1124. URL `https://doi.org/10.1243/09544070JAUTO1124`.

[146] João Pedro Fernandes Trovão and Paulo José Gameiro Pereirinha. Control scheme for hybridised electric vehicles with an online power follower management strategy. *IET Electrical Systems in Transportation*, 5(1):12–23, 2014.

[147] Zachary D Asher, David A Baker, and Thomas H Bradley. Prediction error applied to hybrid electric vehicle optimal fuel economy. *IEEE Transactions on Control Systems Technology*, 26(6):2121–2134, 2017.

[148] Rui Wang and Srdjan M Lukic. Dynamic programming technique in hybrid electric vehicle optimization. In *2012 IEEE International Electric Vehicle Conference*, pages 1–8. IEEE, 2012.

[149] Namwook Kim and Aymeric Rousseau. Sufficient conditions of optimal control based on pontryagins minimum principle for use in hybrid electric vehicles. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 226(9):1160–1170, 2012. doi: 10.1177/0954407012438304. URL `https://doi.org/10.1177/0954407012438304`.

[150] Alberto Santucci, Aldo Sorniotti, and Constantina Lekakou. Power split strategies for hybrid energy storage systems for vehicular applications. *Journal of Power Sources*, 258:395–407, 2014.

[151] Changle Xiang, Feng Ding, Weida Wang, and Wei He. Energy management of a dual-mode power-split hybrid electric vehicle based on velocity prediction and nonlinear model predictive control. *Applied energy*, 189:640–653, 2017.

[152] Amir Rezaei, Jeffrey B Burl, Bin Zhou, and Mohammad Rezaei. A new real-time optimal energy management strategy for parallel hybrid electric vehicles. *IEEE Transactions on Control Systems Technology*, 27(2):830–837, 2017.

[153] Chao Yang, Sixiong You, Weida Wang, Liang Li, and Changle Xiang. A stochastic predictive energy management strategy for plug-in hybrid electric vehicles based on fast rolling optimization. *IEEE Transactions on Industrial Electronics*, 67(11):9659–9670, 2019.

[154] Namdoo Kim, Michael Duoba, Namwook Kim, and Aymeric Rousseau. Validating volt phev model with dynamometer test data using autonomie. *SAE International Journal of Passenger Cars-Mechanical Systems*, 6(2013-01-1458): 985–992, 2013.

[155] Jongryeol Jeong, Sungwook Choi, Namdoo Kim, Heeyun Lee, Kevin Stutenberg, and Aymeric Rousseau. Model validation of the chevrolet volt 2016. Technical report, SAE Technical Paper, 2018.

[156] Emmanuel Vinot, Julien Scordia, Rochdi Trigui, Bruno Jeanneret, and Francois Badin. Model simulation, validation and case study of the 2004 ths of toyota

prius. *International Journal of Vehicle Systems Modelling and Testing*, 3(3): 139–167, 2008.

[157] Namwook Kim, Aymeric Rousseau, and Eric Rask. Vehicle-level control analysis of 2010 toyota prius based on test data. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 226(11): 1483–1494, 2012. doi: 10.1177/0954407012445955. URL `https://doi.org/10.1177/0954407012445955`.

[158] Jingxian Hao, Zhuoping Yu, Zhiguo Zhao, Peihong Shen, and Xiaowen Zhan. Optimization of key parameters of energy management strategy for hybrid electric vehicle using direct algorithm. *Energies*, 9(12):997, 2016.

[159] Laura V Pérez, Guillermo R Bossio, Diego Moitre, and Guillermo O García. Optimization of power management in an hybrid electric vehicle using dynamic programming. *Mathematics and Computers in Simulation*, 73(1-4):244–254, 2006.

[160] Morteza Montazeri-Gh, Amir Poursamad, and Babak Ghalichi. Application of genetic algorithm for optimization of control strategy in parallel hybrid electric vehicles. *Journal of the Franklin Institute*, 343(4-5):420–435, 2006.

[161] Daming Zhou, Alexandre Ravey, Ahmed Al-Durra, and Fei Gao. A comparative study of extremum seeking methods applied to online energy management strategy of fuel cell hybrid electric vehicles. *Energy Conversion and Management*, 151:778–790, 2017.

[162] Cristian Musardo, Giorgio Rizzoni, Yann Guezennec, and Benedetto Staccia.

A-ecms: An adaptive algorithm for hybrid electric vehicle energy management. *European Journal of Control*, 11(4-5):509–524, 2005.

[163] Dekun Pei and Michael J Leamy. Dynamic programming-informed equivalent cost minimization control strategies for hybrid-electric vehicles. *Journal of Dynamic Systems, Measurement, and Control*, 135(5), 2013.

[164] J Park and J-H Park. Development of equivalent fuel consumption minimization strategy for hybrid electric vehicles. *International Journal of Automotive Technology*, 13(5):835–843, 2012.

[165] Simona Onori, Lorenzo Serrao, and Giorgio Rizzoni. Adaptive equivalent consumption minimization strategy for hybrid electric vehicles. In *Dynamic Systems and Control Conference*, volume 44175, pages 499–505, 2010.

[166] Xiang Tian, Y. Cai, Xiaodong Sun, Zeping Zhu, and Yiqiang Xu. An adaptive ecms with driving style recognition for energy optimization of parallel hybrid electric buses. *Energy*, 189:116151, 2019.

[167] Chao Sun, Fengchun Sun, and Hongwen He. Investigating adaptive-ecms with velocity forecast ability for hybrid electric vehicles. *Applied Energy*, 185:1644–1653, 2017.

[168] Feng Tianheng, Yang Lin, Gu Qing, Hu Yanqing, Yan Ting, and Yan Bin. A supervisory control strategy for plug-in hybrid electric vehicles based on energy demand prediction and route preview. *IEEE Transactions on Vehicular Technology*, 64(5):1691–1700, 2014.

[169] Bo Gu and Giorgio Rizzoni. An adaptive algorithm for hybrid electric vehicle energy management based on driving pattern recognition. In *ASME International Mechanical Engineering Congress and Exposition*, volume 47683, pages 249–258, 2006.

[170] Yuping Zeng, Jing Sheng, and Ming Li. Adaptive real-time energy management strategy for plug-in hybrid electric vehicle based on simplified-ecms and a novel driving pattern recognition method. *Mathematical Problems in Engineering*, 2018, 2018.

[171] Jinming Liu and Huei Peng. Modeling and control of a power-split hybrid vehicle. *IEEE transactions on control systems technology*, 16(6):1242–1251, 2008.

[172] Gino Paganelli, Sebastien Delprat, Thierry-Marie Guerra, Janette Rimaux, and Jean-Jacques Santin. Equivalent consumption minimization strategy for parallel hybrid powertrains. In *Vehicular Technology Conference. IEEE 55th Vehicular Technology Conference. VTC Spring 2002 (Cat. No. 02CH37367)*, volume 4, pages 2076–2081. IEEE, 2002.

[173] Simona Onori, Lorenzo Serrao, and Giorgio Rizzoni. *Hybrid Electric Vehicles Energy Management Strategies*. Springer, 2016.

[174] Antonio Sciarretta, Michael Back, and Lino Guzzella. Optimal control of parallel hybrid electric vehicles. *IEEE Transactions on control systems technology*, 12(3):352–363, 2004.

[175] Fengqi Zhang, Kanghui Xu, Lin Li, and Reza Langari. Comparative study of

equivalent factor adjustment algorithm for equivalent consumption minimization strategy for hevs. In *2018 IEEE Vehicle Power and Propulsion Conference (VPPC)*, pages 1–7. IEEE, 2018.