Contributions to Passenger and Commercial Hybrid Electric Vehicle Energy Management Control

Aman Ved Kalia

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2020

Reading Committee:

Brian C. Fabien, Chair

Santosh Devasia

Philip C. Malte

Program Authorized to Offer Degree:

Department of Mechanical Engineering

University of Washington

Abstract

Contributions to Passenger and Commercial Hybrid Electric Vehicle Energy Management Control

Aman Ved Kalia

Chair of the Supervisory Committee:

Brian C. Fabien

Mechanical Engineering

Passenger and commercial vehicle powertrain electrification are both a technical and a marketing challenge. A common hurdle experienced by the industry and the consumer is limited driving range and a relatively high cost of purchase. Hybrid electric vehicles present a promising solution to meet the ever so stringent fuel economy or energy consumption requirements, as well as comparable driving range to the conventional gasoline vehicles. Though, a challenge with hybrid electric vehicles is the efficient utilization of the available energy to reduce emissions and pitch it as a cost-effective solution.

An experimental research vehicle based on the 2016 Chevrolet Camaro platform is re-built as a Plug-in Series Hybrid Electric or Extended Range Electric Vehicle. In this dissertation, mathematical modeling and optimization methods are used to develop a power loss model of this experimental research vehicle. The model is able to estimate vehicle energy consumption within a coefficient of variation of 1.9%-7.0%. The model serves as a foundation for the development of a novel energy management op-

timal control algorithm termed as Distance Constrained - Adaptive Real Time Dynamic Programming (DC-ARTDP). Evaluation of the algorithm over different drive conditions shows an improvement of 9.8% in the overall energy consumption of the vehicle while meeting required driving range. The algorithm is also able to provide an optimal energy consumption trajectory under powertrain system fault scenarios and meet the required range demand. The algorithms functionality is evaluated against a predictive energy management approach using Model Predictive Control. The novel algorithm improves overall energy consumption of the experimental research vehicle by 4.25% relative to the model predictive control approach.

To understand the impact of electrified powertrain architectures in the context of commercial vehicles, a two-truck and three-truck platoon power loss modeling environment is developed. The impact of platooning on heterogeneous powertrain architecture trucks with and without the proposed novel algorithm is evaluated. Implementation of this algorithm on a Series Parallel hybrid electric truck shows improvements of $\approx 8\%$ for individual runs. Higher improvement is observed for the Series Parallel hybrid electric truck in a lead position comparative to tail position for two and three-truck platoons.

Table of Contents

Page

List of Figures iv

List of Tables ix

Glossary x

Chapter 1: Introduction 1
1.1 Motivation and Research Goals 1
1.2 Fundamentals: Vehicle Modeling & Energy Management Control 3
1.3 Research Contributions 5
1.4 Dissertation Layout 6

Chapter 2: Multi-Objective Energy Management Optimization of a Plug-In Series Hybrid Electric Vehicle 8
2.1 Abstract 8
2.2 Introduction 9
2.3 Methods 10
2.4 Results and Discussion 27
2.5 Conclusion 37

Chapter 3: On Implementing Optimal Energy Management for EREV using Distance Constrained Adaptive Real-Time Dynamic Programming . 41
3.1 Abstract 41
3.2 Introduction 42
3.3 Methods 45
3.4 Results and Discussion 76

3.5 Conclusion 93

Chapter 4: On Fault Tolerant Behavior of Optimal Energy Management Control for Extended Range Electric Vehicle 95
4.1 Abstract 95
4.2 Introduction 96
4.3 Methods 98
4.4 Results and Discussion 106
4.5 Conclusion 113

Chapter 5: A Comparative Analysis of MPC and DC-ARTDP Algorithms for EREV Optimal Energy Management 115
5.1 Abstract 115
5.2 Introduction 116
5.3 Methods 118
5.4 Results and Discussion 128
5.5 Conclusion 139

Chapter 6: Energy Consumption Modeling and Optimization of Heterogeneous Powertrain Semi-Truck Platoon 141
6.1 Abstract 141
6.2 Introduction 142
6.3 Methods 145
6.4 Results and Discussion 162
6.5 Conclusion 181

Chapter 7: Conclusion 183

Bibliography 186

Appendix A: Algorithm Implementation 195
A.1 SHEV/EREV DC-ARTDP Code 195
A.2 SHEV/EREV MPC Code 210
A.3 SPHET DC-ARTDP Code 217

Appendix B: Modeled System Parameters 236

List of Figures

Figure Number Page

1.1 Force diagram for a 1-DOF vehicle longitudinal dynamics model. 3

2.1 GT Suite estimated Honda VFR800 engine Brake Specific Fuel Consumption map. 16

2.2 Controller-Plant diagram for Energy Management System 21

2.3 Urban Dynamometer Driving Schedule (UDDS) 24

2.4 Highway Fuel Economy drive cycle (HWFET) 24

2.5 Self generated combined drive cycle 25

2.6 Drive route used for the optimization study. 25

2.7 Polynomial fit for a) Energy Consumption and b) Total Fuel Consumption objective functions. 27

2.8 Polynomial fit for a) Energy Consumption and b) Total Fuel Consumption objective functions. 28

2.9 Multi-objective optimization solution plot. 29

2.10 Vehicle speed comparison between the Base case, Optimal case 1, and Optimal case 2. 30

2.11 Battery SOC comparison between the Base case, Optimal case 1, and Optimal case 2. 31

2.12 Vehicle speed and Battery SOC comparison between base and optimal cases for city driving section. 33

2.13 Vehicle speed and Battery SOC comparison between base and optimal cases for highway driving section. 34

2.14 Relative change in emissions for Optimal case 1 and Optimal case 2 as determined from simulation. 35

2.15 Energy consumption and power data for Series PHEV Chevrolet Camaro on UDDS drive cycle. Behavior between CTL A and CTL B genset control strategies is compared and shown as (a) Battery SOC, (b) Battery Power, (c) Generator Power, and (d) DCDC Power. 39
2.16 Criteria tail-pipe emissions data for Series PHEV Chevrolet Camaro on UDDS drive cycle. Behavior between CTL A and CTL B genset control strategies is compared and shown for (a) CO_2, (b) CO, (c) NO_x, and (d) HC. 40
3.1 Experimental research EREV system architecture diagram. 46
3.2 Block diagram representation of the experimental research EREV. 48
3.3 Battery equivalent circuit models evaluated for power flow model. (a) Zeroth, (b) Dual Polarization. 50
3.4 Estimated net fuel consumption comparison to measured net fuel consumption. 53
3.5 Estimated BSFC map for E85 fueled VFR800 engine. 54
3.6 Estimated combined efficiency map for the motor-gearbox assembly. . . . 56
3.7 EREV research vehicle ESS energy consumption comparison for Charge Depleting behavior during EEC Cycle. 57
3.8 EREV research vehicle ESS energy consumption comparison for Charge Sustaining behavior during EEC Cycle. 58
3.9 EREV research vehicle ESS energy consumption comparison for Charge Depleting - CHarge Sustaining behavior during UDDS Cycle. 59
3.10 Final ESS SoC (%) achieved matrix for (a) EEC, (b) HWFET, (c) UDDS, and (d) US06 drive cycle simulations using forward propagating Dynamic Programming. 63
3.11 Net optimal energy consumption (kWh/km) matrix for (a) EEC, (b) HWFET, (c) UDDS, and (d) US06 drive cycle simulations using forward propagating Dynamic Programming. 64
3.12 Maximum achievable driving range (km) matrix for (a) EEC, (b) HWFET, (c) UDDS, and (d) US06 drive cycle simulations using forward propagation Dynamic Programming. 65
3.13 Hamiltonian in the feasible set for ESS only and ESS + REx operation during EEC 3 and 9 cycle optimization taken at $t = 663.0$ seconds, respectively. 66

3.14 ESS SoC and maximum driving range comparison for optimal energy consumption (ESS only) and maximum driving range (ESS + REx) amongst (a) EEC, (b) HWFET, (c) UDDS, and (d) US06 drive cycle simulations using Pontryagin's Minimum Principle. 67
3.15 Energy consumption comparison on a) UDDS cycle with b) Vehicle mass normalized ESS current variation in CD mode between Research Vehicle, 2014 BMW i3 Rex and 2016 Chevrolet Volt. 69
3.16 Dynamic Programming with Range Constraint network flow diagram. . . . 72
3.17 Proposed optimal energy management algorithm flowchart. 74
3.18 Estimated speed trace build using Google Directions API. (a) Shows path nodes from Origin to Destination, (b) Average vehicle speed determined from current traffic conditions, and (c) Randomized speed traces based on the average speed. 75
3.19 Distance constrained dynamic programming optimization results on research vehicle for UDDS Cycle. 77
3.20 Distance constrained dynamic programming optimization results on research vehicle for US06 Cycle. 78
3.21 Comparison of vehicle's performance with and without re-optimization using the randomized low speed drive cycle. 82
3.22 Comparison of vehicle's performance with and without re-optimization using the randomized medium speed drive cycle. 83
3.23 Comparison of vehicle's performance with and without re-optimization using the randomized high speed drive cycle. 84
3.24 Net Energy Consumption comparison of with and without re-optimization scenarios for the three randomized speed traces. 85
3.25 Comparison of the conventional and proposed algorithms for UDDS-HWFET-UDDS-US06 single cycle run. 88
3.26 Comparison of the conventional and proposed algorithms for UDDS-HWFET-UDDS-US06 double cycle run. 89
3.27 Comparison of the conventional and proposed algorithms for UDDS-HWFET-UDDS-US06 triple cycle run. 90
3.28 Net Energy Consumption comparison of conventional and proposed algorithms for the single, double and triple cycles of UDDS-HWFET-UDDS-US06 combination. 91
3.29 Fault insertion behavior comparison of conventional and proposed algorithm for the UDDS-HWFET-UDDS-US06 triple cycle. 92

4.1 EREV research vehicle power flow model schematic. 99
4.2 Fault tolerant DC-ARTDP algorithm execution flow chart. 101
4.3 Fault injection response trace for (a) REx system deration, (b) REx system shutdown, and (c) propulsion motor system deration. 103
4.4 Simulated 163 km round-trip drive with origin at A and turnaround at B. . 104
4.5 Vehicle speed trace for the simulated round-trip. 105
4.6 EREV battery SOC performance with REx deration fault occurring at ~ 17.9 km marker. 107
4.7 EREV battery SOC performance with REx deration fault occurring at ~ 131.6 km marker. 108
4.8 EREV battery SOC performance with motor deration fault occurring at ~ 17.9 km marker. 110
4.9 EREV battery SOC performance with motor deration fault occurring at ~ 131.6 km marker. 111

5.1 EREV power flow model architecture diagram used for simulations. 127
5.2 Evaluation route chosen for EREV energy management study 129
5.3 Vehicle speed trace and road grade targets for the evaluation route 130
5.4 Cruise vehicle speed of 26.82 m/s and road grade variation of $\pm 1\%$. 131
5.5 Effect on P_{gen} with varying MPC horizons for a sinusoidal road grade variation with different horizon windows for the $\sim 5.5 - 7.5$ km section . 132
5.6 Effect on P_{gen} with varying MPC SOC_{ess} penalty p_2 for a sinusoidal road grade variation 133
5.7 SOC_{ess} comparison for different Energy Management Control approaches during evaluation route simulation with MPC SOC penalty $p_2 = 1e6$. 134
5.8 EC_{net} comparison for different Energy Management Control approaches during evaluation route simulation with MPC SOC penalty $p_2 = 1e6$. . . . 135
5.9 SOC_{ess} comparison for different Energy Management Control approaches during evaluation route simulation with MPC SOC penalty $p_2 = 1e8$. 136
5.10 EC_{net} comparison for different Energy Management Control approaches during evaluation route simulation with MPC SOC penalty $p_2 = 1e8$. . . . 137

6.1 Semi-Truck powertrain architecture power flow diagram. (a) Conventional Truck, (b) Series-Parallel (P2-P2) Hybrid Electric Truck, and (c) Battery Electric Truck. 145

6.2 Aerodynamic drag coefficient variation as a function of Inter-Truck distance. (a) Two-Truck Platoon, (b) Lead Truck in a Three-Truck Platoon, (c) Middle Truck in a Three-Truck Platoon, and (d) Tail Truck in a Three-truck Platoon. 148
6.3 P2-P2 hybrid gear planetary gear system diagram modeled for Series Parallel Hybrid Electric Truck. 153
6.4 Vehicle speed response of different semi-truck powertrain architectures on NREL Metro Highway Cycle. 159
6.5 (a) Highway Step and (b) Portland - Hood River drive cycle speed trace and road grade used for simulations. 161
6.6 Google Map route trace of the Portland,OR to Hood River,OR Route Cycle. 161
6.7 Semi-truck speed response on the custom Highway Step Cycle. (a) Overview, (b) Initial Acceleration, (c) Acceleration over $\pm$3% road grade, and (d) Deceleration over $\pm$6% road grade. 164
6.8 Battery SOC comparison for SPHET architecture over PH Route with DC-ARTDP and Rule Based Energy Management Algorithms. 165
6.9 Semi-truck speed response on the real-world Portland,OR to Hood River,OR Route. (a) Overview, (b) Initial Acceleration, and (c) Acceleration over positive road grade. 166
6.10 Energy Consumption improvement comparison between heterogeneous two truck platoon configurations simulated over PH Route for target separation distances of (a) 15m, (b) 20m, (c) 30m, and (d) 40m. 169
6.11 Distance to Empty improvement comparison between heterogeneous two truck platoon configurations simulated over PH Route for target separation distances of (a) 15m, (b) 20m, (c) 30m, and (d) 40m. 170
6.12 Energy Consumption improvement comparison between heterogeneous three truck platoon configurations simulated over PH Route for target separation distances of (a) 15m, (b) 20m, (c) 30m, and (d) 40m. 174
6.13 Distance to Empty improvement comparison between heterogeneous three truck platoon configurations simulated over PH Route for target separation distances of (a) 15m, (b) 20m, (c) 30m, and (d) 40m. 175
6.14 Energy consumption improvement comparison for two-truck platoon configurations CS and SC at (a) 15m, (b) 20m, (c) 30m and, (d) 40m target separation distance with DC-ARTDP optimal energy management. 179
6.15 Energy consumption improvement comparison for two-truck platoon configurations CBS and SBC at (a) 15m, (b) 20m, (c) 30m and, (d) 40m target separation distance with DC-ARTDP optimal energy management. . 180

List of Tables

Table Number Page

2.1 Plug-in SHEV powertrain component parameters 12
2.2 Estimated scaling factors for tail-pipe emissions. 17
2.3 Simulation Parameters 19
2.4 Multi-Objective Optimization constraints. 23
2.5 Coefficient values for polynomial fit of Energy Consumption (EC) and Total Fuel (TF) used. 26

3.1 Experimental research EREV system parameters. 47
3.2 Molar masses for key E85 combustion reactants and products. 52
3.3 Number of simulated cycles for Initial ESS SOC and Driving Range combinations. 61

6.1 Power Loss Model Component Parameter Table 146
6.2 Semi-Truck evaluation parameters. 162
6.3 Energy Consumption and DTE Results for Individual Truck Simulations on HS Cycle. 163
6.4 Energy Consumption and DTE Results for Individual Truck Simulations on PH Route Cycle. 165
6.5 Two-Truck Platoon configurations evaluated on PH Route Cycle. 168
6.6 Three-Truck Platoon configurations evaluated on PH Route Cycle. 173

B.1 Series Hybrid Electric Vehicle Power Loss Model Parameters 237
B.2 Conventional Truck Power Loss Model Parameters 238
B.3 Series Parallel Hybrid Electric Truck Power Loss Model Parameters 239
B.4 Battery Electric Truck Power Loss Model Parameters 240

Glossary

1DOF: One Degree of Freedom.

15S3P: Cell configuration inside the experimental research vehicle module. 15 cells in series and 3 parallel rows of those 15 cells.

API: Application Programming Interface.

BEV: Battery Electric Vehicle.

BSFC: Brake Specific Fuel Consumption.

CD: Charge Depleting.

CS: Charge Sustaining.

DC-ARTDP: Distance Constrained Adaptive Real Time Dynamic Programming.

DP: Dynamic Programming.

DTE: Distance to Empty.

E85: 85% Ethanol and 15% Gasoline fuel blend.

EC: Energy Consumption.

ECMS: Energy Consumption Minimization Strategy.

ECOCAR: Competition series sponsored by US Department of Energy and General Motors.

ECU: Electronic Control Unit.

EEC: Emissions and Energy Consumption.

EMRAX: Brand name of the propulsion motors used in experimental research vehicle.

EREV: Extended Range Electric Vehicle.

ESS: Energy Storage System.

ETE: Emissions Testing Event.

EV: Electric Vehicle.

HEV: Hybrid Electric Vehicle.

HWFET: Highway Fuel Economy Testing.

PMP: Pontryagin's Minimization Principle.

REX: Range Extender.

SHEV: Series Hybrid Electric Vehicle.

SOC: State of Charge.

UDDS: Urban Dynamometer Driving Schedule.

UHC: Un-burnt Hydrocarbons.

US06: A supplemental federal test procedure cycle.

"Keep Calm and Cycle 12 Volts"

powerful words that helped bring our experimental research vehicle to life.

(UW EcoCAR Lab)

Acknowledgments

I would like to express the deepest gratitude to my advisor Prof. Brian Fabien. This doctoral research has come to a fruition with his guidance and support. My research and work ethics have improved working with him as a Graduate Research and Teaching Assistant at University of Washington, Seattle.

I would like to extend thanks to my committee members Prof. Santosh Devasia, Prof. Philip Malte, and Prof. Radha Poovendran who provided crucial feedback and guidance on my work. Their expertise and insight helped shape my work for its maximum impact to the domain of Hybrid Electric Vehicle Energy Management. A sincere thanks to the faculty and staff in the Department of Mechanical Engineering at University of Washington, Seattle. The access to world renowned academicians and professional support during my research work was unparalleled.

A huge appreciation for the Advanced Vehicle Technology Competitions (AVTC) and their EcoCAR series sponsored by U.S. Department of Energy, General Motors, MathWorks and manageed by Argonne National Laboratory. The inception of this doctoral research topic wouldn't have been feasible without this program and participation by the University of Washington EcoCAR Team. Thank you to all the organizers and sponsors of the EcoCAR competition. Thank you to all the undergraduate and graduate University of Washington EcoCAR team members during EcoCAR 3 and EcoCAR Mobility Challenge competitions.

Thank you to my industry mentors Michael Abowd, Kevin Oshiro, Radu Oprea, Andre Strobel, Lilla Smith and Bryan Ross. The guidance and support provided by my mentors in the industry through EcoCAR competition and internships helped focus my research work contributions for real-world problems. Thank you to fellow research group graduate students for healthy discussions on research problems and their applications.

Thank you to my friends in Seattle for making this long graduate student journey feasible away from home. A special thanks to Rishi Pahuja and Gaurav Mukherjee, for sharing this graduate student journey alongside and great resource for both professional and friendly advice.

Last but not the least, I am grateful to my parents for their unwavering support, love and providing me with the liberty to pursue my academic goals, 7,000 miles away from them.

Dedication

To my parents,
Dr. Pritam Kalia and Dr. (Mrs.) Deepak Kalia
and my grandparents,
Late Shri. Shyam Lal Kalia and Late Smt. Saraswati Devi Kalia
and
Late Shri. Nitya Nand Sharma and Late Smt. Santosh Sharma.

Chapter 1

Introduction

As of year 2020, passenger and commercial road vehicles with partially or completely electrified powertrains are a reality. This transition though slow in the past decade, is projected to grow exponentially in the next decade and beyond. The rate of increase in passenger vehicles is noted to be higher than commercial vehicles. Among the electrified powertrain vehicles, Hybrid Electric Vehicles (HEVs) and Battery Electric Vehicles (BEVs) are the two major classifications. Primarily motivated with the intent to reduce greenhouse gas (GHG) emissions from vehicles, the journey towards completely electrified vehicles or BEVs is a long one. The major challenges in this path are: (a) the purchase cost to the consumer, and (b) obtaining a competitive driving range or distance-to-empty (DTE) to the fuel powered conventional vehicles. While research towards developing cost-effective, high energy density electrified energy storage systems progresses, inclusion of HEVs is a promising step towards this transition.

1.1 Motivation and Research Goals

Passenger and Commercial vehicle electrification is a challenging problem, both in terms of technical and marketing aspects. As of today, the cost associated with this is substantial, thus posing a hurdle in its acceptance by a majority of the consumers. Hybrid Electric Vehicles or HEVs are a promising bridge in this transition towards complete electrification. Through the U.S. Department of Energy and General Motors sponsored

multi-year competition EcoCAR, I had the opportunity to work on design, build and optimization of 2016 Chevrolet Camaro Plug-In Series Hybrid Electric/ Extended Range Electric Vehicle. This challenge showed the level of complexity an HEV's energy system has and was the key motivation to develop and evaluate energy management control strategies that can improve the vehicle's overall energy consumption while ensuring the vehicle can meet the desired driving tasks. During this period of research, I also had the opportunity to learn about commercial vehicle energy management concerns and methods to evaluate these concerns in case of heavy-duty semi-truck platooning. This experience, motivated to expand my research on hybrid electric vehicle energy management optimization into the commercial vehicle domain. Finally, a common theme to combine these two motivations was to develop an energy management strategy that is real-time implementable on a standard vehicle electronic control unit (ECU).

Hence, based on these motivations the following research goals were drawn to fulfill the requirements of a doctoral research.

1. The first research goal, was to develop and validate mathematical models representing the experimental research vehicle using basic principles, collected test data and supplier provided data sheets. As access to vehicle is limited, this goal was to setup the foundation for energy management control and optimization evaluations.

2. The second research goal, was to understand the behavior of state of the art energy management algorithms on the research vehicle and develop a novel energy management optimal control algorithm. The novel algorithm was to be real-time implementable and cost-effective in the sense, that a standard automotive ECU can be used as the operation platform.

3. The third research goal, was to extend this algorithm to hybrid electric commercial

vehicles, such as a Class 8 Long-Haul 18-wheeler tractor-trailer combination truck. Additionally, the intent was to understand the behavior of such trucks in a platoon of trucks with entirely different powertrain architectures.

This dissertation focuses on these three goals and showcases the methods used and results obtained to achieve these research goals.

1.2 Fundamentals: Vehicle Modeling & Energy Management Control

Mathematical modeling of a vehicle's longitudinal behavior i.e. motion along a straight line, is derived from the fundamentals of evaluating forces occurring on a block of mass under the influence of an external force. To include the forces on the vehicle due to road inclination, a one degree of freedom (1DOF) longitudinal vehicle dynamics model is generally considered as shown in Figure 1.1. This model diagram shows the different forces acting on a vehicle driving on an incline. The net resistive force determines the propulsion force required by the vehicle's prime mover to generate and overcome resistance to motion. This force can be used to determine the amount of power required and upon summation the total energy required. This 1DOF model serves as a fundamental building block for the power loss models developed and discussed in this dissertation.

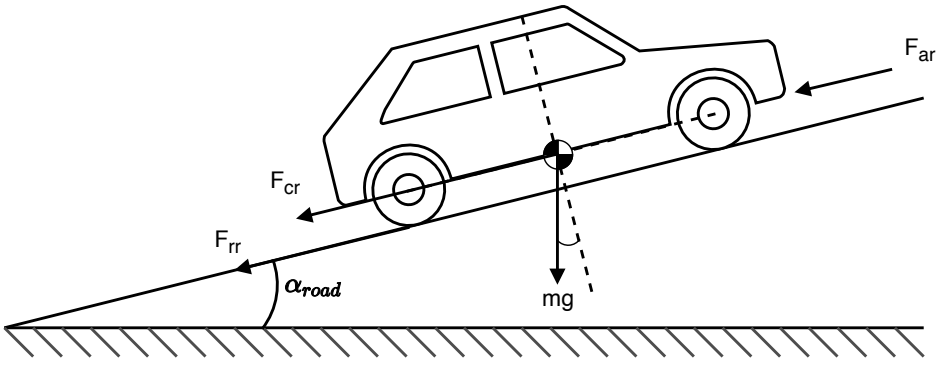


Figure 1.1: Force diagram for a 1-DOF vehicle longitudinal dynamics model.

The force or power demand determined from the 1DOF model has to be fulfilled by the energy sources being used by the prime mover on a vehicle. The term *prime mover* refers to either an internal combustion engine or an electric motor that provides propulsive force at the wheels. The power flow path from prime mover to the wheels of a vehicle converts a significant amount into losses which are dissipated as heat or noise. Thus, these expected losses from the different powertrain components in power flow path have to be added to the power demand calculated by the 1DOF model. The resulting power demand is used to determine the amount of energy source depletion or recuperation feasible during the vehicle's desired motion. In addition to these losses, vehicle's have auxiliary power demands which do not assist in propulsion but are responsible for vehicle's proper functionality or user's demands. These power requests are significant when operating low on available energy and thus are crucial to be added to the net power demand. Hence, in a power loss model, presented in this dissertation, this fundamental approach of modeling is pursued. System level power loss models of powertrain components are described in detail in the respective Chapters.

Now, that the net power demand is determined through vehicle modeling, the next step is to determine strategies that utilize the energy source efficiently while meeting the required power demand. This process is called as Energy Management. For a conventional gasoline vehicle, this process focuses on determining the most efficient operating point of the engine. This implies determining the engine torque and speed value that uses the least amount of fuel to meet the demand. In battery electric vehicles, this process focuses on determining the motor torque and speed value that operates the motor with least amount of losses while meeting the power demand. In addition to that, it also determines the amount of regenerative braking power that can be recuperated during braking scenarios.

The process of energy management is immensely crucial for hybrid electric vehicles due to them having more than one sources of energy to meet the power demand. The

complexity further increases as hybrid vehicles are available in various powertrain configurations, each with their own constraints and operational domain. Irrespective of the powertrain configuration, the main goal is to minimize overall energy consumption by the vehicle while meeting the requested power demand. Due, to more than one sources of energy, the process of energy management is challenging as the functional feasibility varies with operating conditions and parameters for a vehicle. Various offline and online optimization methods are used to determine the most efficient operating strategies for such vehicles. These methods are described and further discussed in Chapters as they become more relevant to the problem.

1.3 Research Contributions

The core contributions made to hybrid electric vehicle energy management research domain by the author, in the descending order of impact are listed as follows.

1. **Distance Constrained - Adaptive Real Time Dynamic Programming Energy Management Algorithm**: During this research work, a novel energy management algorithm based on the concept of *Forward Propagating Dynamic Programming* was developed for the research vehicle used in the study. The novelty of this algorithm comes from the fact that it optimizes for energy consumption minimization while ensuring that the desired target distance is achieved. The algorithm also is capable of adapting to unknown and unmeasured disturbances during a drive such as vehicle speed variations, road grade variation, system faults and route changes. Real-time, in-vehicle application of this algorithm was determined to be feasible without extreme computational capabilities. Additionally, the algorithm allows human interaction as the user can override target distance requirement based on personal preference or future driving requirements. The algorithm is first introduced in Chapter 2 and is then evaluated over its various aspects and capabilities through Chapter 6.

2. **Real-World Drive Cycle Generator**: To ensure that the research vehicle and the energy management algorithms were being evaluated for realistic real-world scenarios, it was crucial to obtain road traffic and grade information. Additionally, the novel algorithm developed in this research is operable only if prior estimates of speed and road elevation are available. Thus, leveraging the Google API for Directions and Elevation, a *Real-World Drive Cycle Generator* was created. This tool takes in user's origin and destination information and computes the estimated speed and elevation map for the drive. This information is crucial in estimating the energy consumption by the vehicle. The tool also shows the potential of integrating on-board navigation systems with the novel algorithm for in-vehicle implementation of the energy management algorithm.

3. **Power Loss Models**: Though the concept of power loss modeling has existed previously, during this research frameworks to simulate energy consumption by a passenger and a commercial vehicle platoon were created. A library of model subsystems was also created for future development of models or modeling frameworks. These include custom functions that improve simulation run-time and are intuitive to use. This contribution enables future researchers in this domain to leverage the model framework and adapt it to their specific research requirements.

1.4 Dissertation Layout

The research conducted in the domain of passenger and commercial vehicle energy consumption modeling and optimization is presented as a compendium of published, submitted and in-preparation conference and journal publications. Beginning with Chapter 2, the initial approach to model energy consumption for the experimental research hybrid Chevrolet Camaro is presented. Significant assumptions pertaining to system efficiencies and behavior are made in accordance with published literature. In the presented work, a Multi-Objective Optimization approach is considered to deter-

mine optimal operating points for the on-board generator system. Limitations of the work were observed upon comparison with in-vehicle testing results.

To improve upon the shortcomings of the research vehicle model in Chapter 2, such as effectiveness and run-time, the modeling approach was changed to that of a power loss model and system parameters were identified from actual data. In Chapter 3, this approach is presented and the model is validated. The core premise of this Chapter though lies in a novel energy management optimization algorithm termed *Distance Constrained - Adaptive Real Time Dynamic Programming*. The algorithm is presented in context of the research vehicle and compared against state-of-the-art optimization algorithms. Chapter 4 evaluates the fault-tolerant capabilities of the novel algorithm in context of the research vehicle. Faults are introduced in key powertrain sub-systems and the algorithm's functionality is compared against a more standard Charge Depleting - Charge Sustaining algorithm. To further understand the capabilities of the novel algorithm in context of road elevation variations, a comparative study between the novel algorithm and more prevalent Model Predictive Control algorithm is presented in Chapter 5. The evaluation is simulated in a real-world scenario and shows the capabilities of the algorithm over Model Predictive Control. Based on the observed capabilities of the novel algorithm in context of a passenger vehicle, it's capabilities were evaluated for a Series Parallel Hybrid Electric Semi-Truck in Chapter 6. This chapter focuses on effectiveness of the algorithm over more standard approaches for the hybrid truck for a solo run and in a platoon of trucks.

Chapter 7 summarizes the outcomes and contributions of the research projects conducted and their alignment with the research goals presented in the beginning of this chapter.

Chapter 2

Multi-Objective Energy Management Optimization of a Plug-In Series Hybrid Electric Vehicle

Publication: *A. V. Kalia and B. C. Fabien, "Development of Optimal Control Strategy for a Plug-In Series Hybrid Electric Vehicle With an On-Board Engine-Generator System for Overall Fuel Economy Improvement and Reduction in Tail-Pipe Emissions," presented at the ASME 2017 Internal Combustion Engine Division Fall Technical Conference, Oct. 2017, p. V002T05A002-V002T05A002, doi: 10.1115/ICEF2017-3526.*

2.1 Abstract

This research study focuses on determining optimal points of operation for the engine-generator system and regenerative braking at the wheels in a plug-in series hybrid electric Chevrolet Camaro. The goal is to improve overall fuel economy of the vehicle as well as reducing overall tail-pipe emissions. An abstract mathematical model of the series hybrid electric Chevrolet Camaro is being used to simulate the overall energy consumption of the vehicle. Previously tested and published control algorithms and strategies are studied, discussed and a viable scheme is chosen for optimization. The results from the optimal strategy considered are compared against the un-optimized results. An im-

provement of ∼8.9% in fuel economy and ∼8.2% reduction in tail-pipe emissions is estimated.

2.2 Introduction

Hybrid electric vehicles (HEVs) have started gaining some traction in the market and amongst consumers. With ride sharing services on the high and tax benefits, consumers prefer a hybrid vehicle over a conventional gasoline powered vehicle. HEV sales in the US show a significant rise in the past two decades [7]. HEVs employ different system architectures based on the power flow from energy storage system to different propulsive components. The series hybrid electric vehicle (SHEV) architecture with an on-board engine-generator system is currently least adopted in production hybrid vehicles. Unlike conventional HEVs, SHEVs operate as a battery electric vehicle (BEV) in the charge depleting mode and switch into a hybrid mode in the charge sustaining mode. The on-board engine generator system operates with the sole objective to sustain the high voltage battery state of charge (SOC) and thereby extend the range of the vehicle.

Application based studies of on-board electrical generator systems have been conducted to determine the functionality and effectiveness. Capaldi et al. formed a 10kW power output electrical generator system for their study. They determined that though their system was efficient for its single cylinder engine architecture it cannot extend range [17]. To improve the electrical generator system behavior numerous studies targeting optimization of the energy flow in a series hybrid architecture have been published [39, 16, 54, 27, 24, 59, 48]. Kaneko et al. in their research determined the impact of variable and constant control approaches for the on-board electrical generator system. The variable control approach they presented varies power output as per the requested power at the accelerator pedal. The constant control approach operates the electrical generator system at a constant power output set point [39] . Their research determined that the constant control approach reduces engine loss but adversely affects battery per-

formance. These results though could not be generalized for all vehicles due to their dependency on the engine and battery loss characteristics of the vehicle under study. Advanced optimization studies have been presented by Styler et al. with their learned optimal control strategy [63], Herrera et al. with their rule based control in conjunction with genetic algorithm [31], Menezes et al. with the selective evolutionary generation system [48] and Serrao et al. with their work on comparing Dynamic Programming, Pontryagin's Minimum Principle and Equivalent Consumption Minimization Strategy[59].

This paper focuses on determining optimal operation set point for the series hybrid vehicle architecture considered for the study. Detailed description of the abstract mathematical model used for simulating longitudinal dynamics of the vehicle is presented. The model is based on previously published and validated models as shown by the work of Herrera et al. and Sampathnarayanan et al. [57]. Further, an assessment of previously published control strategies is made and a suitable strategy is employed and tested with the model. Results of this study and a discussion of their impact is also provided.

2.3 Methods

2.3.1 System Description

The vehicle considered for this study has a Plug-in series hybrid electric vehicle architecture with an on-board engine generator system. This architecture as shown in Figure 3.1 in Chapter 3, is presently implemented on a 2016 Chevrolet Camaro by the University of Washington EcoCAR 3 team. It comprises of three major sub-systems a) the energy storage system, b) the propulsion system, and c) the generator system. The energy storage system comprises of a seven module 15s3p architecture A123 high voltage battery pack. A minimum total of 18.9 kWh of energy is provided through this energy storage system. The propulsion system uses two EMRAX axial flux permanent magnet motors driving the rear wheels of the vehicle. Each motor drives the individual rear wheel. A custom built single speed planetary gearbox with a 4.2:1 reduction assists in delivering

upto 2100 Nm of peak torque to each wheel. Finally, the electrical generator system uses a 0.8L V4 Honda VFR Interceptor engine directly coupled through the stock gear train to 180 kW Bosch synchronous motor generator. The engine is fuelled by E85 instead of regular gasoline from a 7-gallon tank. The 3-phase current generated by the synchronous motor generator is converted into direct current by an inverter converter module. This power generated is primarily used to sustain the charge of the energy storage system. It also assists in supplementing power to the propulsion system during excess power demand situations.

Table 2.1 provides additional information about the modelled components from the described architecture. The vehicle is intended to drive in the charge depleting (CD) mode until the state of charge (SOC) of the battery pack goes below the set limit. After that limit, the supervisory controller requests the on-board engine-generator system to operate at a commanded power set point. This is the charge sustaining (CS) mode which helps in increasing the range of the vehicle. This study at present focuses on a single set point operation of the electrical generator system. This paper uses a mathematical model of the vehicle described to simulate various drive cycles to estimate performance and apply optimization strategy for improvement.

2.3.2 Mathematical Model

The full vehicle mathematical model comprises of sub models for the different components listed previously. These include the energy storage system, the propulsion system and the generator system. A longitudinal vehicle model is formulated using the road load equation to determine the amount of power required for desired vehicle propulsion. Equation(2.1) depicts the road load equation,

$$F_{veh} = F_{trac} - F_{ar} - F_{rr} - F_{cr} \tag{2.1}$$

where, F_{ar} is aerodynamic resistance force, F_{rr} is rolling resistance force, F_{cr} is climbing resistance force, F_{trac} is traction force at the rear wheels and F_{veh} is the net propulsive

Table 2.1: Plug-in SHEV powertrain component parameters

Parameter	**Value**	**Units**
Traction Motor Torque, peak	500	Nm
Traction Motor Torque, continuous	250	Nm
Traction Motor Efficiency	$93-98$	%
A123 Battery Pack Voltage, nominal	340	V
A123 Battery Pack Capacity, minimum	18.9	kWh
A123 Battery Pack Discharge Current, peak	612	A
A123 Battery Pack Discharge Current, continuous	180	A
A123 Battery Pack Charge Current, peak	300	A
A123 Battery Pack Charge Current, continuous	60	A
Engine Power, peak	78	kW
Engine Efficiency *	$35-40$	%
Synchronous Motor Generator Torque, peak	200	Nm
Synchronous Motor Generator Torque, continuous	95	Nm
Synchronous Motor Generator Efficiency *	$83-95$	%

* Estimated value.

force on the vehicle. The values of F_{ar}, F_{rr}, F_{cr} are pre-computed using the data from the drive trace being simulated. The values of these resistive forces show minor variations once actual velocity is computed. The mathematical relations for these resistive forces are given in Eqs.(2.2 - 2.4) with v_{veh} considered in mi/h.

$$F_{ar} = 0.0386\rho_{air}C_dA_fv_{veh}^2 \tag{2.2}$$

$$F_{rr} = C_{rr}m_{tot}g \tag{2.3}$$

where,

$$C_{rr} \;=\; 0.005 + \frac{1}{p_{tire}}\Big(0.01 + 0.0095\Big(\frac{v_{veh}}{100}\Big)^2\Big)$$

,and

$$F_{cr} \;=\; m_{tot}gsin(atan(z)) \;\approx\; m_{tot}gsin(z) \tag{2.4}$$

The overall mass of the vehicle (m_{tot}) is computed as the sum of the kerb weight (m_kerb), net mass of the passengers (m_{pas}) and the net inertial mass of all rotary components (m_i). A total of 4 passengers are considered for simulation purposes. The inertial mass (m_i) was computed as per Eq.(2.5) which only considers wheels, axles and motors as the major rotary elements in the vehicle. The engine and generator were not considered as they were functional only during the CS mode which does not have a significant impact on the overall vehicle performance. The efficiency of planetary gear system (η_{gear}) is considered as a constant value of 90% in the model. This value was computed as an average from the preliminary data collected with the vehicle running in electric mode. Reduction ratio of the gear (i_{gear}) was 4.2 as per the design parameters of the gearbox.

$$m_i \;=\; I_{wh}\left(\frac{1}{r_{wh}}\right)^2 + I_{mot}\eta_{gear}\left(\frac{i_{gear}}{r_{wh}}\right)^2 \tag{2.5}$$

A pre-computation of the required tractive force (F_{trac}) at the wheels is done using the road load equation and speed values from the drive trace. The F_{trac} values are then used in the traction motor model to determine torque request at the motors and by extension the electric power request to run the motors. As F_{trac} is for the entire vehicle, then the torque for a single motor can be computed by considering equal split between the wheels. This model is only longitudinal hence torque vectoring is not considered. Thus, the required torque at one motor can be obtained using Eq.(2.6) and the power request from the motor from Eq.(2.7).

$$\tau_{mot_{req}} \;=\; 0.5F_{trac}r_{wh}\left(\frac{1}{i_{gear}}\right) \tag{2.6}$$

$$P_{mot_{req}} = \tau_{mot_{req}} \omega_{mot} \left(\frac{1}{\eta_{mot} \eta_{gear}} \right) \tag{2.7}$$

The traction motor model also includes regeneration capabilities during coasting and braking events. These are included in the model to adjust the torque request at the motors with regeneration considered. The algorithm applied for regeneration events in the base model is provided in Algorithm 1.

Algorithm 1: Coasting regen and Braking regen algorithm.

Input: a_{veh}, K_{CR}, K_{BR}, τ_{mot}, τ^{p}_{mot}

Output: τ^{*}_{mot}

while $\tau_{mot} \leq 0$ **do**

  if $a_{veh} \geq -0.268 m/s^2$ **then**

    $\tau^{*}_{mot} = -K_{CR} * \tau^{p}_{mot}$;

  else if $a_{veh} \geq -0.5 m/s^2$ ***and*** $a_{veh} \leq -0.268 m/s^2$ **then**

    $\tau^{*}_{mot} = -K_{BR} * \tau^{p}_{mot}$;

  end if

end

The K_{CR} and K_{BR} values are variable and can be adjusted as required. In the event of regeneration being disabled, the torque request to the propulsion system is only dependent on the road load equation. A speed correction is also added to the model to prevent excessive speed reduction due to regeneration. The actual motor torque computed for the drive trace is used to determine the net propulsion power demand ($P_{mot-elec}$). In addition, auxiliary power demand ($P_{aux-elec}$) from accessory electrical components in the vehicle is assumed to be a constant 700 W. This value is based on the power demand from the DC-DC convertor that sustains the low voltage supply. Thus, net power demand from the energy storage system (P_{ess}) is obtained as shown in Eq.(2.8).

$$P_{ess} = P_{mot-elec} + P_{aux-elec} \tag{2.8}$$

This power demand continuously depletes the high voltage energy storage system. The

state of charge value of the energy storage system reduces overtime and is determined by Eq.(2.9). This relation is obtained from the fact that state of charge at any point of time is the ratio of change in charge over the total charge of the energy storage system. A time derivative of state of charge will provide a relation to determine variation of SOC with current as a function of time i.e.

$$\begin{aligned} SOC(t) &= \frac{Q(t) - Q_0}{Q_0} \\ \frac{dSOC}{dt} &= \frac{1}{Q_0}\frac{dQ}{dt} \\ Q &= I(t) \cdot t \\ \frac{dQ}{dt} &= I(t) \\ \frac{dSOC}{dt} &= \frac{I(t)}{Q_0} \end{aligned} \tag{2.9}$$

The high voltage energy storage system is modelled as a voltage source [73]. It comprises of an open circuit voltage (V_{OC}) with a resistance (r_{ess}) in series. Equating power across this modelled circuit provides Equation(2.10) which is used to determine the current flow in and out of the energy storage system. With the reduction in SOC over time, the net voltage across the energy storage system sags. Equation(2.11) shows a linear relation between V_{OC} and SOC. Complete depletion of the energy storage system is counter-productive and thus an on-board generator system is modelled to sustain its charge. The power requested from the on-board generator (P_{gen}) is configurable and modelled as a single set point. Functional limits pertaining to the performance of energy storage system are also added to the model [31].

$$I(t) \;=\; \frac{1}{2r_{ess}}\Big[V_{OC}(SOC) - \sqrt{V_{OC}(SOC)^2 - 4r_{ess}P_{ess}(t)}\Big] \tag{2.10}$$

where,

$$V_{OC}(SOC) \;=\; A \cdot SOC(t) + B \tag{2.11}$$

The genset system is represented as a power demand (P_{gen}) requested by the supervisory controller. The possible operating points are determined based on the BSFC map generated by Wittenbecher et al. overlaid with engine and generator motor power curves [74]. The generator is operated at a single power set point instead of load following for this study. The BSFC map also assists in estimating fuel flow rate at the chosen electrical generator operation set point. Equation(2.12) shows mass flow rate of fuel is determined using the BSFC map.

$$\dot{m}_{fuel} = \frac{P_{gen} \times BSFC}{3600} \tag{2.12}$$

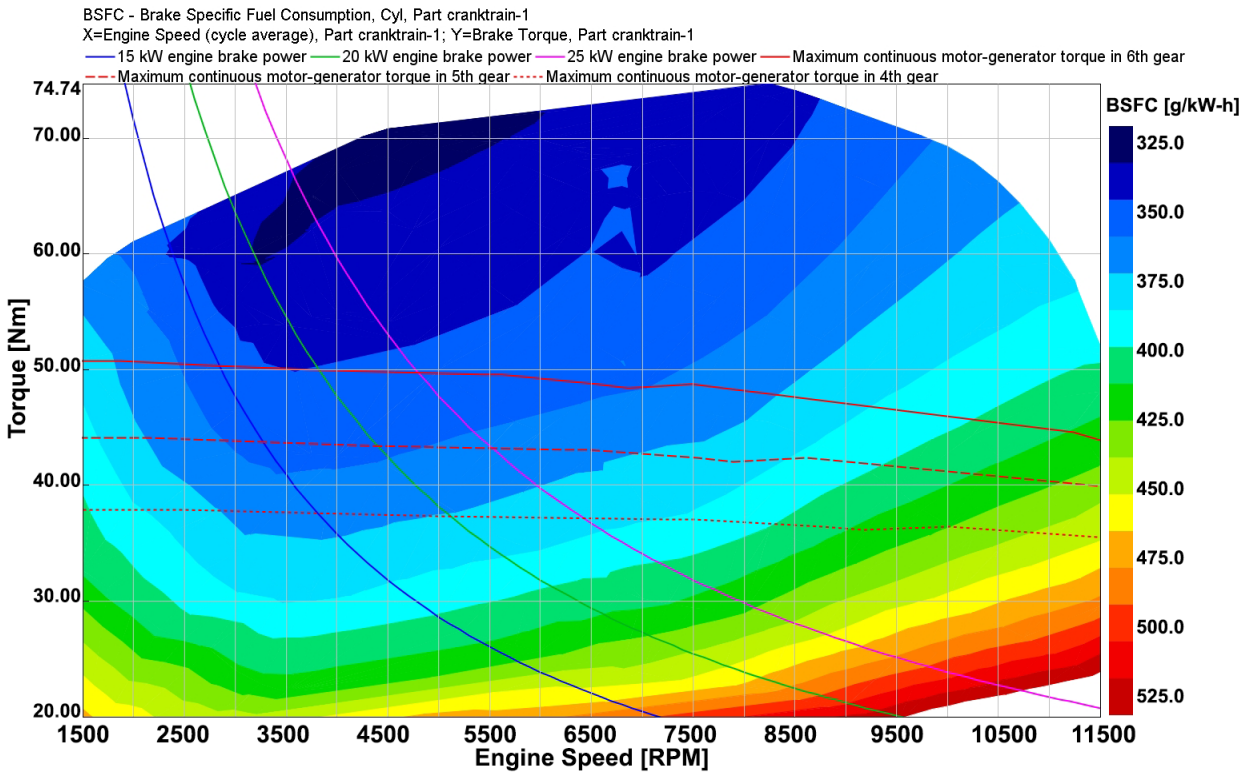


Figure 2.1: GT Suite estimated Honda VFR800 engine Brake Specific Fuel Consumption map.

Tail pipe emissions from E85 combustion in spark-ignition engines comprise of the

usual components CO_2, UHC, NO and CO. In addition to these, certain carbonyl compounds are also found in emissions which are unique to oxy-hydrocarbon fuels. Results compiled from recorded data suggest that these carbonyl compounds do form a considerable component of the total emissions [23, 21, 41]. Due to unavailability of recorded tail-pipe emission data from engine modeled for the study, data from referenced papers is used to determine the fraction these emissions are of the fuel burnt. For this model carbonyl compounds are not considered a component of the emissions. As the engine is operated at a fixed set-point load, cruise mode of engine operation is used as a basis to determine these fractions. Frey et. al. in their study determined mean emission rates for an E85 fueled vehicle. The data, normalized with mass flow rate of fuel is used to obtain the required fraction for the model[23] . These fractions are provided in Table 2.2 for reference.

Table 2.2: Estimated scaling factors for tail-pipe emissions.

Pollutant	**Scaling Factor** $\left[\frac{g}{g_{fuel}}\right]$
CO_2	2.05
CO	5.2e-3
NO	6.6e-4
UHC	1.88e-4

Tail-pipe emissions depend on several parameters and operation conditions. As a three-way catalyst is used downstream of the engine in the exhaust line cold start can lead to higher emission values. To model this effect, temperature variation in the exhaust with its impact on emissions determined from published work is used in the model. Additionally, spark timing can also affect emissions but is out of scope for this study as spark timing is not varied. Finally, the total energy consumption for the simulated vehicle is computed using the energy consumption in both CD and CS modes. A

utility factor is computed based on the relation provided in SAE J2841 per Eq.(2.14). This relation is obtained based on the CD range of the vehicle which is computed through the simulation. Hence, the total energy consumption for the vehicle is determined from the Eq.(2.13) and used for comparison in the results section.

$$EC_{combined} = UF \cdot EC_{CD} + (1 - UF) \cdot EC_{CS} \tag{2.13}$$

$$UF = \frac{\Sigma_{k=1}^{N} min(d(k), R_{CD})}{\Sigma_{k=1}^{N} d(k)} \tag{2.14}$$

With the mathematical model developed, the next stage is to determine the cost function associated with the model for optimization.

2.3.3 Evaluation Setup

The three drive traces considered for the study include: UDDS, HWFET and a combination of UDDS and HWFET speed traces overlaid on a city and expressway grade data. The model was simulated with base parameters on all the three drive traces to ascertain proper functionality. For this paper, the combined drive trace with road grade was considered for optimization. The simulated drive trace is shown in Figure 2.5. A small set of configurable parameters are kept unchanged for the simulations to provide some consistency in base functionality of the modeled system. These parameters are populated in Table 2.3 and based on the requirements pre-defined for the system. The optimization strategies were then implemented using the cost function obtained in the previous section. The results obtained upon simulating the model with optimized parameters were then compared with the baseline results and among themselves. These results are discussed at length in the next section of the paper.

2.3.4 Energy Management Control Strategy

Hybrid electric vehicles like gasoline powered vehicles require optimization of the powertrain components to attain the maximum possible efficiency. In case of HEVs the cost

Table 2.3: Simulation Parameters

Parameter	**Value**	**Units**
Number of drive cycle repetitions	5	-
Vehicle kerb weight	1825	kg
Initial SOC	100	%
SOC Lower Limit, Charge Sustaining	30	%
SOC Upper Limit, Charge Sustaining	35	%
Speed Band, Coast Regen	± 1.5	km/h
Speed Band, Braking Regen	± 3	km/h

function obtained for optimization is more complex due to significantly more power-train components which independently impact the functionality of the vehicle. Several control optimization strategies have been developed, tested and implemented on the vehicles with different architectures. Broadly these strategies are grouped into,

1. *Numerical strategies,* where techniques like dynamic programming (DP) and genetic algorithm are used to map out all the possible operating points for the entire drive cycle. These techniques are perfect for simulation as prior knowledge of the drive cycle exists.

2. *Analytical strategies,* also consider the entire drive cycle but are implemented using a simplified analytical problem. Pontryagin's minimum principle (PMP) employs an analytical approach for optimization.

3. *Minimization strategies*, like the equivalent consumption minimization strategy (ECMS) are perfect for real time implementation for it involves minimization at each time step of the considered drive.

4. *Heuristic strategies*, employ usage of rules and algorithms determined from functionality constraints of the components. These strategies are highly computation efficient but are limited in exploiting the full potential of the system for optimization. cycle.

Substantial work involving HEV optimization have used one or more of these optimization strategies and formulated comparison amongst them. Numerical and Analytical strategies are found to be the preferred choice for optimization which is consistent with the fact that they are perfect for situations with prior knowledge. Serrao et al. in their comparative analysis of energy management strategies for HEVs compared DP, PMP and ECMS strategies. Their work ascertains the results obtained from the three strategies to be quite similar and ECMS as a better option for real-time implementation[59] . The simulation model developed for this paper assumes that the control inputs affecting the cost function are not varied during a drive cycle. Hence, the approach taken for determining the optimal operating input set points computes the cost function for all possible combinations of the control inputs within the specified bounds.

2.3.5 Cost Function and Optimization Strategy

The plug-in series hybrid electric vehicle architecture uses two energy modes of operation, CD and CS as mentioned in the previous section. In the CD mode, vehicle's energy storage system or high voltage battery serves as the only energy source with very minimal energy recuperation with regenerative braking. On the contrary, in the CS mode, in conjunction with the energy storage system, the generator system supplements additional energy and thereby maintaining available capacity on the battery and by extension improving vehicle range. Thus, the goal for the energy management is to utilize these two available modes in the best possible combination. For the targeted application of the control strategy on a real plug-in series hybrid electric vehicle and due to implementation limitations, the deterministic rule-based control strategy was chosen for

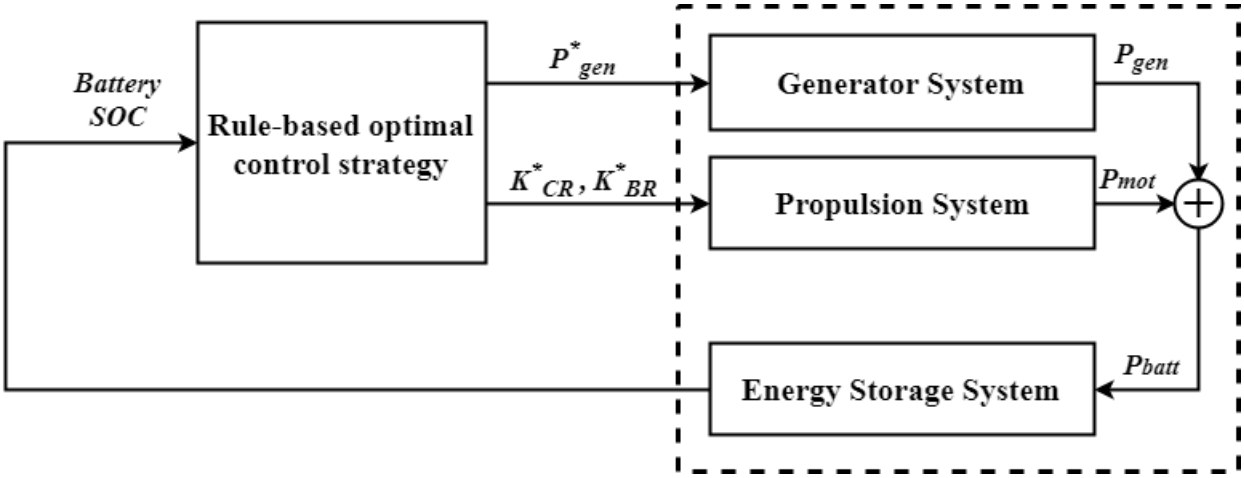


Figure 2.2: Controller-Plant diagram for Energy Management System

optimization. For this optimization process it was assumed that prior knowledge of the drive cycle existed. Further, to narrow down the scope of optimization, a non-blended CD-CS operation mode was chosen with pre-defined battery capacity or battery SOC limit to switch modes from CD to CS. Figure 2.2 shows the controller-plant diagram for energy management system. Thus, the goal of this optimization process was to obtain operating power set-point (P^*_{gen}) for the generator system and determining the amount of regenerative braking (K^*_{CR}, K^*_{BR}). These parameters were then correlated against the net fuel economy of the vehicle over a trip to get the best fuel economy.

2.3.6 Multi-Objective Optimization

P^*_{gen}, K^*_{CR} & K^*_{BR} were decided to be the parameters to be optimized as mentioned earlier. The primary goal of optimization was decided as reduction of overall energy consumption of the plug-in series hybrid electric vehicle and as a secondary goal reduction of net tail-pipe emissions. Hence, the optimization problem can be represented as,

$$min(f(x), g(x)) \; s.t. \; x \in X \tag{2.15}$$

where, $f(x), g(x)$ are the objective functions to be minimized, x is a vector of parameters to be optimized and X a set of all such possible vectors. The form of optimization problem mentioned in Eq.(2.15) is similar to that of a Multi-Objective Optimization problem. Herrera et al. present a similar approach for optimizing energy distribution with genetic algorithm for a hybrid electric bus [31]. As mentioned previously, the first optimization objective $f(x)$ is overall energy consumption ($EC_{combined}$) and the second optimization objective $g(x)$ is net tail-pipe emissions (TPE_{net}). Elaborating on the relation shown in Eq.(2.13),

if $SOC_{current} > \ SOC_{CD-CS} + SOC_{band-CS}$

$$EC_{CD} = \frac{[SOC_{init} - SOC_{CD-CS}] \times E_{batt}}{Total\ distance\ travelled\ during\ CD\ mode} \left[\frac{kWh}{km}\right] \tag{2.16}$$

elseif $SOC_{current} <= \ SOC_{CD-CS} + SOC_{band-CS}$

$$EC_{CS} = \frac{(m_{fuel-total} \times LHV_{E85}) + \left(\frac{[SOC_{CD-CS}+SOC_{band-CS}-SOC_{current}]\times E_{batt}}{\eta_{gen}}\right)}{Total\ distance\ travelled\ during\ CS\ mode} \left[\frac{kWh}{km}\right] \tag{2.17}$$

endif

Equations(2.16,2.17) show energy consumption calculations for CD and CS mode. As evident from these relations $EC_{combined}$ depends on amount of fuel used and amount of battery energy used. Based on the phenomenological emissions model and scaling factors for four key tail-pipe emissions listed in Table 2.2, the net tail-pipe emissions are computed as,

$$TPE_{net} = [K_{CO_2} + K_{CO} + K_{NO} + K_{UHC}] \times m_{fuel-total} \left[\frac{g}{km}\right] \tag{2.18}$$

With the objective functions set as shown in Eqs.(2.13, 2.18), we setup up some constraint relations to obtain the optimized solution. These constraint relations are applied on the parameters chosen to be optimized i.e. P^*_{gen}, K^*_{CR} & K^*_{BR}. Table 2.4 lists down these constraints based on the component performance limitations.

Table 2.4: Multi-Objective Optimization constraints.

Constraint Label	**Relation**
C1	$0 \leq P_{gen} \leq 25\ [kW]$
C2	$0 \leq K_{BR} \leq 15\ [\%]$
C3	$0 \leq K_{CR} \leq 10\ [\%]$
C4	$0 \leq m_{fuel} \leq V_{fuel\ tank\ capacity} * \rho_{E85}\ [kg]$

2.3.7 Simulation Setup

For this simulation, the plug-in series hybrid electric vehicle described in this chapter is operated against three different drive cycles i.e. UDDS (Urban Dynamometer Driving Schedule), HWFET (Highway Fuel Economy drive cycle) and a combination of these two drive cycles. Figures 2.3 - 2.5 display these drive cycles along with the associated road grade. The combined cycle drive trace was generated as a UDDS-HWFET-HWFET-UDDS combination with the road grade profile for a round-trip in Seattle, WA as shown in Figure 2.6.

For the simulation, vehicle parameters chosen were those similar to the plug-in series hybrid electric Chevrolet Camaro for which this study is targeted. These are listed in Table 2.4 . Full vehicle mathematical model along with vehicle control strategy was developed on MATLAB/Simulink platform. Baseline simulations were first done on the UDDS and HWFET cycles separately. This helped debug the vehicle model as well as the baseline control algorithm. Energy management optimization simulations were performed using the combined cycle as shown in Figure 2.8. To enable the vehicle to operated in both CD and CS energy modes, the cycle was repeated 5 times which made the entire trip length close to 89 miles. A parameter sweep for P_{gen}, K_{BR}, K_{CR} within the chosen constraints for overall energy consumption and total fuel consumption was

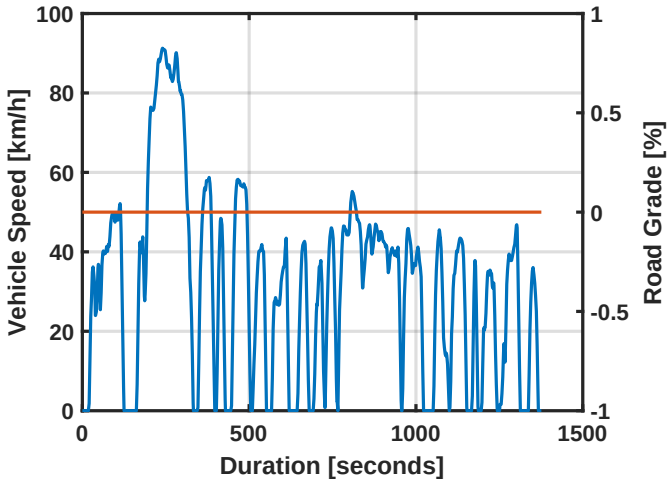


Figure 2.3: Urban Dynamometer Driving Schedule (UDDS)

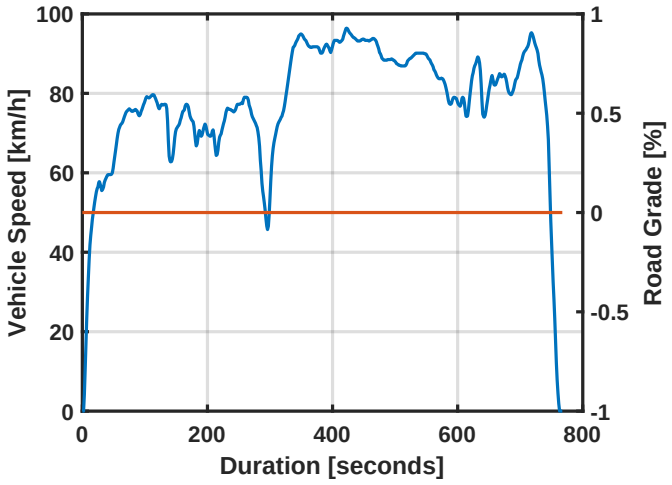


Figure 2.4: Highway Fuel Economy drive cycle (HWFET)

done. Parameter sweep results showed that the variation in overall energy consumption and total fuel consumption was inconsequential due to coasting regenerative braking (K_{CR}). Thus, the focus was shifted towards determining an optimal generator power set-point (P^*_{gen}) and optimal simple regenerative braking coefficient (K^*_{BR}).

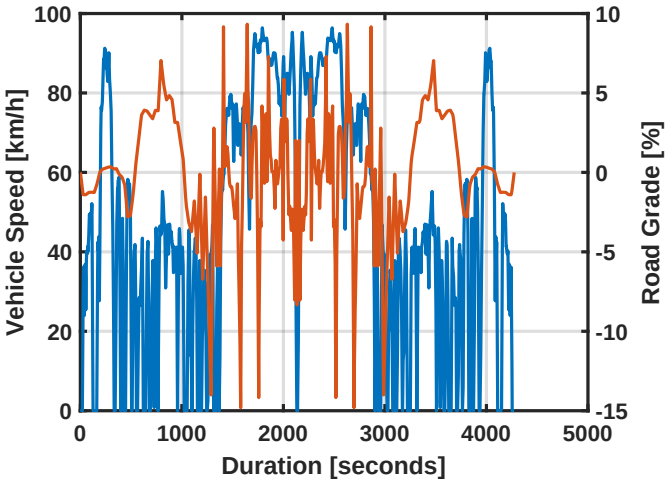


Figure 2.5: Self generated combined drive cycle

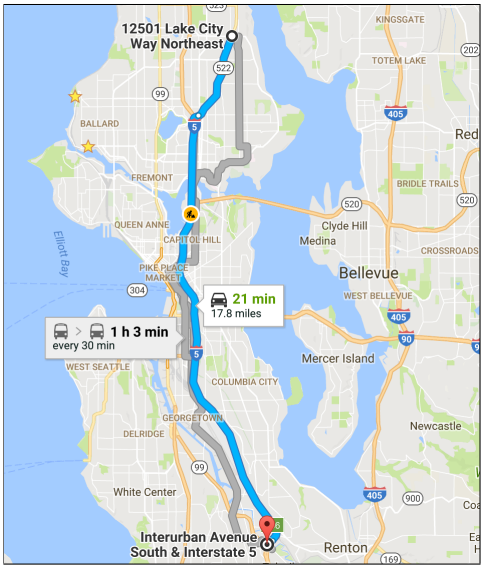


Figure 2.6: Drive route used for the optimization study.

The parameter sweep results were used to generate the multi-objective cost function. Overall energy consumption and total fuel used objective function sweep results were curve fitted to second order polynomial in two variables, x and y which relate to P_{gen}

and K_{BR}, respectively. Equation 2.19 shows the polynomial type and the coefficients for the fit are listed in Table 2.5.

$$f(x,y) = a + bx + cy + dx^2 + exy + fy^2 \tag{2.19}$$

Table 2.5: Coefficient values for polynomial fit of Energy Consumption (EC) and Total Fuel (TF) used.

Coefficient	**EC Fit Values**	**TF Fit Values**
a	0.2987	-0.1439
b	-1.82e-5	1.94e4
c	6.185e-5	2.093e2
d	4.705e-10	-4.986e-9
e	9.211e-8	-9.544e-7
f	-1.753e-4	-2.165e-3

Fitted polynomials for the two objective functions are shown in Figures 2.7 and 2.8. As evident from the figures, if equally weighted the joint minima for the two objective functions will not be the optimal solution for the objective functions individually. Hence, out of the two objective functions more weightage is given to the overall energy consumption objective. This choice is supported by the key goal of an optimal energy management strategy to reduce overall energy consumption. Figure 2.9 shows the multi-objective optimization solution plot along with the possible solutions.The results are further discussed in the next section.

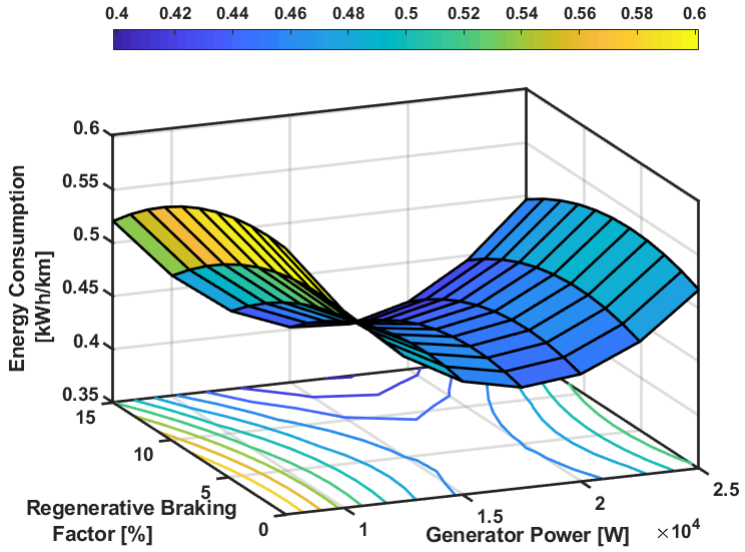


Figure 2.7: Polynomial fit for a) Energy Consumption and b) Total Fuel Consumption objective functions.

2.4 Results and Discussion

On the basis of the Multi-Objective Optimization plot with overall energy consumption and total fuel used as the objective functions, two points “A" and “B" as labelled in Figure 2.9 were obtained as possible solutions. Both of the points have the similar simple regenerative braking (K_{BR}) value of approximately $15\ \%$. This was expected from the simulations, as higher the recuperation of energy from regenerative braking, lesser the overall energy consumption by the vehicle. The limit of 15% was established as per Table 2.4 to avoid degradation in driveability. Amongst the solutions obtained from simulation sweeps, point “A" labelled in Figure 2.9 was chosen as the solution that provides a good compromise between overall energy consumption and total fuel used. Another possible solution for the least possible overall energy consumption number is point “B" labelled in the figure. This operating point reduced overall energy consumption by ap-

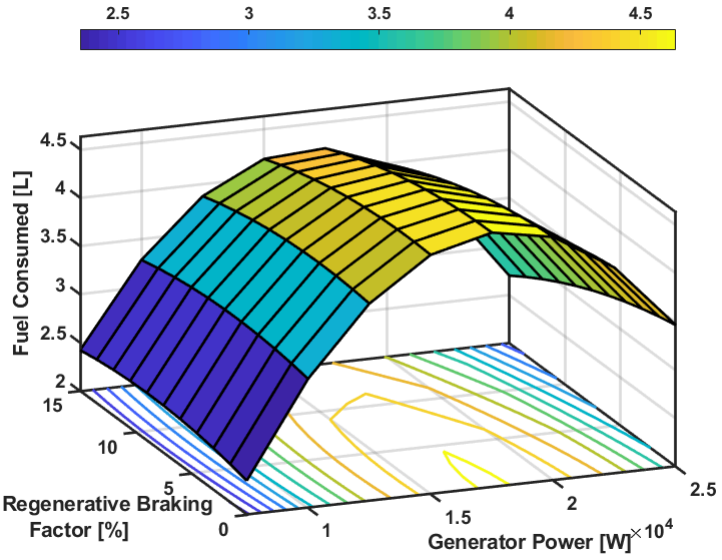


Figure 2.8: Polynomial fit for a) Energy Consumption and b) Total Fuel Consumption objective functions.

proximately $13\ \%$ but increased total fuel used by $16\ \%$.

To better compare the two optimal operation points labelled “A" and “B", a baseline performance set point is chosen. With the intent of operating the generator system at a fixed set point, the baseline P_{gen} was fixed at 15 kW while both K_{BR} & K_{CR} were kept zero. So, the Base case has $P_{gen} = 15$ kW, $K_{CR} = K_{BR} = 0$ %, Optimal case 1 has $P_{gen} = 12.5$ kW, $K_{CR} = 10$ %, $K_{BR} = 15$ % and Optimal case 2 has $P_{gen} = 17.5$ kW, $K_{CR} = 10$ %, $K_{BR} = 15$ %. Here Optimal case 1 refers to point labelled “A" and Optimal case 2 refers to point labelled “B" in Figure 2.9.

As per Figure 2.10, vehicle speed for the Optimal case 1 & 2 overlap each other but with some deviation from the Base case. As both coasting and braking regeneration are kept zero in Base case, such a behavior is expected from the vehicle. Additionally, as the deviation occurs close to coasting and braking events, it further confirms that

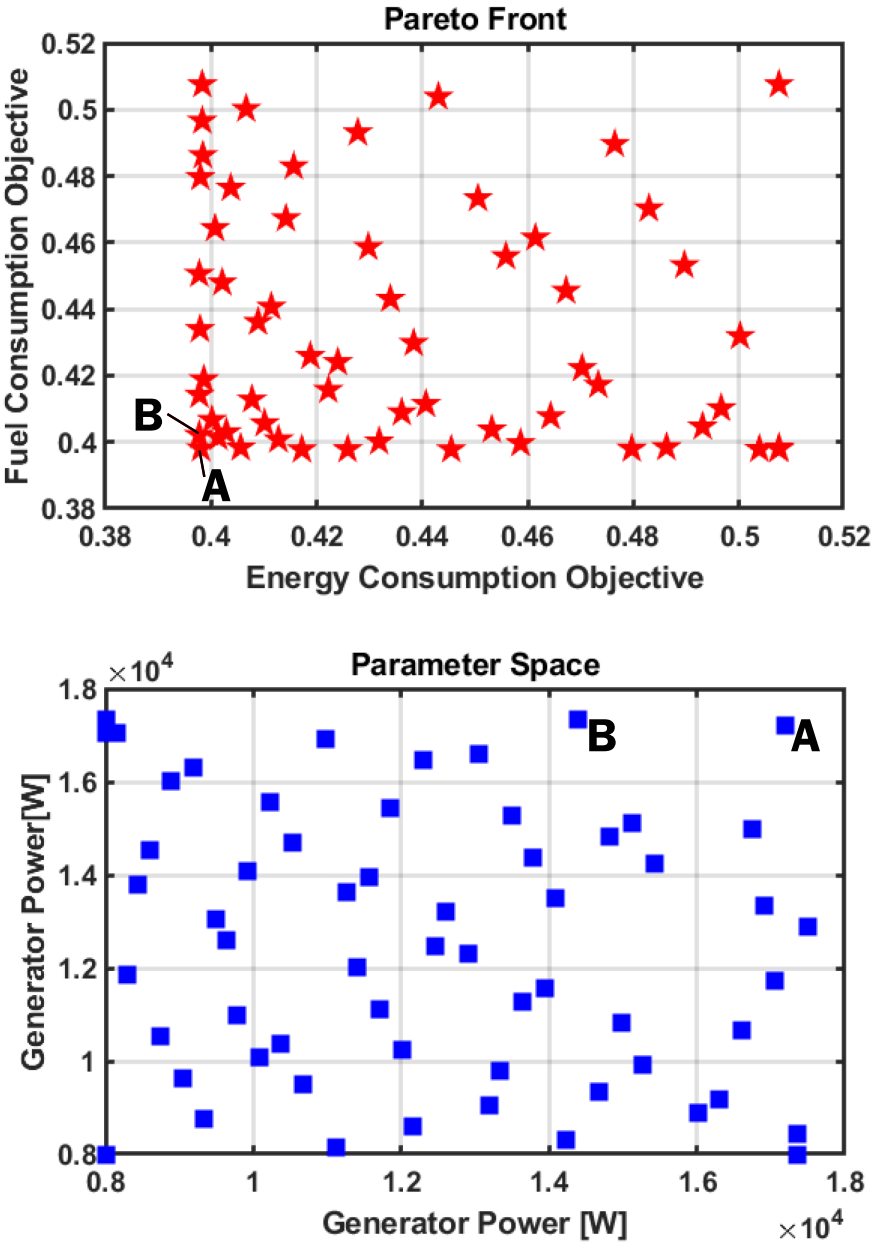


Figure 2.9: Multi-objective optimization solution plot.

regenerative braking control was adequately implemented. One of the shortcomings of using speed drive trace along with a fixed drive duration as an input is the inconsistency

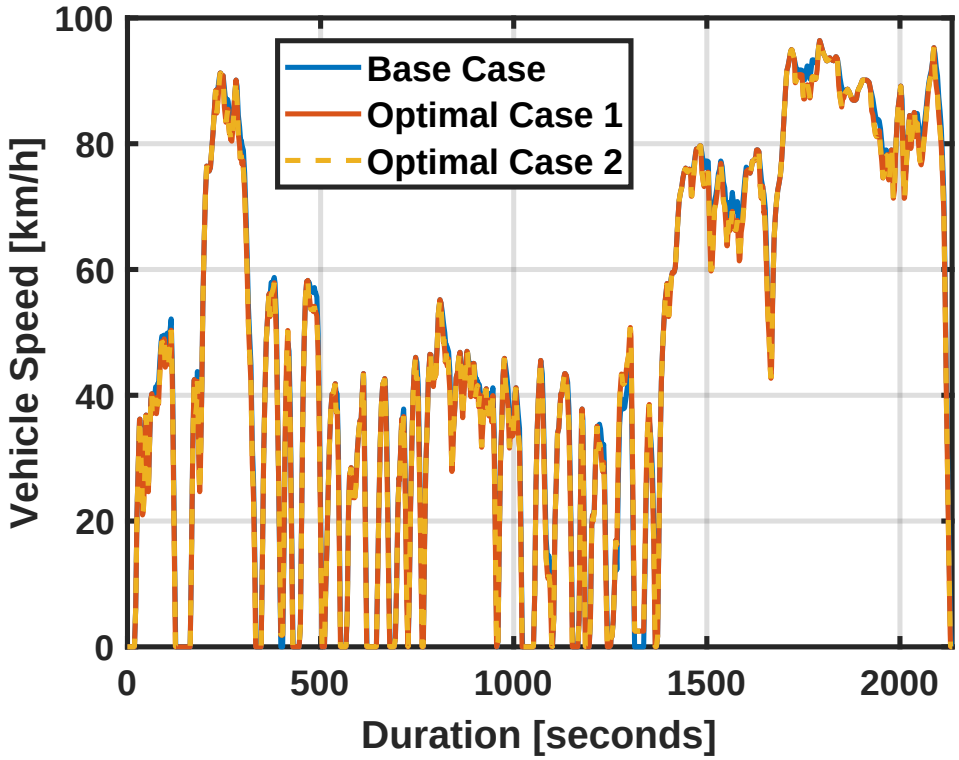


Figure 2.10: Vehicle speed comparison between the Base case, Optimal case 1, and Optimal case 2.

with the total distance travelled. As evident from the speed trace in Figure 2.10, due to speed deviations from Base case, the net distance travelled in Base case is slightly greater than that travelled in the Optimal case 1 & 2. For the simulations this was found to be approximately 285 km and 280 km for Base case and Optimal cases, respectively. As energy consumption is computed with normalization against total distance travelled, this offset in total distance covered doesn't impact final simulation results. Additionally, in reality such a discrepancy between vehicles with and without regenerative braking will be observed if the total duration is fixed.

The deviation in vehicle speeds as discussed earlier, shows a positive impact on the

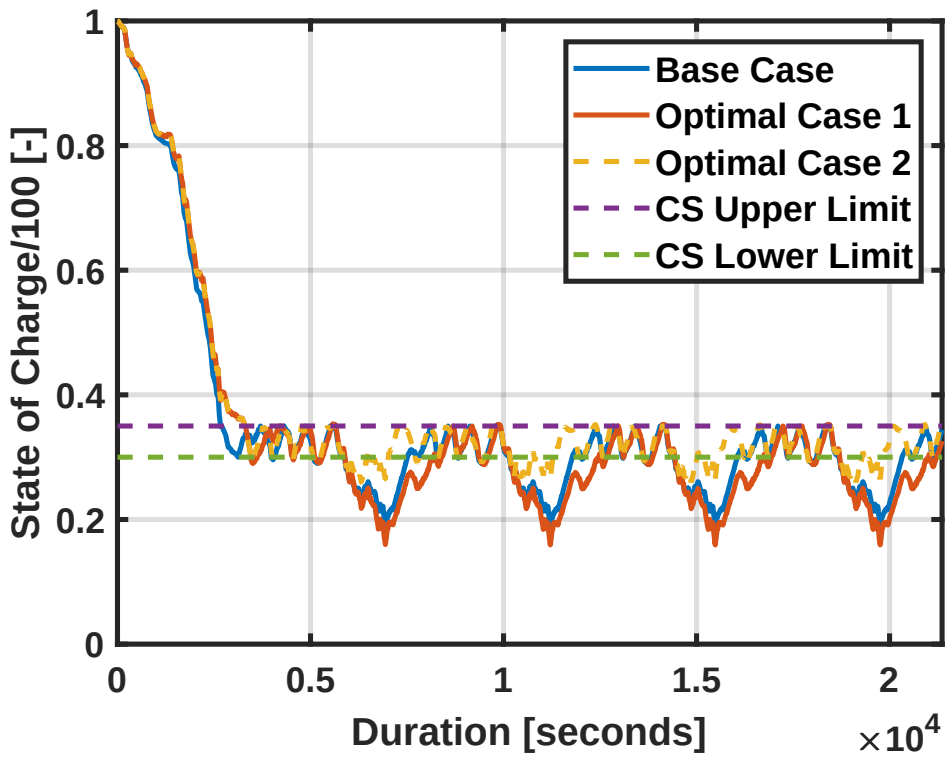


Figure 2.11: Battery SOC comparison between the Base case, Optimal case 1, and Optimal case 2.

vehicle's battery energy consumption. With regenerative braking events occurring in Optimal cases, battery SOC was expected either to stay unchanged/constant or increase with a positive slope. As per Figure 2.11, such a behavior is observed in the Optimal case SOC plots than the Base case SOC plots. Subsequently, the CS energy mode starts earlier in the Base case vehicle than in the Optimal cases, which was expected. The resulting CD range benefit with active regenerative braking in Optimal cases is that of $\approx$ 2 km. Though this benefit is not significant but due to limitations of the driver behavior model and conservative constraints on regenerative braking, the range benefit is still significant. With different generator system operating set points i.e. 15 kW for Base case, as

well as 12.5 kW and 17.5 kW for the Optimal cases, substantial deviations in battery SOC were observed. The city drive section of the trace shows approximately similar battery SOC behavior in all three cases due to the low net energy demand which is easily met by the generator power set-points in all three cases. A much closer observation of SOC trends shows relatively faster SOC rise for Optimal case 2 as per expectations, due to a higher P_{gen} set point. Though a significant difference was not observed for the chosen P_{gen} set points that corresponded to a charging current of approximately 37 Amperes, 44 Amperes, and 51 Amperes (i.e. 0.65C, 0.78C, and 0.9C charge rates). The low energy demand during the city driving section due to stop-and-go type of speed profile as shown in Figure 2.12, allows the battery SOC to be maintained within the 5 % battery SOC band. In comparison, during the highway driving section, none of the three cases being compared maintains the battery SOC within the 5 % battery SOC band. The largest drop in battery SOC is observed for the Optimal case 1 followed by the Base case and the least for Optimal case 2. The difference in P_{gen} values are in agreement with this trend but the deviations in battery SOC from the 5% battery SOC band are asymmetric. As shown in Figure 2.13, for Optimal case 2 the battery SOC (shown in green) is maintained closest to the CS Mode - Battery SOC lower limit of 0.3, with the minimum value of 0.26. In contrast, for Optimal case 1 minimum value for the battery SOC was about 0.16 and that for Base case about 0.19. This asymmetry shows that energy demand during the highway driving section is at the very least met by P_{gen} set point for the Optimal case 1. Thus to meet this high energy demand, P_{gen} needs to be set at a higher value than 17.5 kW. Studies published on $LiFePO_4$ battery degradation show that with a repeated SOC $\leq$ 0.2 in every discharge cycle, battery life diminishes. Hence, from a battery life longevity perspective both the Base case and Optimal case 1 scenarios are not suitable for the vehicle. Now, the simulation results were evaluated with respect to the objective functions i.e. Overall Energy Consumption (kWh/km) and Net Tail-pipe Emissions (g/km). The Base case results were used to determine relative trends for the Optimal case solutions. For overall energy consumption values it was observed that Optimal case 2 showed a

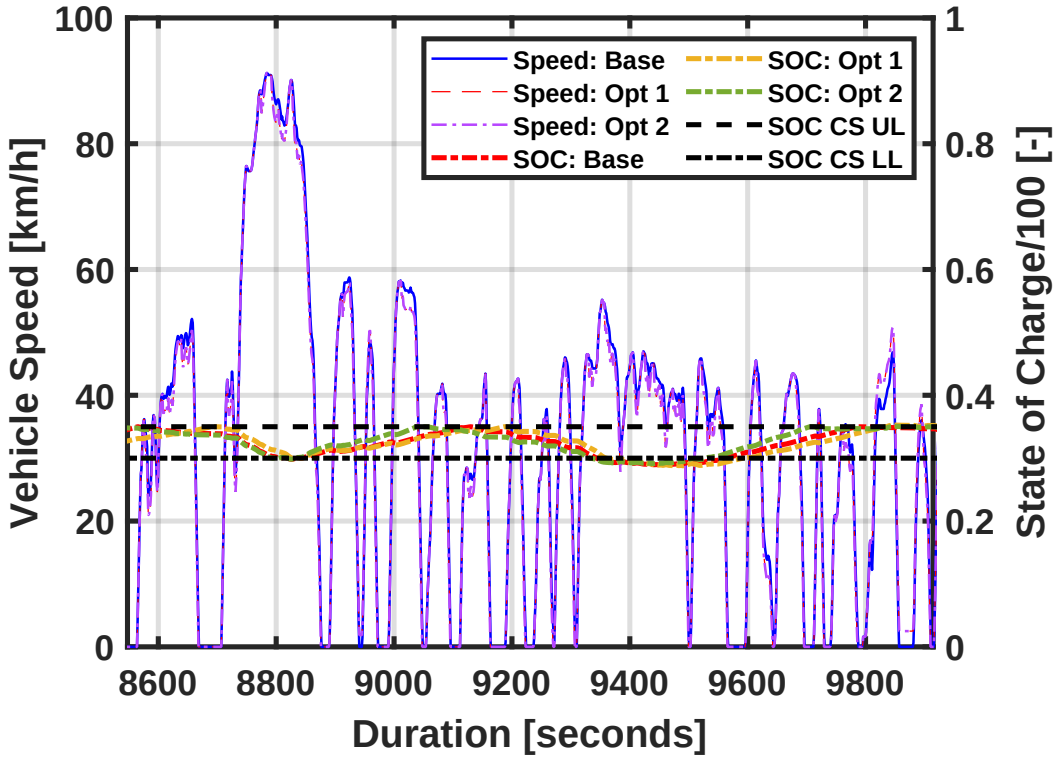


Figure 2.12: Vehicle speed and Battery SOC comparison between base and optimal cases for city driving section.

relative reduction of about 8.91% in overall energy consumption, while Optimal case 1 showed a reduction of about 4.46%. These trends are in accordance with the behavior of battery SOC observed in the CS mode as analyzed earlier. With the generator system being active for relatively longer duration in Optimal case 1, fuel consumption is higher, making overall consumption higher for that scenario. For net tail-pipe emissions we had considered total fuel consumption as the objective function due to direct correlation as per Eq.(2.13). Based on the phenomenological emissions model used for this study as discussed in earlier section, the estimated specie-wise emission mass fractions relative to the base case are compared in Figure 2.16. Optimal case 2 was found to have

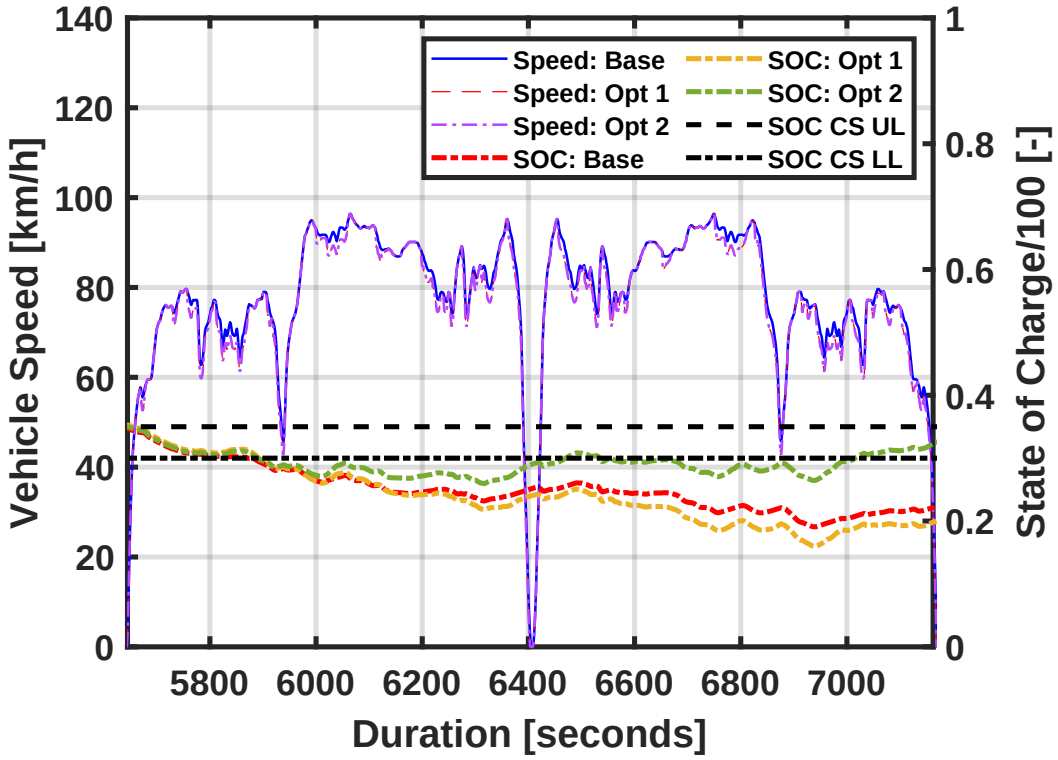


Figure 2.13: Vehicle speed and Battery SOC comparison between base and optimal cases for highway driving section.

relatively larger reduction in UHC and CO species than Optimal case 1 with reference to Base case. For CO_2 and NO species, larger reduction was observed in the Optimal case 1. Overall both the Optimal cases considered estimate reduction in emissions which complies with total fuel consumption results and overall energy consumption trends discussed previously. A closer assessment of the bar plot depicting relative change in mass of emission species shows large swings in Optimal case 2 than Optimal case 1.

This study was conducted with the intent to determine an optimal set of operating parameters to reduce overall energy consumption and net tail-pipe emissions for a plug-in series hybrid electric vehicle. Based on the results obtained from model simula-

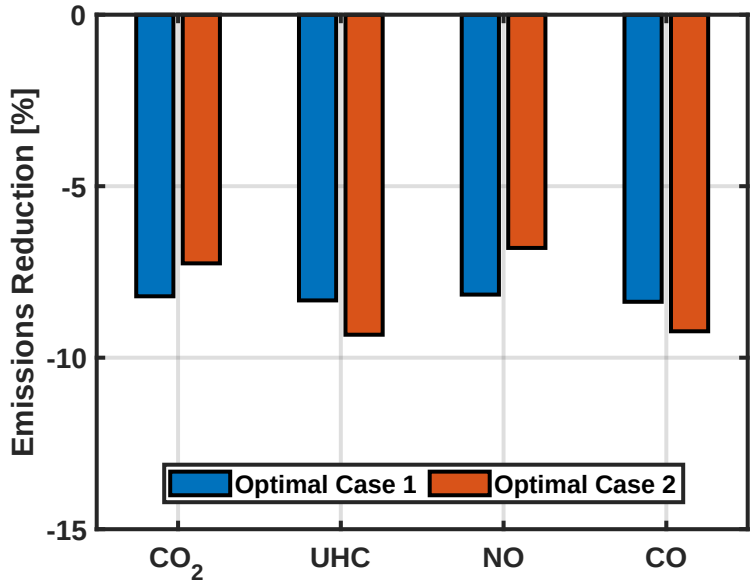


Figure 2.14: Relative change in emissions for Optimal case 1 and Optimal case 2 as determined from simulation.

tion and Multi-Objective offline optimization strategy, two feasible optimal solutions i.e. Optimal case 1 and Optimal case 2 were further evaluated. Upon comparing the effect of these two solution sets on overall vehicle performance and objective functions, Optimal case 2 was found to be the most optimal solution with an overall tail-pipe emission reduction of $\approx$ 8.2% over the Base case.

2.4.1 In-Vehicle Implementation Results

Genset system operation on the Series Hybrid Electric Chevrolet Camaro was tested for performance at criteria emissions at Transportation Research Center Inc. in East Liberty, Ohio. The test involved operating the vehicle in charge depleting (CD), charge sustaining (CS) and transition mode. The test was conducted on a two-wheel chassis dynamometer with a hot-soak at 23.8 ^{o}C. Two different genset control strategies as listed

below were tested.

1. *Constant genset operation (CTL A)*
   In this control strategy, a single genset output power set-point of 15.5kW was used.

2. *Load-following genset operation (CTL B)*
   In this control strategy, three different genset output power set-points of 11.5kW, 13kW and 15.5kW were tested. These set-points were enabled in three different vehicle speed bands of 0-48 km/h, 49-97 km/h and 97+ km/h, respectively.

The results comparing overall energy consumption and criteria tail-pipe emissions of the vehicle for the UDDS (Urban Dynamometer Driving Schedule) drive cycle were compared. The UDDS drive cycle is 8.9 km long as is shown in Figure 2.3. In this test, SOC_{CD-CS} is 50% and $SOC_{band-CS}$ is 10%. Battery state of charge (SOC) and power (kW) of battery, generator and DC-DC for the two control strategies is shown in Figure 2.15. For battery power data, negative power denotes battery discharge and positive power battery charge event. In case of generator power, positive power denotes charge power and for DC-DC power, negative power denotes low voltage discharge. Criteria tail-pipe emissions (CO_2, CO, NO_x, HC) in (g/s) for the two genset control strategies are shown in Figure 2.16.

The overall energy consumption is computed as per Equation 2.13 , using m_{fuel} values of 2.03kg and 2.32kg, respectively. The resulting energy consumption values were obtained to be 1.3 kWh/km and 2.2 kWh/km, respectively. These results are in accordance with the estimated BSFC chart shown in Figure 2.1. For lower generator power outputs at the recorded engine speed, bsfc increases thus resulting in higher fuel consumption. The emission trends were found to be as expected for the two different generator operation regions. The data also shows that at lower generator set-points the combustion is more lean resulting in considerable CO emissions for CTL B. Also, NO_x

emissions are higher for CTL A which based on other emissions data shows relatively rich combustion.

Thus, based on the in-vehicle test results, it is further evident that a higher generator power set-point as in CTL A results in better overall energy consumption. Additionally, in terms of combustion, the variation in criteria emissions is not directly correlated with fuel consumption thus an optimal generator power set-point can be chosen based on having different weights for the emission species.

2.5 Conclusion

In this paper we focused on evaluating a preliminary mathematical model and energy management optimization strategy on a research Plug-in Series Hybrid Electric Vehicle built upon the 2016 Chevrolet Camaro platform. Our goal with this paper was to determine optimal operating points for the on-board generator system with respect to overall energy consumption and tail-pipe emissions. The optimal generator set point was determined by posing the overall energy consumption problem as a Multi-Objective Optimization.

The optimization process with respect to system operational constraints provided two feasible optimal operating points for the generator system. These operating points were evaluated in simulation with a Rule-Based Energy Management approach on the vehicle mathematical model. A peak improvement of 8.9% in overall energy consumption and 8.2% in net tail-pipe emissions was obtained with generator system operating point of P_{gen}=17.5kW.

The Rule-Based approach was evaluated on the actual research vehicle with two different implementations and generator power set points. Vehicle's overall energy consumption and tail-pipe emissions were measured and recorded with a chassis dynamometer testing setup. The results showed lower energy consumption with higher generator

power. This result was partially in accordance with our simulation results i.e. for Optimal 2 strategy. A lower generator power set point showed higher overall energy consumption.

Based on the simulation and in-vehicle implementation results, the following conclusions can be made,

1. The losses considered in the mathematical model do not accurately represent the vehicle's losses.

2. The estimated engine BSFC map referenced in this study is partially accurate. Specifically in the high power regions as compared to the low power regions.

3. Thermal limitations of the vehicle further limited the peak power output from the generator system to 15 kW.

Future work concerning this research vehicle and the energy management optimization problem will focus on improving the vehicle mathematical model using actual vehicle data for different powertrain subsystems. Additionally, also assessing a feasibility of a novel energy management optimization approach for hybrid electric vehicles with a series architecture.

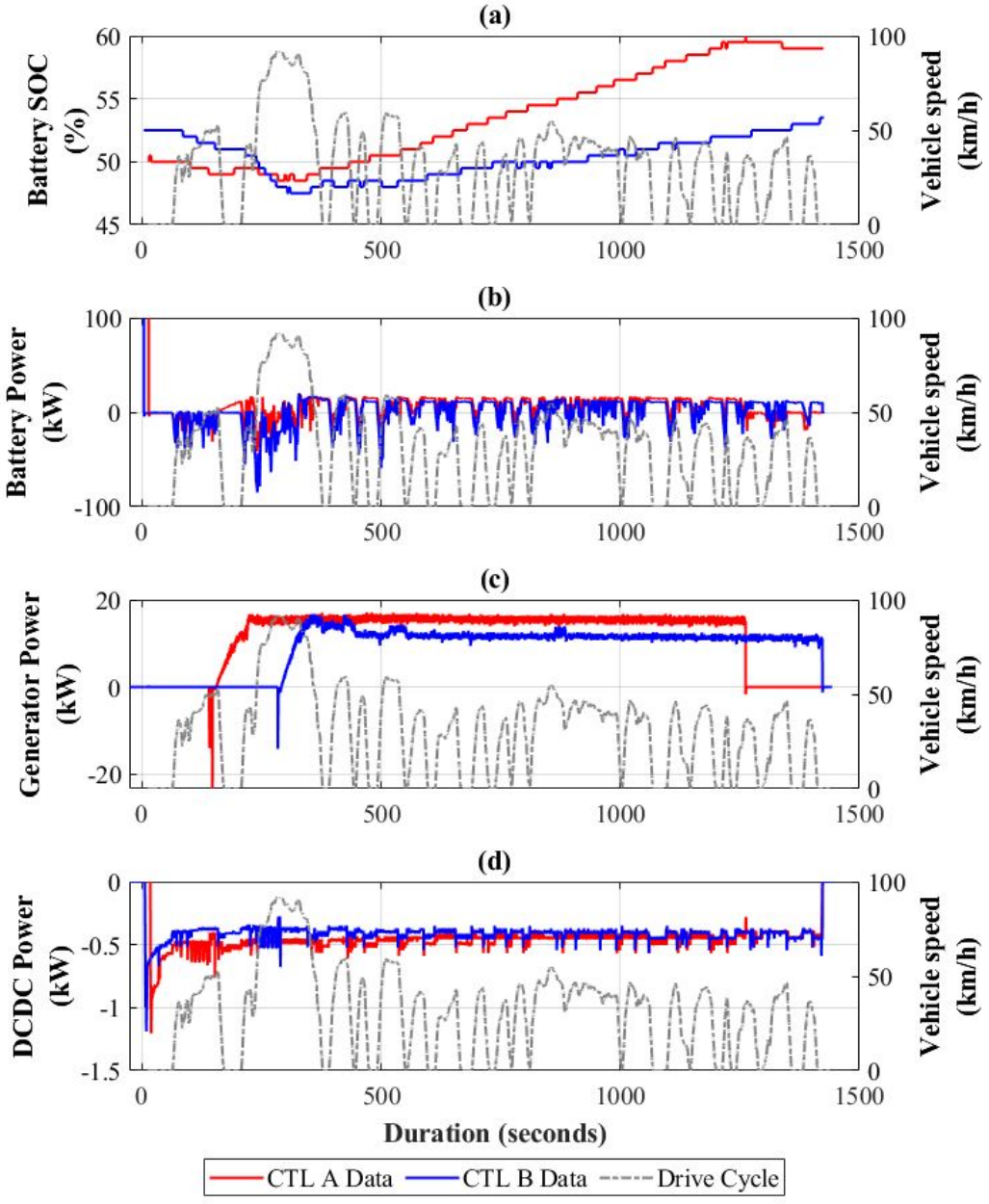


Figure 2.15: Energy consumption and power data for Series PHEV Chevrolet Camaro on UDDS drive cycle. Behavior between CTL A and CTL B genset control strategies is compared and shown as (a) Battery SOC, (b) Battery Power, (c) Generator Power, and (d) DCDC Power.

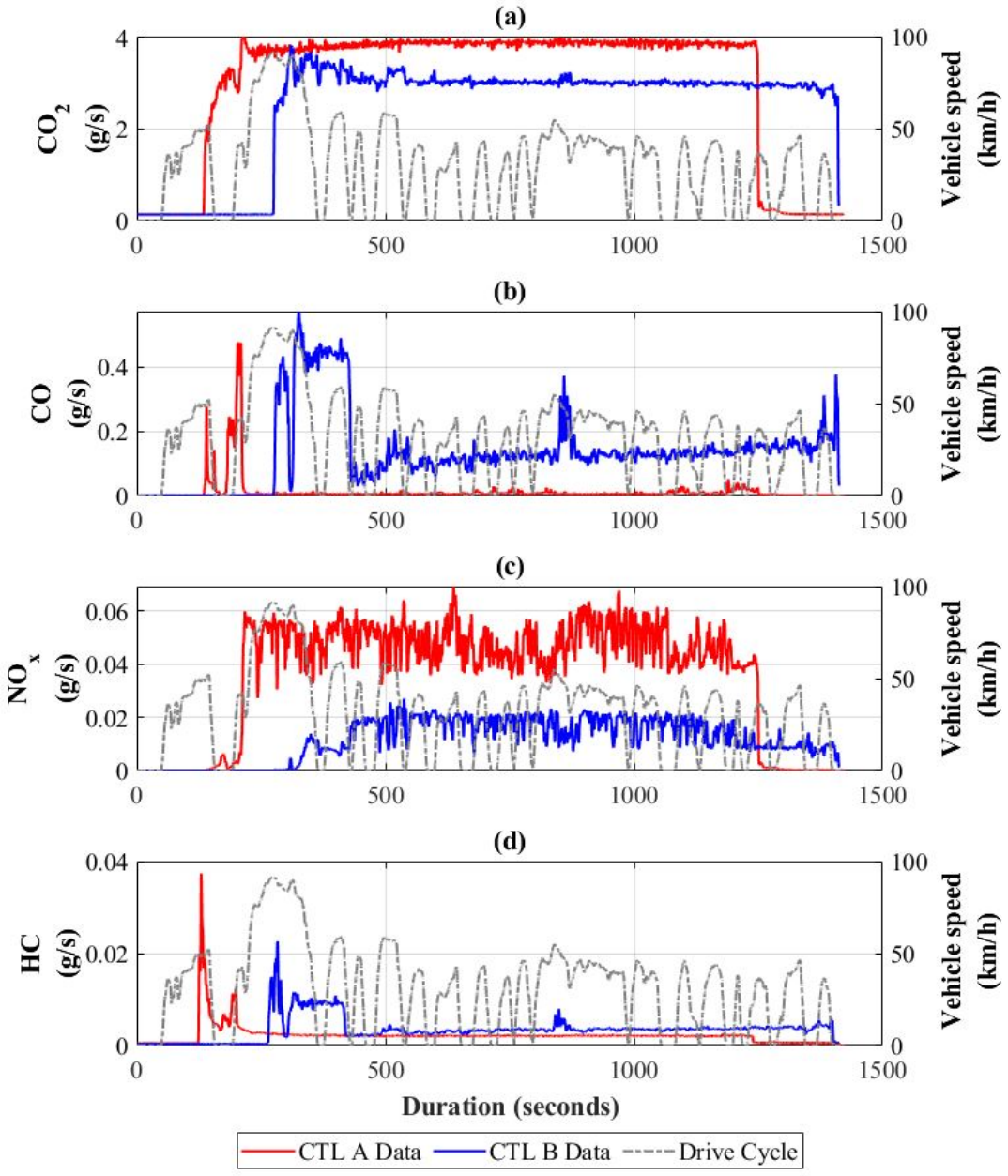


Figure 2.16: Criteria tail-pipe emissions data for Series PHEV Chevrolet Camaro on UDDS drive cycle. Behavior between CTL A and CTL B genset control strategies is compared and shown for (a) CO_2, (b) CO, (c) NO_x, and (d) HC.

Chapter 3

On Implementing Optimal Energy Management for EREV using Distance Constrained Adaptive Real-Time Dynamic Programming

Publication: *A. V. Kalia and B. C. Fabien, "On Implementing Optimal Energy Management for EREV Using Distance Constrained Adaptive Real-Time Dynamic Programming," Electronics, vol. 9, no. 2, p. 228, Feb. 2020, doi: 10.3390/electronics9020228.*

3.1 Abstract

Extended range electric vehicles (EREVs) operate both as an electric vehicle (EV) and as a hybrid electric vehicle (HEV). As a hybrid, the on-board range extender (REx) system provides additional energy to increase feasible driving range. In this paper, we evaluate an experimental research EREV based on the 2016 Chevrolet Camaro platform for optimal energy management control. We use model-in-loop and software-in-loop environments to validate the data-driven power loss model of the research vehicle. A discussion on the limitations of conventional energy management control algorithms is presented. We then propose our algorithm derived from Adaptive Real-Time Dynamic

Programming (ARTDP) with a distance constraint for energy consumption optimization. To achieve a near real-time functionality, the algorithm recomputes optimal parameters by monitoring ESS state of charge deviations from the previously computed optimal trajectory. The proposed algorithm is adaptable to variability due to driving behavior or system limitations while maintaining the target driving range. Net energy consumption evaluation shows a maximum improvement of 9.8% over the conventional CD-CS algorithm used in EREVs. Thus, our proposed algorithm shows adaptability and fault tolerance while being close to the global optimal solution.

3.2 Introduction

Hybrid Electric Vehicles (HEVs) and Electric Vehicles (EVs) are worthy alternatives to the conventional, gasoline-only powered vehicles. As per a recent IEA (International Energy Agency) report on Global EV Outlook, these eco-friendly alternatives have grown upto 4% of global vehicle market share from 2013 to 2018 and are expected to go beyond 40% by 2030 [5]. Relatively lower net energy density amongst HEVs and EVs, makes efficient and optimal utilization of energy necessary. Energy management algorithms enable these vehicles to perform efficiently and at optimal operation points within- certain constraints.

HEVs at a minimum have more than one source of propulsive energy which makes energy management imperative and challenging. Variation in powertrain architectures among HEVs further diversifies usable energy management strategies. In this article, we have focused on an Extended Range Electric Vehicle (EREV) for energy management optimization research. An EREV can operate both as an EV and an HEV based on its battery or energy storage system's (ESS) state of charge [22]. The state of charge (SoC) value represents available energy in the ESS relative to its maximum energy storage capacity. While operating as an EV, the EREV utilizes only the ESS for its energy demands and operates in a charge depleting (CD) mode. Whereas, during its operation as an HEV, the

EREV uses both the ESS and the range extender (REx) system to supply the required energy and operates in a charge sustaining (CS) mode. EREVs either have an on-board REx system or a trailer-mounted detachable REx system [6]. The trailer-mounted detachable REx system allows the user to use REx only for long distance drives thus increasing opportunities for less fuel consumption and emissions [86, 75].

As mentioned earlier, a distinct CD and CS operating mode is a conventional EREV energy management approach. Implementations of this strategy with a Rule-Based (RB) algorithm is well documented in the literature [32, 62]. Researchers have extended on this work by determining optimal operating points through offline optimization schemes [18, 10, 9, 20, 87, 34]. To further improve system efficiency and operate close to optimal performance, real-time online optimization strategies have also been evaluated. Energy Consumption Minimization Strategy (ECMS) is one such well documented and validated strategy [16]. This strategy has proven useful for EREV energy management in the CS mode. In-production EREVs like the BMW i3 Rex and Chevrolet Volt also implement the distinct CD and CS strategy with an additional load-following strategy in CS mode. Moura et al. presented the blended CD-CS strategy as an alternative to the distinct CD-CS strategy [50]. Here, the optimization algorithm determines feasible combination of ESS and REx operation points which satisfy the cost function and additional constraints. Moura et al. also evaluated stochasticity in driving behavior and concluded that a blended strategy consistently provided a better fuel economy in the range of 3.2% to 11.8% over the conventional CD-CS strategy for the different driving behaviors. With no state feeedback, this approach had limited real-time online implementation [44, 55, 43]. For real-time online implementation, a combination of Markov Decision Process (MDP) and Stochastic Dynamic Programming (SDP) has been implemented and validated in MiL environment. The outcome of this approach showed $\sim 24\%$ improvement over a simple RB strategy but was $\sim 4\%$ worse than DP [45, 81]. Recently published research has leveraged machine learning techniques to obtain opti-

mal solutions for HEV energy management. Son et al. proposed an advanced RB mode control strategy for plug-in HEVs that used DP results for training and forming a predictive mode control map for real-time applications [61]. Vatanparvar et al. [69] developed a context-aware NARX model to predict driver behavior for energy management and Wang et al. [71] implemented deep reinforcement learning on RB strategies for EREVs. Both of these research publications and other similar works have improved energy management algorithm functionality for real-time application when there is limited future drive information. Chen et. al. presented a dynamic programming and neural network based energy management strategy which was evaluated in simulation for known and unknown driving distance and duration conditions [19]. Xi et. al. improved on this method and were also able to adjust the strategy for changes in driving distance during the drive [76].

Based on our review, majority of the existing solutions are sub-optimal. To obtain a global optimal solution, dynamic programming is the most sought after approach. Though, due to its requirement for *priori* information, a real-time online implementation is challenging. Stochastic approaches to estimate *priori* information have enabled real-time online usage but estimation errors seldom result in a sub-optimal solution. Estimations can be improved using machine learning techniques by training models using vehicle data. Though the initial computational expense and any future re-training of models for other scenarios poses a scalability challenge for on vehicle implementation. Therefore, a method that can provide a good estimate of future driving behavior and can adapt to changes in real-time while being close to the optimal solution is required. Additionally, the solution needs to be computationally inexpensive relative to more sophisticated techniques like machine learning.

To meet these requirements we propose a real-time online implementable energy management algorithm based on the Adaptive Real-Time Dynamic Programming (ARTDP) approach. The proposed algorithm uses a forward propagating dynamic programming

technique along with Google Directions API to determine average speed trajectory for the upcoming drive. The algorithm performs re-optimization calculation when the vehicle behavior deviates from initial estimate. We tested our proposed algorithm on an average speed trajectory obtained from Google and its variants. Thus, the aim for our EREV energy management research is to: (i) achieve the maximum feasible driving distance without recharging or refueling and, (ii) maintain the minimum feasible net energy consumption. The novelty in our proposed approach is in the implemented variant of ARTDP that uses a driving distance constraint and integration of Google Direction API for future speed estimate.

This article provides an in-depth description of the EREV power loss model and its validation in Section 3.3. It is followed by an evaluation and comparison of prominent EREV energy management algorithms. This section also presents our proposed algorithm in detail. In Section 3.4, the results from our study and their interpretation and implications are discussed. Finally, a summary of the results draws a conclusion to this article in Section 3.5.

3.3 Methods

3.3.1 EREV Modeling Methodology

The experimental research EREV referenced in this article is based on the 2016 Chevrolet Camaro platform. Figure 3.1 represents the vehicle system architecture as implemented on the experimental research EREV. The key features of this vehicle are, (i) independent dual-motor rear wheel drive with custom gearbox and, (ii) 0.8L motorcycle engine retrofitted to run on E85 fuel [35]. Table 3.1 lists out the modelled operating parameters for the propulsion system components on the vehicle. The experimental research EREV modelled in this study performs within operational limits on its traction motors and REx system.

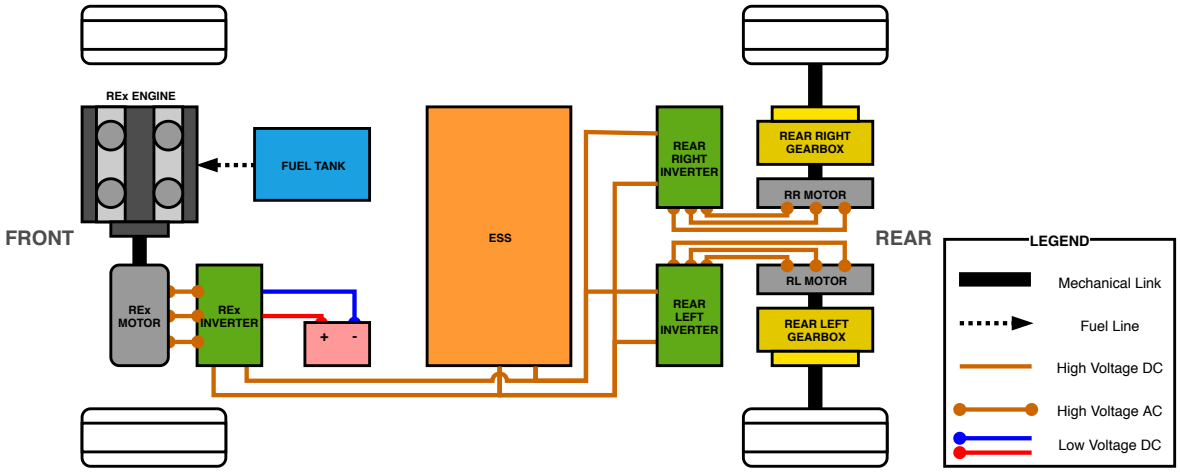


Figure 3.1: Experimental research EREV system architecture diagram.

Model-based design approach is extensively used for controls development of complex systems. This process is an integral step in vehicular system controls formulation. For the presented work, both Model-in-Loop (MiL) and Software-in-Loop (SiL) simulation environment was used in the development, verification and validation stages of the process. A power loss model for the EREV as shown in Figure 3.2 was developed on MATLAB/Simulink platform to simulate the key propulsion and energy storage systems on the research EREV. The model discards effects of tires and propulsion system dynamics to reduce computational complexity. This approach is deemed sufficient as the core premise is to effectively model power flow relative to the power demand during the simulated drive cycle. The power flow model accepts vehicle speed target (v_{target}) in m/s and road grade (α_{road}) in $\%$, as inputs to compute the total road load forces that the propulsion system needs to overcome. Equations 3.1 and 3.2 show the discrete-time road load equations in terms of power, as used in this model.

$$P_{demand}[k] = P_{traction}[k] + P_{aero}[k] + P_{roll}[k] + P_{climb}[k] \tag{3.1}$$

Table 3.1: Experimental research EREV system parameters.

System/Component	Parameter	Value	Units
Engine	Displacement	0.8	L
	Max. Power	80.7	kW
	Max. Torque	80	Nm
ESS	Max. Capacity	18.9	kWh
	Nominal Pack Voltage	340	V
	Discharge Power Limit (Peak)	208	kW
	Discharge Power Limit (Cont.)	61.2	kW
	Charge Power Limit (Peak)	102	kW
	Charge Power Limit (Cont.)	20.4	kW
Generator	Max. Power (Peak)	83.8	kW
	Max. Power (Cont.)	58.6	kW
	Max. Torque (Peak)	200	Nm
	Max. Torque (Cont.)	95	Nm
Traction Motor	Max. Power (Peak)	230	kW
	Max. Power (Cont.)	110	kW
	Max. Torque (Peak)	500	Nm
	Max. Torque (Cont.)	250	Nm
	Gear Ratio	4.2	-
Vehicle	Gross Weight	1875	kg
	Passenger (x2) Weight	160	kg
	Tire Rolling Radius	0.346	m
	Frontal Area	0.4	m^2
	Fuel Tank Capacity	26.5	L

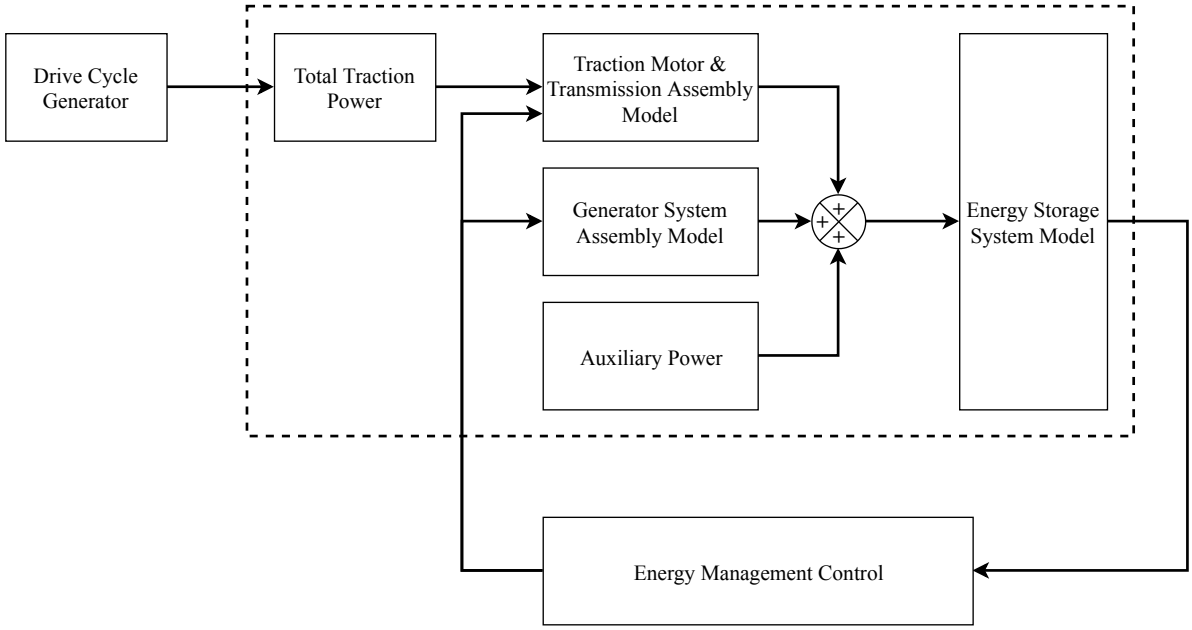


Figure 3.2: Block diagram representation of the experimental research EREV.

$$
\begin{aligned}
P_{demand}[k] =& (m_{vehicle} + m_{pass}) \times v_{target}[k] \times \frac{v_{target}[k] - v_{target}[k-1]}{\Delta k} \\
&+ \frac{1}{2} \times C_d \times \rho_{air} \times A_{front} \times v_{target}^3[k] \\
&+ C_{rr}(p_{tire}, v_{target}[k]) \times (m_{vehicle} + m_{pass}) \times g \times v_{target}[k] \\
&+ (m_{vehicle} + m_{pass}) \times g \times sin(\alpha_{road}[k]) \times v_{target}[k]
\end{aligned} \tag{3.2}
$$

As mentioned earlier, an EREV has an all-electric final propulsion to the wheels. Thus, the actual electric power demand from the ESS, i.e. P_{motor}^{E}, was computed as per Equation 3.3. Here η_{prop} i.e. propulsion system efficiency, is an estimated combined efficiency of the propulsion motor and the custom planetary gearbox used in the research vehicle. The efficiency map was developed as a function of motor torque (τ_{motor}) and motor speed (ω_{motor}). Details of this efficiency map estimation are provided in subsection 3.3.3.

$$P^{E}_{motor}(\eta_{prop}, P_{demand}) = \frac{P_{demand}[k]}{\eta_{prop}(\tau_{motor}, \omega_{motor})} \tag{3.3}$$

The range extension on an EREV is feasible due to the REx system that serves as an APU. The research vehicle referenced in this study uses an E85 fueled 0.8L ICE which is directly coupled to a Bosch SMG 180 electric motor-generator. This REx system supplies additional power to the electric traction motors and ESS to meet the electric power demand denoted by P^{E}_{motor} and/or recharge the ESS. Electric power from REx i.e. P^{E}_{REx} can be formulated in terms of fuel power (P_{fuel}) used to generate it. Equation 3.4 shows the correlation between these two power terms.

$$\begin{aligned} P^{E}_{REx}(\eta_{engine}, \eta_{generator}, P_{fuel}) = \eta_{engine}(\tau_{engine}, \omega_{engine}) \times \cdots \\ \eta_{generator}(\tau_{generator}, \omega_{generator}) \times P_{fuel}[k] \end{aligned} \tag{3.4}$$

With limited $\eta_{generator}$ data available and the feasible value being in the range of i.e. $\eta_{generator} \in$ [0.82, 0.95], $\eta_{generator}$ was assumed to be a constant at 0.86 in this model. Engine efficiency or brake specific fuel consumption map for the E85 fueled engine was infeasible to obtain on an engine dynamometer due to integration limitations. To circumvent this obstacle, REx electric power data and tail-pipe emissions data obtained during in-vehicle operation were used to estimate the BSFC map. The details of this method and validation of estimated fuel flow rate is explained in subsection 3.3.2. Additionally, low-voltage auxiliary electric loads were noted to have a significant impact on the ESS energy levels. The auxiliary electric power i.e. P^{E}_{REx} was modeled as a constant power demand of 520 W with additive noise. The signal-to-noise ratio (SNR) of 25 was used for the model and determined from auxiliary power load tests on the vehicle.

Finally, the ESS model used in this study was derived from discharge and charge data collected during vehicle operation. The earlier version of this model was based

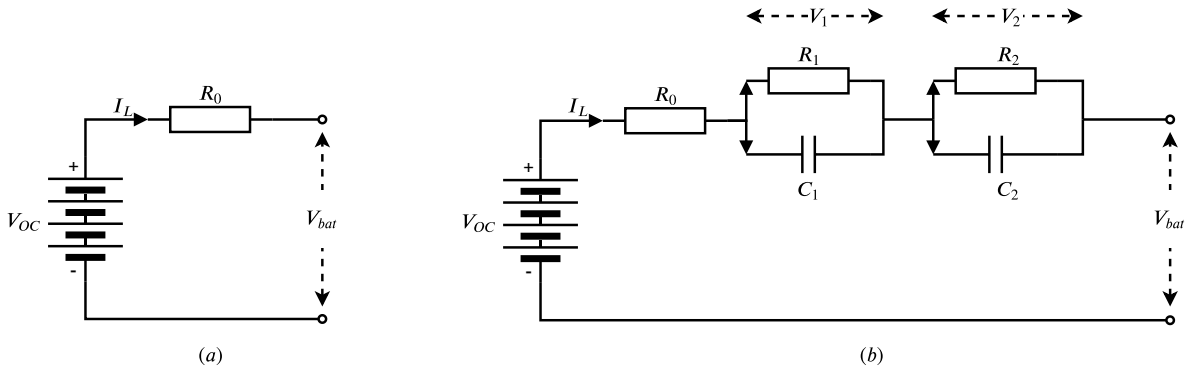


Figure 3.3: Battery equivalent circuit models evaluated for power flow model. (a) Zeroth, (b) Dual Polarization.

on the zeroth order equivalent circuit as shown in Figure 3.3(a) and represented by the Eq.(3.5). In this study, we used a dual polarization battery model as show in Figure 3.3(b) and Eq.(3.6). In case of the zeroth order equivalent circuit, if internal resistance (R_0) is considered a constant, then Eq.(3.5) is linear in voltage. Alternatively, in the dual polarization battery model, the additional resistance and capacitance representing electro-chemical and concentration polarization, is capable of capturing transient behavior during charge and discharge cycles as well as due to ESS SOC non-linear variation. Comparison of these two models showed that the dual polarization battery model captures variation in battery voltage more accurately than zeroth order battery model during discharge and charge current transients [29, 72]. This variation was observed to have an impact on the battery SoC estimation. The dual polarization battery model is represented as per Eq.(3.6) and the resistance and capacitance values were estimated using the System Identification Toolbox in MATLAB.

$$V_{bat}(SOC,k) = V_{OC}(SOC) - I_L[k] \times R_0(SOC) \tag{3.5}$$

$$V_{bat}(SOC,k) = V_{OC}(SOC) - V_1(SOC,k) - V_2(SOC,k) - I_L[k]timesR_0(SOC) \tag{3.6}$$

$$\frac{SOC[k] - SOC[k-1]}{\Delta k} = -\frac{I_L[k]}{Q_{max}} \tag{3.7}$$

The ESS SOC computation is based on coulomb counting method as per Eq.(3.7), where discharge current is considered positive and charge current negative. Available ESS power limits were noted to be a function of ESS SOC and ESS operating temperature. An ESS thermal model wasn't integrated for being out of scope for this study. Hence, the available ESS power limits were modeled as constants. Furthermore, the available range of ESS SOC was modeled between 0% and 100% SOC i.e. SOC $\in$ [0,1.0] with an ESS EPO limit at 8%. These limits were set to accurately model available energy as observed during vehicle evaluation tests.

3.3.2 Engine Efficiency Map Estimation

Limited performance information of the 0.8L engine used in REx system, posed a challenge in accurately modeling power flow. In addition to this, the engine was retrofitted to operate with E85 fuel instead of gasoline. In our earlier work [35], engine efficiency was considered to be a constant due to unknown system state values. In this article, we present an estimate of the engine efficiency or BSFC, for its region of operation.

As mentioned earlier, the engine was directly coupled to a generator-motor in this research vehicle. This setup is quite similar to an engine dynamometer and was useful in determining engine output power. The fuel power corresponding to the output was determined by estimating the input fuel flow rate deduced from the tail-pipe emissions. The E85 fuel used is an oxy-hydrocarbon that is 85% Ethanol and 15% Gasoline which has an equivalent chemical formula of $C_{2.55}H_{7.23}O$. The key tail-pipe emissions

measured for E85 fuel combustion were CO_2, CO, HC and NO_x. For this estimate, the amount of HC emissions were considered negligible and the associated terms weren't considered in the simplified combustion equation as shown in Eq.(3.8).

$$y_1 C_a H_b O_c + y_2(O_2 + 3.773N_2) \longrightarrow k_1 CO_2 + k_2 CO + k_3 H_2O + k_4 NO_x \tag{3.8}$$

Equating the amount of C,H,O and N atoms amongst reactants and products provided relation between y_1, y_2 and k_1, k_2, k_3, k_4. With the goal to estimate amount of E85 consumed, the only important relation for estimation is the one between E85 ($C_{2.55}H_{7.23}O$) and CO_2, CO as per Eq.(3.9). This being a molar relation, it was converted into a mass-based relation by including molar masses as per Table 3.2 and considering $y_1 = 1$ i.e. 1 mol. Without loosing the equality relation and considering mass conservation of carbon atoms, the resulting relation is provided in Eq.(3.10). The mass of carbon in a mole of E85 fuel as well as CO_2 and CO is correlated to determine the mass flow rate of E85 fuel.

$$C: \quad a.y_1 = k_1 + k_2 \tag{3.9}$$

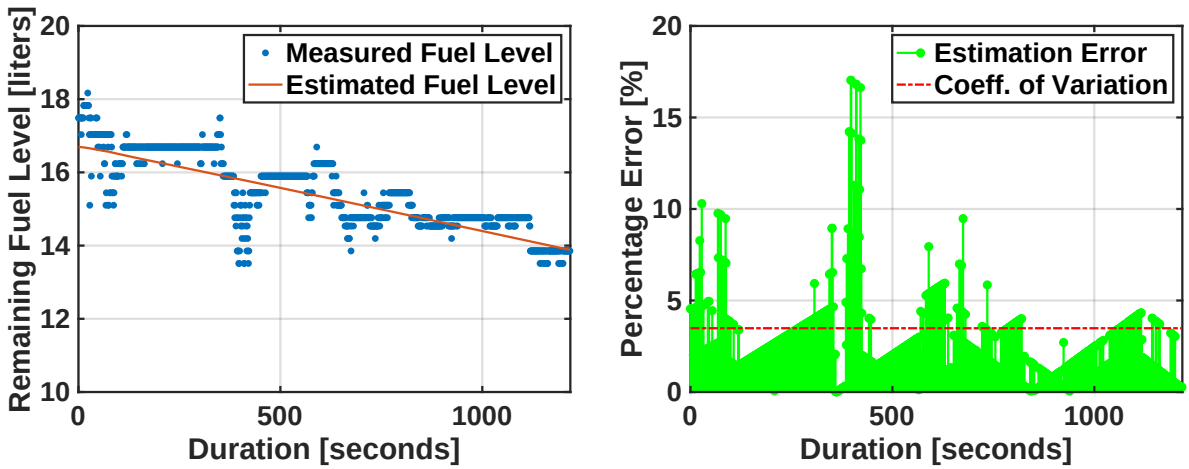
$$\dot{m}_{E85} = \left(\frac{N_{C_{2.55}H_{7.23}O} M_{C_{2.55}H_{7.23}O}}{m_{C \in C_{2.55}H_{7.23}O}}\right) \times \left[\left(\frac{m_{C \in CO_2}}{N_{CO_2} M_{CO_2}}\right)\dot{m}_{CO_2} + \left(\frac{m_{C \in CO}}{N_{CO} M_{CO}}\right)\dot{m}_{CO}\right] \tag{3.10}$$

Table 3.2: Molar masses for key E85 combustion reactants and products.

Substance	**Molar Mass** [g/mol]
$C_{2.55}H_{7.23}O$	53.83
CO_2	44.0
CO	28.0

REx system operating power data along with the corresponding tail-pipe emissions

10Hz modal data set was used to generate the estimated brake specific fuel consumption map for the engine. The estimated fuel flow rate as per Eq.(3.10) was validated against net fuel consumption measured from fuel tank level sensor in the research vehicle. Figure 3.4 shows the comparison between measured and estimated volume of fuel in the tank. The mean estimation error was computed to be with in $\pm 2\%$ with an RMSE of 0.96. The estimated fuel mass flow rate was thus deemed validated and was subsequently used to generate the BSFC map for the engine.



(a) (b)

Figure 3.4: Estimated net fuel consumption comparison to measured net fuel consumption.

The estimated BSFC map for the E85 fueled VFR800 engine is shown in Figure 3.5. The REx system was operated within power limits defined by [0, 17.5] kW in the research vehicle. This limit was primarily defined by the continuous charge power limit of the ESS. The operating speed of the engine was bounded between [4000, 7000] RPM while the load torque bounds were [0,50] Nm. Figure 3.5 also denotes the engine peak brake torque, maximum operating torque and torque associated with ESS power limit. The

map shows the lowest BSFC values in the high torque and low speed region which coincide with the maximum operating power, as shown by dashed white curves. Thus, for lowest fuel consumption rate the REx system must operate near the maximum operating power with high torque and low speed.

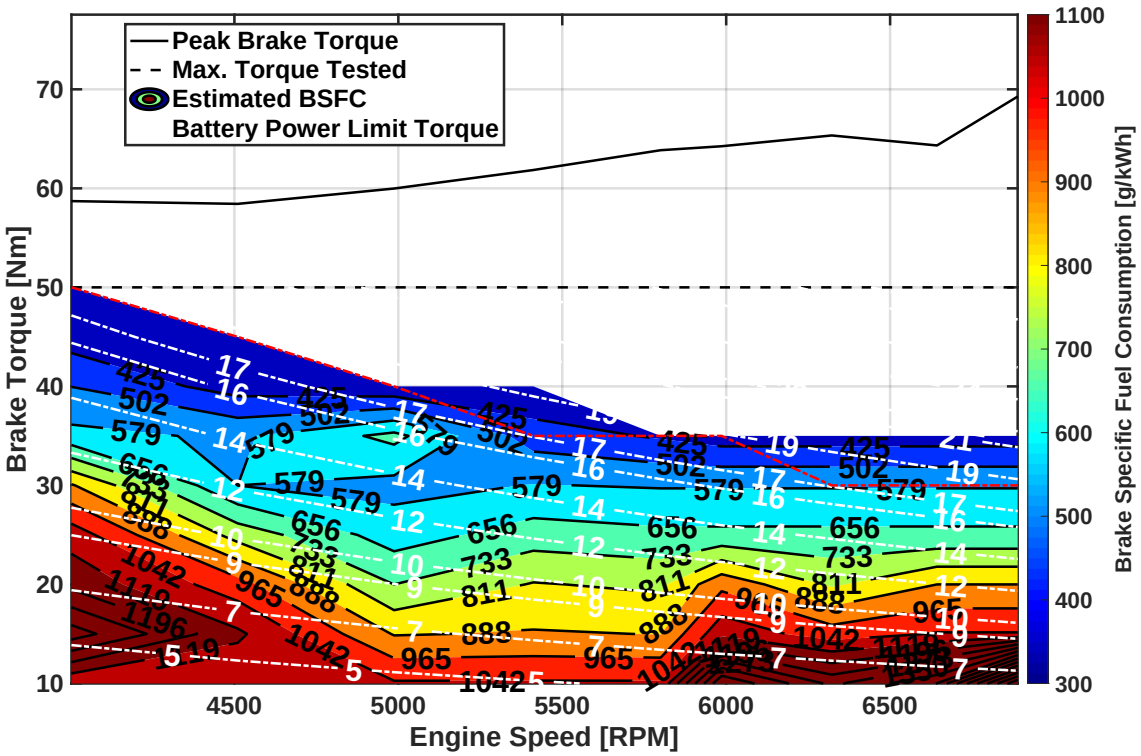


Figure 3.5: Estimated BSFC map for E85 fueled VFR800 engine.

3.3.3 Transmission Efficiency Map Estimation

The propulsion system on the EREV research vehicle represented a rear wheel drive architecture as shown in Figure 3.1. The system includes two independent motors, one on each of the rear wheels, coupled to a custom single speed straight cut planetary gearbox. The output from the individual gearboxes is connected to the corresponding rear wheel

through a half-shaft. The total power loss in this system is due to the inefficiencies in the motor and the custom gearbox. Constant efficiency assumption for the system reduced model accuracy and resulted in under estimating net energy consumption for low speed drive cycles. Chassis dynamometer testing results were used to estimate a combined efficiency map for the motor and gearbox assembly on the vehicle.

The propulsion system efficiency (η_{prop}) was computed using the power flow relation as shown in Eq.(3.3) earlier. Here, the motor electric power (P^{E}_{motor}) was determined from ESS discharge current and voltage while operating in charge depleting mode. The power output at the wheels (P_{demand}) was obtained from the chassis dynamometer. By aggregating recorded data from several dynamometer runs, the combined efficiency map as shown in Figure 3.6 was obtained. Half-shaft safety and ESS continuous power discharge limits prevented exceeding a commanded maximum torque of 180 Nm. This resulted in a partial estimation of the efficiency map for the propulsion system.

3.3.4 EREV Power Flow Model Validation

The EREV model discussed in the previous sub-sections was validated against measured power flow and energy consumption data from the EREV research vehicle. As ESS SOC is the main energy consumption state variable for an EREV, the validation process compares measured and model simulated ESS SOC data. Figures 3.7, 3.8, and 3.9 show these comparisons for EEC Cycle in Charge Depleting (CD) mode, Charge Sustaining (CS) mode and UDDS Cycle in a combined Charge Depleting - Charge Sustaining (CD-CS) mode. Simulation time step was observed to effect model accuracy. The time step was varied in the range of 0.05, 0.1, 0.2, 0.5, and 1.0 seconds. The accuracy was found to deteriorate with increasing time step. A time step of 0.2 seconds was chosen to balance out simulation time and model accuracy values.

ESS SOC comparison for EEC Cycle - CD as per Figure 3.7 establishes the presented

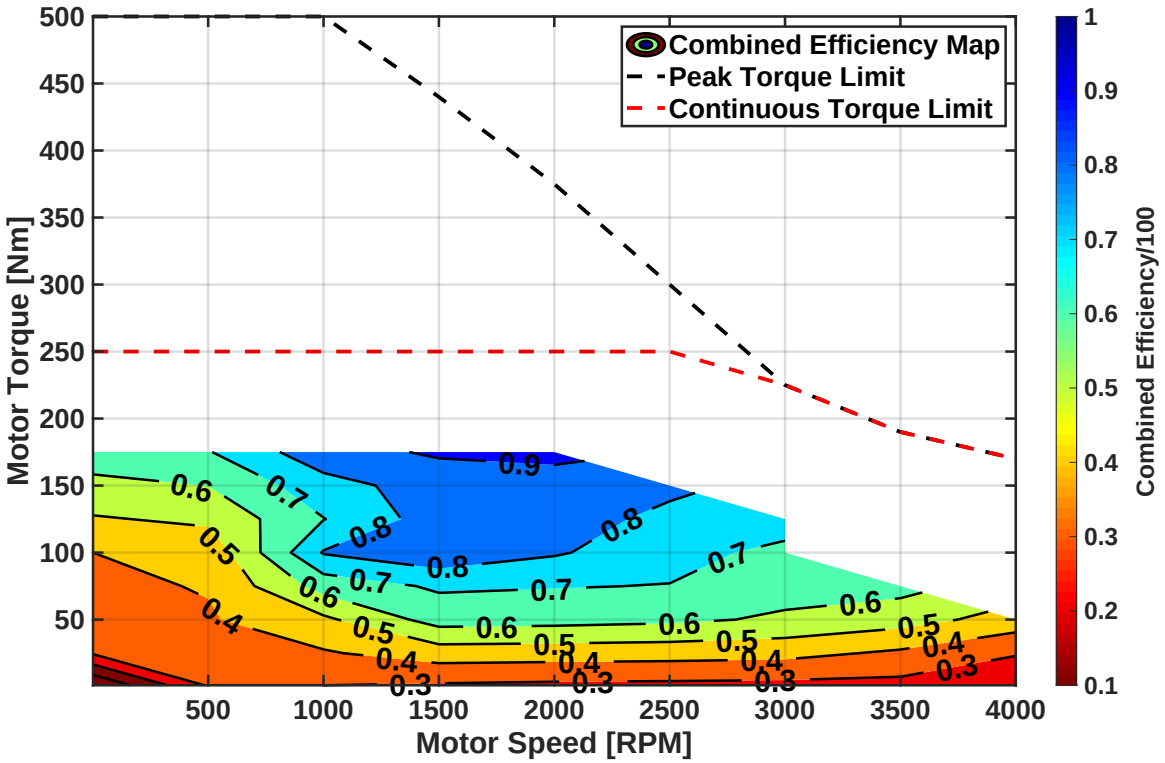
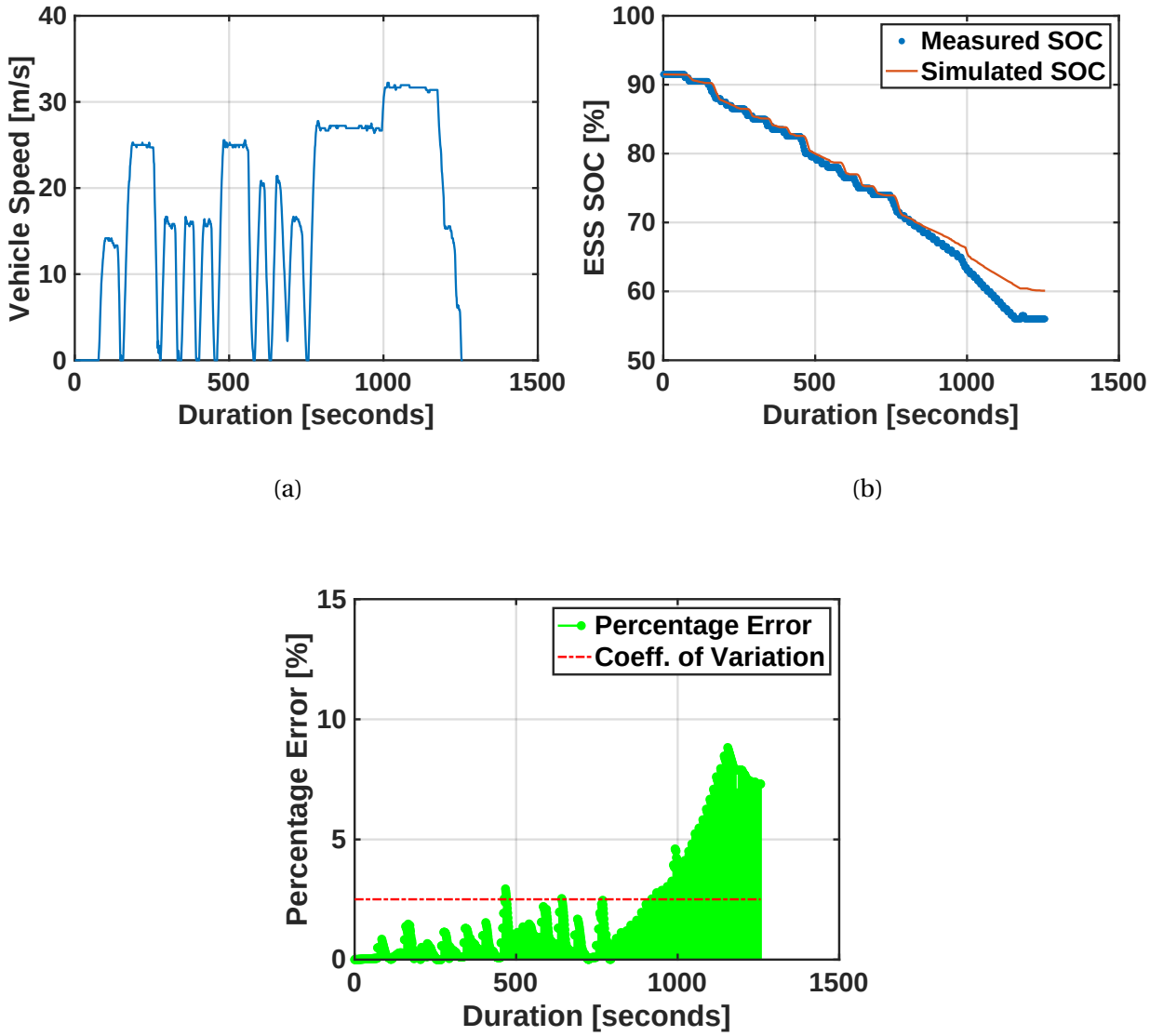


Figure 3.6: Estimated combined efficiency map for the motor-gearbox assembly.

EREV model with a high accuracy. The percentage error was computed as per Eq.(3.11) for all validation scenarios presented in Figures 3.8 and 3.9. Coefficient of Variation (C.V.) is also calculated as per Eq.(3.12) and shown as a dotted red line overlay.

$$Error\ [\%] = \frac{|SOC_{measured} - SOC_{simulated}|}{SOC_{measured}} \times 100 \tag{3.11}$$

$$C.V.\ [\%] = \frac{SOC_{RMSE}}{\overline{SOC}_{measured}} \times 100 \tag{3.12}$$

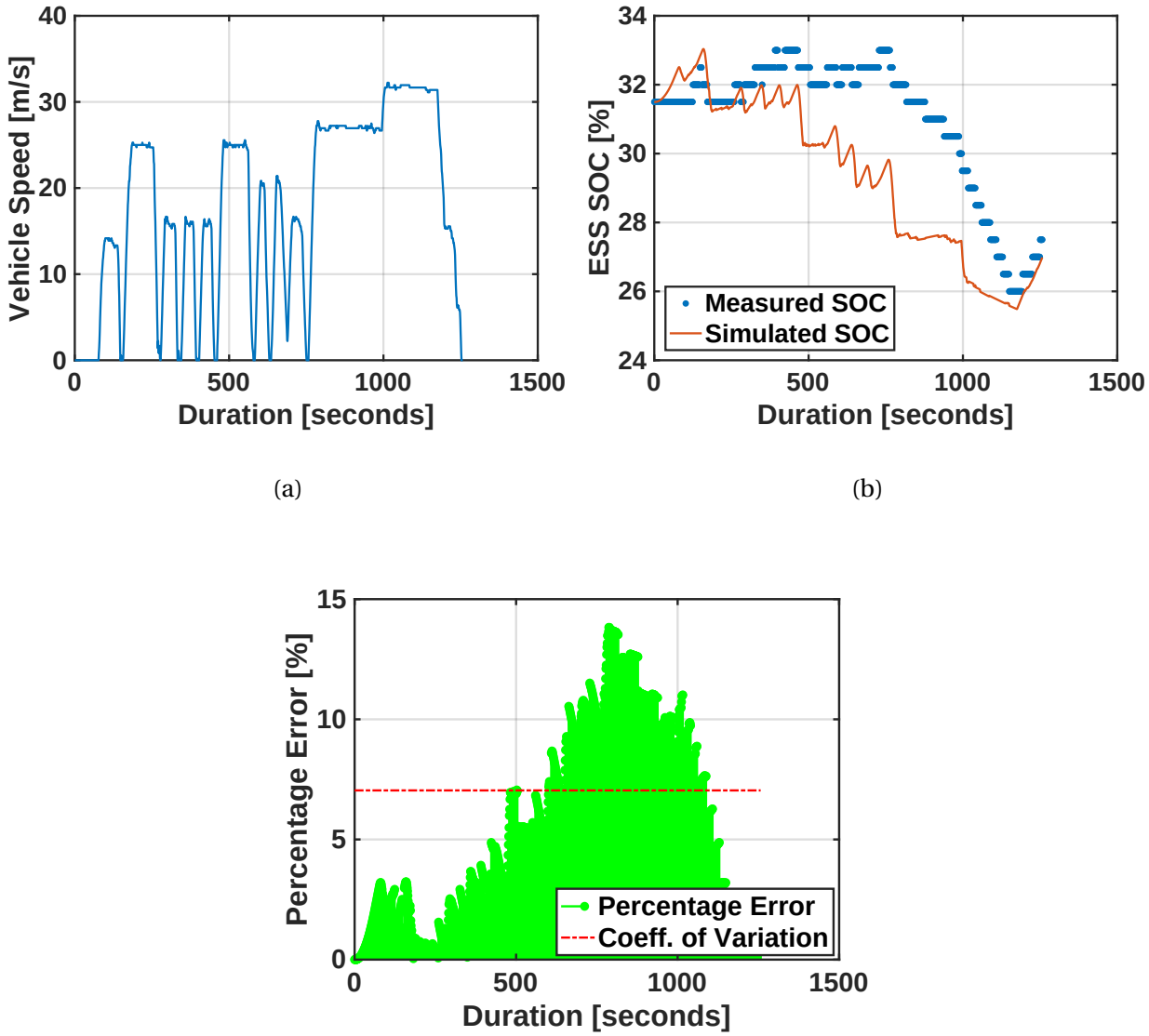


(a)

(b)

(c)

Figure 3.7: EREV research vehicle ESS energy consumption comparison for Charge Depleting behavior during EEC Cycle.

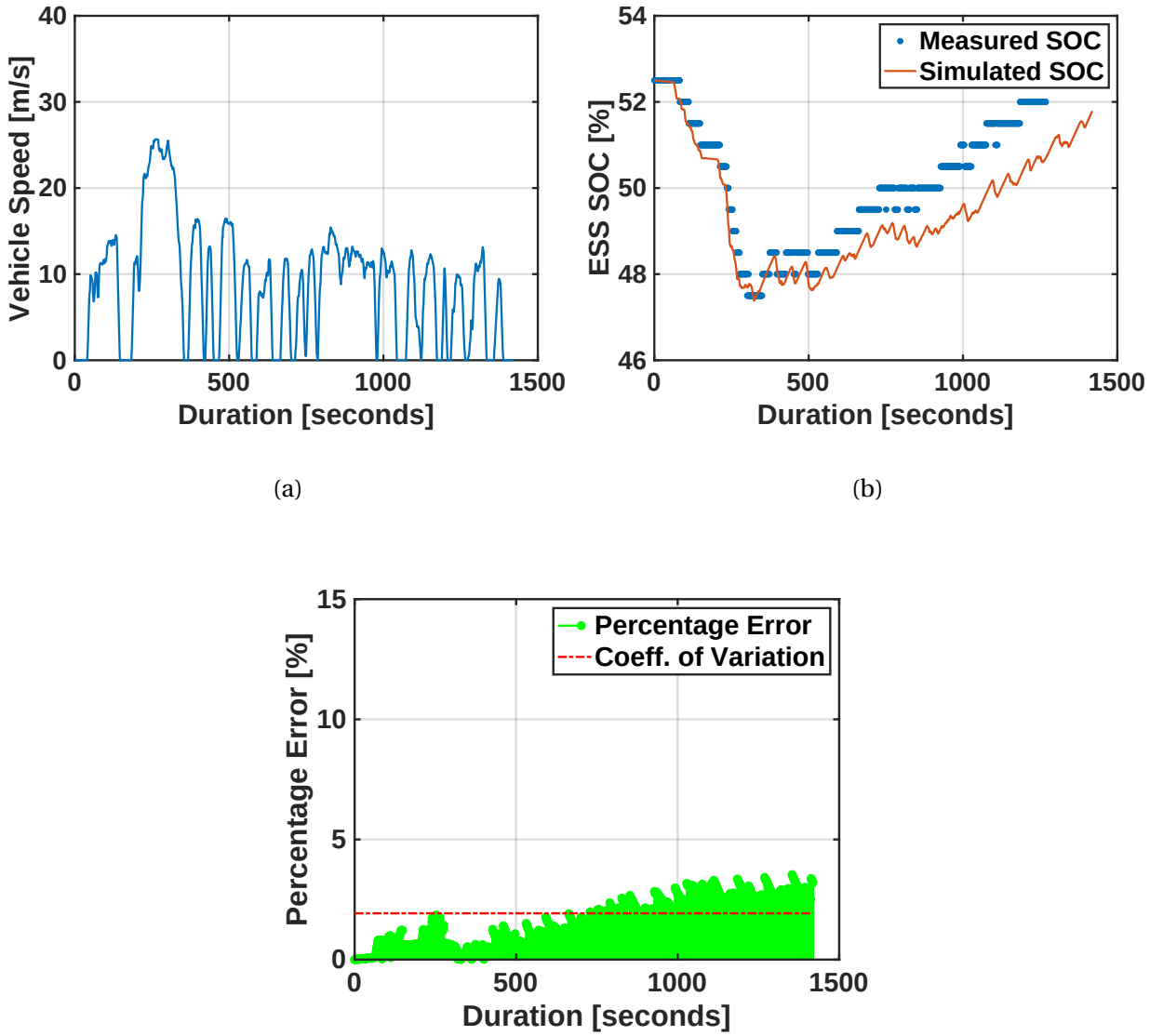


(a)

(b)

(c)

Figure 3.8: EREV research vehicle ESS energy consumption comparison for Charge Sustaining behavior during EEC Cycle.



(a)

(b)

(c)

Figure 3.9: EREV research vehicle ESS energy consumption comparison for Charge Depleting - CHarge Sustaining behavior during UDDS Cycle.

C.V. computed for the validation simulation results ranged between 1.9 - 7.0 %. Larger deviation in model accuracy was observed during charge sustaining mode in the drive. Discharge current comparison between measured and simulated values for CD and CS modes showed better accuracy in CD mode than CS. Thus, the deviation can be attributed to battery charge dynamics. As the computed C.V. values are still low, the model was considered validated for our study. In the next section, we present our work on energy management optimization for the EREV research vehicle.

3.3.5 Energy Management Optimization Algorithm

An EREV is capable of operating in two different energy modes, CD and CS, as mentioned earlier. Both literature and in-production EREVs show either a well-defined transition between these two modes or a blend of these as an energy management strategy. The intent of the on-board REx system is to only extend the driving range when required. An investigation of widely used energy management optimization algorithms was conducted for this EREV research vehicle. It showed that to attain the minimal energy consumption solution, one had to compromise on the feasible driving distance. Vice-versa, to obtain the maximum driving distance, the energy consumption was found to be sub-optimal.

Dynamic programming algorithm was used to determine global optimal solution for net energy consumption using the EREV research vehicle model. This algorithm is generally defined as backward propagating, where for n equally spaced time intervals between k stages, the process is started at k-th stage and traversed backwards. We implemented this algorithm in the forward propagating sense where the process started at the zeroth stage and traversed forwards. Equation (3.13) shows the optimal cost function (J_k^*) used in dynamic programming, called the Hamilton-Jacobi-Bellman equation in discrete time [51]. Here x_k and u_k correspond to the state and control variables corresponding to stage k, while $\hat{f}$ represents the cost associated with traversing from stage

$k-1$ to k and $\hat{g}$ represents the net cost associated till stage $k-1$. In this study, we use a time increment of 1.0 second for optimization, hence stage k also represents the time step.

$$J_k^*(SOC_k) = \min_{(P_{fuel},P_{ESS})_k \in U_{(P_{fuel},P_{ESS})_k}} \Big\{ \hat{f}(SOC_k,(P_{fuel},P_{ESS})_k) + J_{k-1}^*\big(\hat{g}(SOC_k,(P_{fuel},P_{ESS})_k)\big) \Big\} \forall k : 0 \leq k \leq n-1 \quad (3.13)$$

Four different drive cycles, namely EEC, HWFET, UDDS and US06 that emulate different power demand conditions were used. The ESS SOC state variable and the Driving Range state constraint were varied to perform 45 simulations per drive cycle. The simulated driving distance was varied by repeating the number of drive cycles as shown in Table 3.3 for reference. The final ESS SOC, maximum achievable driving range and net optimal energy consumption results are shown in Figures 3.10 - 3.12.

Table 3.3: Number of simulated cycles for Initial ESS SOC and Driving Range combinations.

Driving Range	**EEC Cycle**	**HWFET Cycle**	**UDDS Cycle**	**US06 Cycle**
50 mi($\approx$ 80.5 km)	3	5	7	6
75 mi($\approx$ 120.7 km)	5	7	10	9
100 mi($\approx$ 160.9 km)	7	10	13	12
125 mi($\approx$ 201.2 km)	9	12	17	16
150 mi($\approx$ 241.4 km)	11	15	20	19

The final ESS SOC results per Figure 3.10 show that for high power demand drive cycles such as EEC and US06, a feasible solution exists only for limited combinations

of initial ESS SOC and target driving distance. From HWFET and UDDS evaluation results, we observed that the EREV was able to meet the final ESS SOC requirements for more simulation scenarios in case of HWFET than UDDS. This outcome is counter intuitive to our initial expectation, due to an expected lower energy demand per cycle in case of UDDS. To maintain similar driving distances, the number of UDDS cycles was higher than HWFET resulting in a higher net energy demand and complete depletion of fuel reserve. On overlaying ESS SOC results with net optimal energy consumption results per Figure 3.11, lowest values are achievable only for 50mi driving range and 100% initial ESS SOC amongst the simulated drive cycles. As the driving range requirement increases or initial ESS SOC value decreases, the energy consumption values increase substantially. This result is supported by maximum driving range results as per Figure 3.12. Thus based on the dynamic programming based simulations, an optimal energy consumption for the EREV is highly dependent on driving range requirement and initial ESS SOC.

With a baseline available for our work using dynamic programming, we utilized another energy management optimization algorithm based on Pontryagin's Minimization Principle (PMP) [51]. This approach is also applicable only with *priori* knowledge of the vehicle behavior. In this algorithm the hamiltonian as shown in Eq.(3.14) is computed at every time step and minimized to obtain the optimal control parameters.

$$H(P_{fuel}[k], P_{ESS}[k], \lambda[k], p_1(SoC)) = \\ P_{fuel}[k]\big|_{P_{fuel}\in U_{P_{fuel}}} + \Big(\lambda[k] + p_1(SoC)\Big) \cdot P_{ESS}[k]\big|_{P_{ESS}\in U_{P_{ESS}}} \quad (3.14)$$

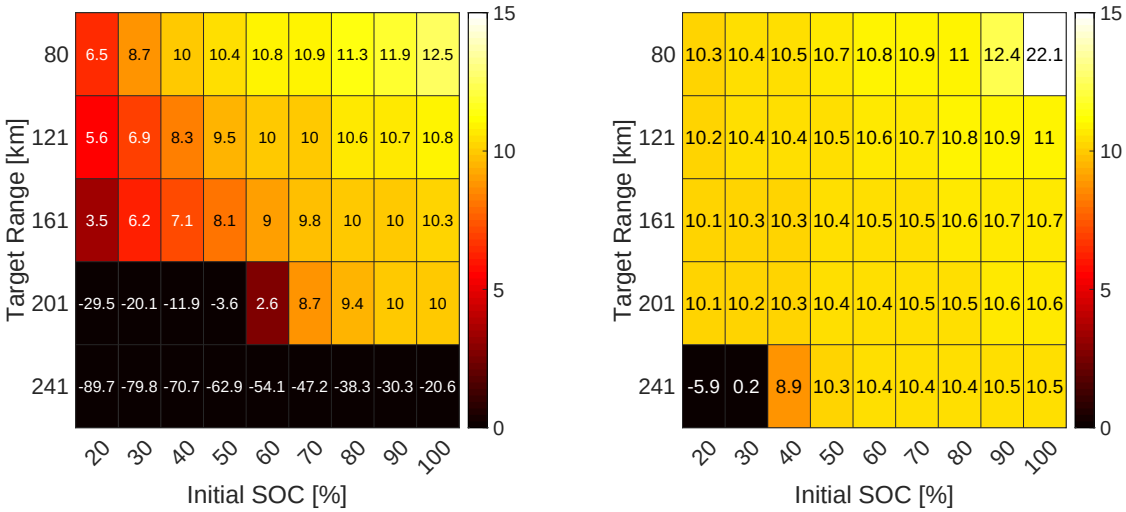
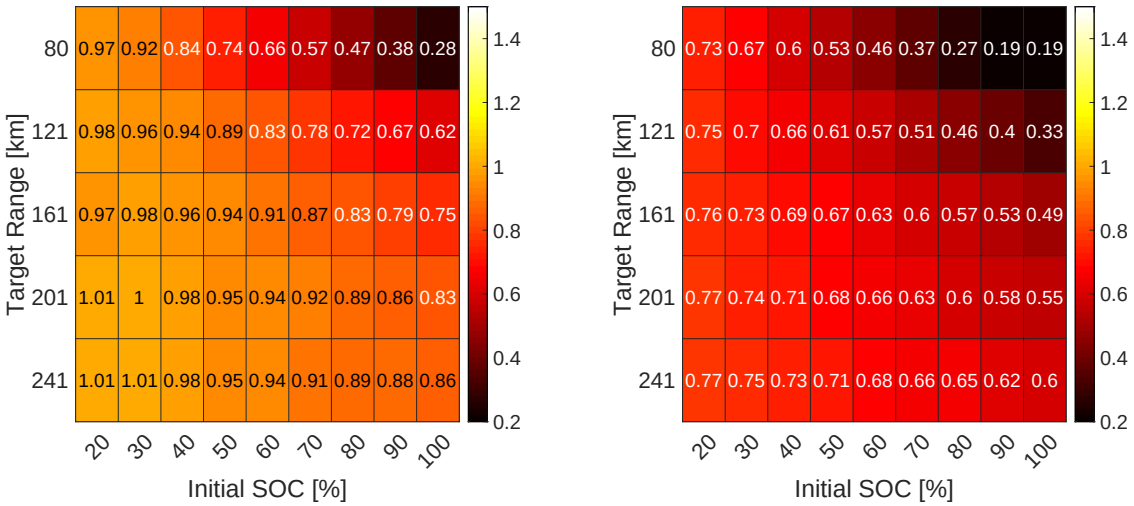
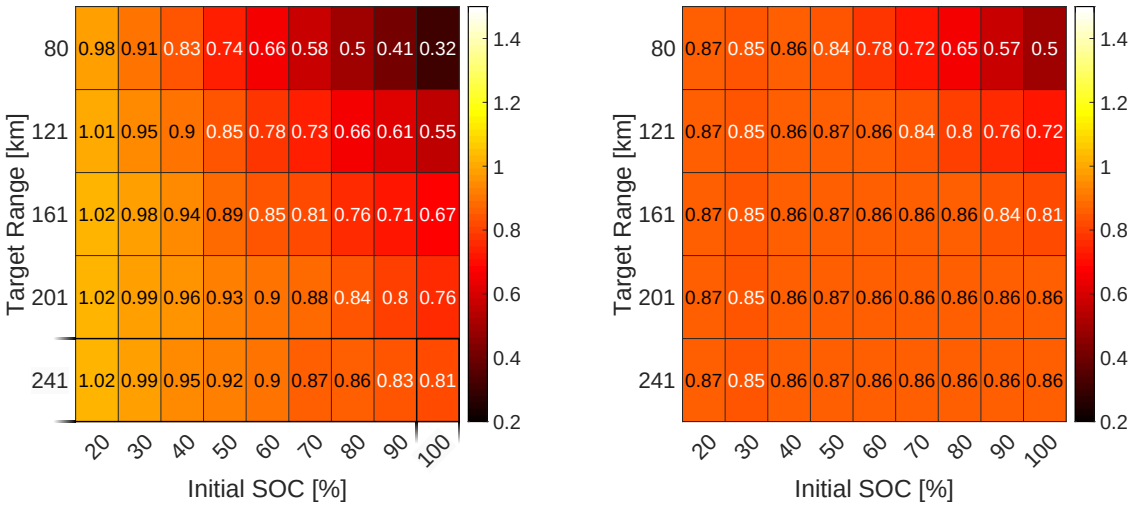


Figure 3.10: Final ESS SoC (%) achieved matrix for (a) EEC, (b) HWFET, (c) UDDS, and (d) US06 drive cycle simulations using forward propagating Dynamic Programming.



(a)

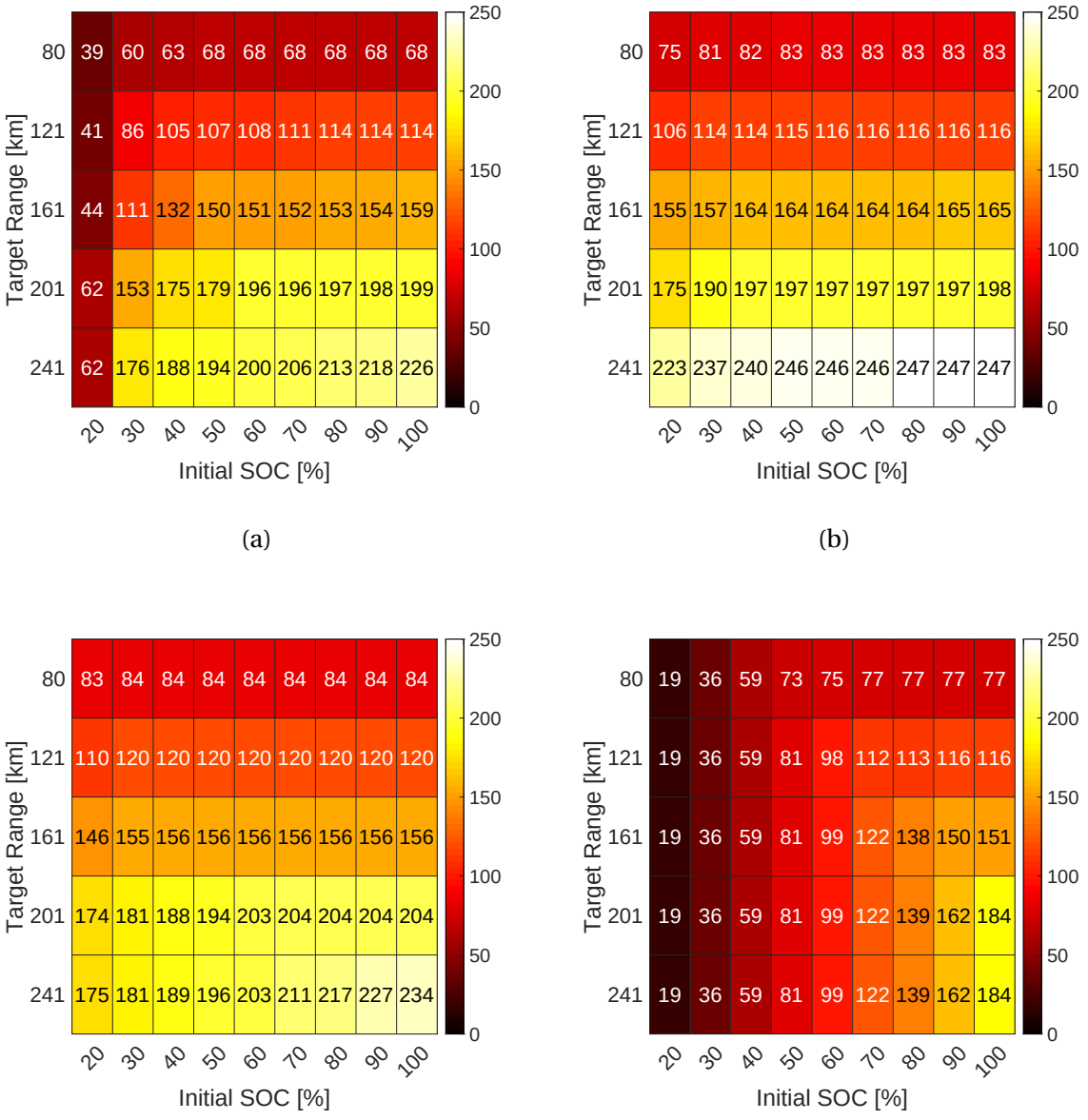
(b)



(c)

(d)

Figure 3.11: Net optimal energy consumption (kWh/km) matrix for (a) EEC, (b) HWFET, (c) UDDS, and (d) US06 drive cycle simulations using forward propagating Dynamic Programming.

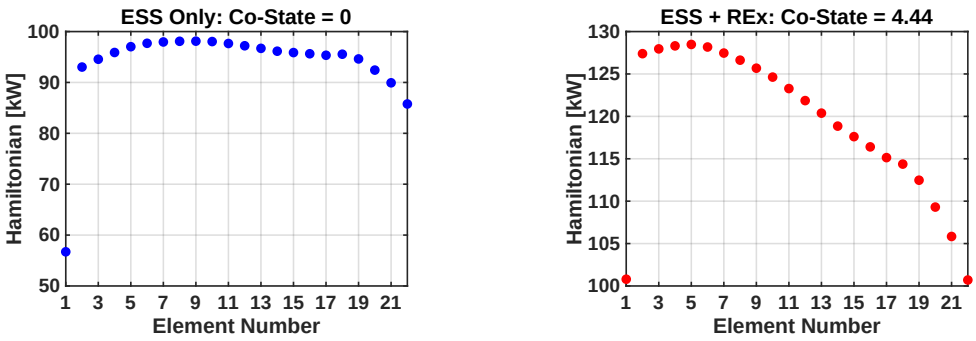


(a) (b)

(c) (d)

Figure 3.12: Maximum achievable driving range (km) matrix for (a) EEC, (b) HWFET, (c) UDDS, and (d) US06 drive cycle simulations using forward propagation Dynamic Programming.

Based on the feasible power values in $U_{P_{fuel}}$, $U_{P_{ESS}}$ and the co-state parameter (λ) that satisfies SOC constraint the hamiltonian obtained is shown in Figure 3.13. The plot shows hamiltonian corresponding to a feasible set of power with 27 elements computed at time step $k = 1150$ seconds for a single cycle of EEC. As per the figure, at any time-step k there are only two feasible options for an optimal solution. One corresponding to ESS only operation and the other corresponding to ESS + REx operation. These simulations were performed on the EEC, HWFET, UDDS, and US06 drive cycles and the observed ESS SOC behavior is shown in Figure 3.14. For the US06 drive cycle, due to the high power demand (as noted through dynamic programming results earlier), even with maximum REx operation the ESS SOC approaches the lower limit. In case of HWFET and UDDS drive cycles, a sharp drop in ESS SOC is observed for ESS + REx operation due to fuel reserve depletion. A key takeaway from these results is that based on the final SOC constraint, only two feasible operating solutions exist. The ESS only solution provides the best energy consumption, while the ESS + REx solution provides the maximum range. This limits the range of operation of the REx and adaptability to variability in drive power demand.



(a) (b)

Figure 3.13: Hamiltonian in the feasible set for ESS only and ESS + REx operation during EEC 3 and 9 cycle optimization taken at $t = 663.0$ seconds, respectively.

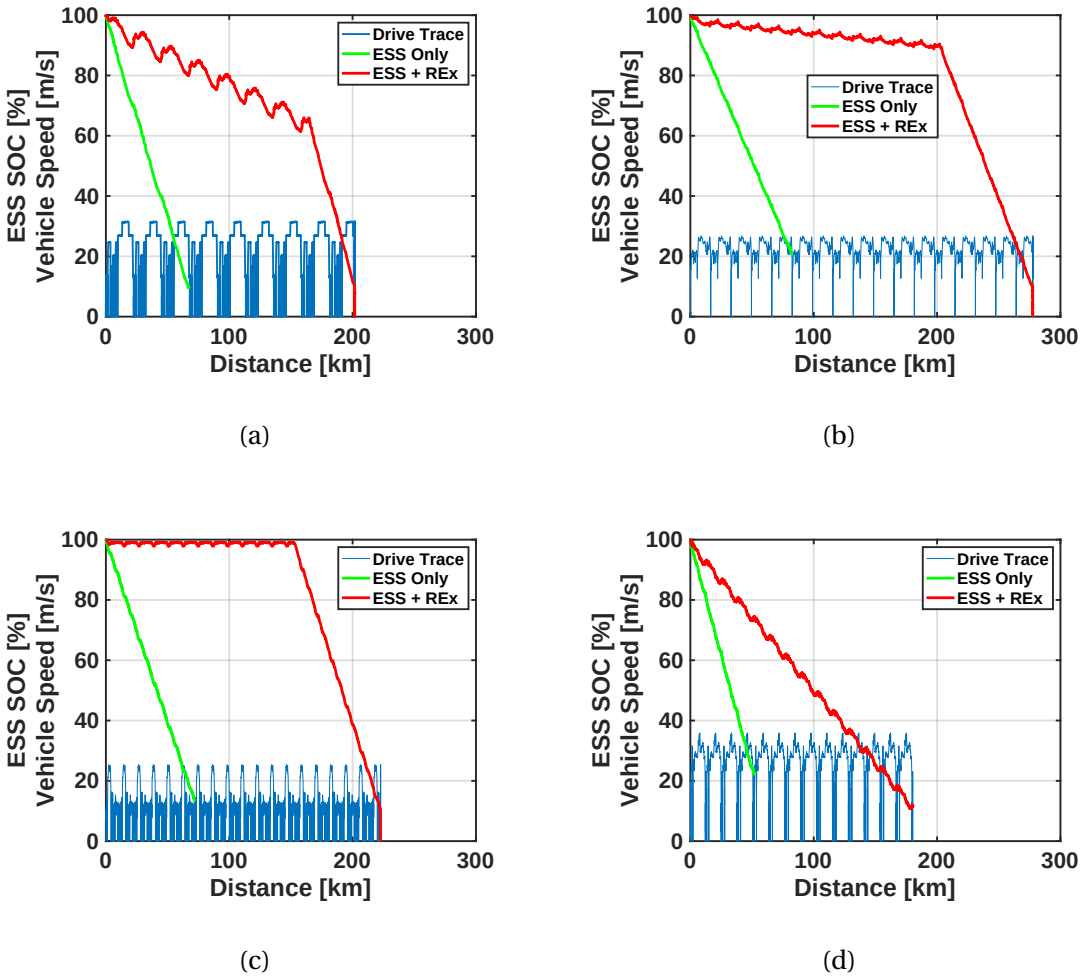


Figure 3.14: ESS SoC and maximum driving range comparison for optimal energy consumption (ESS only) and maximum driving range (ESS + REx) amongst (a) EEC, (b) HWFET, (c) UDDS, and (d) US06 drive cycle simulations using Pontryagin's Minimum Principle.

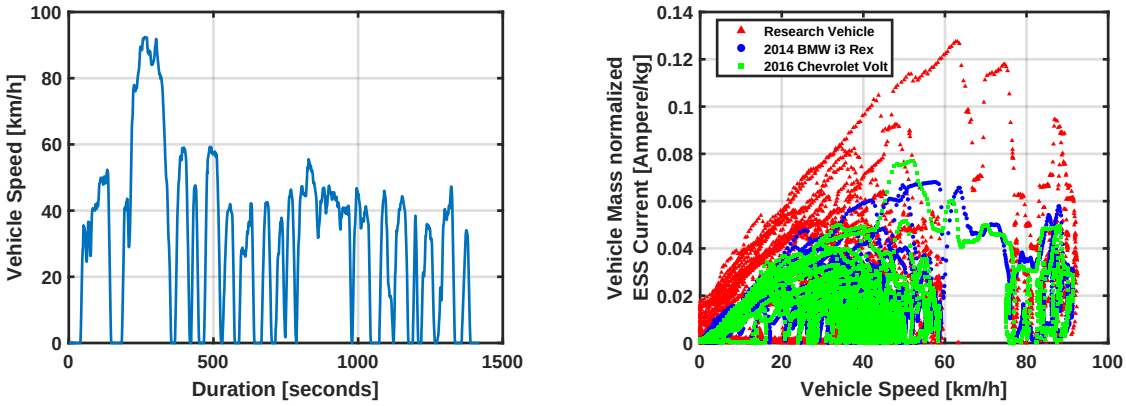
With the aim to implement an optimal control algorithm on the research vehicle, real-time online energy management optimization algorithms were evaluated as mentioned in the Introduction section. ECMS and its variants have been proven in literature

to have similar outcome as PMP and are also applicable only for HEVs operating in CS mode. As the PMP evaluation showed only two optimal REx operation set points and the vehicle operates in both CD and CS modes, ECMS was discarded as a possible optimization algorithm.

Amongst the in-production EREVs, 2014 BMW i3 Rex and 2016 Chevrolet Volt, vehicle dynamometer test data was obtained from Argonne National Laboratory (ANL) and analyzed for energy management strategy. Both of these vehicles perform in a distinct CD and CS mode and at an ESS SOC level of $\sim$15.7%, the CS mode was observed to initiate. ESS power analysis and its comparison with vehicle speed showed that the REx system follows an offline optimized energy management strategy which targets maintaining ESS SOC level close to 15.7%. For the UDDS cycle, discharge current normalized with vehicle mass as a function of vehicle speed was compared between our research vehicle and the two in-production EREVs mentioned earlier as shown in Figure 3.13. It is evident from the figure that our research vehicle normalized current demand is nearly two times greater than the in-production vehicles. Due to the operation power limit on the REx system in the research vehicle, implementing a generic CS mode as observed in the in-production vehicles will be inconsequential. Thus, for an energy management strategy similar to in-production vehicles, the research vehicle has a higher chance of getting depleted out of it's maximum capacity, thus limiting the feasible driving range. Therefore, a more blended CD-CS strategy is required for the research vehicle to be operational within the imposed constraints.

Stochastic dynamic programming for energy management has been shown to operate in a real-time online environment but requires offline optimization over a variety of drive cycles to generate optimal transition policies while traversing between different energy states. In our evaluation, dynamic programming optimization results showed high adaptability to changes in drive scenarios. This motivated us to develop an optimal energy management algorithm based on Adaptive Real-Time Dynamic Program-

ming, that can generate optimal control parameters for power distribution in our EREV. To overcome the challenge of obtaining *priori* information about the drive, we have integrated Google API functions to create an estimated drive trace for the planned drive. The two key elements of our proposed algorithm are: i) using a range constraint parameter to variate power distribution between ESS and REx and, ii) adapting to changes during actual driving by monitoring energy consumption residuals.



(a) (b)

Figure 3.15: Energy consumption comparison on a) UDDS cycle with b) Vehicle mass normalized ESS current variation in CD mode between Research Vehicle, 2014 BMW i3 Rex and 2016 Chevrolet Volt.

3.3.6 Distance Constrained Dynamic Programming

The results from dynamic programming evaluation in Figures 3.10 - 3.12, showed that the net energy consumption worsened as simulated driving distance increased. If the required driving range is known prior to the drive or can be treated as a user-input variable, the optimal power distribution between the ESS and REx within these new constraints can be obtained. For this variant of the algorithm, we still use the cost func-

tion as shown in Eq.(3.13). Though the feasible set of operating parameters is subjected to the driving range constraint condition. Here, the feasible sets of P_{fuel} and P_{ESS} i.e. $U_{P_{fuel}}$ and $U_{P_{ESS}}$, are derived from the feasible set of ESS SOC at each time step k i.e. $U_{SOC_k} \vdash U_{P_{fuel_k}}, U_{P_{ESS_k}}$. U_{SOC_k} itself is determined from the net power demand and range extender available power i.e. P_{demand_k} and P_{REx_k}, where $P_{REx} \vdash P_{fuel}$. With the additional range constraint, the feasible set of ESS SOC is represented as $U^d_{SOC_k}$ such that $U^d_{SOC_k} \subseteq U_{SOC_k}$. The range constraint changes the lower limit of the set U_{SOC_k} i.e. at time step k, if $U_{SOC_k} \in [\alpha, \beta]$ such that $\alpha < \beta$, then $U^d_{SOC_k} \in [\alpha^{'}, \beta]$ such that $\alpha < \alpha^{'} < \beta$, to form the new feasible subset of operating parameters. The range constrained feasible set $U^d_{SOC_k}$ is computed at every time step k as per Algorithm 2.

At any time step k, the algorithm takes, SOC range precision factor (γ), distance travelled (d), distance remaining (d_{rem}), total energy capacity of ESS (E_{ESS}), ESS energy consumption (EC), power demand P_{demand} and the optimal SOC i.e. SOC^*, from the previous time step $k-1$ and initialization. Using these inputs, the lower and upper limits to U_{SOC_k} i.e. $\alpha[k]$ and $\beta[k]$ are computed. Here $\alpha[k]$ corresponds to ESS only operation, while $\beta[k]$ corresponds to ESS and maximum REx operation. With γ = 2e-5, which is equivalent to 0.002% SOC and provides a power precision of 1360.8 W every time step k and the limits, U_{SOC_k} is obtained. In Figure 3.14 at time step k = 1 the blue circles represent the feasible ESS SOC set, U_{SOC_k}. To determine $U^d_{SOC_k}$, we next compute U_{EC_k} which forms a set of EC at time step k and is further used to compute $U_{d_{fsbl_k}}$ which is a set of feasible distance the vehicle can travel. Here, the feasible distance computation is based only on the remaining ESS SOC. This provides the worst case scenario solution, as ESS energy is the limiting factor for an EREV. Finally, $U^d_{SOC_k}$ is compared against the desired distance target (d_{target}) and infeasible $SOC[k]$ values are changed to ∞. In Figure 3.16 at time step k = 2 the blue circles represent the range constrained feasible ESS SOC set, $U^d_{SOC_k}$ and the yellow circles represent SOC values which are infeasible due to the range constraint. This process continues $\forall k \in [0, n]$. The impact of adding range constraint to

dynamic programming optimization is elaborated in the Results section.

Algorithm 2: Compute feasible set of ESS SOC subject to driving range constraint at time step k.

Input: $\gamma, d[k], d_{rem}[k], E_{ESS}, EC[k], P_{demand}[k], SOC^*[0], SOC^*[k-1]$

Output: $U^d_{SOC_k}$

$\alpha[k] \longleftarrow SOC^*[k-1] - \left(\frac{P_{demand}[k]}{E_{ESS} \cdot 3600}\right)$;

$\beta[k] \longleftarrow SOC^*[k-1] - \left(\frac{P_{demand}[k] + max(P_{REx})}{E_{ESS} \cdot 3600}\right)$;

$U_{SOC_k} \longleftarrow \Big[\alpha[k] + n\gamma \mid n \in \mathbb{Z}^+, max(n) = \frac{\beta[k]-\alpha[k]}{\gamma}, \alpha[k], \beta[k] \neq \infty\Big]$

$U_{EC_k} \longleftarrow \frac{1}{d[k]}\left[\Big(SOC^*[0] - SOC[k]\Big) \cdot E_{ESS}\right] \qquad \forall SOC[k] \in U_{SOC_k}$

$U_{d_{fsbl_k}} \longleftarrow \left(\frac{SOC[k]-SOC_{min}}{EC[k]}\right) - d_{rem}[k] \qquad \forall SOC[k] \in U_{SOC_k}, \forall EC[k] \in U_{EC_k}$

for d *in* $U_{d_{fsbl_k}}$ **do**

  if $d < d_{target}$ **then**

    $SOC[k] \longleftarrow \infty$

  end

end

$U^d_{SOC_k} \longleftarrow \Big[\alpha'[k] + n\gamma \mid n \in \mathbb{Z}^+, max(n) = \frac{\beta[k]-\alpha'[k]}{\gamma}, \alpha'[k], \beta[k] \neq \infty\Big]$

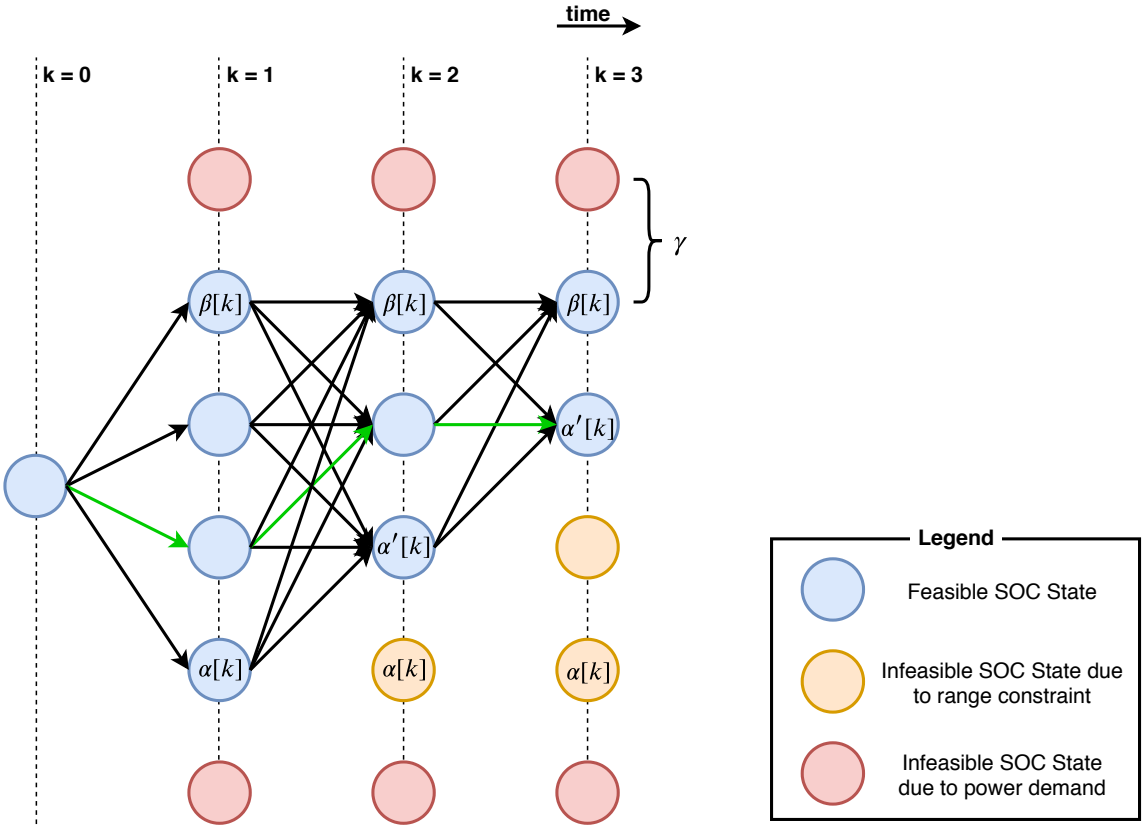


Figure 3.16: Dynamic Programming with Range Constraint network flow diagram.

3.3.7 Adaptive Distance Constrained Dynamic Programming

The estimated average vehicle speed obtained from Google API is helpful for the proposed optimization. Though the variability in actual driving and real-time changes in traffic conditions can cause the energy consumption to deviate from the pre-drive optimal solution. The effect of this variability was evaluated over a randomized drive trace bounded within speed limits with reference to average vehicle speed obtained from Google. Equation(3.15) shows the relation to obtain this random new speed from a normal distribution $\mathcal{N}(\mu, \sigma^2)$, where μ is the new speed at previous time step $k-1$ i.e. $v_{new}[k-1]$ and σ^2 is the variance. The variance was selected to be 0.0625 to limit the

magnitude of instantaneous acceleration within drive comfort i.e. 0.2G. A speed limit parameter is also used to prevent the new randomized speed deviate from the average vehicle speed obtained from Google API.

$$v_{new}[k] \sim \mathcal{N}(v_{new}[k-1], \sigma^2) \qquad \forall k \in [1, t_f] \tag{3.15}$$

To compare the effect of variation in vehicle speed against the average vehicle speed used for optimization, we monitor the ESS SOC state variable. It was observed that for a randomized vehicle speed obtained from Eq.(3.15), a deviation of $\sim 3\%$ ESS SOC can occur at the very least. This deviation can cause a violation of the driving range constraint or can increase the net energy consumption. Thus, the optimal solution needs to be re-computed to avoid these violations. For this we calculated the absolute difference in magnitude between the optimal ESS SOC and actual ESS SOC, called the ESS SOC residual, as the vehicle covers required distance. If the ESS SOC residual exceeds the limit, it re-computes the optimal solution for the remaining ESS energy and distance.

3.3.8 Proposed Real-Time Application of Adaptive Distance Constrained Dynamic Programming

The proposed energy management optimization scheme is subdivided into two phases, 1) Pre-drive optimization, and 2) Real-time re-optimization. As the proposed scheme is based on Dynamic Programming algorithm, *priori* knowledge of the drive is required for both the phases. Thus, we implemented a tool that uses Google Directions API to generate an estimated speed trace of the journey based on user inputs for Origin and Destination locations. The API uses Google's "best guess" traffic model to determine the duration of travel between different path nodes that is crucial in building the estimated speed trace. Figure 3.18 shows the output of the estimated speed trace generating tool. In this example scenario, the speed trace is generated from coordinates [47.6536,-

122.3046] to [47.4247,-121.5834], which results in an 84.5 km long travel. To emulate variability in driving, we used the two randomized speed traces shown in Figure 3.18(c).

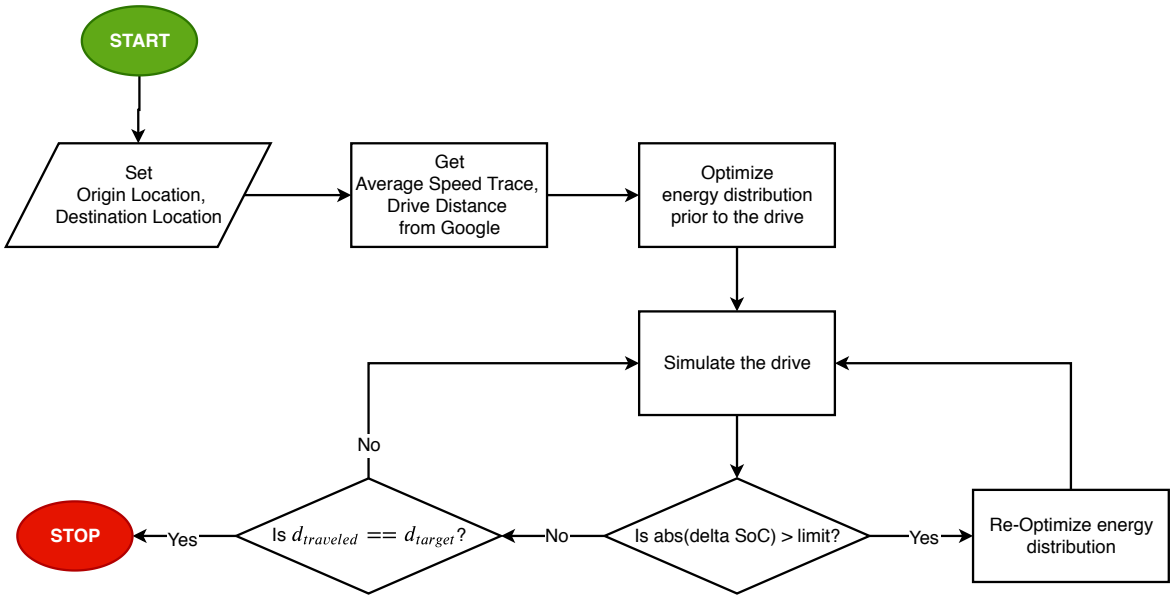
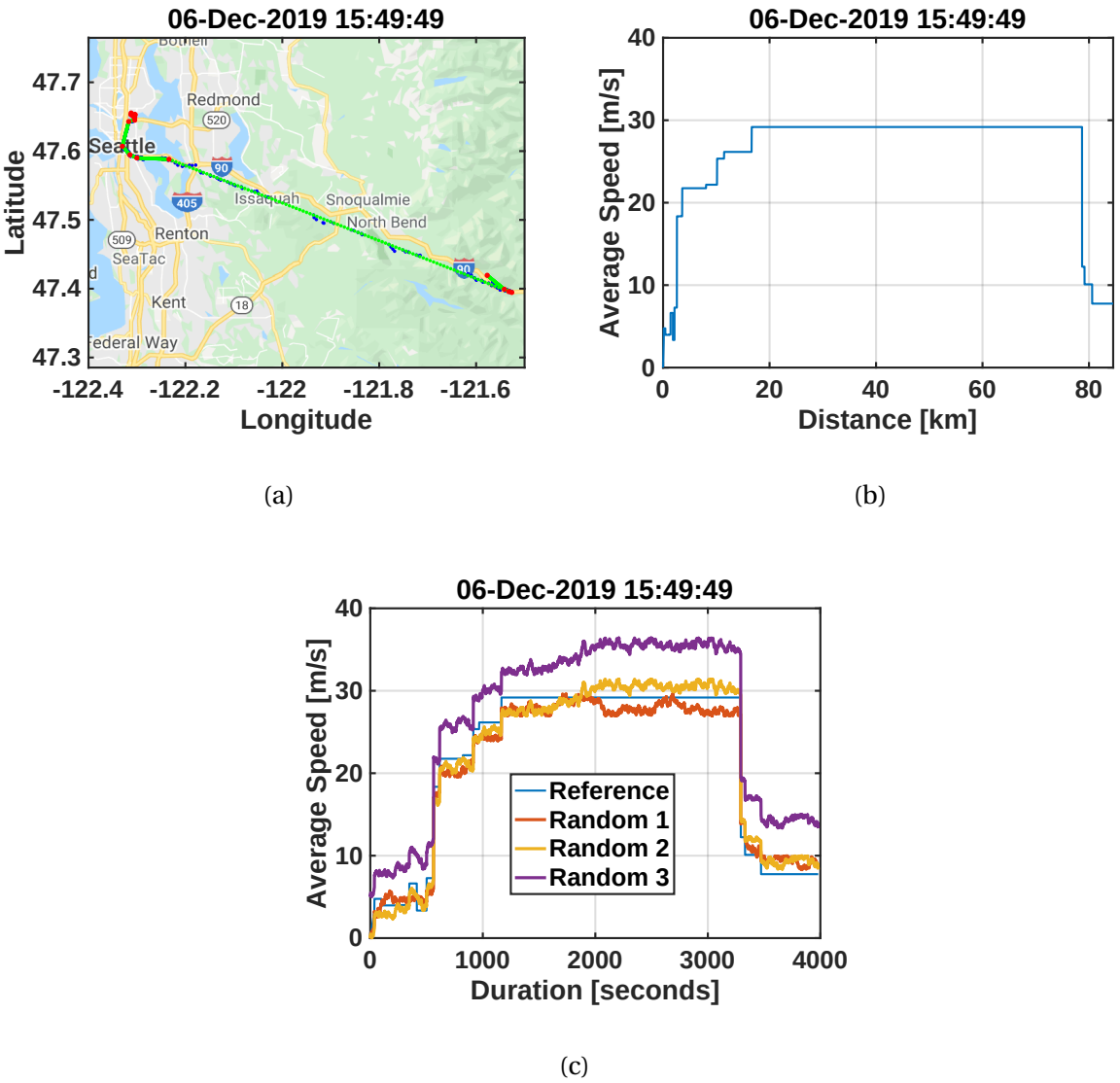


Figure 3.17: Proposed optimal energy management algorithm flowchart.

Using this *priori* knowledge about the drive, the proposed scheme performs Phase 1 and generates an optimal set of control parameters for the EREVs REx system. As the simulation proceeds, the ESS SOC residual is computed and compared against a pre-defined limit. If the residual exceeds the limit, it initiates Phase 2 i.e. re-optimization of control parameters. This re-optimization only occurs for the remaining travel distance and available energy. The updated control parameters are now used for the remaining drive. This process of re-optimization is part of an iterative decision loop and thus can occur as many times the ESS SOC residual exceeds the limit. The process flowchart for this proposed scheme is represented in Figure 3.17.



(a)

(b)

(c)

Figure 3.18: Estimated speed trace build using Google Directions API. (a) Shows path nodes from Origin to Destination, (b) Average vehicle speed determined from current traffic conditions, and (c) Randomized speed traces based on the average speed.

3.4 Results and Discussion

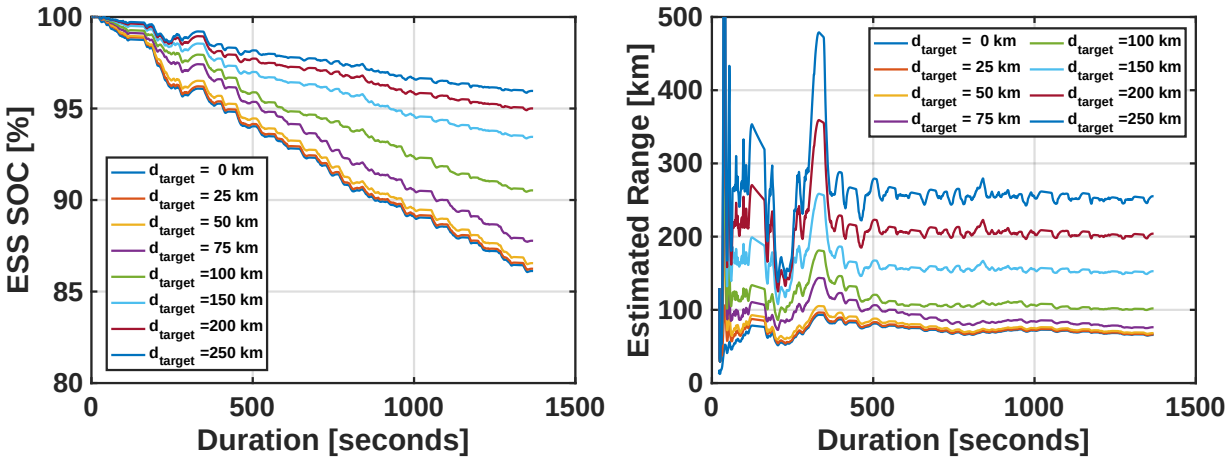
In this section we provide simulation results showing key stages of evaluation of this proposed algorithm. First we present the implementation of dynamic programming algorithm with a distance constraint variable and its impact on energy consumption of the research vehicle. We then present results which show a comparison of the proposed algorithm with and without the re-optimization process. Finally, a comparison of the proposed algorithm against the conventional CD-CS algorithm used in EREV energy management is provided. Interpretation of these results, the assumptions made and limitations are also discussed.

3.4.1 Implementation of Distance Constrained Dynamic Programming

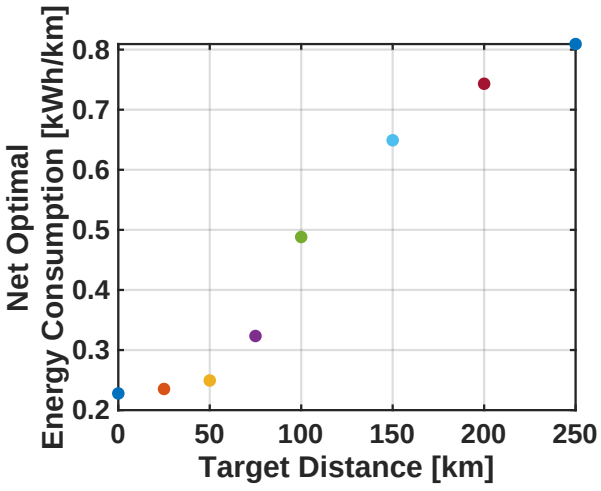
The dynamic programming algorithm with distance constraint as described in the previous section was implemented on the research vehicle model. A simulation sweep for the UDDS and US06 cycles with varying target distances, d_{target} = [0, 25, 50, 75, 100, 150, 200, 250] km was conducted. For the UDDS cycle scenario as shown in Figures 3.19(a)-(c), the algorithm was able to optimize for all values of d_{target}. The net optimal energy consumption values increased with an increasing d_{target}. A minimum of 0.23 kWh/km and a maximum of 0.81 kWh/km net energy consumption was obtained for the UDDS cycle evaluation.

For the US06 cycle scenario as shown in Figures 3.20(a)-(c), the algorithm was able to optimize for all values of d_{target} except 150, 200 and 250 km as shown by triangle (▲) markers in Figure 3.20(c). The net optimal energy consumption in this case also showed a similar increasing trend as in the case of UDDS cycle. A minimum of 0.25 kWh/km and a maximum of 0.53 kWh/km value was obtained for successfully optimized US06 cycle evaluations. A key difference between the UDDS and US06 cycle was a consistently high energy consumption value for all optimized d_{target} values. This observation is consistent

with US06 being a relatively higher speed cycle than UDDS.

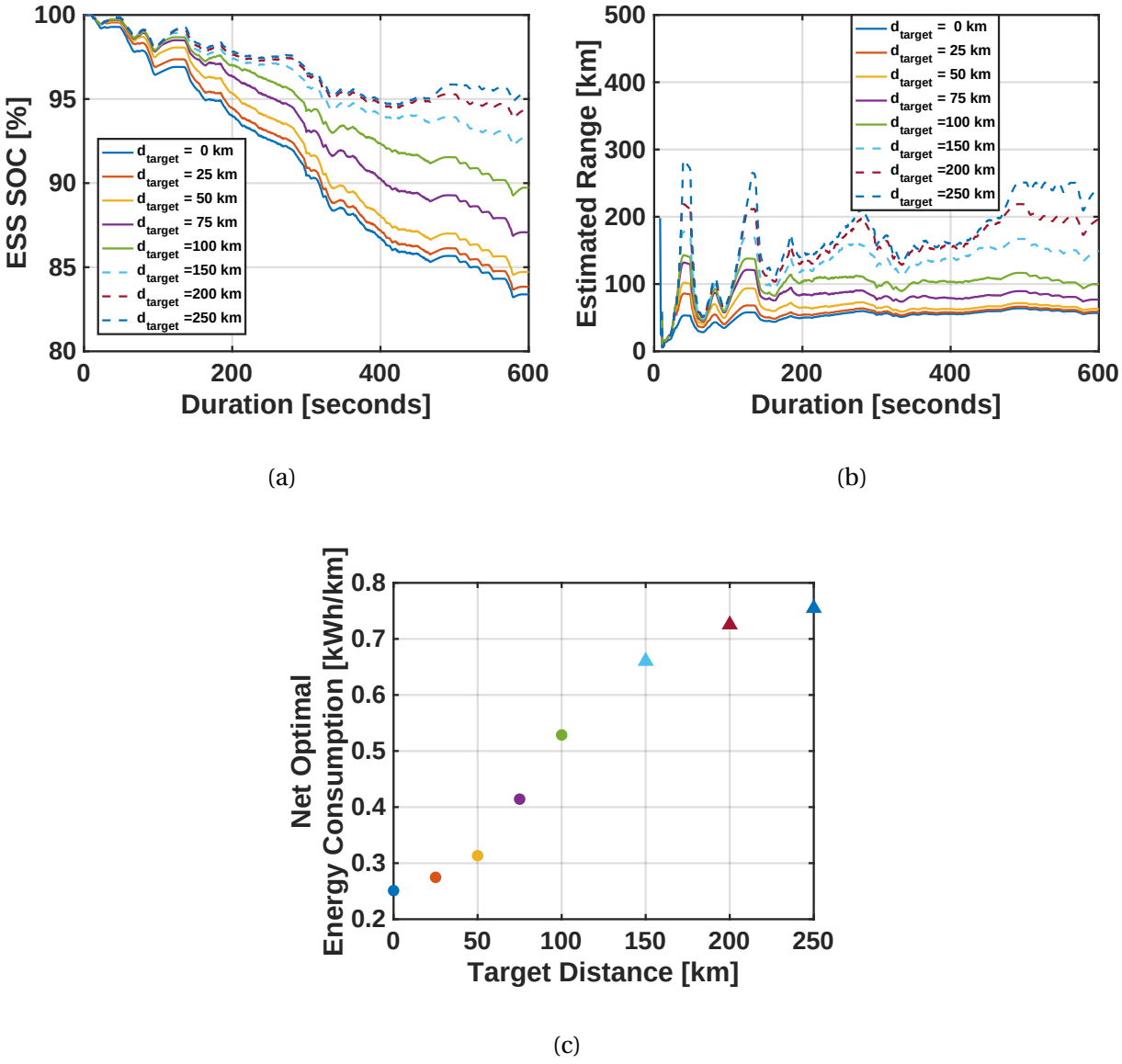


(a) (b)



(c)

Figure 3.19: Distance constrained dynamic programming optimization results on research vehicle for UDDS Cycle.



(a)

(b)

(c)

Figure 3.20: Distance constrained dynamic programming optimization results on research vehicle for US06 Cycle.

The vehicle energy management inherently prefers to only use ESS energy in order to maintain a low net energy consumption. Such an outcome reduces EREVs feasibility to

meet the required target distance. Hence, our implementation of distance constrained dynamic programming for EREV energy management showed that the feasible set of solutions at each stage of DP can be varied with the target distance parameter. This allowed the vehicle to meet the target distance while being optimal in net energy consumption. Based on the UDDS and US06 cycle simulation results shown in Figures 3.19 and 3.20, the final ESS SOC can be observed to be higher as target distance increases. This was expected, as a higher target distance would require the REx system to be operational and provide supplemental energy. In Figure 3.20(a), the optimal solutions for target distances 150, 200 an 250 km are in violation of the distance constraint but the optimal solution for d_{target} = 250 km does not violate d_{target} = {150, 200} km. This implies that in some cases for a solution to not violate both final ESS SOC and target distance constraints, it will be sub-optimal for net energy consumption.

An assumption for computing the feasible driving distance was made based on EREV behavior. We assumed that the available energy at any point of time during the drive is the energy available in ESS only. An EREVs available driving range in the worst case scenario i.e. if REx system is non-functional, is solely dependant on the available ESS energy. Thus, if we optimize with the worst case scenario, we can always ensure that the vehicle will be able to meet the ESS SOC and target distance requirements, if successfully optimized. Based on these results, we were able to establish a functional dynamic programming algorithm with distance constraint.

3.4.2 Adaptive Distance Constrained Dynamic Programming

The distance constrained dynamic programming algorithm was integrated with an ESS SOC residual based re-optimization. This approach made the algorithm adaptive to variation in driving and by extension energy usage. This approach was evaluated for standard drive cycles and for custom drive cycles obtained from Google Directions API. The results presented in this section are for a custom drive cycle as shown in Figure

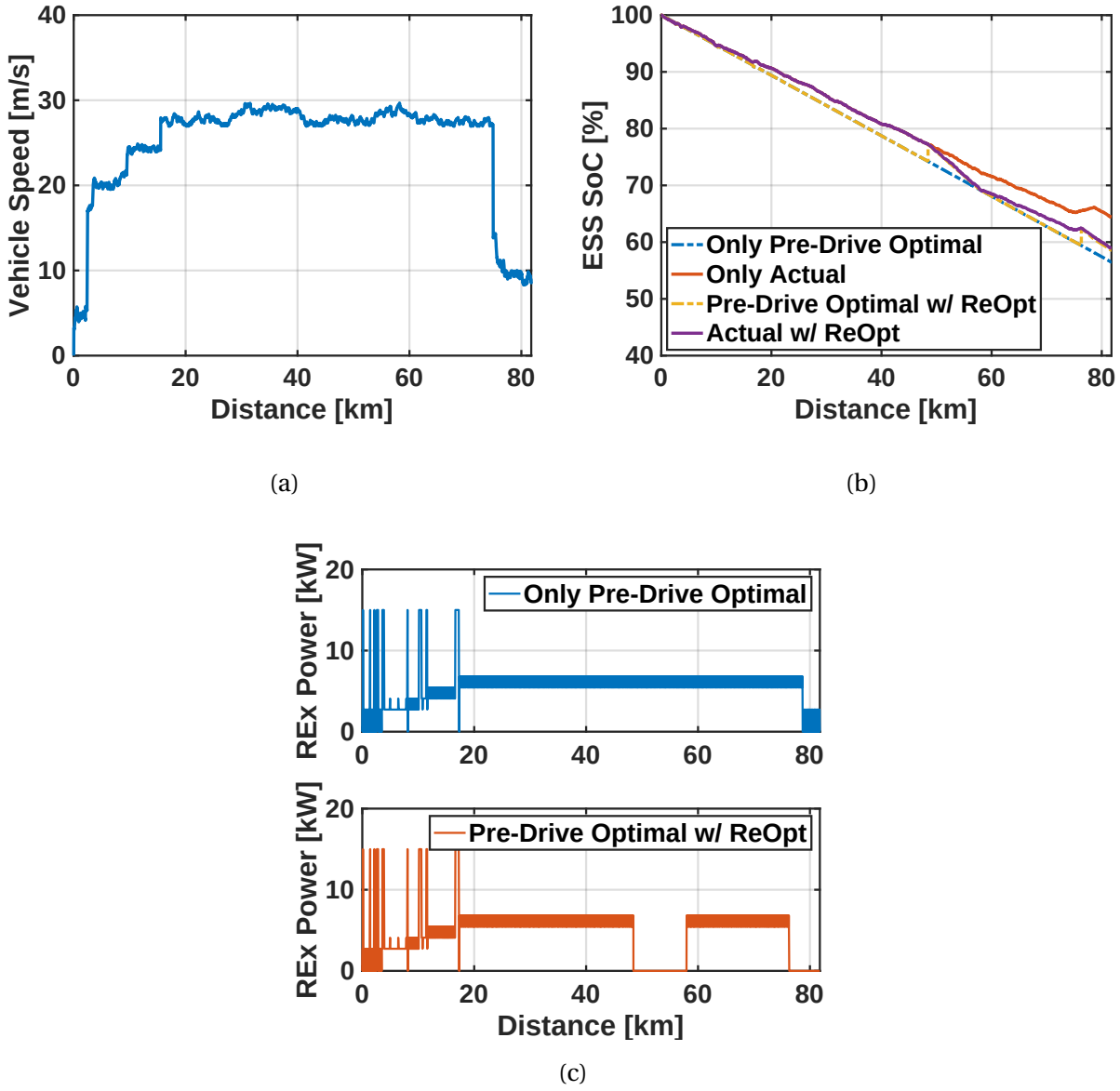
3.18 and the randomized speed traces derived from it. The ESS SOC deviation limit to initiate re-optimization was chosen to be 3%. This value was chosen to prevent unwanted re-optimizations due to modeling errors while preventing excessive deviations from an optimal solution. Figures 3.21(a)-(c) show results for the Random 1 speed trace, where the actual energy demand was much lower than the pre-drive optimal estimate. Similarly, Figures 3.22(a)-(c) show results for Random 2 speed trace, where the actual energy demand was within the deviation limit of the pre-drive optimal estimate and Figures 3.23(a)-(c) show results for Random 3 speed trace with actual energy demand much higher than the pre-drive optimal estimate.

For the Random 1 speed trace, ESS SOC was observed to deviate by 8% from the pre-drive optimal solution, resulting in an excess usage of fuel by the REx system. With re-optimization, the solution was observed to adapt at 49km and 75km distance markers resulting in a $< 1\%$ deviation from the new optimal solution and a 2.4% deviation from the pre-drive optimal solution. In case of Random 2 speed trace, ESS SOC was observed to stay within the 3% limit, thus resulting in no re-optimization events. Finally, for the Random 3 speed trace, ESS SOC was observed to deviate by 9% from the pre-drive optimal solution, resulting in excess usage of ESS energy. With re-optimization the solution was observed to adapt at 53km, 71km, 82km and 84km distance markers resulting in a 2% deviation from the pre-drive optimal solution. In this case, the re-optimization resulted in an increased power demand from the REx system than originally optimized. Net energy consumption values for the three randomized speed traces with and without re-optimization are shown in Figure 3.24. Random 1 speed trace showed an improvement of $\sim 15.6\%$ in the net energy consumption, while Random 3 speed trace increased the energy consumption by $\sim 4.4\%$ but satisfied the driving distance constraint. Random 2 speed trace shows a nearly similar net energy consumption in either of the cases.

Driving behavior tends to vary due to several system and environmental disturbances. Hence, it is challenging to have a deterministic knowledge of future driving speeds.

Stochastic approach to obtain an estimate of future driving speeds are discussed in literature but they increase computational load. In our proposed approach we leveraged the average speed information obtained from Google Direction API which uses a proprietary traffic estimation model along with live traffic data. The average speed trace served as an initial baseline though possible deviations can occur as observed with the three random speed traces. Speed deviations result in energy demand variations which are used to initiate re-optimization. The re-optimization process only computes optimal operation parameters for the remaining drive. Based on the Random 1 speed trace results, if energy demand is lower due to lower than expected speed, the re-optimization process reduces REx system run-time for the drive. By extension, if the vehicle comes to an unexpected complete stop, as the residual calculation is not a function of time but of distance travelled, the re-optimization process is expected to handle such variations.

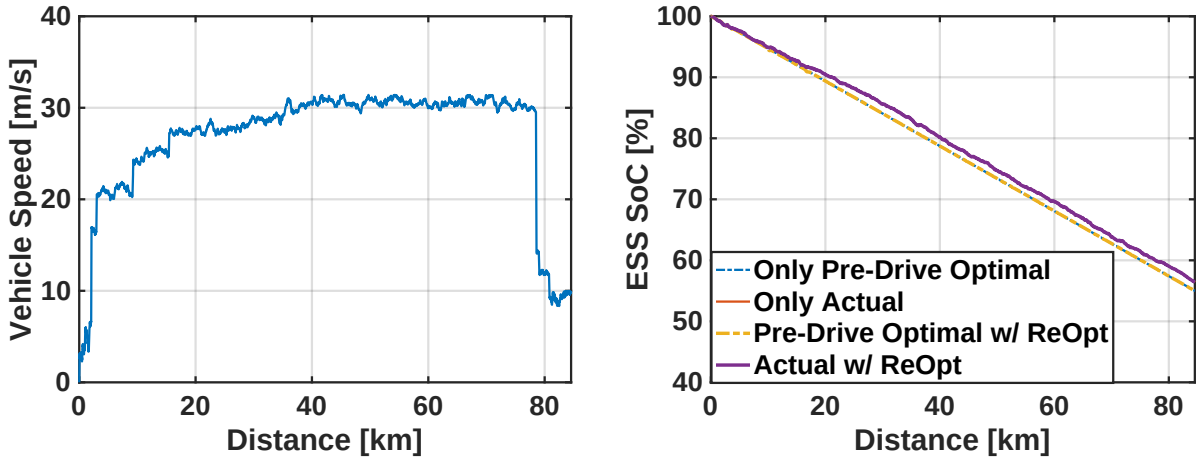
Vice-versa based on Random 3 speed trace results, if the energy demand is higher due to higher than expected speed, the re-optimization process causes an increase in REx system run-time. Such an outcome would result in an increased net energy consumption than expected initially to meet final ESS SOC and target distance requirements. This was expected based on our presented hypothesis and subsequent validation of the distance constrained dynamic programming explained in the previous subsection. This is further confirmed by net energy consumption results as shown in Figure 3.24. With re-optimization the EREV was able to get optimal net energy consumption results without violating final ESS SOC and target distance requirements. Thus, the adaptive distance constrained dynamic programming is able to provide optimal results in case of driving disturbances.



(a) (b)

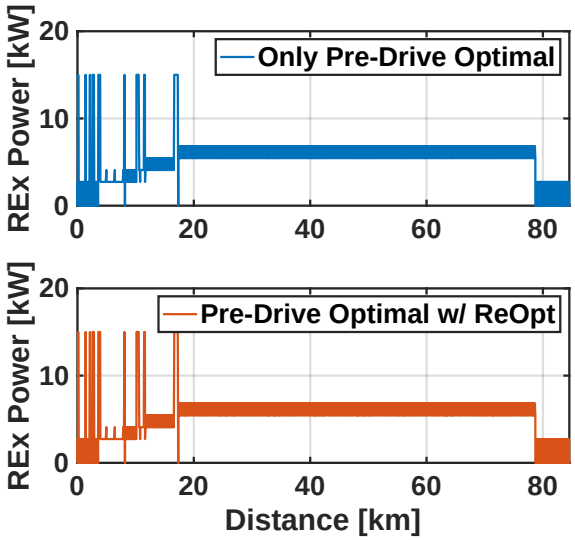
(c)

Figure 3.21: Comparison of vehicle's performance with and without re-optimization using the randomized low speed drive cycle.



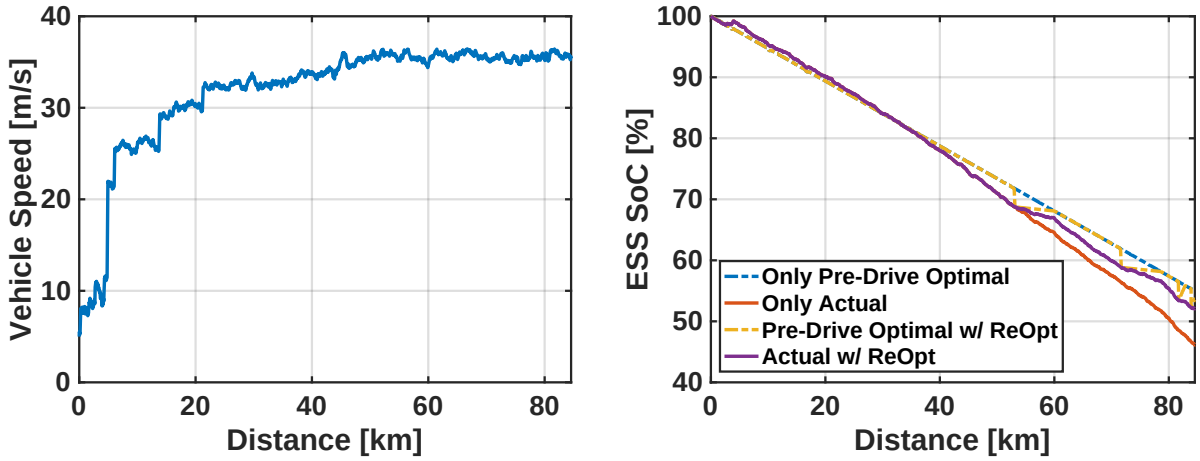
(a)

(b)

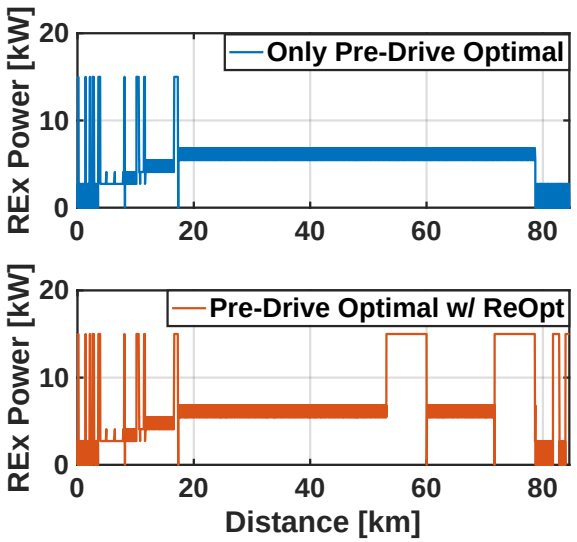


(c)

Figure 3.22: Comparison of vehicle's performance with and without re-optimization using the randomized medium speed drive cycle.



(a) (b)



(c)

Figure 3.23: Comparison of vehicle's performance with and without re-optimization using the randomized high speed drive cycle.

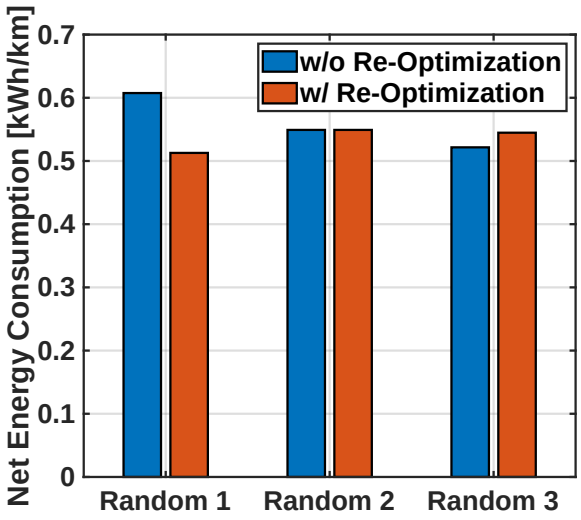


Figure 3.24: Net Energy Consumption comparison of with and without re-optimization scenarios for the three randomized speed traces.

3.4.3 Comparison of Adaptive Distance Constrained Dynamic Programming

We evaluated the performance of our proposed algorithm against the more conventional energy management approach for EREVs. This was based on operating the vehicle in two distinct modes, CD and CS, depending on the ESS SOC value. For the comparison presented we used UDDS-HWFET-UDDS-US06 cycle, a combination of the standard cycles used by EPA. The ESS SOC band for the CS mode was chosen to be between 15% to 25%. These values ensured that when the REx operated at its maximum output power, the minimum ESS SOC limit was not violated. The minimum ESS SOC limit was chosen to be 8% as per vehicle tests. The ESS was found to perform an Emergency Power Off (EPO) below that limit. In Figures 3.25, 3.26, and 3.27 we have provided the results showing a comparison between the two algorithms, corresponding to a single cycle run, double cycle run and triple cycle run, respectively.

For a single cycle run, both the conventional algorithm and the proposed algorithm

showed a similar behavior with vehicle operating in the CD mode i.e. using only ESS energy. Thus, the net energy consumption is of the same magnitude. In case of a double cycle run, the conventional algorithm covers ~ 60 km distance in CD mode and the remaining in CS mode. Our proposed algorithm showed an earlier activation of REx system to meet the driving distance and minimum ESS SOC requirements. A similar behavior was observed for the triple cycle run case also where our proposed algorithm was able to generate optimal control parameters for the REx system that satisfied driving distance and minimum ESS SOC requirements. The double and triple cycle run scenario results show the ability of our proposed algorithm to adapt to driving distance changes. The net energy consumption comparison for these three cases as per Figure 3.28, shows an improvement by our proposed algorithm over the conventional algorithm. Amongst the three scenarios shown, the largest improvement of 9.8% is observed for the double cycle run and a 1.6% for the triple cycle run. The single cycle run had the same net energy consumption due to operating only in the CD mode.

The conventional EREV energy management approach was obtained from our evaluation of two in-production EREV vehicles i.e. 2014 BMW i3 Rex and 2016 Chevrolet Volt. Both the vehicles were found to operate in distinct CD and CS modes. The CS mode was found to initiate when ESS SOC was lower than $\sim 15.7\%$. Both vehicles implemented a proprietary CS mode energy management strategy. Thus for our comparison, we formulated a distinct CD and CS mode energy management strategy for the experimental research vehicle, inspired by the in-production vehicles strategies we evaluated. This derived conventional strategy also switched to CS mode near 15.7% ESS SOC. With limitations on REx power, the strategy operated REx at its maximum feasible power output when required. To ensure the vehicle didn't violate the ESS EPO cutoff and by extension the distance constraint, the CS mode ESS SOC upper limit was determined to be 25%. This representative conventional EREV strategy was evaluated in the MiL and SiL environment along with our proposed algorithm.

In single cycle simulation, the vehicle operates in CD mode for both of the algorithms. This implies that for short distance drives, the proposed algorithm is able to prioritize ESS only usage over REx system. A difference between the conventional and proposed algorithm is more evident for the double and triple cycle simulations. The conventional algorithm in both the cases is able to effectively charge sustain and maintain ESS SOC higher than the EPO limit as marked by the dashed black line in the ESS SOC plots. The proposed algorithm on the other hand was observed to employ a blended CD-CS strategy of operation with the REx system. The corresponding ESS SOC trend for the experimental research vehicle was observed to fulfill both ESS SOC and distance constraints. As observed from prior evaluations, such an outcome was expected from the proposed algorithm over the conventional algorithm. In case of the conventional algorithm, fuel usage was computed to be higher than that in case of proposed algorithm. For the triple cycle simulation, the conventional algorithm used 17.8 L of fuel against 10.1 L in case of the proposed algorithm. This shows the vehicle can travel farther with the proposed algorithm. With respect to our research goal of implementing and evaluating an optimal energy management strategy for EREV, the net energy consumption results as shown in Figure 3.28 shows that the proposed algorithm at a minimum matches and in other cases lowers the value in comparison to the conventional algorithm. Thus, the proposed adaptive distance constrained dynamic programming algorithm performs better than the conventional algorithm.

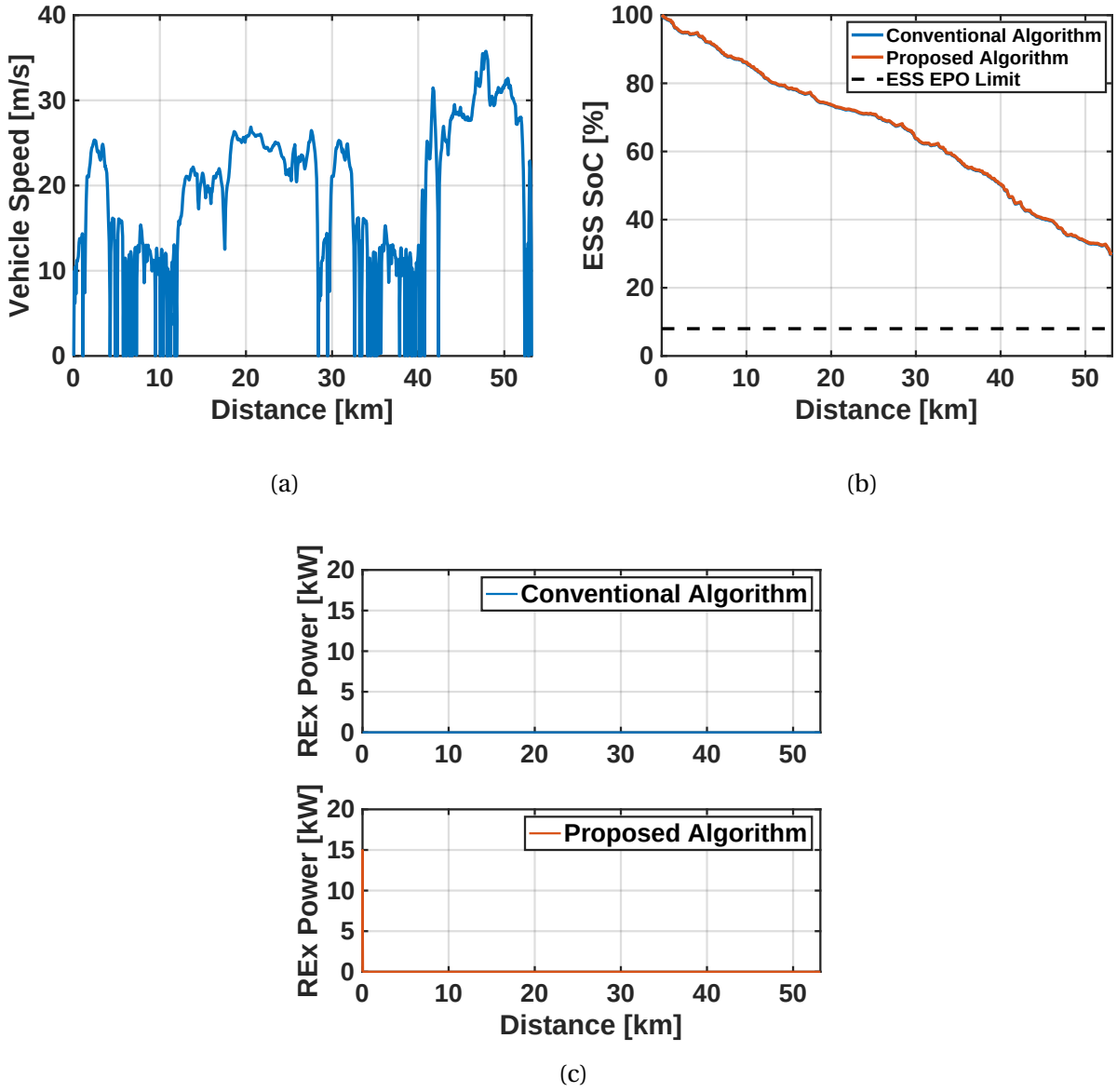
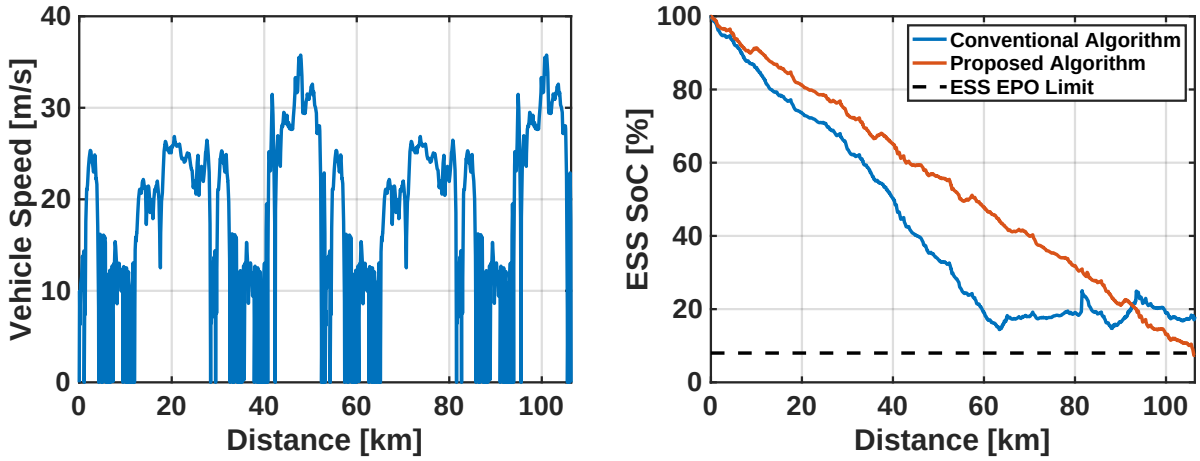
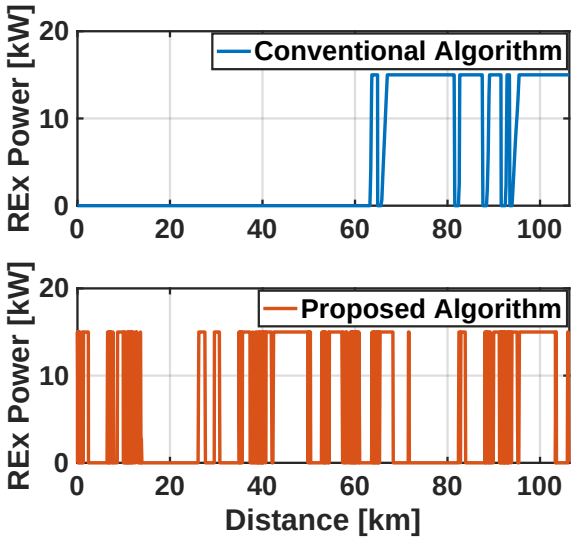


Figure 3.25: Comparison of the conventional and proposed algorithms for UDDS-HWFET-UDDS-US06 single cycle run.



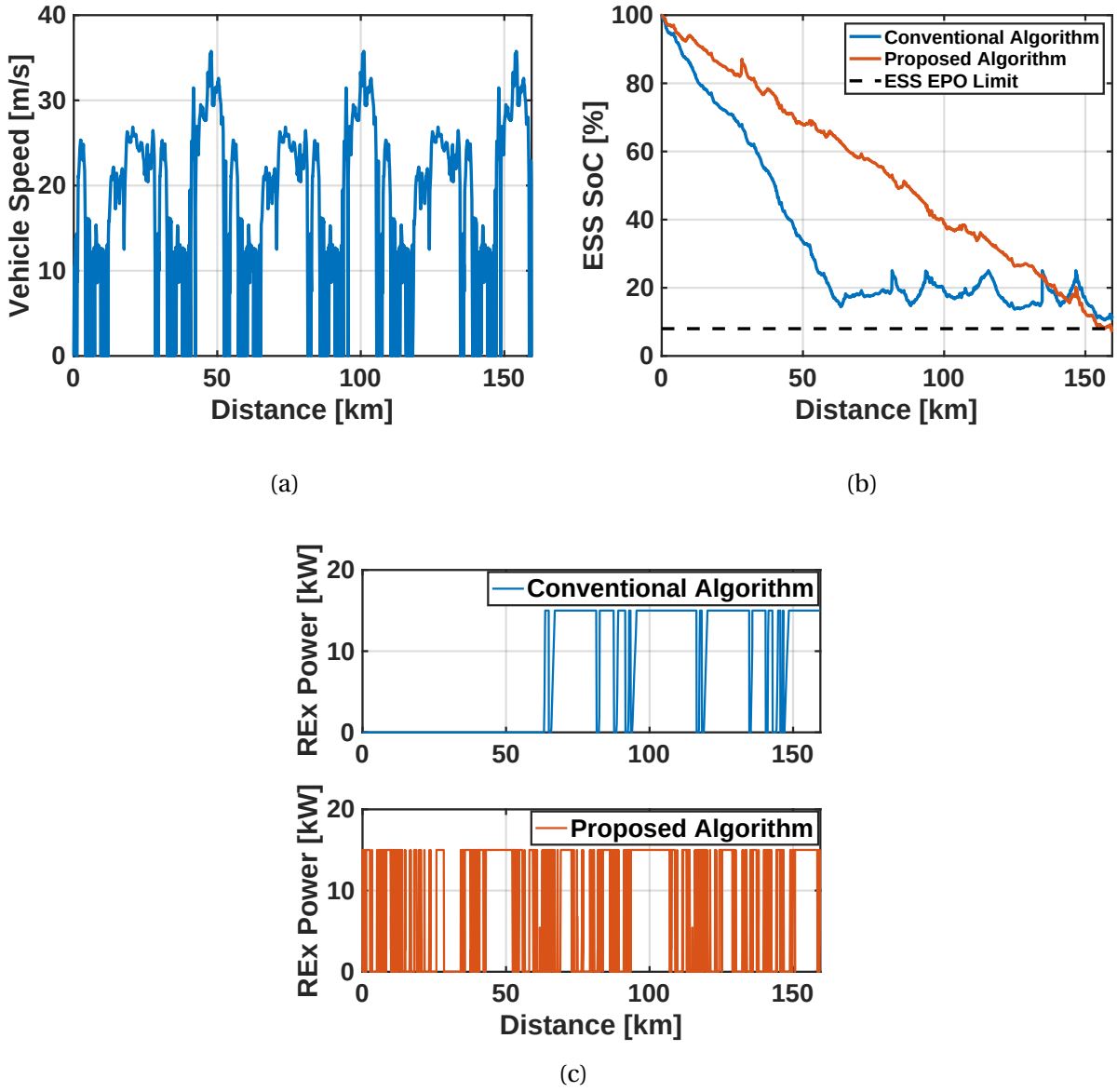
(a)

(b)



(c)

Figure 3.26: Comparison of the conventional and proposed algorithms for UDDS-HWFET-UDDS-US06 double cycle run.



(a)

(b)

(c)

Figure 3.27: Comparison of the conventional and proposed algorithms for UDDS-HWFET-UDDS-US06 triple cycle run.

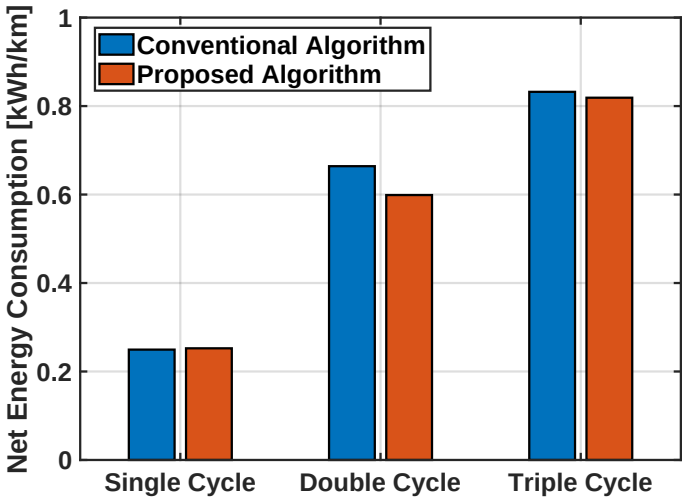


Figure 3.28: Net Energy Consumption comparison of conventional and proposed algorithms for the single, double and triple cycles of UDDS-HWFET-UDDS-US06 combination.

The adaptive nature of the proposed algorithm introduces the possibility of it being fault tolerant. In a scenario, where a propulsion system component on the research EREV experiences fault or functional limitation, the vehicle performance will be limited or the feasible driving distance will reduce. We evaluated a fault scenario causing the REx system maximum power to reduce from 15 kW to 8 kW. Under this limited power capability, the conventional CD-CS algorithm and our proposed Adaptive Distance Constrained Dynamic Programming algorithm were evaluated. The ESS SOC comparison as per Figure 3.29(b) shows that the proposed algorithm was able to adapt to the fault and optimize REx system operation to meet the final ESS SOC and target distance requirements. Unfortunately, in case of the conventional algorithm, ESS SOC drops below the ESS EPO Limit near 90 km marker, resulting in the vehicle being immobile.

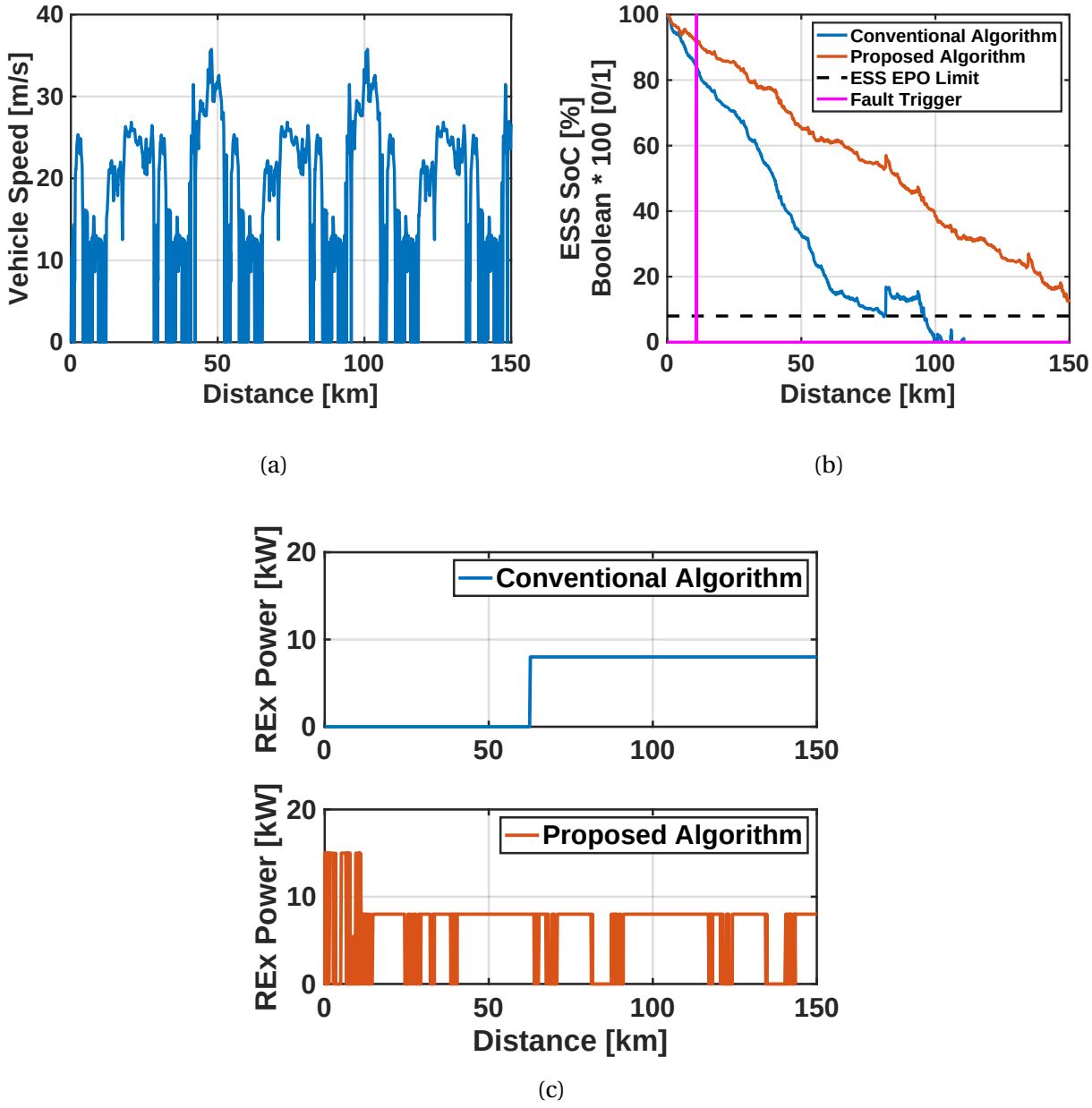


Figure 3.29: Fault insertion behavior comparison of conventional and proposed algorithm for the UDDS-HWFET-UDDS-US06 triple cycle.

Based on these results, our proposed algorithm shows fault tolerance over the conventional algorithm but within certain limitations. In the presented results, the fault insertion occurred early in the drive, thus allowing considerable time for the vehicle to adjust to the new limitations. We expect the algorithm to violate target distance constraint if the fault was inserted much later in the drive. Such an outcome is possible as the vehicle ESS SOC would be much closer to the ESS EPO Limit. With ADAS features in modern vehicles, the fault tolerant behavior of the algorithm can also be used to inform the driver of the updated available driving distance.

3.5 Conclusion

This article presented a novel implementation of an energy management algorithm for the experimental research EREV. The development and application of this algorithm was directed by two major goals (i) achieving the maximum feasible driving distance without recharging or refueling and, (ii) maintaining the minimum feasible net energy consumption. We also targeted for a real-time online application of this algorithm.

To achieve these goals, we first formulated a power loss model of the experimental research EREV. The model was validated against vehicle test results conducted on a chassis dynamometer. This followed an evaluation of frequently used energy management optimization algorithms. Dynamic Programming approach was chosen for development of our proposed algorithm. The proposed algorithm was a variation of dynamic programming with forward propagation and a distance constraint. To ensure the algorithm is resilient to external disturbances, a re-optimization feature was added that made the algorithm adaptive. The application of this algorithm was evaluated against different standard speed traces and real-time average speed trace obtained from Google Directions API.

Our evaluation of the more prevalent energy management optimization algorithms showed that the dynamic programming approach provided more optimal results. We

implemented our variant of dynamic programming with forward propagation, by adding a target distance constraint. This approach resulted in ensuring the energy management is optimal while the target distance requirement is achieved. The results from the adaptive version of the proposed algorithm validated our hypothesis of real-time online implementation while being optimal. Finally, a functional comparison of our proposed algorithm with the conventional CD-CS algorithm was conducted on the model of our experimental research EREV vehicle. The results proved that our algorithm at the very least performs at par with the conventional algorithm and outperforms it by 9.8% on the other extreme.

In conclusion, the presented work showed that for EREV energy management, in addition to optimizing for minimum net energy consumption, we also need to optimize for maximum feasible driving distance. We also showed an implementation of our proposed adaptive distance constrained dynamic programming algorithm with live-traffic information from Google Directions API. This verified our hypothesis of the proposed algorithm being real-time online implementable. By extension our proposed algorithm is also able to provide an optimal solution even if one or more of the propulsion systems are power limited or faulting. Our proposed algorithm was verified and validated in MiL and SiL environments in this article and additional system and environmental disturbances such as road elevation, system ramp-up and ramp-down delays as well as thermal effects were not considered. Future extension of this work will include the impact of these limitations as well as implementation of the proposed energy management strategy on the experimental research EREV.

Chapter 4

On Fault Tolerant Behavior of Optimal Energy Management Control for Extended Range Electric Vehicle

Publication: *Submitted to IEEE Vehicle Power and Propulsion Conference 2020, A. V. Kalia and B. C. Fabien, "On Fault Tolerant Behavior of DC-ARTDP Optimal Energy Management Algorithm for Extended Range Electric Vehicle"*

4.1 Abstract

Energy system failure or reduced power operation is a plausible scenario in extended range electric vehicles (EREVs). Such an occurrence causes performance degradation of these systems, and can impact vehicle distance-to-empty (DTE) and overall energy consumption. In this article, we present the fault tolerant optimal energy management control behavior of our proposed Distance Constrained Adaptive Real-Time Dynamic Programming (DC-ARTDP) algorithm. We evaluate this behavior with model-in-loop environment for a 2016 Chevrolet Camaro extended range electric research vehicle. Fault scenarios limiting (a) propulsion motor output power, and (b) range extender system output power are evaluated. A discussion on EREV's response to these faults while comparing DC-ARTDP algorithm with the conventional charge depleting - charge sustain-

ing (CD-CS) algorithm is presented. The DC-ARTDP algorithm can meet the 163 km DTE target in all of the fault scenarios with net energy consumption between 0.53 - 0.89 kWh/km. Thus, our proposed DC-ARTDP algorithm shows adaptability to fault scenarios over the conventional CD-CS EREV energy management algorithm.

4.2 Introduction

Extended range electric vehicles (EREVs) comprise of multiple energy systems in their powertrain architecture. The core energy systems include (i) Energy Storage System (ESS), (ii) Electric Propulsion Motor(s), and the (iii) Range Extender System (REx). Optimal energy flow amongst these systems is necessary to obtain a minimal overall energy consumption while targeting the maximum feasible distance-to-empty (DTE). Off-line and on-line optimized energy management algorithms for EREVs have been thoroughly discussed but under the assumption that these systems perform flawlessly. In this article, we focus on fault occurrence scenarios on the earlier mentioned energy systems in an extended range electric vehicle and their implications on the vehicle's performance. A fault occurrence in one or more of these energy systems can at minimum impose power limitations and in worse case scenarios impede vehicle movement.

Fault detection and diagnostic strategies or fault-tolerant control systems (FTCS) have been a part of vehicle component and system electronic control units (ECUs) [11]. Based on the severity level of fault, a safe operating mode for the system/component is determined. In conventional powertrain vehicles, a mild to moderate level fault causes the vehicle to operate in a limp home mode [14]. In such a scenario, the powertrain system output is derated to protect the components. For a high severity fault, the systems are completely shutdown to prevent excessive damage. For hybrid electric vehicles (HEVs), both sensor-based measurements and analytical models have been used for fault detection and diagnostics. Application of these methods on propulsion motors, engine, transmission, energy sources, and power-converters have been extensively

studied [65, 78, 49]. The FTCS strategies studied and implemented on HEVs are predominantly focused on improving control robustness at a component level. These strategies have a limited impact on the high-level vehicle energy management. Aouzellag et al. implemented a fault-tolerant energy management control strategy on a fuel-cell and ultra-capacitor based hybrid electric vehicle. Their approach defined different operation combinations for energy systems based on state of charge (SOC) and vehicle speed [8]. Oubellil et al. studied fault-tolerant capabilities of a standard PI and two H_∞ controllers in series with an optimal energy management strategy providing reference loads to the fuel-cell and battery in parallel hybrid vehicle. Fault in voltage measurement was introduced to assess system response and robustness of energy management control [52]. Kamal et al. in their work focus on battery faults in hybrid electric bus. They also aim to improve component level energy management with a fuzzy control approach [38]. These solutions approach the energy management problem in a fault scenario by introducing a robust controller at the component level. Furthermore, the strategies use offline optimized or safe operating parameters which can be sub-optimal in real test/-operating scenarios.

Therefore, we present the fault-tolerant behavior of our Distance Constrained Adaptive Real-Time Dynamic Programming (DC-ARTDP) algorithm. This algorithm is applicable on a supervisory controller level that can monitor all energy systems status as well as vehicle's DTE. The algorithm was evaluated on a 2016 Chevrolet Camaro EREV research vehicle model using MATLAB/Simulink. Fault scenarios targeting two core EREV energy systems i.e. electric propulsion motor(s), and REx system were simulated. This novel approach provides optimal energy management control parameters during fault scenarios at a supervisory level.

The article further comprises of four sections Section 4.3 describes the Methodology used for vehicle modeling, control and fault scenarios. The Results and Discussion is presented in Section 4.4 with Conclusion in Section 4.5.

4.3 Methods

Model-in-Loop (MiL) simulation environment was chosen as a verification and validation approach for this study. A power flow model representing the core energy systems of the EREV research vehicle was used for simulations. Additionally, a drive scenario generator, the proposed fault-tolerant DC-ARTDP algorithm and active fault insertion nodes are integrated into the MiL environment. The model was verified and validated against real research vehicle data and presented in our previously published article [37].

In this study, we used a custom drive cycle obtained using Google Directions API for a user provided source and destination location. The vehicle's round trip travel was simulated for no fault and fault induced scenarios. DC-ARTDP algorithm's behavior was compared against the conventional CD-CS EREV energy management algorithm for net energy consumption and DTE performance. The simulation model, DC-ARTDP algorithm and fault insertion implemented, is further explained in sub-sections below.

4.3.1 EREV Power Flow Model

The mathematical model representing our research vehicle was designed to focus on power flow among the core energy systems. Figure 4.1 shows the model schematic with fault insertion locations and the energy management control algorithm in a feedback loop. Model input is reference speed (v_{ref}) obtained from the drive cycle block. The power demand calculator uses v_{ref} along with Eq.(4.1) to determine propulsion power demand (P_{dem}). In this model, road grade α_{road} was kept constant at 0%.

$$\begin{aligned} P_{dem}[k] =& (m_{veh} + m_{psg}) \times v_{ref}[k] \times \left[\frac{v_{ref}[k] - v_{ref}[k-\Delta k]}{\Delta k}\right] \\ &+ \frac{1}{2} \times C_d \times \rho_{air} \times A_{front} \times v_{ref}^3[k] \\ &+ C_{rr}(p_{tire}, v_{ref}[k]) \times (m_{veh} + m_{psg}) \times g \times v_{ref}[k] \\ &+ (m_{veh} + m_{psg}) \times g \times sin(\alpha_{road}[k]) \times v_{ref}[k] \end{aligned} \tag{4.1}$$

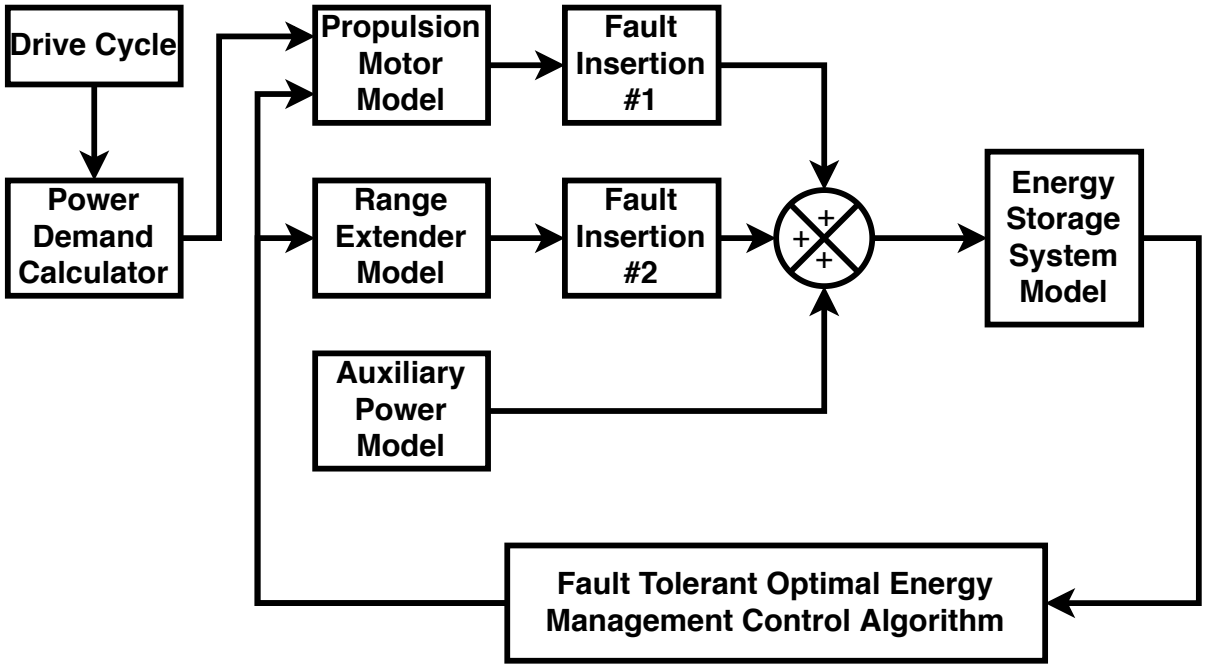


Figure 4.1: EREV research vehicle power flow model schematic.

As the prime propulsion system on an EREV, the motor system model determines electric propulsion power (P_{elec}) demand from energy sources i.e. battery system and REx system. An efficiency map, $\eta_{mot}(\tau_{mot}, \omega_{mot})$ for the motor system was used to determine P_{elec}. The value of η_{mot} includes gearbox efficiency and was computed for the tested operating range. The REx system model employs a combined efficiency map, $\eta_{REx}(\tau_{gen}, \omega_{gen})$ for the system and to obtain P_{REx}. Here, η_{REx} was computed from the engine BSFC map and constant generator motor efficiency of 0.86. Finally, the auxiliary power model is represented with a normally distributed power demand about a fixed value of 520 W, similar to that observed in the research vehicle. The energy storage system or battery system model is a dual-polarization model and represented by Eq.(4.2) and Eq.(4.3). The unknown model parameters i.e. V_1,V_2, and R_0, were derived with system identification methods using collected test data [37].

$$V_{bat}(SOC,k) = V_{OC}(SOC) - V_1(SOC,k) - V_2(SOC,k) - I_L[k] \cdot R_0(SOC) \tag{4.2}$$

$$\frac{SOC[k] - SOC[k-\Delta k]}{\Delta k} = -\frac{I_L[k]}{Q_{max}} \tag{4.3}$$

The power flow model was validated against vehicle performance and energy consumption data gathered from on-road tests. This process has been described in our previous publication in detail [37]. Model estimated battery SOC ($\widehat{SOC}$) was compared against measured battery SOC (SOC_z) to obtain a coefficient of variation in the range 1.9%-7.0% [37]. A low value was indicative of a good fitting model.

4.3.2 Fault Tolerant Optimal Energy Management

The optimal energy management algorithm discussed in this article is termed as the Distance Constrained Adaptive Real-Time Dynamic Programming (DC-ARTDP) algorithm. The algorithm was designed to operate on-line on the vehicle, with *priori* knowledge of estimated vehicle speed obtained from Google Directions API for a known origin and destination location [37]. During the drive, vehicle battery SOC was monitored for deviations from reference SOC trajectory determined from pre-drive optimization by the algorithm. SOC deviation limit magnitude was chosen to be $\pm 3\%$ i.e. equivalent to 570 Wh, to discard deviations due to model errors and prevent significant diversion from the optimal SOC trajectory. In an event of deviation beyond the limit, the algorithm re-optimizes for the remaining planned drive.

High level implementation of the algorithm is represented by the flowchart schematic in Figure 4.2. The section of the flowchart bounded by a dashed-box obtains estimated vehicle speed and computes optimal SOC reference trajectory within energy and DTE constraints. The remaining section of the flowchart operates as a loop driven by SOC

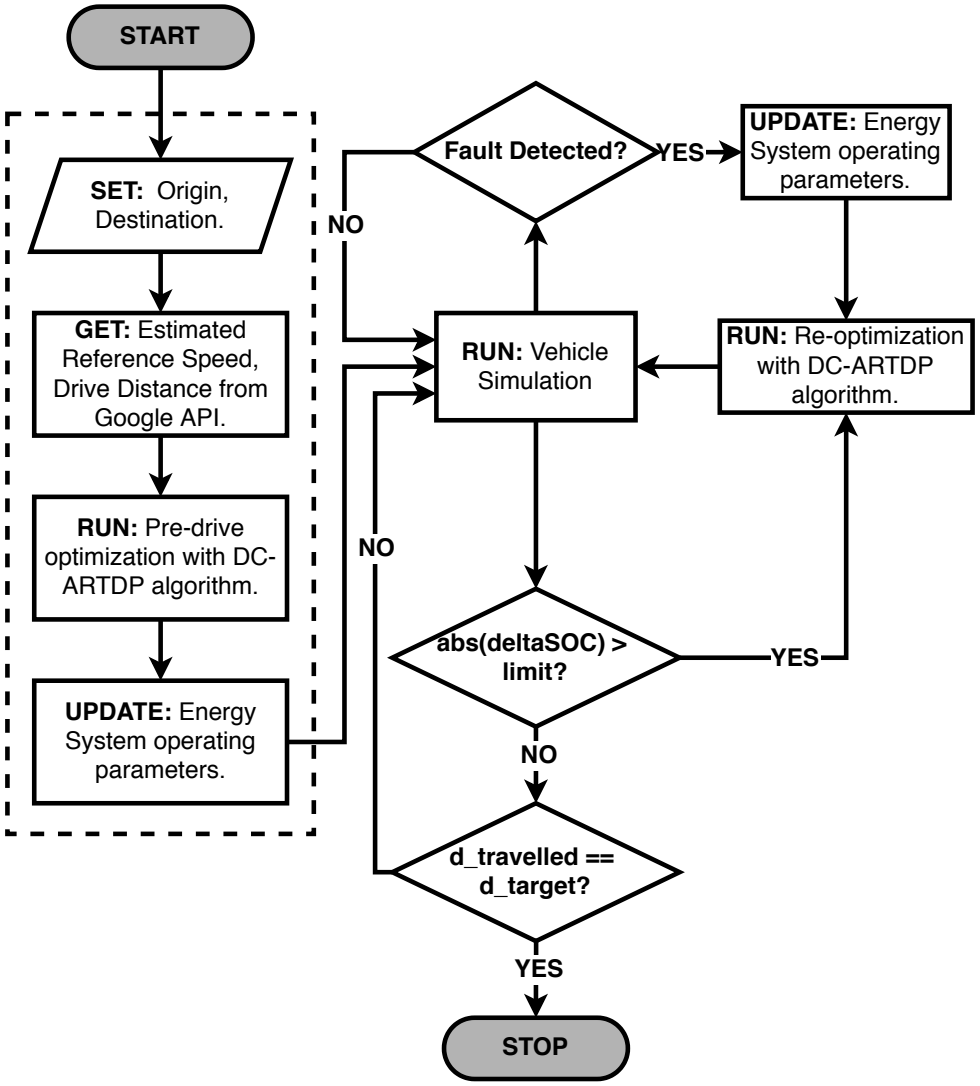


Figure 4.2: Fault tolerant DC-ARTDP algorithm execution flow chart.

deviation and fault detection assertion conditions. In this study we assumed that a core energy system fault detection mechanism exists. Vehicle on-board fault diagnostic software for energy systems, generally provides fault severity level as well as updated safe operable power limits. The fault tolerant behavior of our algorithm was designed and

verified per this assumption.

Algorithm 3: Fault-triggered parameter renewal

Input: FA_Type, REx_Pwr_Lim, MOT_Pwr_Lim, MOT_Trq_Cont, MOT_Eff, MOT_Regen_Pwr_MAX

Output: MAX_Spd_Lim, MAX_Regen_Pwr, REx_Min_Lim, REx_Max_Lim, MOT_Min_Lim, MOT_Max_Lim

$R_{tire} \leftarrow 0.346$;

$N_{gbx} \leftarrow 4.2$;

while *FA_Type != 0* **do**

  if *(FA_Type == 1) or (FA_Type == 2)* **then**

    REx_Min_Lim $\leftarrow$ 0;

    REx_Max_Lim $\leftarrow$ REx_Pwr_Lim;

  endif

  if *FA_Type == 3* **then**

    MOT_Min_Lim $\leftarrow$ -MOT_Pwr_Lim;

    MOT_Max_Lim $\leftarrow$ MOT_Pwr_Lim;

    MAX_Spd_Lim $\leftarrow \frac{1}{MOT\_Trq\_Cont}\left[R_{tire} \times \text{MOT\_Eff} \times \text{MOT\_Max\_Lim}\right]$;

    MAX_Regen_Pwr $\leftarrow$ min(MOT_Regen_Pwr_MAX, MOT_Min_Lim);

  endif

end

The schematic in Figure 4.2 shows a parameter update step upon positive fault assertion. The process of parameter updation follows per Algorithm 3 as shown. The updation algorithm was designed around the four fault scenarios studied. The fault type has four feasible numerical values i.e. FA_Type $\in$ {0,1,2,3}, where 0: No Fault, 1: REx system deration fault, 2: REx system offline fault, 3: Motor system deration fault. The updated power and speed limits were then used as inputs to the DC-ARTDP algorithm

to compute a feasible set of SOC at each stage in the process of Dynamic Programming.

4.3.3 Faults and Drive Scenario

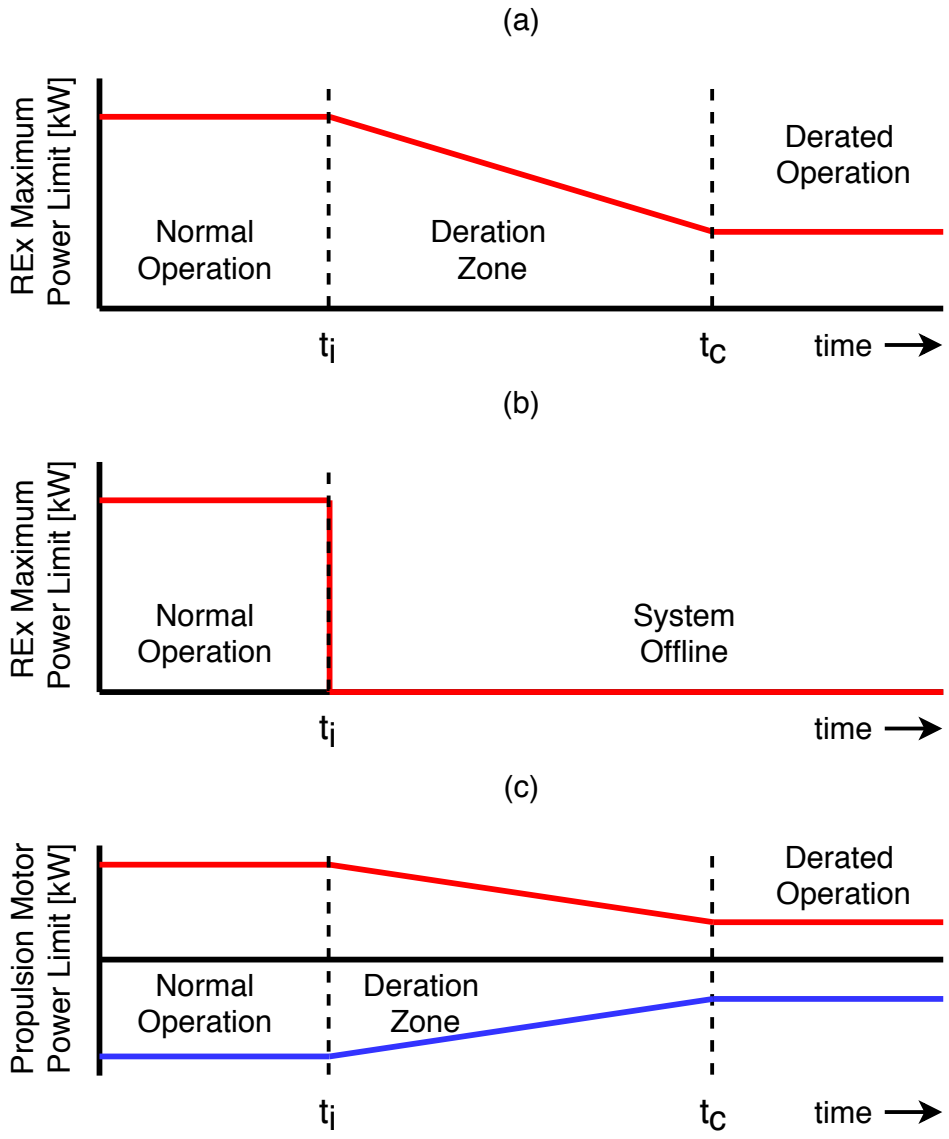


Figure 4.3: Fault injection response trace for (a) REx system deration, (b) REx system shutdown, and (c) propulsion motor system deration.

Among the EREV energy systems we implemented four fault scenarios, (a) REx sys-

tem deration fault, (b) REx system offline fault, and (c) Motor system deration fault. Figure 4.3 shows the fault injection response as a function of time on system power output. The fault is injected at time t_i, followed by a deration zone in case of faults (a) and (c) till time t_c, that denotes deration completion time. For fault scenario (b) the system turns off upon fault injection at time t_i. These faults were introduced in the MATLAB/Simulink model of the EREV research vehicle as per the schematic in Figure 4.1.

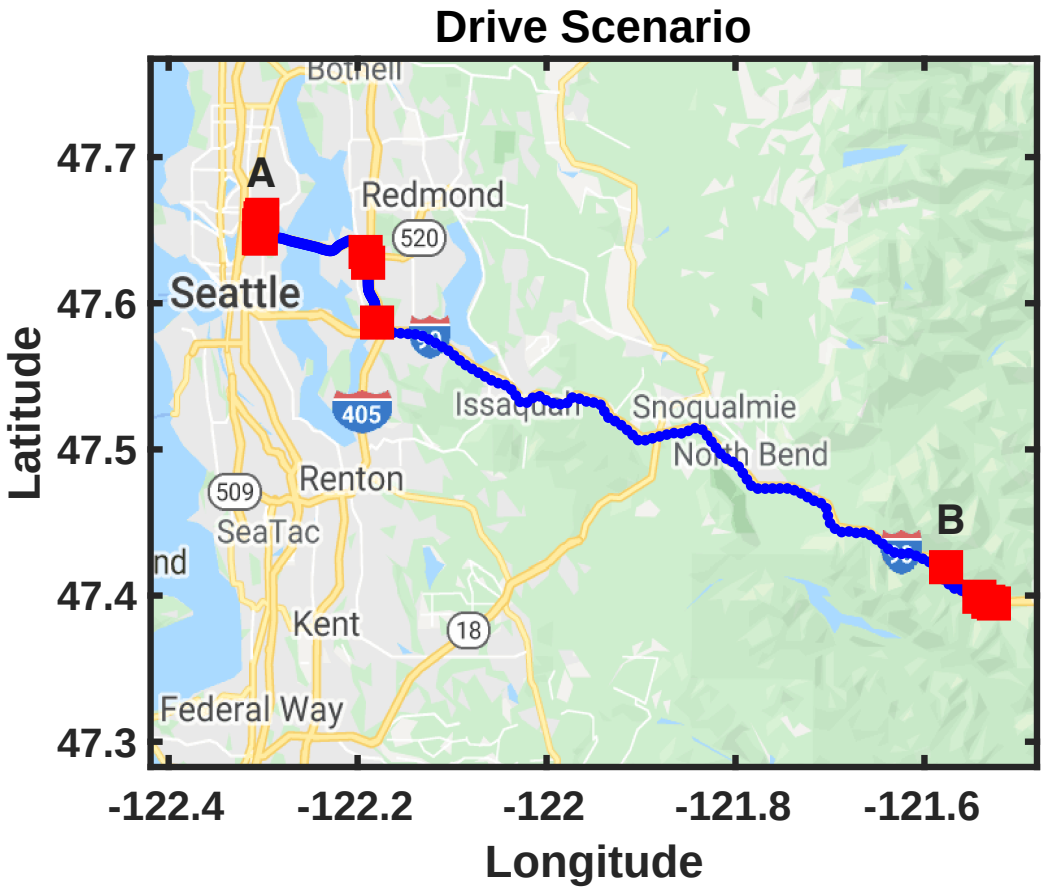


Figure 4.4: Simulated 163 km round-trip drive with origin at A and turnaround at B.

The impact of these faults on the EREV research vehicle performance was studied over a real-world drive scenario. Figure 4.4 shows the route simulated for a $\sim$ 163 km round trip from point A to point B and back. The speed trace as shown in Figure 4.5 was

obtained by generating normally distributed random values around the average speed obtained from Google Directions API [37]. In this study, the simulations use the random vehicle speed trace to emulate realistic driving while the DC-ARTDP uses the average vehicle speed for optimization purposes.

In this study, the EREV research vehicle was evaluated on the basis of, (i) it's ability to travel the target driving distance without a battery EPO and (ii) operate close to global optimal battery SOC trajectory. The Results and Discussion sections provide these evaluations in context of the faults studied.

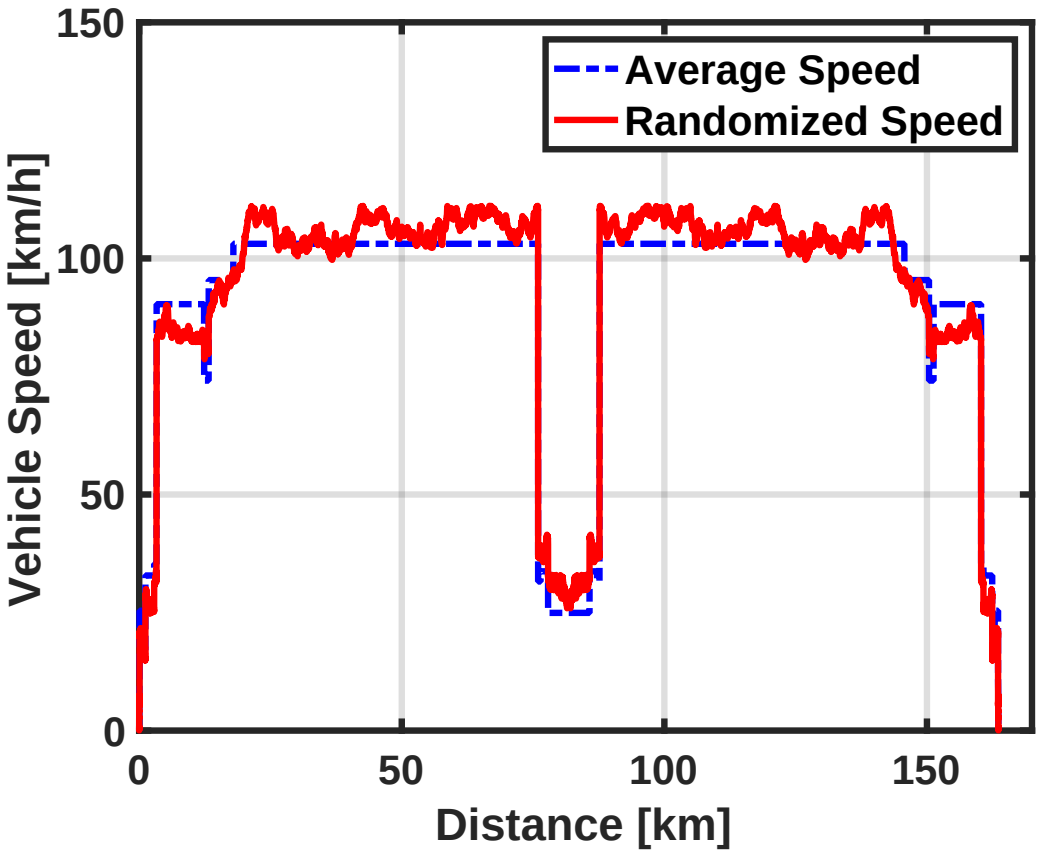


Figure 4.5: Vehicle speed trace for the simulated round-trip.

4.4 Results and Discussion

The three faults scenarios described earlier were evaluated through simulations through the model-in-loop environment. All the faults were evaluated one-by-one for two insertion events at $t_i = 1000$ seconds i.e. ~ 17.9 km marker and at $t_i = 6000$ seconds i.e. ~ 131.6 km marker. The deration time for deration faults i.e. $t_c - t_i$, was fixed at 10 seconds. The results presented in this section compare EREV battery SOC performance with the conventional algorithm, DC-ARTDP algorithm and the Global Optimal trajectory obtained from DP. These results also show the battery EPO limit set at 8% SOC, determined from in-vehicle testing and the fault trigger shown as the impulse $1 - 0 - 1$ i.e. No Fault - Fault - No Fault.

4.4.1 REx System Faults

Two variants of REx system faults were studied on the EREV model. (i) *REx deration fault* (FA_Type = 1), where the REx system observed a reduction from its maximum power output of 15 kW to 8 kW after fault insertion, thus limiting supplemental power for range extension and impacting distance-to-empty of the EREV. (ii) *REx shutdown fault* (FA_Type = 2), where the REx system shuts down after fault insertion resulting in no supplemental power. In case of a REx shutdown fault the vehicle functions as an EV, therefore there is no difference in battery SOC performance between the conventional and DC-ARTDP algorithms.

The impact of REx deration fault studied at two different time events is shown per Figure 4.6 and Figure 4.7, respectively. In Figure 4.6, the fault was inserted at $t_i = 1000$ seconds (~ 17.9 km marker) resulting in varying responses between the conventional and DC-ARTDP algorithm cases. For the conventional algorithm, the vehicle operates in charge depleting (CD) mode till battery SOC is lower than 15%. At this limit, the vehicle starts operating in charge sustaining (CS) mode and is expected to sustain SOC

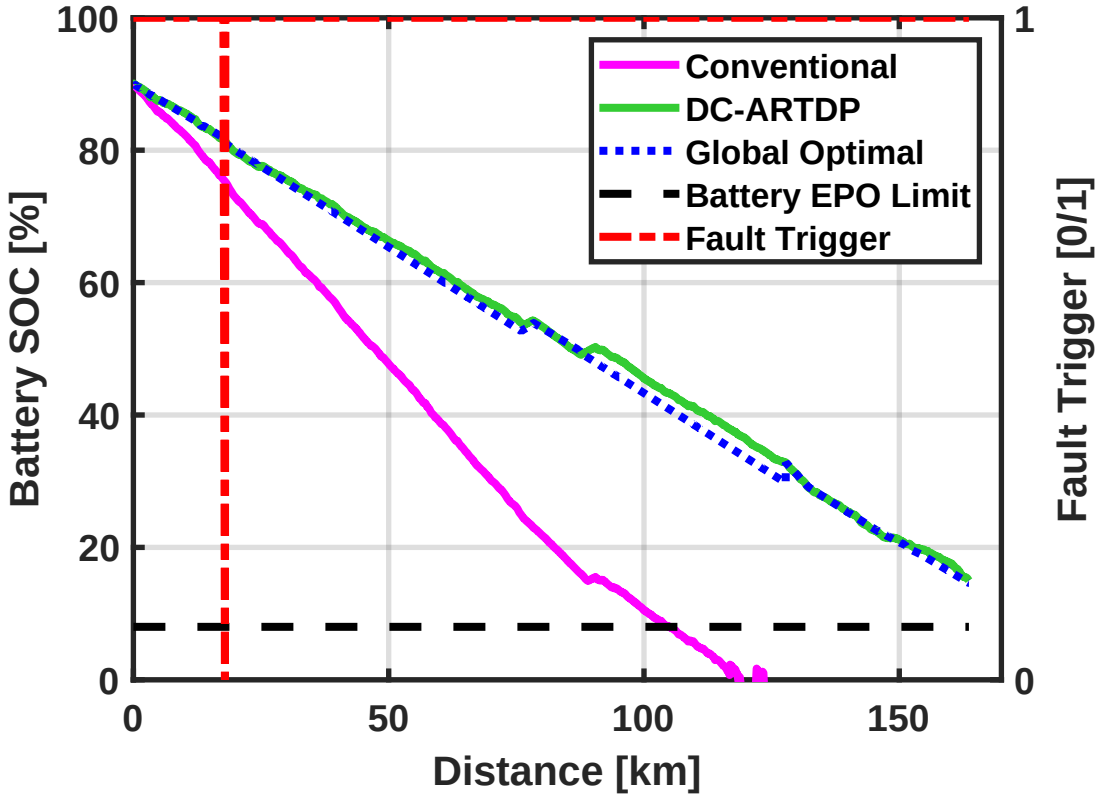


Figure 4.6: EREV battery SOC performance with REx deration fault occurring at $\sim$ 17.9 km marker.

within the limit i.e. [15, 25]%. With the REx deration fault introduced, the battery SOC performance was observed to perform as expected in the CD mode but was unable to sustain charge in the CS mode. Battery SOC was observed to fall below the battery EPO limit $\sim$ 100 km marker. Thus, the EREVs distance-to-empty in this case was determined to be 105 km only. With the fault tolerant DC-ARTDP algorithm operating per algorithm execution flow chart as shown in Figure 4.2, the EREVs battery SOC was observed to stay higher than the battery EPO limit. The DC-ARTDP algorithm computed the optimal SOC trajectory prior to the drive and fault insertion. The, algorithm was observed

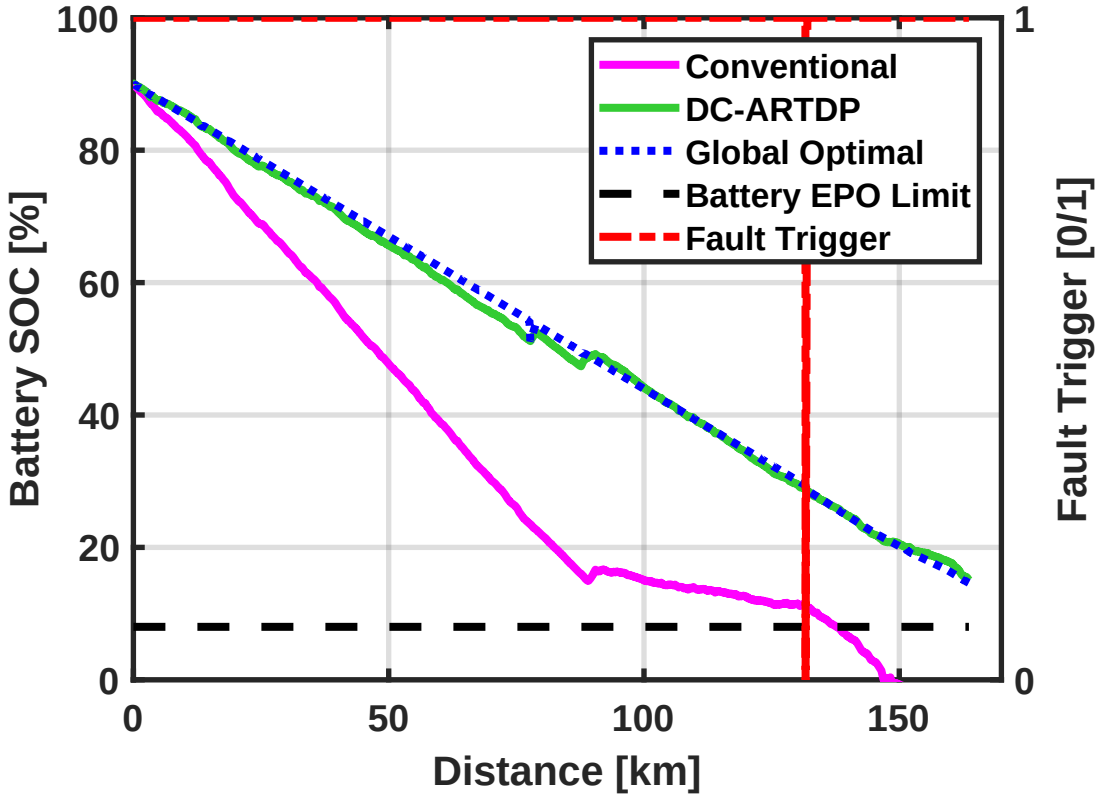


Figure 4.7: EREV battery SOC performance with REx deration fault occurring at ~ 131.6 km marker.

to re-compute the optimal trajectory with updating operating parameters as shown in Algorithm 3. In Figure 4.6, the global optimal trajectory can be observed to undergo two more events of re-optimization to meet the distance and final SOC constraints. The EREV was observed to successfully complete the simulated 163 km drive with the final battery SOC of 15%. The REx deration fault insertion event at $t_i = 6000$ seconds (~ 131.6 km marker) as shown in Figure 4.7, also showed the EREV being unable to complete the simulated drive with the conventional algorithm. The distance-to-empty in this case was determined to be 138 km. The fault tolerant DC-ARTDP algorithm was again able

to complete the simulated drive with the final battery SOC of 15%.

4.4.2 Motor System Fault

In *motor system deration fault* (FA_Type = 3), a limit on total motor output power of 60kW from a total maximum continuous power of ~ 140 kW was introduced. The impact of the fault was emulated by limiting maximum vehicle speed to meet the derated power limit and allowing the maximum continuous torque of 180 Nm. Figure 4.8 and Figure 4.9 refer to the results from fault insertion at the two time events described earlier. In both the cases after fault insertion, the vehicle speed is limited to ~ 50 km/h. As expected, with reduced speed limit, the EREV is able to complete the simulated drive with both conventional and DC-ARTDP algorithms. The results presented were obtained for a battery initial SOC of 40%.

For the fault insertion at $t_i = 1000$ seconds (~ 17.9 km marker), the EREV battery SOC performance can be observed per Figure 4.8. With the conventional algorithm the vehicle was able to perform its CD and CS modes of operation and complete the drive. The vehicle was also able to complete the simulated drive with DC-ARTDP algorithm. The results show that due to reduced power demand with speed limiting, both the algorithms provide distance-to-empty greater than the target distance. For fault insertion at $t_i = 6000$ seconds (~ 131.6 km marker), results per Figure 4.9 show the EREV was unable to reach the target distance with conventional algorithm. Though with the DC-ARTDP, the EREV was able to successfully complete the simulated drive. Based on the results obtained for the motor deration fault, an earlier introduction of fault allows the EREV to have a DTE greater or equal to the target distance. With fault being introduced later in the drive, due to low battery SOC and high power demand, the EREV battery SOC can fall below the 8% limit and cut short the DTE. Though in both the fault insertion events, the DC-ARTDP algorithm is able to achieve a DTE greater than or equal to the target distance.

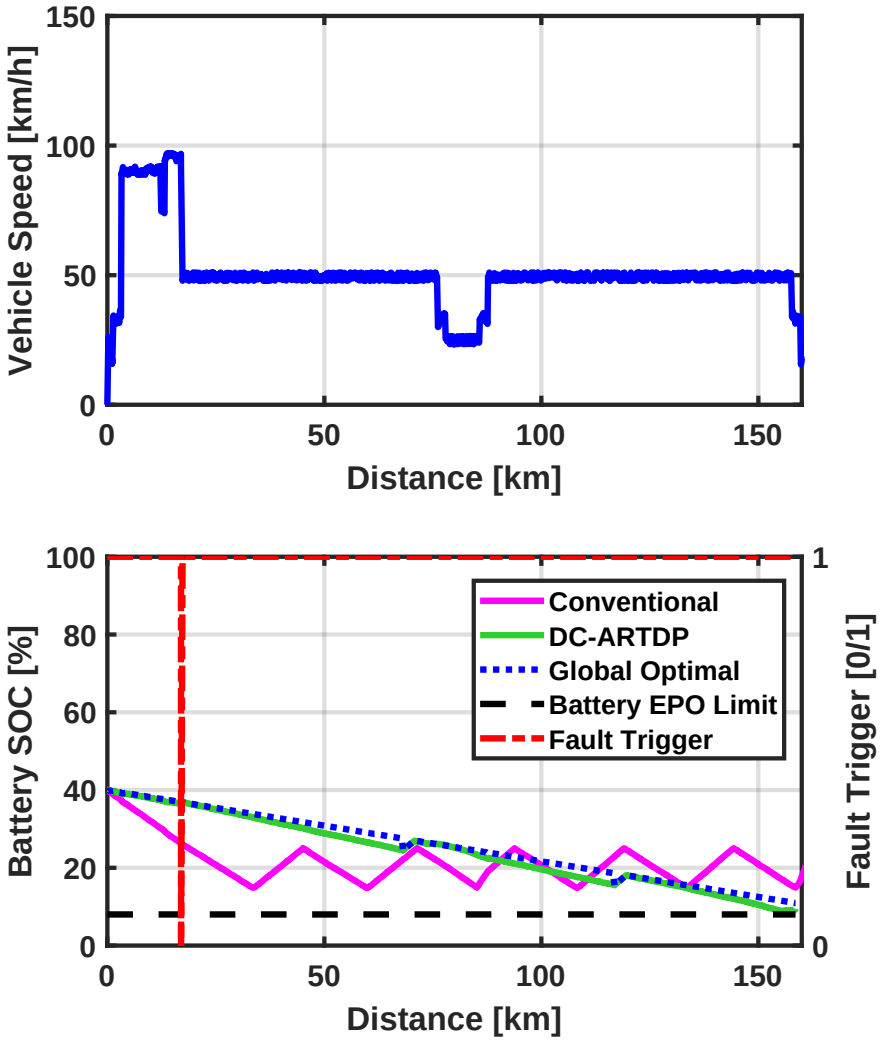


Figure 4.8: EREV battery SOC performance with motor deration fault occurring at $\sim$ 17.9 km marker.

DC-ARTDP algorithm's fault tolerant behavior is best exemplified with the REx deration fault scenario as observed from the results. In a REx deration scenario, the REx system on an EREV is incapable of providing maximum rated supplemental power. Such

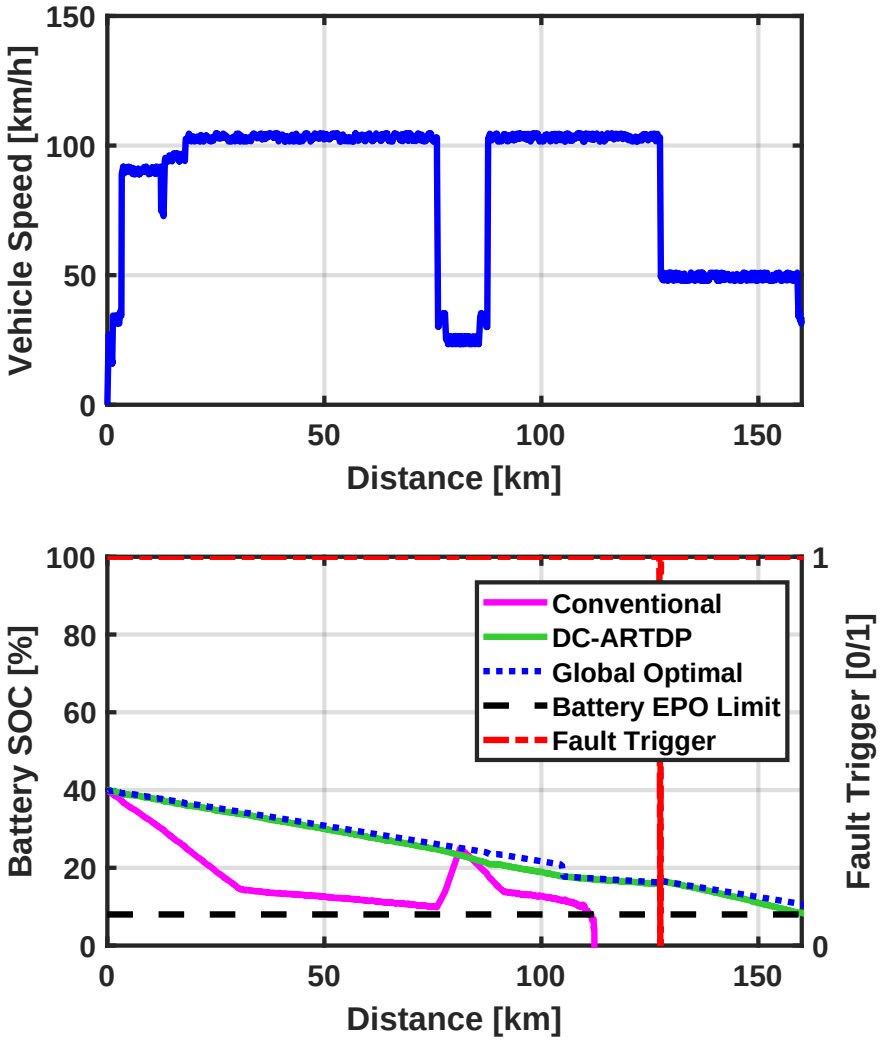


Figure 4.9: EREV battery SOC performance with motor deration fault occurring at $\sim$ 131.6 km marker.

a behavior can prevent EREVs battery to sustain charge effectively in the CS mode with a conventional energy management algorithm. An impact of this is reduced DTE due to the battery SOC dropping below the battery EPO limit. The results presented in Figure

4.6 and Figure 4.7, for such a fault align with the expected behavior. The DC-ARTDP algorithm, with its ability to adapt to changes in REx system performance, is able to operate the EREV in a blended CD-CS mode. This is helpful in reducing the possibility of a rapid reduction of battery SOC to EPO and thus provide larger DTE than the conventional algorithm.

Motor system deration fault scenario impacts total power demanded by the EREV. The deration fault results in reduced power demand, causing the EREV's DTE to extend. We found that during motor system deration fault, vehicle's performance varies between conventional and DC-ARTDP algorithm as battery initial SOC and motor deration power change. For a constant motor deration value, both the algorithms are able to attain a DTE greater than or equal to target for higher initial SOC values. Though for low initial SOC values, the DC-ARTDP algorithm is able to provide the EREV a high DTE. For the motor system deration fault shown in Figure 4.9, the initial SOC was varied between [20,100]% with a step of 20%. The sweep simulation study showed that only for EREV with DC-ARTDP algorithm, the target distance was attained for all initial SOC values. The conventional algorithm was unable to attain the target distance for initial SOC of 20% and 40%.

Based on the EREV's energy performance results in this study, the DC-ARTDP was able to adapt to system performance variations caused by inserted faults. The DC-ARTDP algorithm also provided a global optimal SOC trajectory to the EREV while meeting DTE requirements. The net energy consumption for DC-ARTDP in fault scenarios was determined to vary between 0.53 - 0.89 kWh/km. Net energy consumption values showed that for low REx power output, the REx efficiency was low but the fuel used was significant. This behavior can be attributed to the E85 fueled engine used in the REx system. The engine bsfc map showed peak efficiency near 17.5kW output from the engine which corresponds to 15kW output from the REx system [37]. For lower power outputs the engine uses more fuel compared to the energy output. This results in an increased

net energy consumption by the EREV. Such a behavior is a limitation of this EREV research vehicle. Thus, a variation in engine type and fuel can improve the net energy consumption while using DC-ARTDP.

4.5 Conclusion

In this article, we presented the fault tolerant behavior of our proposed novel DC-ARTDP algorithm on a research EREV. The algorithm was developed to operate in a real-time scenario with two major goals. These were, (i) to adapt EREVs response to energy system performance changes in case of a fault and (ii) to meet DTE requirements while being energy optimal. We improved our DC-ARTDP algorithm to use updated system parameters and perform re-optimization for optimal battery SOC trajectory upon fault detection. Using the model-in-loop approach of evaluation, we evaluated fault scenarios with the motor and REx system on the research EREV model. The EREVs performance during a fault was evaluated with the conventional and DC-ARTDP energy management algorithm. This approach was taken to validate and verify our proposed algorithm's fault tolerant behavior and benefit over the conventional EREV algorithm.

In conclusion, the model-in-loop study results presented in this article, showed that the DC-ARTDP algorithm was able to meet DTE requirements in all fault scenarios. Whereas, the conventional algorithm was found to be only effective in case of motor deration fault while initial SOC was higher than 50% SOC. Hence, our goal to develop an energy management algorithm that is fault tolerant and able to meet DTE requirements was achieved. A major limitation of the DC-ARTDP algorithm determined from this evaluation was with a relatively higher overall energy consumption in comparison to the conventional algorithm. This limitation persists only during extreme motor power deration fault with battery initial SOC greater than 50%. This behavior is attributed to the significantly low efficiency of E85 fuel operated engine in the research EREV at low power outputs. An improvement in the REx system can alleviate this challenge. A future

extension of this work includes, a processor-in-loop study to evaluate computational time and memory management of DC-ARTDP algorithm, that will assist in progressing towards in-vehicle applications.

Chapter 5

A Comparative Analysis of MPC and DC-ARTDP Algorithms for EREV Optimal Energy Management

Publication: *Submitted to ASME Journal of Dynamic Systems, Measurement and Control, A. V. Kalia and B. C. Fabien, "A Comparative Analysis of MPC and DC-ARTDP Algorithms for EREV Optimal Energy Management"*

5.1 Abstract

Intelligent energy management of hybrid electric vehicles is feasible with a priori information of route and driving conditions. Model Predictive Control (MPC) with finite horizon road grade preview has been proposed as a viable predictive energy management approach. We propose that our novel Distance Constrained - Adaptive Real Time Dynamic Programming (DC-ARTDP) approach can provide better energy management than MPC without any road grade information in context of an Extended Range Electric Vehicle (EREV). In this article, we have evaluated and compared the MPC and DC-ARTDP energy management strategies for a real-world driving scenario. The simulations were conducted for a 160km drive with road grade variation between $+4\%$ and -1%. Results show that the DC-ARTDP approach is optimal and at max 4.25% better

than the simple MPC with a finite horizon road grade preview implementation. Additionally, a higher value for energy storage system SOC tracking penalty p_2, results in the net energy consumption for MPC to converge towards that of DC-ARTDP. A combination of the MPC and DC-ARTDP approach is also evaluated with only 1.25% maximum improvement over simple MPC.

5.2 Introduction

EREV energy management problem is analogous to a resource allocation problem with a goal to efficiently and effectively match available resources to demand. For EREVs, a prior knowledge of future energy demands can help improve effectiveness and efficiency. Vehicle weight, speed, acceleration and road grade have a majority impact on vehicle's net power demand. Except for vehicle weight, the remaining factors can vary during the drive. Road grade variation is deterministic spatially i.e. for a given location on the road, the elevation will not change. Road grade data can be obtained through online Geographic Information System (GIS) resources or through active sensing with a combination of on-board IMU, GPS and/or perception systems. For an average sedan weighing 1520 kg with two average passengers, a variation of $+0.5\%$ in road grade at 26.82m/s (60 mi/h) can incur an additional power demand of $\sim$ 2.2kW. Thus, a prior knowledge of road grade among other factors can help effective allocation of energy resources in EREVs.

Predictive energy management methodologies for hybrid electric vehicles, using a *priori* route and driving condition data, have been extensively discussed in literature. Early implementations use model predictive control (MPC) to estimate future SOC behavior without road grade prediction [12, 13, 57, 53]. These implementations either assume prior knowledge of vehicle speed or use historic drive data and real-time drive data to make vehicle speed predictions. Sun et al. improved vehicle speed estimation methods by introducing exponential varying, stochastic Markov chains and neural-

network based approaches in the MPC framework [64]. Integration of traffic flow information [15] and use of preceding vehicle's movement for speed estimation [84] approaches, shows an emergence of Intelligent Predictive energy management methods. Though all these approaches show significant improvement in predictive energy management, they do not consider road grade variation in their prediction. Zhang et al. evaluated impact of road grade preview in context of a parallel-hybrid electric vehicle using Rule-Based and ECMS energy management strategies. Their research concluded that a road grade preview in hilly regions improved fuel economy by $1 - 4\%$ for the parallel-hybrid vehicle [83]. He et al. implemented a road grade prediction method along with MPC for predictive energy management of a split-parallel hybrid and reported an improvement of 4.7% over methods without preview [28]. Zeng and Wang also show benefits of estimate road grade with a stochastic approach in the MPC framework for a parallel-hybrid electric vehicle [82]. Limited implementations of predictive energy management strategies were found in context of EREVs or series hybrid electric vehicles (SHEV). Shen et al. presented benefits of predictive energy management with their implementation two MPC control algorithms for headway and energy management, respectively on an SHEV [60]. Hajimiri et al. show battery life extension and energy management improvement for SHEV with a fuzzy logic predictive control approach [26]. Wang et al. present their implementation and comparison of MPC based predictive energy management for a series hybrid electric bulldozer. Their results showed significant improvement over a simple Rule-Based strategy and achievement of 98% optimality [70].

Based on our literature review we ascertained that predictive energy management with road grade preview has shown improvement in overall energy management for hybrid electric vehicles. Additionally, the MPC framework for predictive energy management is a widely recognized approach. We also determined that a limited application of predictive energy management on EREVs and SHEVs is presented. Hence, in this

presented article we begin at sub-section 5.3.1 with the energy management problem in context of a EREV research vehicle [37] and an explanation of the implementation of the three different energy management algorithms considered. These include the basic CD-CS algorithm, MPC algorithm with road grade preview and our proposed DC-ARTDP algorithm. In sub-section 5.3.2 we describe the EREV research vehicle power loss model used for simulations. Our simulation scenario is presented in sub-section 5.3.3. We focus our study on energy consumption benefits amongst these algorithms and the importance of road grade preview for predictive energy management in comparison to our DC-ARTDP algorithm. We also determine implications of MPC horizon window and SOC tracking penalty variations on net energy consumption of the EREV. We finally summarize are findings for the EREV research vehicle studied over a real-world evaluation route in Section 5.4 and Section 5.5.

5.3 Methods

5.3.1 Energy Management Problem

Extended Range Electric Vehicle architecture uses two energy sources at a minimum for propulsion and auxiliary power demands. These include a high voltage battery or energy storage system (ESS) and a fuel operated generator system. The propulsion and auxiliary power demands are electric in nature, thus require the energy storage system to be operational at its rated capacity. Additionally, the generator output power is limited by the continuous charging power capability of the energy storage system. Thus,

$$P_{dem}^{t} \;=\; P_{prop}^{\;t} \;+\; P_{aux}^{t} \;=\; P_{ess}^{t} \;+\; P_{gen}^{t} \tag{5.1}$$

where,

$$\overline{P}_{ess}^{\,chg} \;\leq\; P_{ess} \;\leq\; \overline{P}_{ess}^{\,dchg}$$

and

$$0 \;\leq\; P_{gen} \;\leq\; min(\overline{P}_{ess}^{\,chg} \,,\; P_{gen}^{max})$$

The overall energy consumption for an EREV is determined by the relation in Eq.(5.2). This relation is minimized to obtain optimal overall energy consumption. As generator power is a controllable variable, the optimal problem can be formulated as an optimal generator power problem.

$$E_{net} = \sum_{t=0}^{N} \frac{P_{ess}^{\ t} + P_{gen}^{\ t}}{d_{veh}^{t} \times 3600} \tag{5.2}$$

$$P_{gen}^{\ t} = \eta_{MG} \frac{\dot{m}_{fuel} \times 3600}{BSFC(P_{eng}, \omega_{eng})} \tag{5.3}$$

$$\arg\min_{\dot{m}_{fuel}} E_{net} = \sum_{t=0}^{N} \frac{1}{d_{veh}^{t} \times 3600} \left[P_{ess}^{\ t} \ + \ \eta_{MG} \frac{\dot{m}_{fuel} \times 3600}{BSFC\ (P_{eng},\ \omega_{eng})} \right] \tag{5.4}$$

The generator power output is non-linearly dependant on generator fuel consumption as shown in Eq.(5.3). This the overall energy consumption calculation is required to include fuel energy consumption for the generator and not the electric energy. Thus, the optimal energy consumption relation for EREVs is shown in Eq.(5.4). This forms the basis for Energy Management Control approaches evaluated and mentioned in subsections 5.3.1.1 - 5.3.1.4.

5.3.1.1 CD-CS Energy Management Control Approach

The Charge Depleting - Charge Sustaining (CD-CS) energy management control approach is the most widely implemented energy management strategy in production EREVs. This approach divides EREV energy system operation into two operation zones. The charge depleting zone, where only the ESS meets the energy demand and the charge sustaining mode, where ESS and generator both meet the energy demand while maintaining the ESS state of charge within a set of bounds. The CD-CS approach uses optimal generator operating points (P_{gen}^{*}) obtained through offline optimization. The minimization problem as shown in Eq.(5.4) is solved for within the CS mode SOC limits i.e. SOC_{ess}^{UL} and SOC_{ess}^{LL}. In this study, the CD-CS control approach is referred to as

Algorithm 4: CD-CS Energy Management Control Algorithm

Input: SOC_{ess}, SOC_{ess}^{UL}, SOC_{ess}^{LL}, P_{gen}^{*}

Output: P_{gen}

if *(*$SOC_{ess} \geq SOC_{ess}^{UL}$*) and (*$SOC_{ess} \geq SOC_{ess}^{LL}$*)* **then**

  $P_{gen} \leftarrow 0$;

endif

else if *(*$SOC_{ess} \leq SOC_{ess}^{UL}$*) and (*$SOC_{ess} \geq SOC_{ess}^{LL}$*)* **then**

  $P_{gen} \leftarrow P_{gen}^{*}$;

endif

the *baseline* strategy. The ESS SOC limits used in the study are $SOC_{ess}^{UL} = 0.25$ and $SOC_{ess}^{LL} = 0.15$, where $SOC_{ess} \in [0,\ 1.0]$.

5.3.1.2 Predictive Energy Management Control with MPC Approach

Prior knowledge of expected energy demand by a vehicle is beneficial in energy management and optimization. Infinite horizon or finite horizon estimation or prediction approaches are able to implement predictive energy management control methods for hybrid electric vehicles. A receding horizon control or model predictive control (MPC) approach has been extensively evaluated in research for such an application. For an EREV, this energy management control approach is significantly better than CD-CS approach, for its ability to preemptively react to expected future high energy demands.

In this study, the implemented MPC observes EREV SOC_{ess} state while manipulating the P_{gen}^{t} input variable. The observed SOC_{ess} is compared against reference SOC_{ess}^{ref} obtained from Dynamic Programming. A receding horizon energy consumption estimate is made by assuming vehicle speed (v_{veh}) and road grade (α_{road}) as measured disturbances. For a temporal estimation horizon of N and resolution of 1 second, Eq.(5.5) computes the future energy demands and subsequently the estimated ESS SOC ($\widehat{SOC}_{ess}^{t+i}$)

from Eq.(5.6), for the vehicle. The receding horizon optimization is performed on the cost function J per Eq.(5.7) over the horizon window.

$$\begin{aligned} P_{dem} &= P_{gen} + P_{aux} + (m_{veh} + m_{pass})v_{veh}a_{veh} + \frac{1}{2}\rho_{air}A_f C_d v_{veh}^3 \\ &+ C_{rr}(m_{veh} + m_{pass})gcos(\alpha_{road})v_{veh} + (m_{veh} + m_{pass})gsin(\alpha_{road})v_{veh} \end{aligned} \tag{5.5}$$

$$\widehat{SOC}_{ess}^{\,t+1} = SOC_{ess}^{\,t} - \frac{P_{dem}}{E_{ess}^{max} \times 3600} \tag{5.6}$$

$$\arg \min_{P_{gen}, SOC_{ess}} J = \sum_{i=0}^{N-1} \Big[p_1 (w_1 P_{gen}^{\,t+i})^2 + p_2 (w_2 (SOC_{ess}^{\,t+i} - SOC_{ess,\,ref}^{\,t+i}))^2 \Big] \tag{5.7}$$

The cost function J is a strictly convex quadratic problem solving for $P_{gen}^{\,t+i}$ and $SOC_{ess}^{\,t+i}$. Inequality constraints of,

$$0 \leq P_{gen}^{\,t+i} \leq 15000 \tag{5.8}$$

$$\min[\widehat{SOC}_{ess}^{\,t+i}] \leq SOC_{ess}^{\,t+i} \leq \max[\widehat{SOC}_{ess}^{\,t+i}] + \Big(\frac{15000}{E_{ess}^{max} \times 3600} \Big) \tag{5.9}$$

along with an equality constraint of,

$$SOC_{ess}^{\,t+i} - \frac{P_{gen}^{\,t+i}}{E_{ess}^{max} \times 3600} = \widehat{SOC}_{ess}^{\,t+i} \tag{5.10}$$

are applied on the cost function. The terms $w_1,\ w_2$ and $p_1,\ p_2$ in Eq.(5.7) are weights and penalties applied to the two variables being solved.

A strictly convex quadratic problem is represented as $\frac{1}{2}\mathbf{x}^\intercal H\mathbf{x} + f^\intercal \mathbf{x}$, where $\mathbf{x}$ represents the state vector, H represents the Hessian matrix, and f represents the gradient vector. For an N second horizon window $\mathbf{x}$, H and f are determined as follows,

$$\mathbf{x}_{[3N\times 1]} = \begin{bmatrix} P_{gen}^{\,t} \\ SOC_{ess}^{\,t} \\ 1 \\ P_{gen}^{\,t+1} \\ SOC_{ess}^{\,t+1} \\ 1 \\ \vdots \\ P_{gen}^{\,t+N-1} \\ SOC_{ess}^{\,t+N-1} \\ 1 \end{bmatrix}, \; f_{[3N\times 1]} = \begin{bmatrix} 0 \\ -2p_2k_2^2SOC_{ess}^{\,ref} \\ 0 \\ 0 \\ -2p_2k_2^2SOC_{ess}^{\,ref} \\ 0 \\ \vdots \\ 0 \\ -2p_2k_2^2SOC_{ess}^{\,ref} \\ 0 \end{bmatrix}$$

$$H_{[3N\times 3N]} = \begin{bmatrix} p_1w_1^2 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & p_2w_2^2 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & p_2SOC_{ess}^{\,ref} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & p_1w_1^2 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & p_2w_2^2 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & p_2SOC_{ess}^{\,ref} \end{bmatrix}$$

while the inequality constraint $A\mathbf{x} \geq b$ and equality constraint $A_{eq}\mathbf{x} = b_{eq}$ vectors are formed as described below.

$$A_{[4\times 3N]} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 1 & 0 & 0 \\ -1 & 0 & 0 & \cdots & -1 & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 1 & 0 \\ 0 & -1 & 0 & \cdots & 0 & -1 & 0 \end{bmatrix}$$

$$b_{[4\times1]} = \begin{bmatrix} 0 \\ -NP_{gen}^{max} \\ \sum_{i=0}^{N-1} \widehat{SOC}_{ess}^{t+i} \\ -\Big(\sum_{i=0}^{N-1} \widehat{SOC}_{ess}^{t+i} + \frac{1.5\text{e}+4}{E_{ess}^{max}\times 3600}\Big) \end{bmatrix}$$

$$A_{eq_{[1\times3N]}} = \begin{bmatrix} -(\frac{1}{E_{ess}^{max}\times3600}) & 1 & 0 & \cdots & -(\frac{1}{E_{ess}^{max}\times3600}) & 1 & 0 \end{bmatrix}$$

$$b_{eq_{[1\times1]}} = \begin{bmatrix} \sum_{i=0}^{N-1} \widehat{SOC}_{ess}^{t+i} \end{bmatrix}$$

The penalty values of $p_1 = 1$, $p_2 = \{1\text{e}6,\ 1\text{e}8\}$, and weight values of $w_1 = 1.41\text{e}-4$, $w_2 = 1$ are used for this study. A kinematic vehicle model per Eq.(5.11) is used to estimate vehicle speed for the N horizon window. The estimation model assumes that vehicle acceleration is constant over the horizon i.e. $a_{veh}^{t+i} = a_{veh}^{t}\ \forall i \in [0, \cdots, N-1]$. The estimated vehicle speed $\widehat{v}_{veh}$ is saturated between the feasible upper and lower limits of the vehicle i.e. $v_{veh}^{max} = 35.76$ m/s, $v_{veh}^{min} = 0$ m/s.

$$\widehat{v}_{veh_{[N\text{x}1]}}^{t+i} = v_{veh_{[N\text{x}1]}}^{t} + a_{veh_{[N\text{x}N]}}^{t+i} \cdot t_{[N\text{x}1]} \tag{5.11}$$

The control and estimation horizon window for the implemented MPC is chosen to be of the same size. This approach is referred to as a *simple MPC* strategy in this study.

5.3.1.3 Distance Constrained - Adaptive Real Time Dynamic Programming Approach

The *simple MPC* approach mentioned in the previous subsection is a better approach than CD-CS, but provides a sub-optimal solution. Additionally, it requires road grade information to effectively estimate future energy demands and respond accordingly. Increasing the horizon window N to capture significant road grade data leads to an increased computation time. Our novel Distance Constrained - Adaptive Real Time Dynamic Programming (DC-ARTDP) approach provides a global optimal solution while

considering road grade as an unmeasured disturbance to the EREVs energy system.

Algorithm 5: Forward propagating dynamic programming with distance constraint.

Input: $\bar{\mathbf{v}}_{veh},\ \text{init\_}SOC_{ess},\ \text{min\_}SOC_{ess},\ \text{max\_}SOC_{ess},\ \mathsf{t}_s, \text{max\_}d_{veh},\ d_{rem}$

Output: $\mathbf{SOC}^*_{ess},\ \mathbf{P}^*_{gen}$

$N \longleftarrow \text{length}(\bar{\mathbf{v}}_{veh})$

$t \longleftarrow 0$

while $t \leq N$ **do**

  if $(t == 0)$ **then**

    $\mathbf{SOC}^*_{ess}[t] \longleftarrow \text{init\_}SOC_{ess}$

    $d_{veh}[t] \longleftarrow 0$

  end

  $d_{veh}[t+1] = d_{veh}[t] + \left(v_{veh}[t+1] \times \mathsf{t}_s\right)$

  $SOC_{ess}[t+1] \longleftarrow \mathbf{SOC}^*_{ess}[t] - \left(\frac{P_{dem}[t+1] + P_{gen}[t+1]}{E^{max}_{ess} \times 3600}\right),\ \forall\, P_{gen} \in \mathsf{U}_{P_{gen}}$

  $d_{DTE}[t+1] \longleftarrow \left[\frac{SOC_{ess}[t+1] - \text{min\_}SOC_{ess}}{\text{init\_}SOC_{ess} - \mathbf{SOC}^*_{ess}[t]}\right] \times d_{veh}[t]$

  $\arg\min_{P_{gen},\, SOC_{ess}} EC_{net}[t+1] \longleftarrow$

  $\frac{1}{d_{veh}[t+1]}\Big[(\text{init\_}SOC_{ess} - SOC_{ess}[t+1])E^{max}_{ess} + (\eta^{-1}_{gen} P_{gen}[t+1]\ \frac{BSFC \times LHV}{3600})\Big],$

  $\forall\, SOC_{ess} : d_{DTE}[t+1] \geq \text{max\_}d_{veh} + d_{rem}$

  $\mathbf{SOC}^*_{ess}[t+1] \longleftarrow \mathbf{SOC}^*_{ess}[t] - \left(\frac{P_{dem}[t] + \mathbf{P}^*_{gen}[t]}{E^{max}_{ess} \cdot 3600}\right)$

  $t \longleftarrow t + \mathsf{t}_s;$

end

For a known average vehicle speed ($\bar{\mathbf{v}}_{veh}$) between two locations A and B, the DC-ARTDP algorithm computes the optimal ESS SOC ($\mathbf{SOC}^*_{ess}$) and optimal generator power ($\mathbf{P}^*_{gen}$) trajectory. The distance constrained forward propagating dynamic programming algorithm used in DC-ARTDP is shown in Algorithm 5. For each stage of forward propagation, the algorithm computes a feasible ESS SOC vector (SOC_{ess}) subject to distance

to empty (d_{DTE}) constraint condition. Overall energy consumption (EC_{net}) is treated as a cost function and minimized for optimal parameters. Though variation in actual vehicle speed and the unmeasured disturbance from road grade variation can cause the real-time ESS SOC (SOC_{ess}^{t}) to deviate from optimality. To adapt to these deviations, the algorithm re-computes an optimal trajectory of SOC_{ess} and P_{gen} for the remaining distance to be covered. An additional variable distance parameter (d_{rem}) is added to ensure additional driving distance is feasible and can be varied during re-optimization or route changes.

The *DC-ARTDP* approach provides optimal energy management without finite horizon knowledge of road grade. This method is also adaptable to plausible errors in predictive energy demand estimation due to incorrect road grade data.

5.3.1.4 Combined DC-ARTDP and MPC Approach

As mentioned earlier, the *simple MPC* approach uses SOC_{ess}^{ref} to track and determine required P_{gen} at each time step. The approach also predicts future energy demand over a fixed horizon, which assists in the P_{gen} calculation. A limitation to this approach is its adaptability to changes in route and/or speed during the drive. The *DC-ARTDP* approach on the other hand is able to adapt to such changes and thus update the global optimal SOC trajectory. In this article, we also combine these two approaches to evaluate possible benefits to predictive energy management. In this *combined MPC DC-ARTDP* approach the SOC_{ess}^{ref} is frequently updated as SOC_{ess} deviates from the reference. Meanwhile, the MPC component ensures implementation of predictive energy management.

The *baseline*, *simple MPC*, *DC-ARTDP*, and *combined MPC DC-ARTDP* approaches discussed were evaluated on the EREV research vehicle power flow model considered in this research study.

5.3.2 EREV Power Flow Model

The experimental research extended range electric vehicle built on the 2016 Chevrolet Camaro platform was used as the target vehicle for this study. The research vehicle uses a series hybrid electric architecture with an 18.9 kWh ESS, an E85 fueled on-board range extender or generator system with a peak continuous output of 15kW and two rear electric traction motors. The power flow model of the research vehicle uses the 1DOF longitudinal dynamics equation to determine the propulsive power demand. Figure 5.1 shows the model schematic with the energy management control algorithm in a feedback loop as implemented in MATLAB/Simulink. EREV power demand calculator input is vehicle speed (v_{veh}) and road grade (α_{road}) obtained from the drive cycle block. The power demand calculator uses these inputs along with Eq.(5.5) to determine propulsion power demand (P_{dem}).

The EREV propulsion system model block implements rear traction motors, energy storage system, generator system and auxiliary power demand sub-models to compute the net electric power demand P_{elec}. P_{elec} contribution from rear traction motors is calculated using a combined motor and gearbox efficiency map, $\eta_{mot}(\tau_{mot}, \omega_{mot})$. The generator system contribution uses BSFC map for the engine along with constant efficiency value of 0.86 for the generator motor. Auxiliary power model replicates a continuous low voltage power demand on the ESS. The model implements this as a constant power demand of 520 W, determined for research vehicle data, with an Additive White Gaussian Noise having a signal-to-noise ratio of 25. The EREV power flow model was validated against actual vehicle data in a previous publication [37]. Along with P_{elec}, the observed ESS energy state SOC_{ess}, and generator system mass fuel flow rate $\dot{m}_{fuel}$ are treated as inputs to the energy management control algorithm block. This block implements the earlier mentioned *baseline*, *simple MPC*, *DC-ARTDP*, and *combined MPC DC-ARTDP* strategies. The control output from this block is the requested generator power P_{gen}. To simulate the system for a real-time application, the power flow model also implements

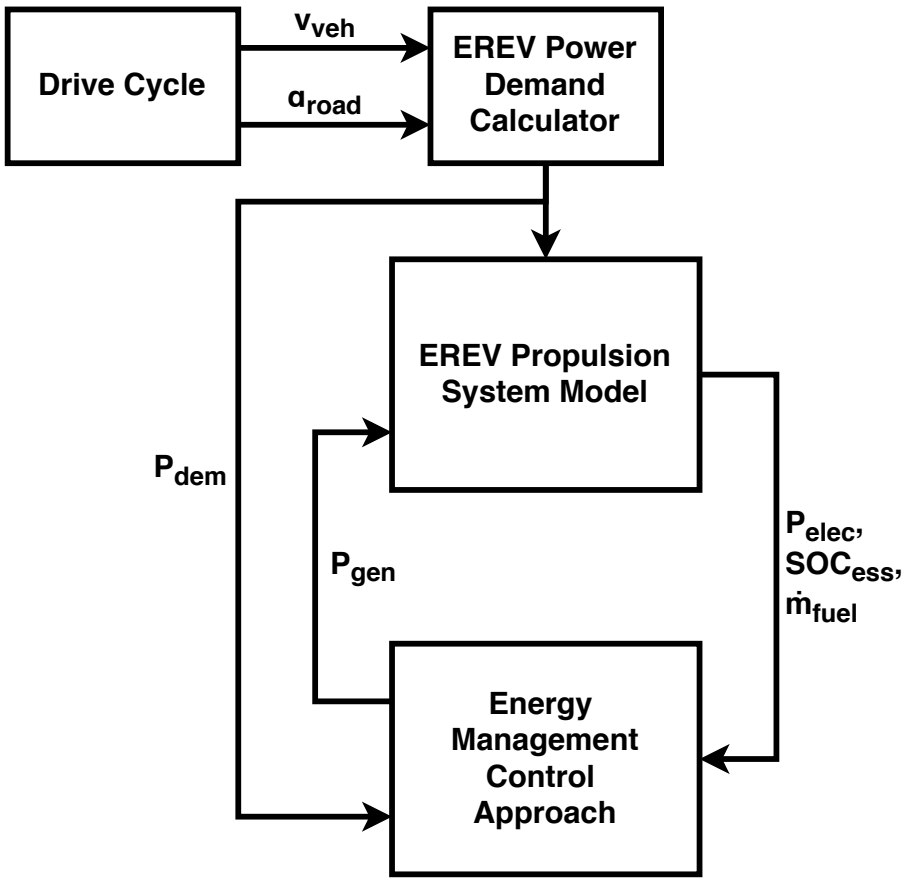


Figure 5.1: EREV power flow model architecture diagram used for simulations.

a drive cycle generator using Google API as well as a distance to empty (DTE) calculator.

The drive cycle generator tool uses origin and destination location information to determine the path, distance, elevation and average driving speed using Google API suite. The average speed information obtained is used as a *posterior* estimated vehicle speed for forward propagating Dynamic Programming used in *DC-ARTDP*. The distance to empty calculation is required to estimate feasible driving range. In this model, we compute a long term average of EREV's electric energy consumption $\overline{E}_{net,elec}$ to determine the DTE using Eq.(5.12). Computed d_{DTE} at each time step is compared against desired driving distance in the *DC-ARTDP* approach.

$$d_{DTE} = \frac{[SOC_{ess} - \text{min\_}SOC_{ess}] \times E_{batt}}{\overline{E}_{net,elec}} \tag{5.12}$$

5.3.3 Simulation Scenario

The goal of this study was to evaluate the energy management approaches for the research EREV in real-world driving environments. Thus, we generated a round-trip real-world evaluation route with origin coordinates [47.6534,-122.3041] and turn-around destination coordinates [47.4194,-121.5771]. Figure 5.2 shows the evaluation route highlighted in blue with origin and turn-around destination represented with triangle markers of red and green color, respectively. The simulated vehicle speed and road grade used for simulation and obtained from the drive cycle generator tool are shown in Figure 5.3.

The 160 km evaluation route is simulated using MATLAB/Simulink with the *baseline*, *simple MPC*, *DC-ARTDP*, and *combined MPC DC-ARTDP* energy management approaches. For these simulations, the initial ESS SOC (init_SOC_{ess}) is 0.9, minimum ESS SOC (min_SOC_{ess}) is 0.1 and ESS SOC emergency power off at 0.08. The EREV research vehicle is assumed to match demanded speed perfectly in these simulations. Hence, it does not consider vehicle's powertrain inertia and dynamics.

5.4 Results and Discussion

time and real-time implementation. In this comparative study on MPC and DC-ARTDP energy management control algorithms for EREV, we have focused on two key attributes i.e. net energy consumption and computation time. We also present exploratory results for MPC application on predictive energy management with variation in the size of horizon window and the cost function penalty on SOC tracking. These results help compare the MPC and DC-ARTDP algorithms as well as understand feasible benefits or disadvantages of a combined MPC DC-ARTDP alogrithm.

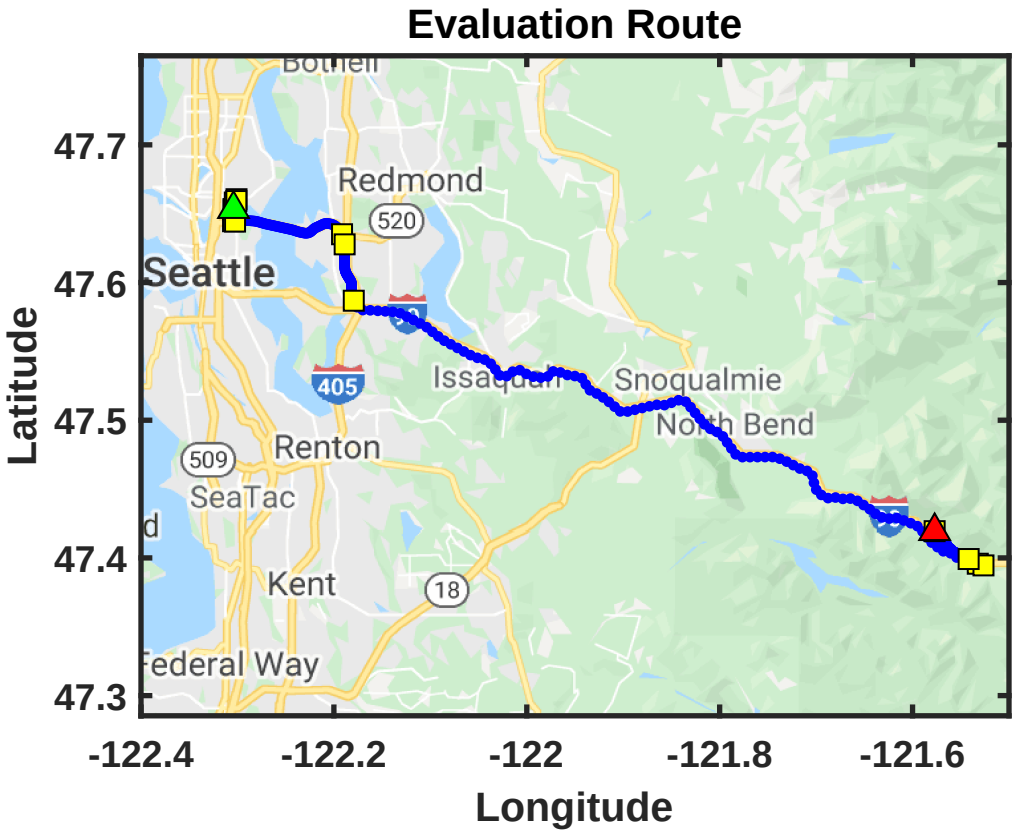


Figure 5.2: Evaluation route chosen for EREV energy management study

5.4.1 Effect of Varying MPC Horizon Window

The MPC horizon window size h dictates controller's predictive estimation and control capability. To determine a reasonable value of h for this study, we evaluated the effect of varying h on P_{gen} set point. For this evaluation we selected a vehicle cruise speed of 26.82 m/s (60 mi/h) and a sinusoidal road grade variation within $\pm 1\%$ limits per Figure 5.4. The *simple MPC* approach with SOC tracking penalty as $p_2 = 1e6$ was evaluated with five different values where $h = \{1, 5, 10, 20, 30\}$. Figure 5.5 shows a comparison of P_{gen} values for different values of h in the $\sim 5.5 - 7.5$km distance section as highlighted in Figure 5.4.

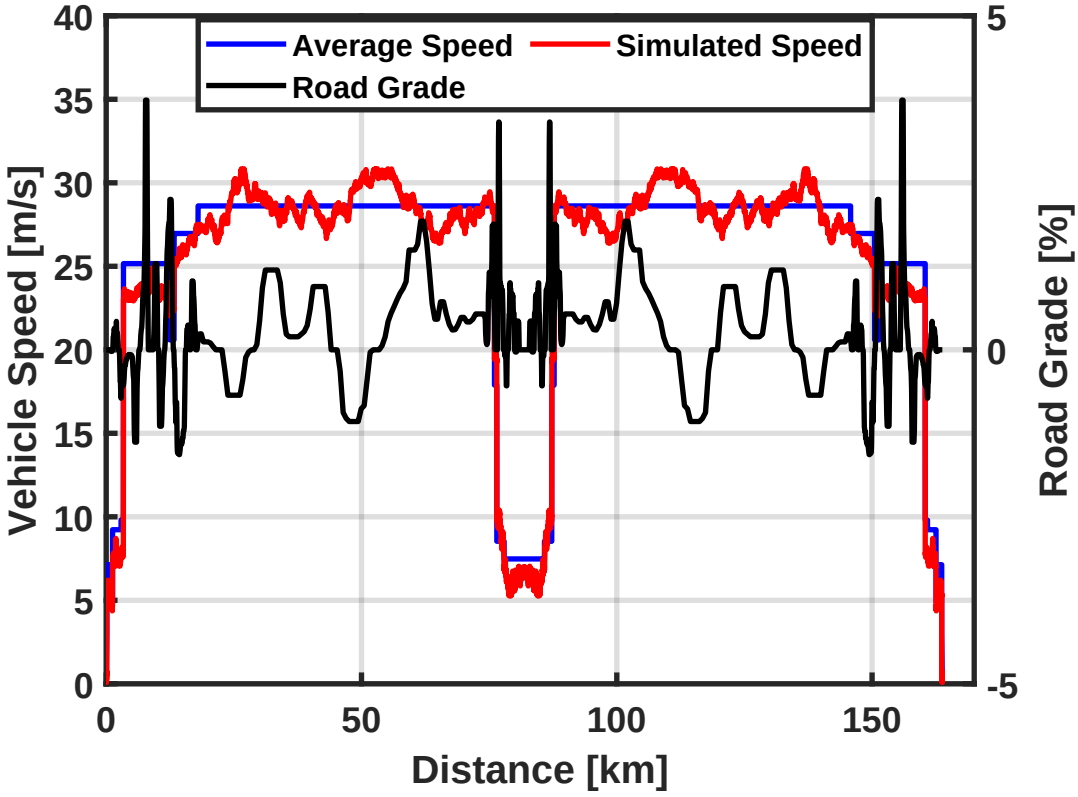


Figure 5.3: Vehicle speed trace and road grade targets for the evaluation route

In the P_{gen} comparison per Figure 5.5, it is evident that a larger value of h results in an earlier rise of P_{gen} value. For $h = 30$ the P_{gen} shows a rise marked by label **A** near distance marker 5.9km, while for $h = 1$ the rise is marked by label **E** near distance marking 6.5km. An effect of this can also be observed in the peak P_{gen} value, where the difference between label **A** and label **E** is nearly 0.8 kW. This result shows that a prior knowledge of energy demand due to a large horizon can reduce peak power demand and result in reduction in net energy consumption. Although certain practical constraints pose a limit on the feasible values of h. As used in this study, the value of $h = 30$, where h represents a temporal horizon of 1.0 second steps, with a cruise speed of 26.82 m/s implies

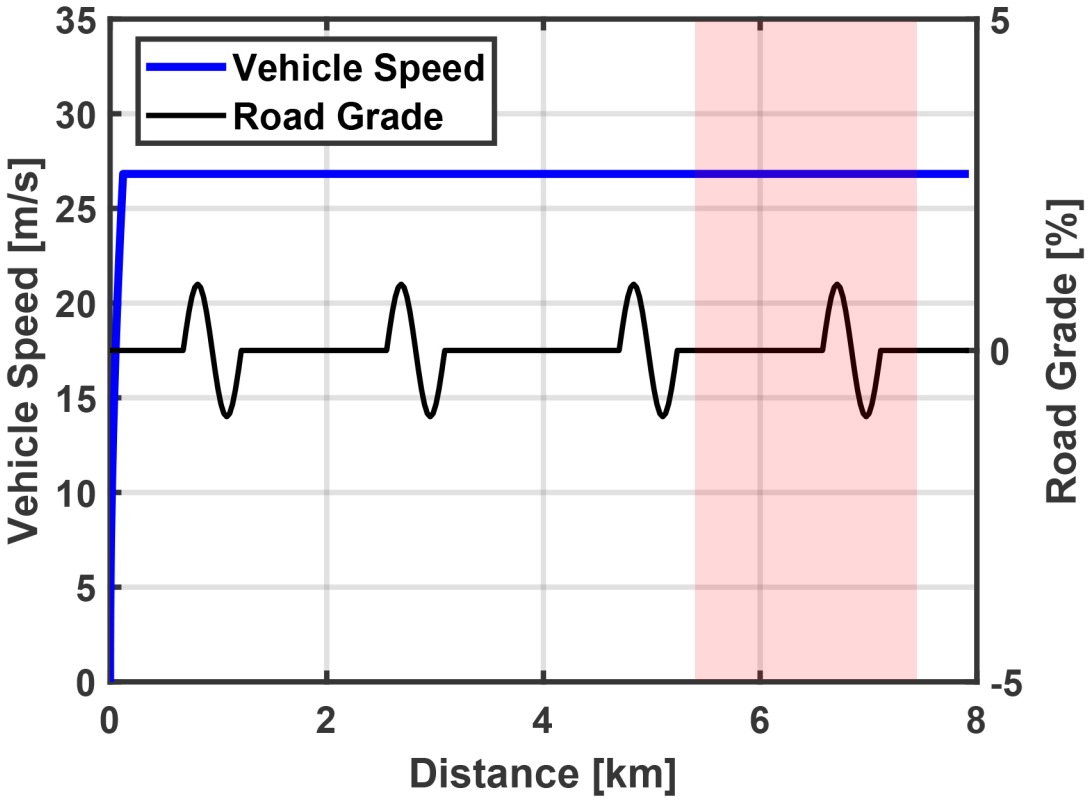


Figure 5.4: Cruise vehicle speed of 26.82 m/s and road grade variation of $\pm$ 1%.

a maximum spatial horizon of $\sim$ 800m. Such a spatial horizon with active perception sensing hardware is not currently feasible, unless a prior knowledge of road elevation is obtained. As the current active perception limit is at 300m, we selected a value of $h = 10$, for this study. From a computation time standpoint, increasing the value of h increases the dimension of the Hessian matrix (H) for the strictly convex quadratic problem solved as a part of MPC. Thus, a lower value of h is also computationally beneficial.

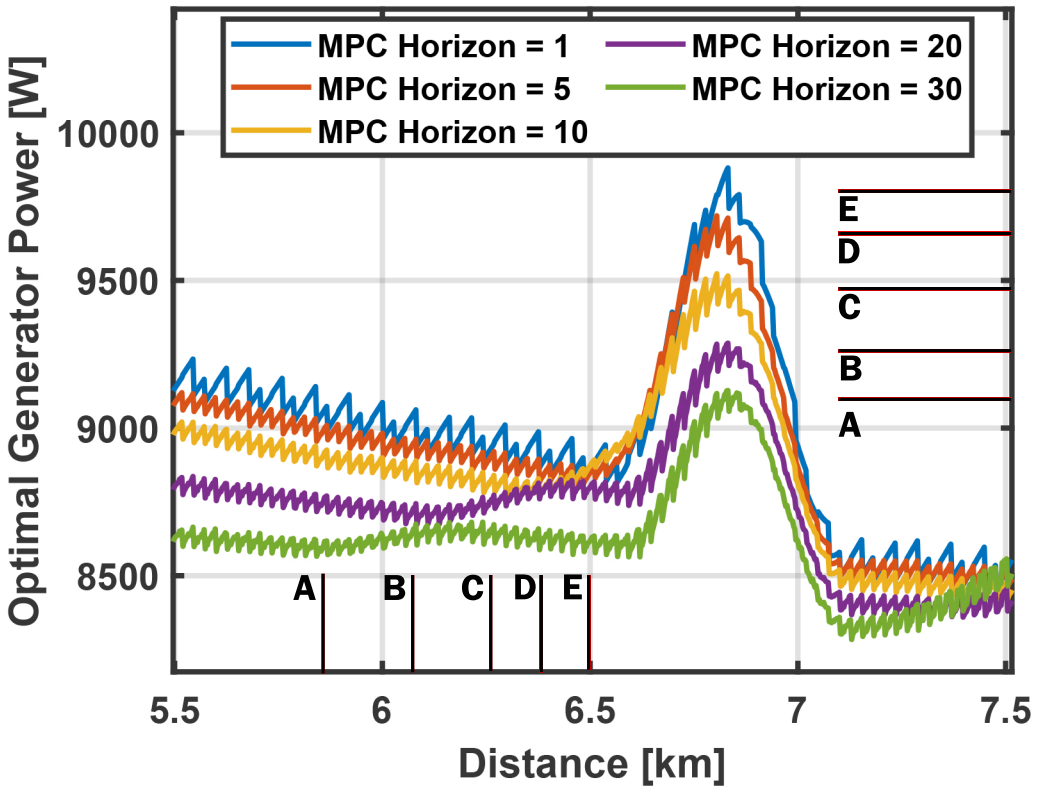


Figure 5.5: Effect on P_{gen} with varying MPC horizons for a sinusoidal road grade variation with different horizon windows for the $\sim$ 5.5 $-$ 7.5 km section

5.4.2 *Effect of Varying MPC SOC_{ess} Penalty*

Another important variable parameter in the *simple MPC* formulation is the SOC_{ess} tracking penalty p_2, as shown in Eq.(5.7). As per the cost function formulation, when the value of p_2 is increased the SOC tracking is favored over fuel consumption by the generator. To understand its impact on P_{gen}, we evaluated five different values where $p_2 = \{1e4, 1e5, 1e6, 1e7, 1e8\}$ for the same drive cycle parameters as shown in Figure 5.4. Figure 5.6 shows the P_{gen} operation for different values of p_2 as obtained from the simulations. In accordance with the expected behavior, as the value of p_2 increases, the

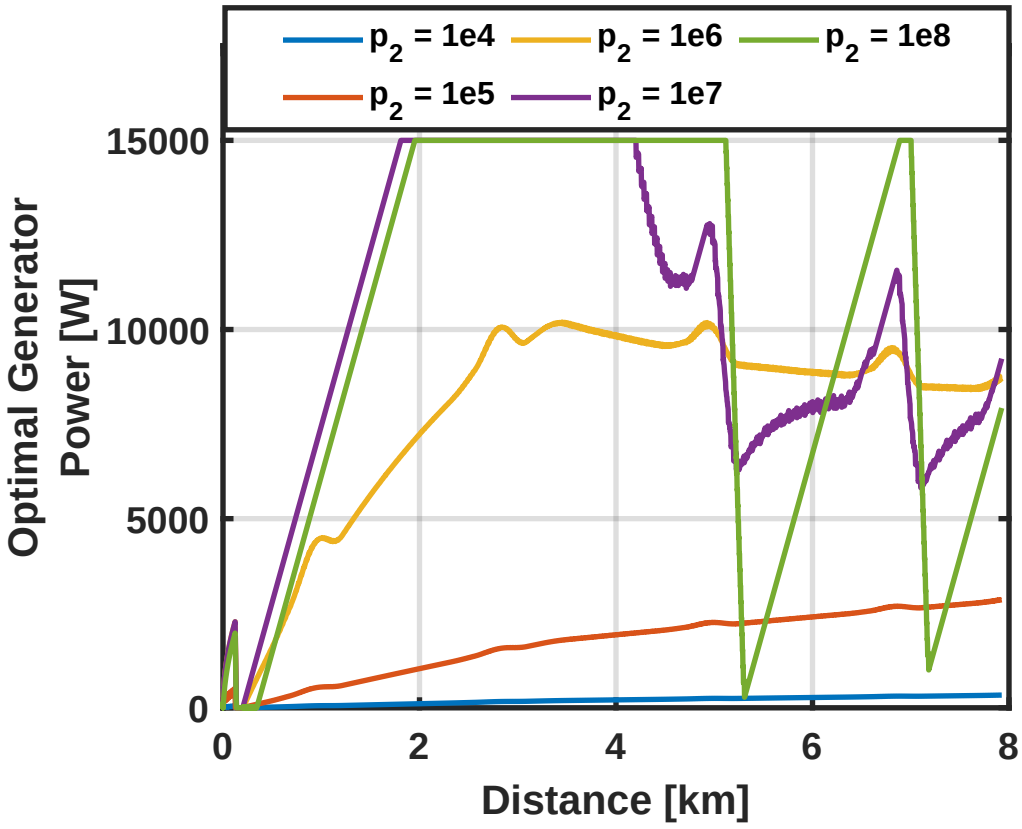


Figure 5.6: Effect on P_{gen} with varying MPC SOC_{ess} penalty p_2 for a sinusoidal road grade variation

peak operating point of P_{gen} is also observed to increase. At $p_2 = 1e6$, P_{gen} can be observed to react to change in road grade. As the value of p_2 is further increased, P_{gen} can be observed to saturate at the maximum limit of 15000 W with the generator response being similar to that of a thermostatic control. We also observed a $7 - 10\%$ increase in computation time with an increase in the value of p_2 from $1e6 - 1e8$. For the study, we selected two p_2 values of $1e6$ and $1e8$, respectively to further assess the impact.

5.4.3 Net Energy Consumption Comparison

The EREV research vehicle evaluated in this study uses a combination of electric and fuel energy to meet the energy demand. The net energy consumption by the EREV was computed per Eq.(5.2) and optimized per Eq.(5.4). We compared net energy consumption for the *baseline*, *simple MPC*, *DC-ARTDP*, and *combined MPC DC-ARTDP* algorithms implemented for the return drive simulated for evaluation route. Figures 5.7 - 5.10 show the change in ESS SOC and Net Energy Consumption as a function of the distance travelled for these algorithms. In Figures 5.7 and 5.8 we use $p_2 = 1e6$ for *simple MPC* and

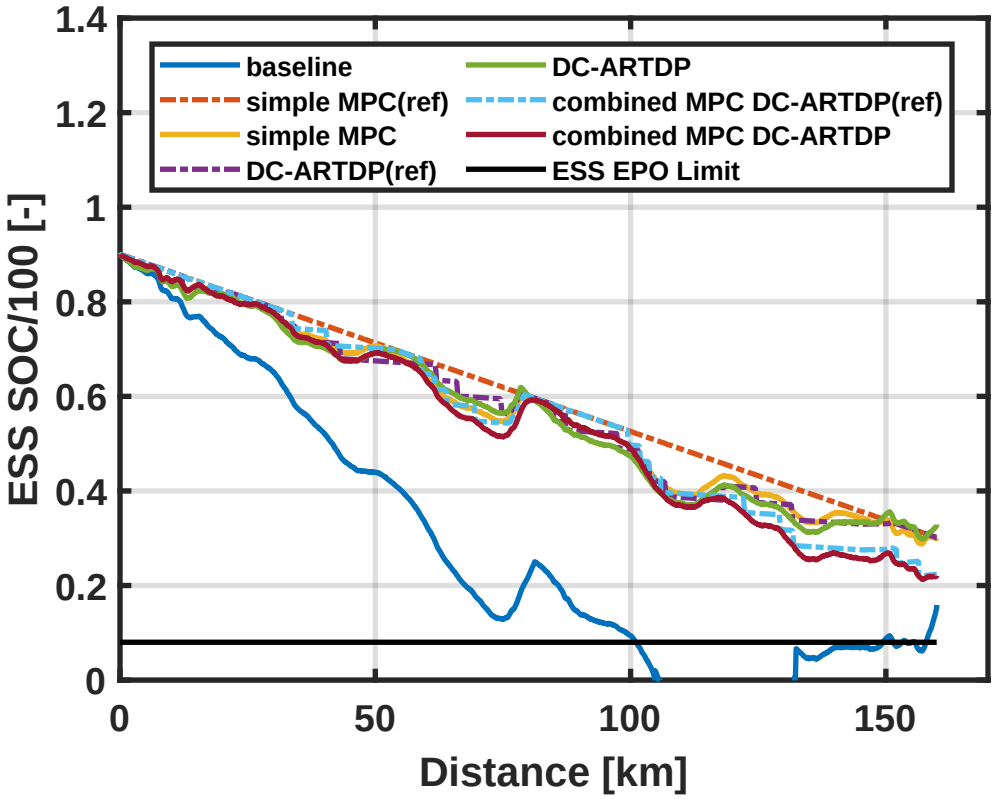


Figure 5.7: SOC_{ess} comparison for different Energy Management Control approaches during evaluation route simulation with MPC SOC penalty $p_2 = 1e6$.

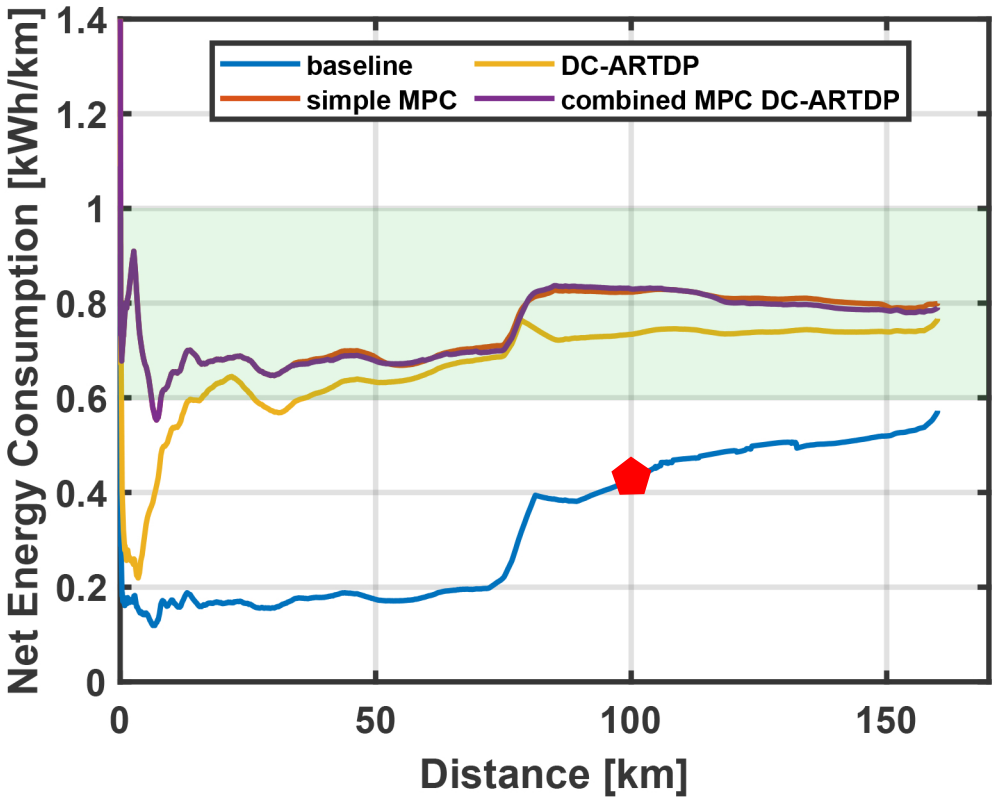


Figure 5.8: EC_{net} comparison for different Energy Management Control approaches during evaluation route simulation with MPC SOC penalty $p_2 = 1e6$.

combined MPC DC-ARTDP algorithms. It is evident from the ESS SOC plot that the *baseline* algorithm is unable to meet excessive energy demands due to extreme road grade conditions. This results in the vehicle's ESS SOC dropping below the ESS EPO Limit, implying the vehicle will not be able to meet the required driving distance. For the other algorithms, ESS SOC stays well above the ESS EPO Limit. While *DC-ARTDP* algorithm is able to attain a final ESS SOC of ~ 0.31, the *simple MPC* algorithm also shows a similar final result. Though, the *combined MPC DC-ARTDP* algorithm attains a relatively lower final ESS SOC of ~ 0.22.

Net Energy Consumption plot for these algorithms shows, the *baseline* having the lowest value, although it is unable to complete the required driving range. Among the other three algorithms *DC-ARTDP* has the lowest value at 0.766 kWh/km, *combined MPC DC-ARTDP* as the next highest at 0.79 kWh/km, and *simple MPC* the highest at 0.8 kWh/km. Thus, the *DC-ARTDP* approach is observed to be the most energy optimal algorithm in comparison. For $p_2 = 1e8$ in MPC related algorithms being evaluated in

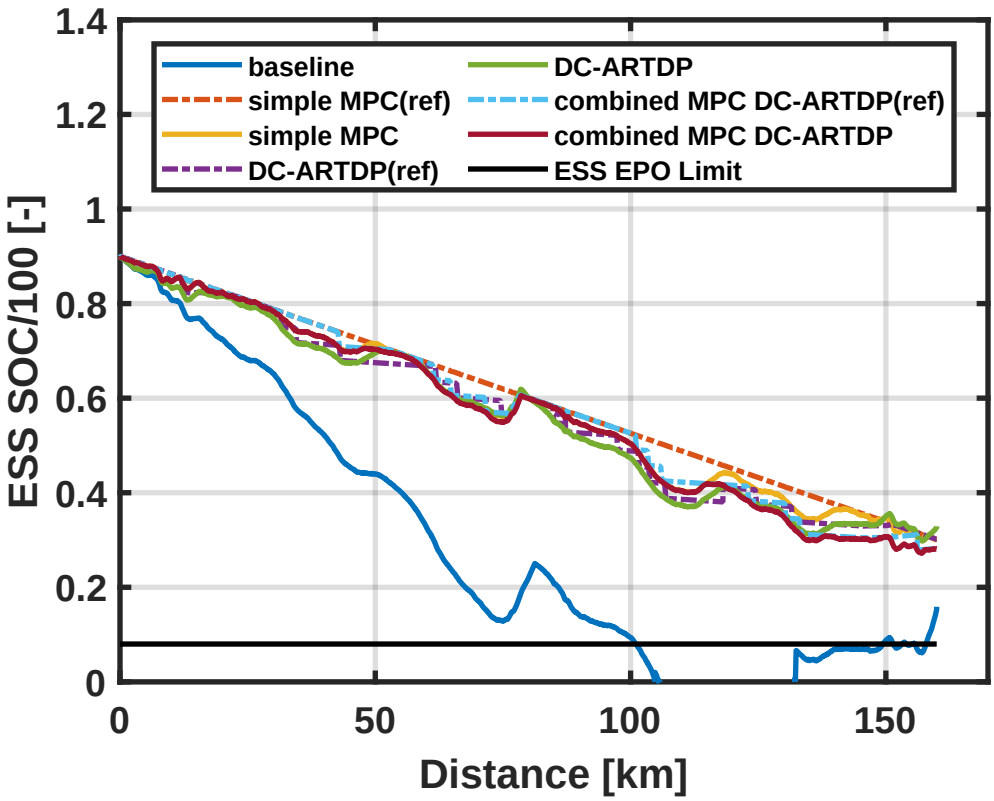


Figure 5.9: SOC_{ess} comparison for different Energy Management Control approaches during evaluation route simulation with MPC SOC penalty $p_2 = 1e8$.

this study, Figures 5.9 and 5.10 show the comparison. The key difference from Figure 5.9 is observed in outcomes from *simple MPC* and *combined MPC DC-ARTDP* algorithms.

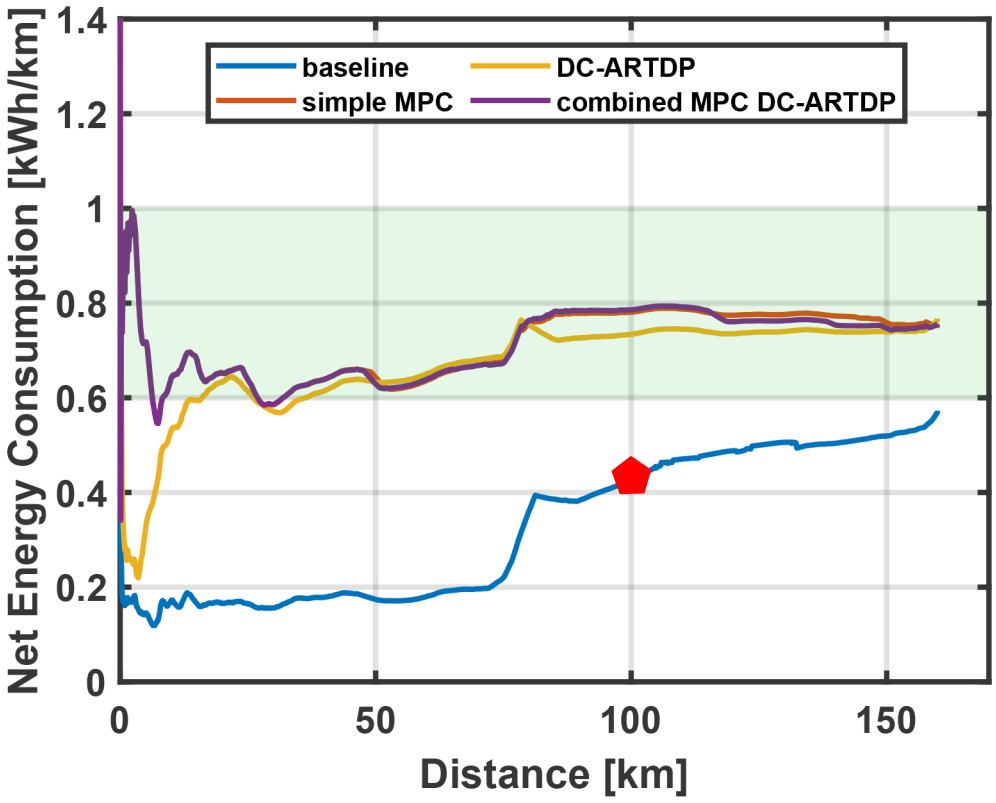


Figure 5.10: EC_{net} comparison for different Energy Management Control approaches during evaluation route simulation with MPC SOC penalty $p_2 = 1e8$.

For *simple MPC* the final ESS SOC only changes by $+0.01$, while for *combined MPC DC-ARTDP* the change is of $+0.06$. As for the Net Energy Consumption, the two algorithms get a similar value as *DC-ARTDP* of ~ 0.766 kWh/km. These results show that except *baseline*, other three algorithms provide similar outcomes.

In our evaluation of MPC based predictive energy management algorithms in contrast with our novel *DC-ARTDP* algorithm, we determined that the *DC-ARTDP* algorithm is capable of providing optimal energy management without prior knowledge of road grade variation. Based on the net energy consumption results presented in sub-

section 5.4.3, the *DC-ARTDP* algorithm can be observed to perform better or at-par with *simple MPC* and *combined MPC DC-ARTDP* algorithms. The *DC-ARTDP* algorithm performs a re-optimization if SOC_{ess} varies by more than 3% in comparison to the optimal SOC_{ess}. Thus, the *DC-ARTDP* algorithm is actually performing ESS SOC tracking with a $\pm 3\%$ deviation bound. Hence, any ESS SOC deviations occurring due to change in vehicle speed or due to unmeasured road grade disturbances, are accounted for by the re-optimization process.

In the "green" highlighted band from the net energy consumption results in Figures 5.8 and 5.10, a change of p_2 from 1e6 to 1e8 converges the net energy consumption values for *simple MPC* and *combined MPC DC-ARTDP* algorithms closer to *DC-ARTDP* outcome. Correspondingly, the ESS SOC for *simple MPC* and *combined MPC DC-ARTDP* algorithms can be observed to nearly overlap *DC-ARTDP* ESS SOC. Hence, by increasing the value of p_2 we are nearly neglecting the P_{gen} term in the cost function J per Eq.(5.7) for MPC problems. Therefore, the MPC optimization problem now becomes an ESS SOC tracking problem, similar to the *DC-ARTDP* algorithm approach.

For the EREV research vehicle evaluated in this study, the on-board generator system operated under power limits. This resulted in only the available engine peak power to coincide with its most fuel-efficient zone. Hence, operating the generator at its peak power by optimizing for ESS SOC tracking also resulted in net energy consumption reduction. This behavior is achievable by *simple MPC* and *combined MPC DC-ARTDP* only with a high p_2 value. This shows an adaptability limitation of the *simple MPC* and *combined MPC DC-ARTDP*, where for a variation in powertrain architecture or power deration fault scenarios, the MPC algorithms would have to update penalty values to perform close to global optimal outcomes.

Our evaluation of *combined MPC DC-ARTDP* algorithm, an approach using a mix of the MPC and DC-ARTDP algorithms, showed that the outcome is not significantly dif-

ferent than that of *simple MPC*. The expected benefit was from the reference SOC being updated upon its $\pm 3\%$ deviation from previous optimal SOC trajectory. This would have allowed for the p_2 penalty to be lower and make the algorithm adaptable to variations. Though a feasible benefit of this approach is with changes in route or final distance targets, as the *simple MPC* operates only on a pre-computed reference SOC trajectory.

5.5 Conclusion

A prior knowledge of driving route, vehicle speed and road elevation is beneficial in efficiently managing energy resources on hybrid electric vehicles. In this article, we focused on comparing a widely used predictive energy management approach based on Model Predictive Control algorithm with our novel Distance Constrained - Adaptive Real Time Dynamic Programming optimal energy management algorithm. A key variant between these two approaches was the prior knowledge or prior estimation of road grade changes. We also compared the behavior of these two algorithms to the CD-CS based energy management algorithm used in production EREVs. Our evaluation was conducted for an experimental research EREV developed on the 2016 Chevrolet Camaro platform. The simulation was conducted on a 160 km real-world driving route with considerable road grade variation.

The study shows that our novel *DC-ARTDP* algorithm is able to achieve the most efficient net energy consumption of 0.766 kWh/km while achieving the desired driving distance. The *DC-ARTDP* approach is found to improve net energy consumption in the range of $0-3\%$ with respect to *combined MPC DC-ARTDP* algorithm and an improvement in the range of $0-4.25\%$ with respect to *simple MPC* algorithm. It is worth noting that a higher penalty value of $p_2 = 1\text{e}8$ for ESS SOC tracking in the MPC cost function (J), is able to provide a similar behavior to our novel approach. Thus, we are able to show that our novel algorithm is able to outperform and at worst perform at-par to an algorithm that uses a *priori* road grade measurement for energy management. Hence,

road grade can be treated as an unmeasured disturbance to the energy management problem for EREVs with our novel algorithm.

Our future research goals will focus on real-time computation load studies of the DC-ARTDP algorithm. This novel algorithm uses Dynamic Programming at its core, which has a time complexity of $\mathcal{O}(mn)$. For a single state, m scales with number of feasible states at a given time step, while n scales with the driving distance. The time complexity can increase with longer travel distances and introduces an optimization challenge for real-time implementation in long distance applications such as for hybrid long haul semi-trucks.

Chapter 6

Energy Consumption Modeling and Optimization of Heterogeneous Powertrain Semi-Truck Platoon

Publication: *In preparation*

6.1 Abstract

Semi-truck fuel economy or energy consumption improvement is an ongoing effort to reduce operational costs and emissions. Alternative powertrain architectures such as hybrid electric and battery electric have been proposed as a feasible solution. A major hurdle with powertrain electrification in semi-trucks is significantly limited driving range. Among these propositions, hybrid electric semi-trucks prove as a viable option in the effort towards electrification. In this chapter, energy consumption and distance to empty behaviors of these alternative powertrain architectures are evaluated. Efforts to improve driving range of electrified powertrains is evaluated under platooning scenarios with heterogeneous powertrain architectures. Results show that Battery Electric Truck (BET) architecture's distance to empty is improved in the range of 12% to 22% with platooning in certain specific configurations. Operating Series Parallel Hybrid Electric Truck (SPHET) architecture with DC-ARTDP optimal energy management algorithm in

a platoon improved platoon efficiency by a maximum of $\approx 8\%$ points when in the lead-truck position. With this study an energy efficient and distance to empty maximizing platoon configuration for a heterogeneous platoon is determined.

6.2 Introduction

Long-haul semi-trucks travel an average of $\approx$ 100,990 km (62,752 mi) annually in the United States [1]. This is almost $4.5\times$ of the annual travel distance by a passenger car. These distances are covered while carrying heavy loads upto a maximum gross vehicle weight of $\approx$ 35,380 kg (78,000 lbs). Hence, the conventional semi-trucks consume significant amount of fuel to meet these requirements with an average fuel economy in the range of 37 - 42 L/100km (6.35 - 5.60 mi/gal) [2]. Improvement in semi-truck powertrain architectures are being made to reduce fuel consumption and associated costs. Recently, hybrid electric and battery electric semi-trucks have been proposed and launched with an expected exponential proliferation till 2050. A major hurdle with hybrid electric and battery electric semi-trucks at present is it's distance to empty (DTE) or available driving range. As of 2018, in-production hybrid electric and battery electric semi-trucks have shown a maximum DTE of $\approx$ 500 km (310 mi). Tests predict an expected DTE of $\approx$ 1000 km (620 mi) for upcoming semi-trucks [3]. Even then the test-based and expected DTE numbers are upto $3\times$ lower than that of a conventional semi-truck. Thus, introduction of hybrid electric and battery electric semi-trucks into a long-haul semi-truck fleet poses a limit to the maximum continuous driving range without refuelling or recharging.

In this article, we develop a model and evaluate the overall fuel economy or energy consumption improvement as well as distance to empty for a platoon of two and three semi-trucks with heterogeneous powertrain architectures. Here, the term *heterogeneous powertrain architecture platoon* implies that all the trucks in the platoon have a different powertrain among the following choices, a) conventional, b) hybrid electric, and c)

battery electric. These three different types of powertrain architectures introduce variations in vehicle longitudinal dynamics and total driving range. Thus, in case of a heterogeneous powertrain architecture platoon, the following feasible concerns exist,

1. The battery electric truck has the lowest individual range among the three variants, thus becoming a limiting factor for platoon's total driving range.

2. The response time for a conventional semi-truck is much higher than the hybrid electric and battery electric variants. This can either cause increased separation from the lead truck or collisions with the rear truck.

3. The potential to overcome extreme road grade variations or elevation changes varies with the semi-truck powertrain architecture.

The concept of vehicle platooning to reduce overall fuel consumption or energy consumption has been a topic of research as noted by early work of Zabat et al. on aerodynamic drag reduction in vehicle platoons. Their work showed reduction in average drag coefficient ($C_{d,avg}$) for the platoon with an increase in the size of platoon and reduction in the inter-vehicle spacing [80]. Since this work, impact of platooning on vehicle aerodynamic drag has been thoroughly studied both theoretically and experimentally. Salari et al. conducted two-truck and three-truck platooning wind-tunnel experiments with scale models to compute aerodynamic drag benefit as a function of inter-truck spacing. These results were computed for wind averaged speeds including yaw upto 6^o [56]. Results reported by Salari et al. showed similar trends to those suggested by Zabat et al. though magnitude variations were observed due to different vehicle types. At highway speeds, aerodynamic drag resistance can cause an added power demand of upto ≈ 64 kW. Hence, semi-truck aerodynamics are an important variable in overall fuel economy improvement. Platooning for heavy-duty vehicles has shown an average fuel economy improvement in the range of 3.5% - 7% for two-truck platoons and 5% - 13% for three-

truck platoons [47]. The maximum improvements were reported for shortest inter-truck separation of $\approx$ 4m. Lammert et al. also report fuel economy improvements in a similar range for a two-truck platoon i.e. 3.7% - 6.4% [42]. Intelligent heavy-duty truck platooning with Co-operative Adaptive Cruise Control (CACC) and Vehicle-to-Vehicle (V2V) communication are some of the more recent approaches in platooning. These methods have shown improvement in overall energy consumption reduction, platoon stability, and vehicle safety at close following distances [67, 25, 46, 77].

Impact of variation in powertrain architectures for a platoon hasn't been extensively studied in literature. A recent work considering a platoon of gasoline operated and electric semi-truck by He et al. provides eco-driving strategies for a such a platoon to improve overall energy consumption. The authors provide a strategy for both an automated truck as a leader as well as a human operated truck as a leader [30]. Other work in alternative powertrain platoons involves hybrid electric vehicle platoons both in passenger and commercial vehicles [79, 33]. Hence, a need to evaluate semi-truck platoons with different powertrain architectures i.e. conventional, hybrid electric, and battery electric is necessitated to further understand the benefits and impact on overall energy consumption as well as total driving range. Additionally, this work can provide some guidance for future implementations of such platoons on the road. Therefore, the goal of this study is to evaluate if platooning can assist in improving overall energy consumption of the platoon and also increase the DTE in comparison to the individual hybrid electric or battery electric truck. The study also determines the best configuration of semi-trucks in such a platoon with respect to overall energy consumption and DTE. The study performs model-in-loop simulation for two-truck and three-truck platoons over standard, custom and real-world driving cycles with road elevation variations.

Powertrain architectures for the three different semi-trucks, their modeling approach and verification is mentioned in Section 6.3. The results for overall energy consumption and DTE for this study are in Section 6.4, followed by concluding remarks in Section 6.5.

6.3 Methods

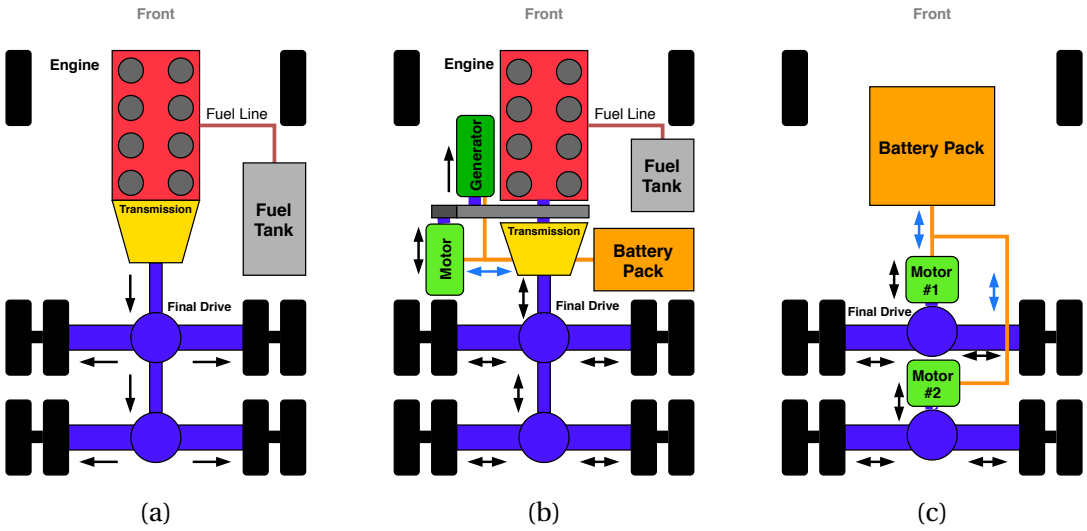


Figure 6.1: Semi-Truck powertrain architecture power flow diagram. (a) Conventional Truck, (b) Series-Parallel (P2-P2) Hybrid Electric Truck, and (c) Battery Electric Truck.

The heterogeneous powertrain architecture platoons are evaluated for energy consumption as well as distance to empty with a model-in-loop (MiL) framework. To achieve this a platoon power loss model environment was developed which can evaluate a two-truck and three-truck platoon scenario. Three different semi-truck powertrain architectures were modeled for this study. These include, a) a conventional diesel powered architecture (CT), b) a P2-P2 series-parallel hybrid electric architecture (SPHET), and c) a battery electric architecture (BET). Mechanical and electrical power flow for these architectures is shown in Figure 6.1 and the modeled component parameters are provided in Table 6.1. The black arrows represent mechanical power flow and blue arrows represent electrical power flow. The aerodynamic drag benefit experienced by the trucks in a platoon was modeled using a power function derived from data presented by Salari et al. [56].

Table 6.1: Power Loss Model Component Parameter Table

Powertrain	**Parameter**	**Value**	**Units**
CT, SPHET	Engine, peak power	360 @ 1800 RPM	kW
CT, SPHET	Engine, peak torque	2100 @ 1300 RPM	Nm
CT, SPHET	Transmission	10 Speed AMT	-
CT, SPHET	Transmission, gear ratios	1^{st} - 12.8:1, 10^{th} - 0.73:1	-
CT, SPHET	Final Drive ratio	2.64:1	-
SPHET, BET	Traction Motor, peak power	350	kW
SPHET, BET	Traction Motor, continuous power	245	kW
SPHET, BET	Traction Motor, peak torque	3400	Nm
SPHET, BET	Traction Motor, continuous torque	2380	Nm
SPHET	Generator Motor, peak power	173	kW
SPHET	Generator Motor, continuous power	63	kW
SPHET	Battery, maximum capacity	145	kWh
SPHET	Battery, nominal voltage	600	VDC
SPHET	Battery, available SOC range	20 - 80	%
SPHET	Gearbox, Carrier-to-Ring ratio	0.7:1	-
SPHET	Gearbox, Sun-to-Carrier ratio	3.35:1	-
SPHET	Gearbox, Motor-to-Ring ratio	1.6:1	-
SPHET	Gearbox, efficiency*	98	%
BET	Battery, maximum capacity	460	kWh
BET	Battery, nominal voltage	600	VDC
BET	Battery, available SOC range	10 - 100	%
BET	Final Drive ratio	5.2:1	-

* Estimated

The drag benefit determined by Salari et al. was adapted for an average drag coefficient ($C_{d,avg}$) of 0.52, used in this study. To simulate platooning, the semi-trucks in this study use forward looking perception system (such as a RADAR or a LiDAR) to determine relative distance of an obstacle for the implementation of Adaptive Cruise Control (ACC). Vehicle-to-Vehicle (V2V) communication wasn't considered in this study due to in-significant impact on semi-truck energy consumption values. The power loss models, energy management strategies and evaluation scenarios are described in the following sub-sections.

6.3.1 Heterogeneous Powertrain Platoon Modeling Environment

In this study, we are focusing on the energy consumption behavior of semi-trucks in a platoon formation which is predominantly affected by their longitudinal dynamics. Thus, the core component of this simulation environment are the power loss models for the different semi-truck powertrain architectures. The platoon modeling environment also integrates a Perception and Communication Layer (PCL), which models the forward looking perception systems and V2V communications for future use. Among the two-truck and three-truck platoons evaluated, the lead truck operates based on a speed target input, whereas the following trucks operate with Adaptive Cruise Control (ACC) enabled with a fixed following distance. The ACC settings used in this study are provided in sub-section 6.3.4. Finally, the aerodynamic drag benefit in platooning was modeled as power functions for the two different platooning scenarios and used in the simulation environment. The aerodynamic drag coefficient variation as a power function of separation distance is shown in Eqs.(6.1, 6.2), which can be visualized per Figure 6.2.

$$C_{d,\,2T} \;=\; -a\,\mathrm{x}_{12}^{-b} \;+ c \tag{6.1}$$

$$C_{d,\,3T} \;=\; -K_a(\mathrm{x}_{12})\,\mathrm{x}_{23}^{-K_b(\mathrm{x}_{12})} \;+ K_c(\mathrm{x}_{12}) \tag{6.2}$$

In Eq.(6.1), x_{12} is the inter-truck separation distance in meters, for a two-truck pla-

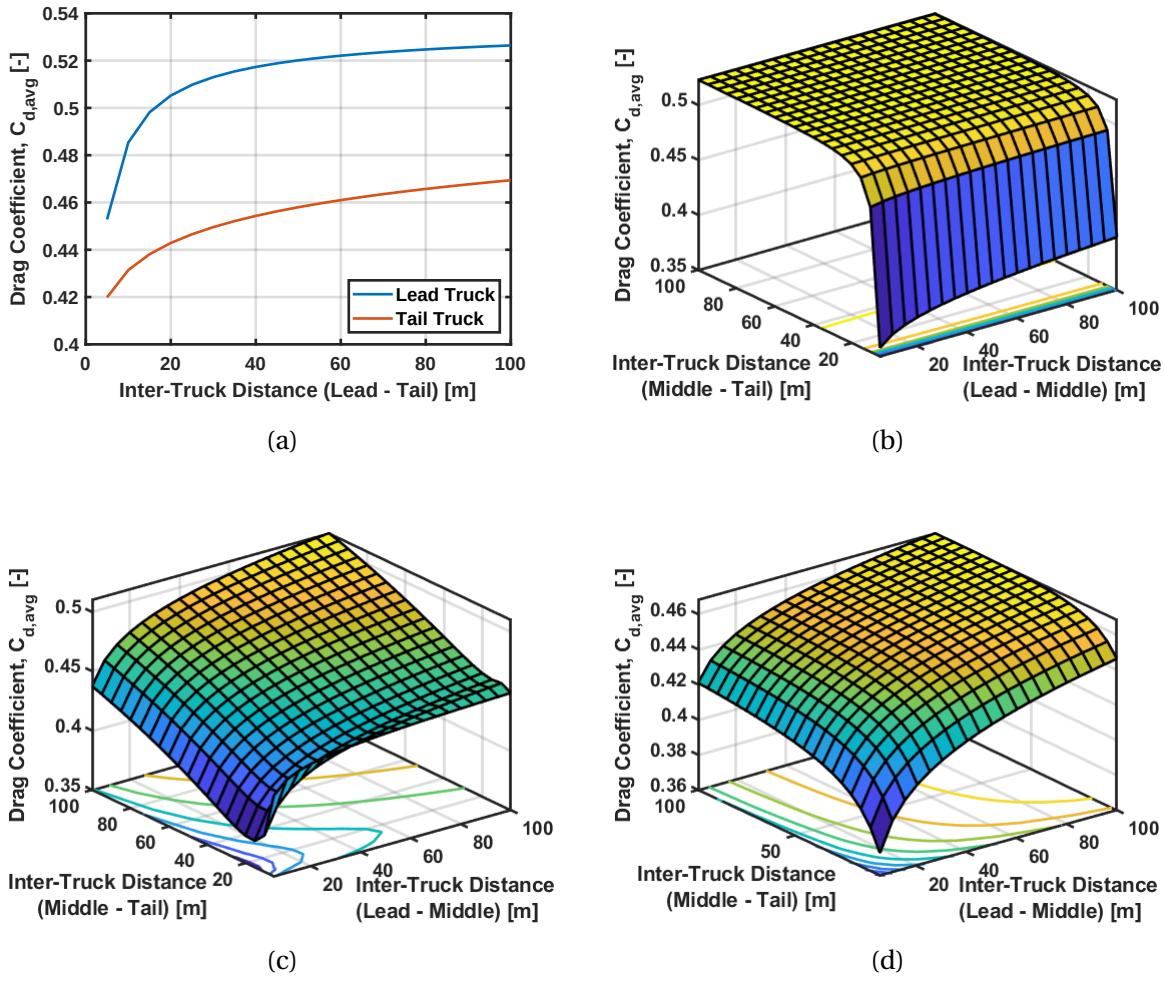


Figure 6.2: Aerodynamic drag coefficient variation as a function of Inter-Truck distance. (a) Two-Truck Platoon, (b) Lead Truck in a Three-Truck Platoon, (c) Middle Truck in a Three-Truck Platoon, and (d) Tail Truck in a Three-truck Platoon.

toon scenario, while a,b, and c are the fitting coefficients for the function. For a three-truck platoon as shown in Eq.(6.2), x_{23} is the inter-truck separation distance in meters for the middle and tail truck while x_{12} is the inter-truck separation distance for the lead and the middle truck. The fitting coefficients K_a, K_b, and K_c are a function of x_{12} and are represented by the same power function as in Eq.(6.1) but with different fitting coefficients for each of them. These relations are in accordance with drag reduction rate (DRR) variations as a function of change in normalized air-speed ($\frac{U_n}{U_\infty}$) in the wake of a platoon lead vehicle. Tadakuma et al. determined aerodynamic drag benefit prediction models with the same approach and showed $\frac{U_n}{U_\infty}$ varies as an inverse power of the distance between vehicles [66].

A common power loss model framework was implemented for all the powertrain architecture models. To emulate semi-truck's longitudinal dynamics, the desired propulsion force was computed as a first order response with vehicle speed as the input. For an effective comparison between the vehicles, the variation in dynamics arose only from the core power source i.e. engine in conventional truck, motors in battery electric truck and a combination of both in hybrid electric truck. A driver model emulated by a PI controller was integrated to generate acceleration and braking demands based on the vehicle speed error (v_{err}). Hence, the target vehicle speed delivered by the propulsion system was computed by solving for the roots of Eq.(6.3). The power demand was calculated based on Eq.(6.4) using the actual vehicle speed output from the solution of Eq.(6.3).

$$Av_{act}^2[t] \; + \; Bv_{act}[t] \; + \; C \; = \; 0 \tag{6.3}$$

where,

$$A \; = \; \frac{1}{2} \times \rho_{air} \times A_f \times C_d(\mathbf{x})$$

$$B \; = \; \frac{m_{veh}}{t_s} + \Big(9.17\text{e}-5 \times c_h \times m_{veh} \times g \times cos(\alpha_{road})\Big)$$

$$C \; = \; \Big(\frac{m_{veh}}{t_s} \times v_{act}[t-ts]\Big) \; - \; F_{prop} + \Big(m_{veh} \times g \times sin(\alpha_{road})\Big) - \Big(4.1\text{e}-3 \times m_{veh} \times g \times cos(\alpha_{road})\Big)$$

and

$$P_{dem} \;=\; F_{dem} \times v_{act}[t] \tag{6.4}$$

where,

$$F_{dem} \;=\; F_{prop} \;+\; F_{ar} \;+\; F_{rr} \;+\; F_{cr}$$

6.3.1.1 Conventional Truck Power Loss Model

In a conventional truck (CT) powertrain architecture, internal combustion engine is the prime mover and thus the only source of propulsive power. The power loss model takes the demand power (P_{dem}) as input and computes the required engine output power (P_{eng}), subject to drive line efficiency (η_{drv}) and auxiliary power demand (P_{aux}). The key output of this model is the fuel flow rate determined from the Brake Specific Fuel Consumption (BSFC) map, which is a function of the engine output torque (τ_{eng}) and engine output speed (ω_{eng}). Another factor determining the fuel flow rate is the commanded gear ratio (N_G). For any value of P_{dem}, more than one feasible pair of τ_{eng} and ω_{eng} exist, a correct gear command (G_{cmd}) allows to achieve the most fuel efficient operation. The engine and transmission data used for this model were obtained from EPA GEM v2.0 Simulation Model data library [68].

$$\dot{m}_{fuel}[t] \;=\; P_{eng}(\tau_{eng}, \omega_{eng})[t] \times BSFC(\tau_{eng}, \omega_{eng}) \times \frac{t_s}{3600} \tag{6.5}$$

where,

$$P_{eng}[t] \;=\; \frac{P_{dem}[t]}{\eta_{drv}} \;+\; P_{aux}[t]$$

$$\omega_{eng}[t] \;=\; N_G \times fd_{ratio} \times \frac{v_{act}[t]}{r_{tire}}$$

Equation(6.5) represents the functionality of conventional powertrain architecture model. Efficient engine operation was obtained by implementing an optimal gear selection algorithm which is discussed in sub-section 6.3.2.

6.3.1.2 Series Parallel Hybrid Electric Truck Power Loss Model

The series parallel hybrid electric truck (SPHET) powertrain architecture is modeled as a P2-P2 hybrid. The term P2-P2 hybrid implies that the generator motor and the propulsion motor are in a P2 configuration in the powertrain i.e. post-clutch and pre-transmission. The powertrain architecture is based on the Toyota's First Generator Hybrid Synergy Drive [58]. Figure 6.1 shows the power flow diagram for this powertrain architecture. This hybrid electric truck uses the same engine as in the conventional truck in addition to a 63 kW continuous power generator motor and a 245 kW continuous power propulsion motor. The on-board energy storage for electrical energy is a 600 VDC, 145 kWh $LiFePO_4$ battery pack configured based on A123 3.2V, 20Ah prismatic cells.

Being a series parallel hybrid electric vehicle, the power demand (P_{dem}) is fulfilled by either the propulsion motor, the engine or a combination of both. This choice is determined by the energy management strategy subject to various operational and optimization constraints. The engine power loss model is similar to the one described for conventional truck. The battery model used in this study uses similar cell resistance and capacitance parameters and model architecture as implemented for the EREV research vehicle by Kalia et al. [37]. The only variation is in the cell configuration to achieve the desired nominal voltage of 600 VDC and energy capacity. Finally, the electric propulsion motor and the generator motor model include their respective efficiency maps as a function of output torque and speed. The auxiliary power load is assumed to be entirely low-voltage load, and thus was added to the net electric power demand fulfilled by the on-board battery.

$$\dot{m}_{fuel}[t] \;=\; \Big(P_{eng}(\tau_{eng},\omega_{eng})[t] \;+\; P_{gen}(\tau_{gen},\omega_{gen})[t]\Big) \times BSFC(\tau_{eng},\omega_{eng}) \times \frac{t_s}{3600} \qquad (6.6)$$

where,

$$P_{eng}[t] \;=\; \frac{P_{dem}[t]}{\eta_{drv} \times \eta_{plnt}}$$

$$\omega_{eng}[t] \;=\; N_{CR} \times N_G \times fd_{ratio} \times \frac{v_{act}[t]}{r_{tire}}$$

$$\omega_{gen}[t] \;=\; \frac{\omega_{eng}[t]}{N_{CS}}$$

and

$$SOC[t] \;=\; SOC[t-1] \;-\; \frac{(P_{mot}(\tau_{mot}, \omega_{mot})[t] + P_{aux}[t] - P_{gen}(\tau_{gen}, \omega_{gen})[t]) \times t_s}{E_{batt} \times 3600} \tag{6.7}$$

where,

$$P_{mot}[t] \;=\; \frac{P_{dem}[t]}{\eta_{drv} \times \eta_{plnt} \times \eta_{mot}}$$

$$\omega_{mot}[t] \;=\; N_{MR} \times N_G \times fd_{ratio} \times \frac{v_{act}[t]}{r_{tire}}$$

Equations(6.6, 6.7) show the key operations of the model. The fuel flow rate $\dot{m}_{fuel}$ is determined based on the combined P_{eng} and P_{gen} demands. In addition to the fuel flow rate, for a hybrid electric truck the battery state of charge (SOC) was also observed. This term was computed at every time step per Eq.(6.7) and was influenced by the power terms P_{mot} and P_{gen}. As mentioned earlier, the hybrid architecture for the truck is that of a P2-P2 hybrid, additional gear ratio terms for the planetary gear system as shown in Figure 6.3 used are observed in the calculations. These include N_{CR}, the ratio between carrier and ring gear, N_{CS}, the ratio between carrier and sun gear, and N_{MR}, the ratio between motor pinion and ring gear.

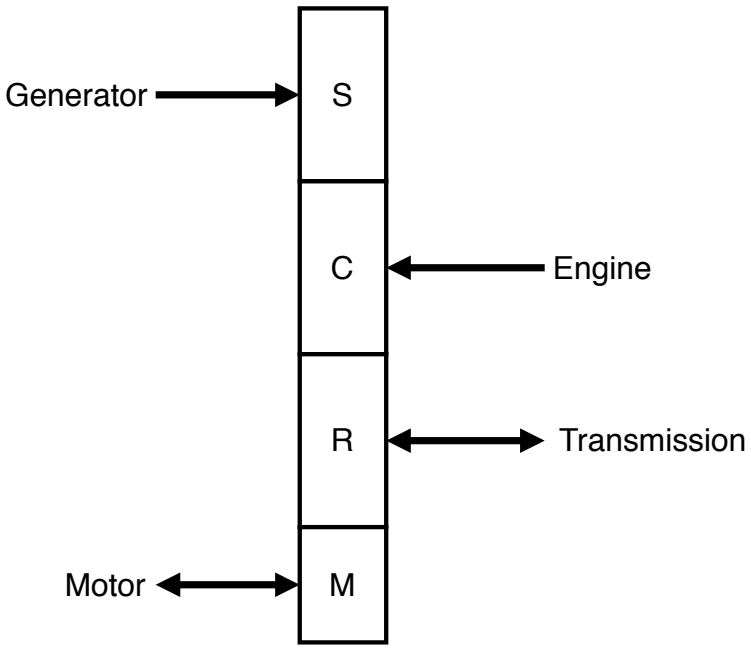


Figure 6.3: P2-P2 hybrid gear planetary gear system diagram modeled for Series Parallel Hybrid Electric Truck.

Battery operation was limited within the SOC range of 0.2 - 0.8 for the hybrid architecture. To maintain these constraints and utilize the various propulsive and energy recuperation systems on-board, the vehicle was operated in five different energy modes i.e. *EV Mode*, *Series Mode*, *Parallel Mode*, *Engine Only Mode*, and *Regenerative Braking Mode*. These modes were selected subject to certain operational constraints and can be determined through offline and online optimization methods.

6.3.1.3 Battery Electric Truck Power Loss Model

Finally, the battery electric truck (BET) powertrain architecture comprised of two 600 VDC, 245 kW continuous power electric motors on each of the rear axles of the tractor. Each motor transfers torque to the two wheels per axle through a single speed gear ratio of 5.2:1. The on-board electrical energy storage system is a 600 VDC, 460 kWh $LiFePO_4$

battery pack configured based on A123 3.2V, 20Ah prismatic cells. The power flow for this architecture is as straightforward as the CT architecture and shown in Figure 6.1.

$$SOC[t] \;=\; SOC[t-1] \;-\; \frac{(P_{mot}(\tau_{mot},\omega_{mot})[t] + P_{aux}[t]) \times t_s}{E_{batt} \times 3600} \tag{6.8}$$

where,

$$P_{mot}[t] \;=\; \frac{P_{dem}[t]}{\eta_{drv} \times \eta_{mot}}$$

$$\omega_{mot}[t] \;=\; N_{gbx} \times \frac{v_{act}[t]}{r_{tire}}$$

The P_{dem} input determines the motor power P_{mot}, which by extension was used to compute the change in battery SOC. In addition, to the propulsive power, auxiliary load power P_{aux} was considered to be all electric and was also fulfilled by the battery through a DC-DC converter. Here, the drive-line efficiency term (η_{drv}) corresponds to the single speed gear efficiency and (N_{gbx}) refers to the gear ratio of the single speed transmission. The battery model architecture and cell parameters were similar to that of series parallel hybrid truck. As the motors can operate only in two modes i.e. motoring, and generation, the energy management strategy for a battery electric truck is a simple rule-based approach determined by acceleration requests.

6.3.2 Energy Management Strategies

Energy management implementation varies with the powertrain architecture of the truck, despite of a similar goal. In case of a conventional truck, the intent is to reduce the amount of fuel consumed for a given distance with efficient operation of the engine. Though for a battery electric truck the goal is to reduce the net energy consumption by efficient usage of motor and energy recuperation with regenerative braking. In this study, we compare energy management among the different powertrain architecture on the basis of overall energy consumption. The energy management strategies used in this study and their implementation in the power loss model are further described.

6.3.2.1 Conventional Truck Energy Management

In a conventional truck the main approach to actively manage and control fuel economy is by performing optimal gear shifts. To achieve such a functionality an optimal gear shift is commanded based on the power demand (P_{dem}) and required vehicle speed (v_{act}). Thus, the cost function to determine optimal gear shift (G^*_{cmd}) at a given time step is given as,

$$G^*_{cmd}[t] \;=\; \arg\min_{G_{cmd}} \left(\dot{m}_{fuel}(P_{dem}[t], v_{act}[t])\Big|_{G_{cmd}=1}, \cdots, \dot{m}_{fuel}(P_{dem}[t], v_{act}[t])\Big|_{G_{cmd}=G_{max}} \right) \tag{6.9}$$

where, the fuel flow rate $\dot{m}_{fuel}$ is computed using Eq.(6.5) and associated relations. The cost function as shown in Eq.(6.9) finds the gear command (G_{cmd}) for which the fuel flow rate is the least at a given time t.

6.3.2.2 Series Parallel Hybrid Electric Truck Energy Management: Rule Based

Among the three different powertrain architectures being evaluated in this study, the series parallel hybrid electric truck architecture is the most complex. As mentioned earlier in the power loss model section, the SPHET can operate in five different energy modes i.e. *EV Mode, Series Mode, Parallel Mode, Engine Only Mode,* and *Regenerative Braking Mode*. These modes were determined per operational constraints on battery SOC and motor output power limit. A rule-based energy management approach as implemented by Zhao et al. for a medium-duty series parallel hybrid electric truck was implemented [85]. The Rule Based algorithm implemented in our study is provided as Algorithm 6.

In this Rule Based algorithm the different energy modes are referred to by the value of variable `MODE_FLAG`. During a drive, if battery SOC is greater than `lim_SOC` = 0.5 and less than `max_SOC` <= 0.8, the truck operates either in *EV Mode* (`MODE_FLAG` = 1) or in *Parallel Mode* (`MODE_FLAG` = 3). *Parallel Mode* is chosen if the P_{dem} is greater than $\texttt{max\_P}_{\text{mot}}$.

Algorithm 6: Series Parallel Hybrid Electric Truck Rule Based Energy Management Algorithm

Input: $P_{dem}[t], v_{act}[t], SOC[t], \text{max\_}SOC, \text{min\_}SOC, \text{lim\_}SOC, \text{max\_}P_{mot}[t], \text{avail\_}P_{gen}[t]$

Output: $P_{eng}[t], P_{gen}[t], P_{mot}[t], \text{MODE\_FLAG}$

if $v_{act}[t] == 0$ **then**
  MODE_FLAG $\longleftarrow 0$;
  $P_{eng}[t] \longleftarrow 0;\ P_{gen}[t] \longleftarrow 0;\ P_{mot}[t] \longleftarrow 0;$
endif

else if $0 < P_{dem}[t] \leq max\_P_{mot}[t]$ && $lim\_SOC \leq SOC[t] \leq max\_SOC$ **then**
  MODE_FLAG $\longleftarrow 1$;
  $P_{eng}[t] \longleftarrow 0;\ P_{gen}[t] \longleftarrow 0;\ P_{mot}[t] \longleftarrow P_{dem}[t];$
endif

else if $0 < P_{dem}[t] \leq max\_P_{mot}[t]$ && $min\_SOC \leq SOC[t] \leq lim\_SOC$ **then**
  MODE_FLAG $\longleftarrow 2$;
  $P_{eng}[t] \longleftarrow 0;\ P_{gen}[t] \longleftarrow \text{avail\_}P_{gen}[t];\ P_{mot}[t] \longleftarrow P_{dem}[t];$
endif

else if $P_{dem}[t] > max\_P_{mot}[t]$ && $lim\_SOC \leq SOC[t] \leq max\_SOC$ **then**
  MODE_FLAG $\longleftarrow 3$;
  $P_{eng}[t] \longleftarrow P_{dem}[t] - \text{max\_}P_{mot}[t];\ P_{gen}[t] \longleftarrow 0;\ P_{mot}[t] \longleftarrow \text{max\_}P_{mot}[t];$
endif

else if $P_{dem}[t] > max\_P_{mot}[t]$ && $SOC[t] \leq lim\_SOC$ || $SOC[t] < min\_SOC$ **then**
  MODE_FLAG $\longleftarrow 4$;
  $P_{eng} \longleftarrow P_{dem}[t];\ P_{gen} \longleftarrow 0;\ P_{mot} \longleftarrow 0;$
endif

else if $P_{dem}[t] < 0$ && $SOC[t] < max\_SOC$ **then**
  MODE_FLAG $\longleftarrow -1$;
  $P_{eng} \longleftarrow 0;\ P_{gen} \longleftarrow 0;\ P_{mot} \longleftarrow P_{dem}[t];$
endif

If battery SOC is less than `lim_SOC` but greater than `min_SOC` = 0.2 and P_{dem} is less than $\texttt{max\_P}_{\text{mot}}$, the truck operates in *Series Mode* (`MODE_FLAG` = 3). *Engine Only Mode* is activated if the battery SOC is less than `lim_SOC` and P_{dem} is high. These modes allow the truck to operate the engine as less as possible for shorter distances and also allows to meet range requirements by using engine for long distance driving.

6.3.2.3 Series Parallel Hybrid Electric Truck Energy Management: DC-ARTDP

The Distance Constrained Adaptive Real Time Dynamic Programming (DC-ARTDP) algorithm developed in context of a passenger Extended Range Electric Vehicle by Kalia et al. [37] was modified for application on this SPHET architecture. The optimization algorithm determines optimal operating points P^*_{gen} and P^*_{mot} for an impending travel based on Google Directions API. The algorithm recomputes optimal energy modes, if a deviation of greater than 3% was observed in the battery SOC from the optimal SOC trajectory. The modified DC-ARTDP algorithm for SPHET architecture is shown in Algorithm 7.

6.3.2.4 Battery Electric Truck Energy Management

Battery electric truck energy management ensures the electric propulsion motors run at their more efficient operation point. This was achieved by minimizing for power loss by the motor while achieving the desired P_{dem} and v_{act}. In this study, the chosen motor efficiency ranges between 93% - 96% and hence, for the ease of computation a constant efficiency at 95% was assumed. The energy management system also implemented a regenerative braking algorithm in parallel with normal braking request by ensuring the battery charging power limits were not violated.

Algorithm 7: Distance Constrained Adaptive Real Time Dynamic Programming for SPHET Architecture.

Input: $\bar{\mathbf{v}}_{veh}$, init_SOC, min_SOC, max_SOC, lim_SOC, init_V_{fuel}, t_s, max_d_{veh}, d_{rem}, max_P_{mot}

Output: $\mathbf{P}^*_{gen}, \mathbf{P}^*_{mot}$

$N \longleftarrow \text{length}(\bar{\mathbf{v}}_{veh})$; $\quad t \longleftarrow 0$; $\quad \mathbf{P}_{veh} \longleftarrow \mathsf{A}\bar{\mathbf{v}}^2_{veh} + \mathsf{B}\bar{\mathbf{v}}_{veh} + \mathsf{C}$;

while $t \leq N$ **do**

  if $(t == 0)$ **then**

    $\mathbf{SOC}^*_{ess}[t] \longleftarrow$ init_SOC_{ess}; $\quad d_{veh}[t] \longleftarrow 0$;

  end

  $d_{veh}[t+1] \longleftarrow d_{veh}[t] + \left(v_{veh}[t+1] \times \mathsf{t}_s\right)$

  if $P_{veh}[t] \leq$ *max_*P_{mot} && $\mathbf{SOC}^*_{ess}[t] >$ *lim_*SOC **then**

    $SOC_{ess}[t+1] \longleftarrow \mathbf{SOC}^*_{ess}[t] - \left(\frac{P_{dem}[t+1] + P_{gen}[t+1]}{E^{max}_{ess} \times 3600}\right), \forall P_{gen} \in \mathsf{U}_{P_{gen}}$

    $d_{DTE}[t+1] \longleftarrow \mathsf{EC}_{net}[t] \times d_{veh}[t]$

    $\arg\min_{P_{gen},\, SOC_{ess}} \mathsf{EC}_{net}[t+1] \longleftarrow$

    $\frac{1}{d_{veh}[t+1]}\Big[(\text{init\_}SOC_{ess} - SOC_{ess}[t+1])E^{max}_{ess} + (\eta^{-1}_{gen} P_{gen}[t+1] \frac{BSFC \times LHV}{3600})\Big],$

    $\forall SOC_{ess} : d_{DTE}[t+1] \geq \text{max\_}d_{veh} + d_{rem}$

    $\mathbf{SOC}^*_{ess}[t+1] \longleftarrow \mathbf{SOC}^*_{ess}[t] - \left(\frac{P_{dem}[t] + \mathbf{P}^*_{gen}[t]}{E^{max}_{ess} \cdot 3600}\right)$

  end

  else

    $SOC_{ess}[t+1] \longleftarrow \mathbf{SOC}^*_{ess}[t] - \left(\frac{P_{dem}[t+1]}{E^{max}_{ess} \times 3600}\right), \forall P_{mot} \in \mathsf{U}_{P_{mot}}$

    $d_{DTE}[t+1] \longleftarrow \mathsf{EC}_{net}[t] \times d_{veh}[t]$

    $\arg\min_{P_{mot},\, SOC_{ess}} \mathsf{EC}_{net}[t+1] \longleftarrow \frac{1}{d_{veh}[t+1]}\Big[(\text{init\_}SOC_{ess} - SOC_{ess}[t+1])E^{max}_{ess} + ((P_{dem} - P_{mot})[t+1] \frac{BSFC \times LHV}{3600})\Big],$

    $\forall SOC_{ess} : d_{DTE}[t+1] \geq \text{max\_}d_{veh} + d_{rem}$

    $\mathbf{SOC}^*_{ess}[t+1] \longleftarrow \mathbf{SOC}^*_{ess}[t] - \left(\frac{\mathbf{P}^*_{mot}[t]}{E^{max}_{ess} \cdot 3600}\right)$

  end

  $t \longleftarrow t + \mathsf{t}_s$;

end

6.3.3 Model Verification

Power loss models for the different semi-truck powertrain architectures evaluated in this study were verified for usability against published simulation and test data. Power loss models were verified for their energy consumption over the NREL Metro Highway Cycle $\times$ 2 as shown in Figure 6.4. This cycle was chosen as it closely emulates highway driving power requirements similar to the HHDDT (Highway Heavy Duty Diesel Truck) Cycle.

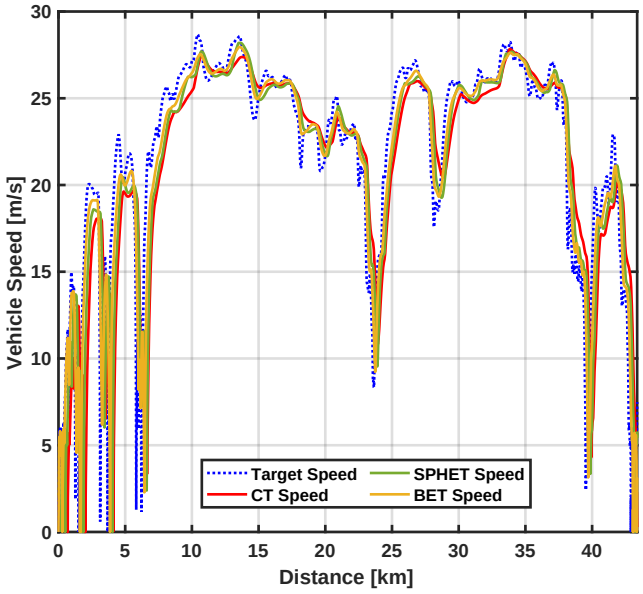


Figure 6.4: Vehicle speed response of different semi-truck powertrain architectures on NREL Metro Highway Cycle.

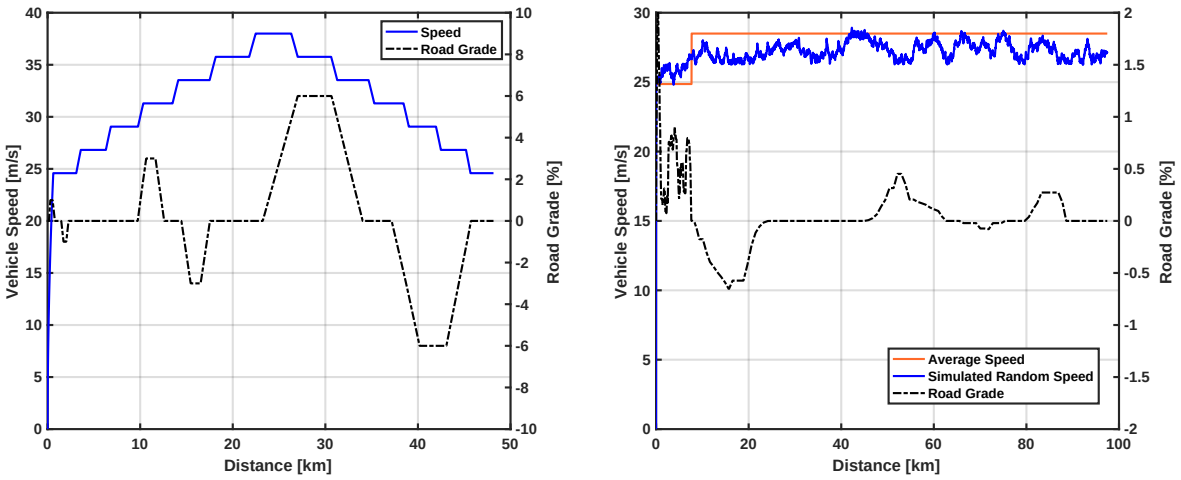
Based on the results of this verification simulation, we observed the overall fuel economy of the conventional truck was 38.13 L/100km or 3.38 Wh/m. This is about 2 $\times$ higher in comparison to the hybrid electric truck's overall energy consumption of 1.69 Wh/m. This relative difference is similar to the ones observed by Zhao et al. for their series parallel hybrid electric medium-duty vehicle for UDDS and HWFET Cycles [85]. Furthermore, the energy consumption for the conventional vehicle is close to the aver-

age fuel economy recorded for trucks in the US [2]. The results presented in this study are normalized against individual truck's fuel economy to focus on the relative benefit of platooning and remove the effects of modeled energy consumption deviation from real-world energy consumption.

6.3.4 Evaluation Scenario

To effectively understand the energy consumption behavior of the three different semi-truck powertrain architectures, individual trucks and all possible completely heterogeneous combinations in a two-truck (T2) and three-truck (T3) platoon configuration were simulated. As semi-truck platooning is logistically feasible and beneficial only on highways and at highway speeds, the individual trucks were first evaluated on a custom highway cycle termed *Highway Step* (HS). The HS Cycle varies between 24.6m/s - 38m/s (55mi/h - 85mi/h) and covers $\approx$ 42.9km (26.65mi). A progressive trapezoidal road grade variation of 1%, 3%, and 6% was also included in this drive cycle. Figure 6.5 shows the speed trace and road grade variation for the HS Cycle.

The T2 and T3 platoon configurations, where each truck in the platoon had a different powertrain architecture were the evaluated. These evaluations were conducted on a feasible real-world drive scenario for an actual semi-truck route corridor. The *Portland,OR to Hood River,OR Route* (PH Route), is a 97.5km (60.5mi) long drive from coordinates [45.5268, -122.6579] to [45.7120, -121.5166] along highway Interstate 84. The route has an average speed of 28m/s (63mi/h) a road grade variation between [-0.65%, 2%]. Figure 6.6 shows the map view of the simulated route and Figure 6.5(b) shows the simulated random speed trace and road grade variation.



(a) (b)

Figure 6.5: (a) Highway Step and (b) Portland - Hood River drive cycle speed trace and road grade used for simulations.

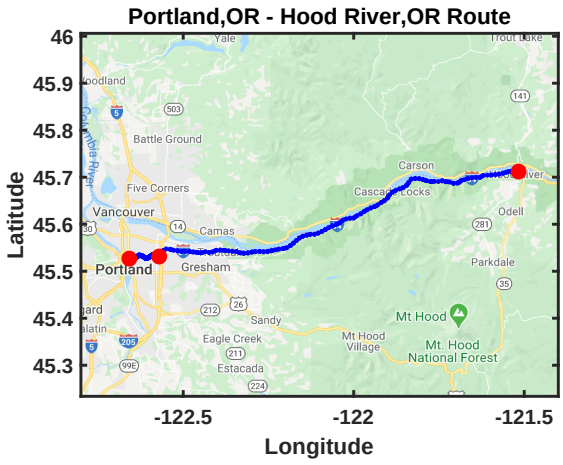


Figure 6.6: Google Map route trace of the Portland,OR to Hood River,OR Route Cycle.

Table 6.2: Semi-Truck evaluation parameters.

Parameter	**Value**	**Units**
Vehicle Weight	35380	kg
Frontal Area	10.4	m^2
Average Drag Coefficient	0.52	-
Tire Radius	0.489	m
Fuel Type	Diesel	-
Fuel Density	0.846	kg/L
Fuel LHV	11.83	kWh/kg
ACC, Set Speed	28	m/s
ACC, Cruise Start Time (from start)	100	s
ACC, Pre-Platoon Target Distance	60	m
ACC, Platoon Start Time (from start)	200	s
ACC, Platoon Target Distance	[15,20,30,40]	m

The simulation parameters used for the semi-trucks in this evaluation are listed for reference in Table 6.2. Apart from the fuel parameters, the rest are common among all the powertrain architectures being evaluated.

6.4 Results and Discussion

6.4.1 Individual Truck Evaluation

The three powertrain architectures, *conventional* (CT), *hybrid electric* (SPHET) and *battery electric* (BET) were individually evaluated on the HS Cycle and the PH Route Cycle. The trucks were simulated to match the desired speed trace. We first observe the speed response of the three vehicles as shown in Figure 6.7. The overall energy consumption

results obtained for the HS Cycle as shown in Table 6.3, confirms that BET architecture has the best overall energy consumption followed by SPHET and then CT. The distance to empty estimation values though show a higher driving range for the SPHET followed by CT and then BET as shown in Table 6.3.

Table 6.3: Energy Consumption and DTE Results for Individual Truck Simulations on HS Cycle.

Truck Type	**Energy Consumption** [Wh/m]	**Distance to Empty** [km]
CT	4.63	2207
SPHET	2.14	2376
BET	2.13	149.4

For the individual truck behavior and energy consumption on the PH Route Cycle, a similar trend as HS Cycle was observed. We also evaluated SPHET both with the Rule Based energy management strategy and the DC-ARTDP strategy. The DC-ARTDP strategy improved overall energy consumption by $\approx$ 7.65% for the SPHET architecture. In terms of DTE, BET has the lowest value as expected and CT the highest in this case. As for SPHET, the DTE improves when the DC-ARTDP strategy was implemented. These energy consumption and distance to empty values are shown in Table 6.4 for reference. Observed speed response as shown in Figure 6.9 was relatively similar as in case of HS cycle. Extreme variations were not observed as the road grade magnitude was not as high as HS Cycle. The CT can be observed to have a slower response to road grade changes as well as speed changes in comparison to SPHET and BET.

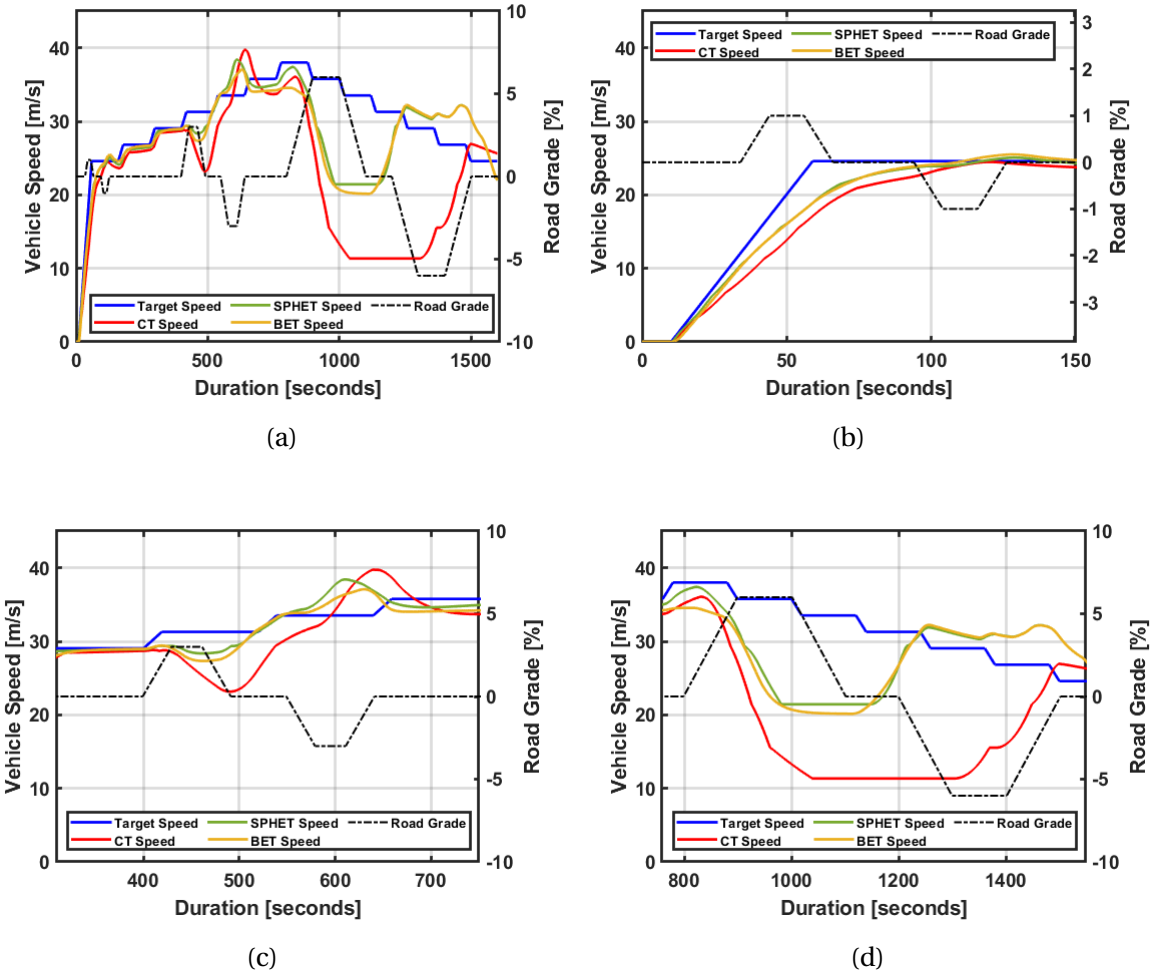


Figure 6.7: Semi-truck speed response on the custom Highway Step Cycle. (a) Overview, (b) Initial Acceleration, (c) Acceleration over ±3% road grade, and (d) Deceleration over ±6% road grade.

At high magnitude road grade changes, such as the 6% as shown in Figure 6.7(d), the CT architecture speed response can be observed to deviate significantly from the target speed. In comparison, SPHET and BET architectures do not show a deviation of similar magnitude or duration. The extreme deviation in case of CT can be attributed to the power limit at the most fuel efficient gear selection during the steep climb. In case of

SPHET and BET architectures, the relatively less deviation in speed is primarily due to electric motor's capability of producing high torque instantaneously.

Table 6.4: Energy Consumption and DTE Results for Individual Truck Simulations on PH Route Cycle.

Truck Type	**Energy Consumption** [Wh/m]	**Distance to Empty** [km]
CT	3.52	2943
SPHET, Rule Based	1.96	2553
SPHET, DC-ARTDP	1.81	2774
BET	1.92	133.2

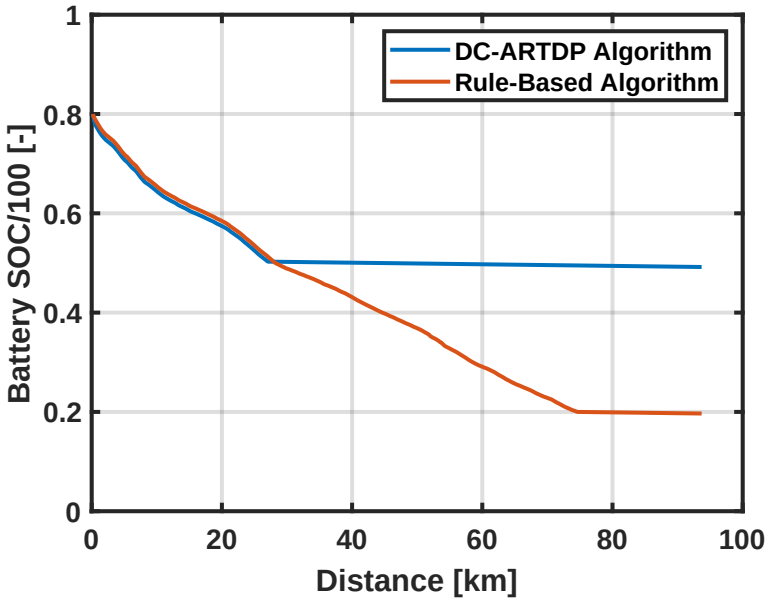


Figure 6.8: Battery SOC comparison for SPHET architecture over PH Route with DC-ARTDP and Rule Based Energy Management Algorithms.

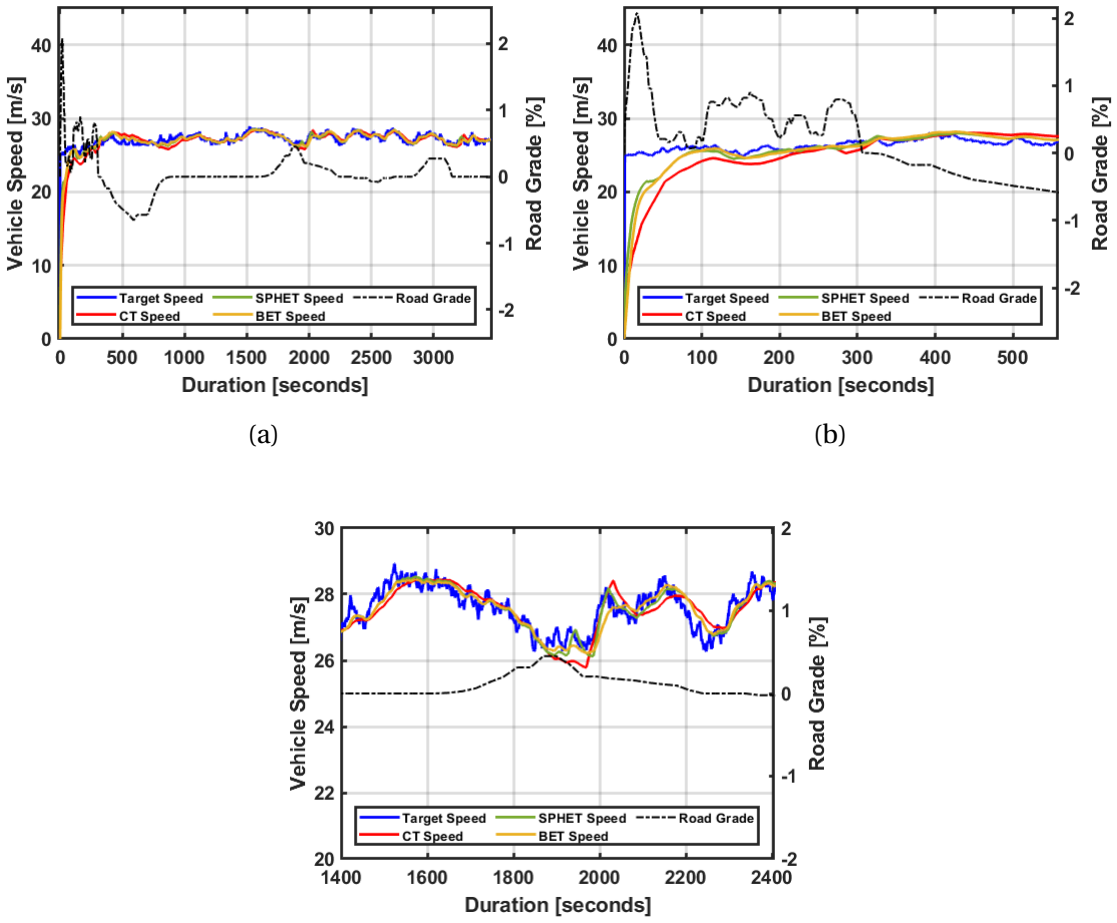


Figure 6.9: Semi-truck speed response on the real-world Portland,OR to Hood River,OR Route. (a) Overview, (b) Initial Acceleration, and (c) Acceleration over positive road grade.

Based on these observations made for individual truck behaviors on two different cycles, the following conclusions can be drawn about the truck behaviors and energy consumption trends,

1. Among the three powertrain architectures evaluated on the two routes, CT is consistently found to have a higher response time in comparison to SPHET and BET architectures. Among the SPHET and BET, the response time is almost close and depends significantly on the chosen prime mover. Though the similarity in response mostly exists because of the transmission gear command in case of SPHET not being varied as frequently as in case of CT.

2. Energy consumption increases significantly with vehicle speed and road grade as expected due to increased power demand. Though, as the power demand keeps rising, the energy consumption for CT increases significantly as observed in case of HS Cycle. The behavior of distance to empty though is highly dependant on road grade. A significant drop in road grade and increased frequency of such occurrences creates more chances for regenerative braking to recuperate energy. This impact can be observed for BET architecture with respect to DTE value by comparing responses of HS Cycle to PH Route Cycle.

3. SPHET architecture shows energy consumption values close to BET and distance to empty values close to CT. Energy management optimization further improves these trends for a SPHET.

Hence, based on this study both for a high power demanding cycle like HS Cycle and a relatively low power demanding cycle as PH Route Cycle, the SPHET architecture provides the best overall energy consumption and the distance to empty.

6.4.2 Two-Truck Platoon Evaluation

In the two-truck platoon evaluation, six feasible configurations were evaluated. The goal was to ensure that neither of the trucks have a similar powertrain architecture. The evaluated configurations are identified as Lead Truck - Tail Truck e.g. Conventional - Series Parallel Hybrid (CS) as shown in Table 6.5. These configurations were evaluated on the PH Route Cycle with varying target following distance settings. The variation in distance settings was evaluated because platooning benefit reduces with increase in inter-truck separation distance as evident from Salari et al. and Figure 6.2(a). The results shown in Figure 6.10 and Figure 6.11 are computed as relative improvements over individual truck performances on the same route. A positive value shows improvement and a negative value shows degradation. Each bar group in the plot corresponds to the configuration code mentioned in the X-Axis label.

Table 6.5: Two-Truck Platoon configurations evaluated on PH Route Cycle.

Configuration	**Identifier**
Conventional - Battery Electric	CB
Conventional - Series Parallel Hybrid	CS
Series Parallel Hybrid - Conventional	SC
Series Parallel Hybrid - Battery Electric	SB
Battery Electric - Conventional	BC
Battery Electric - Series Parallel Hybrid	BS

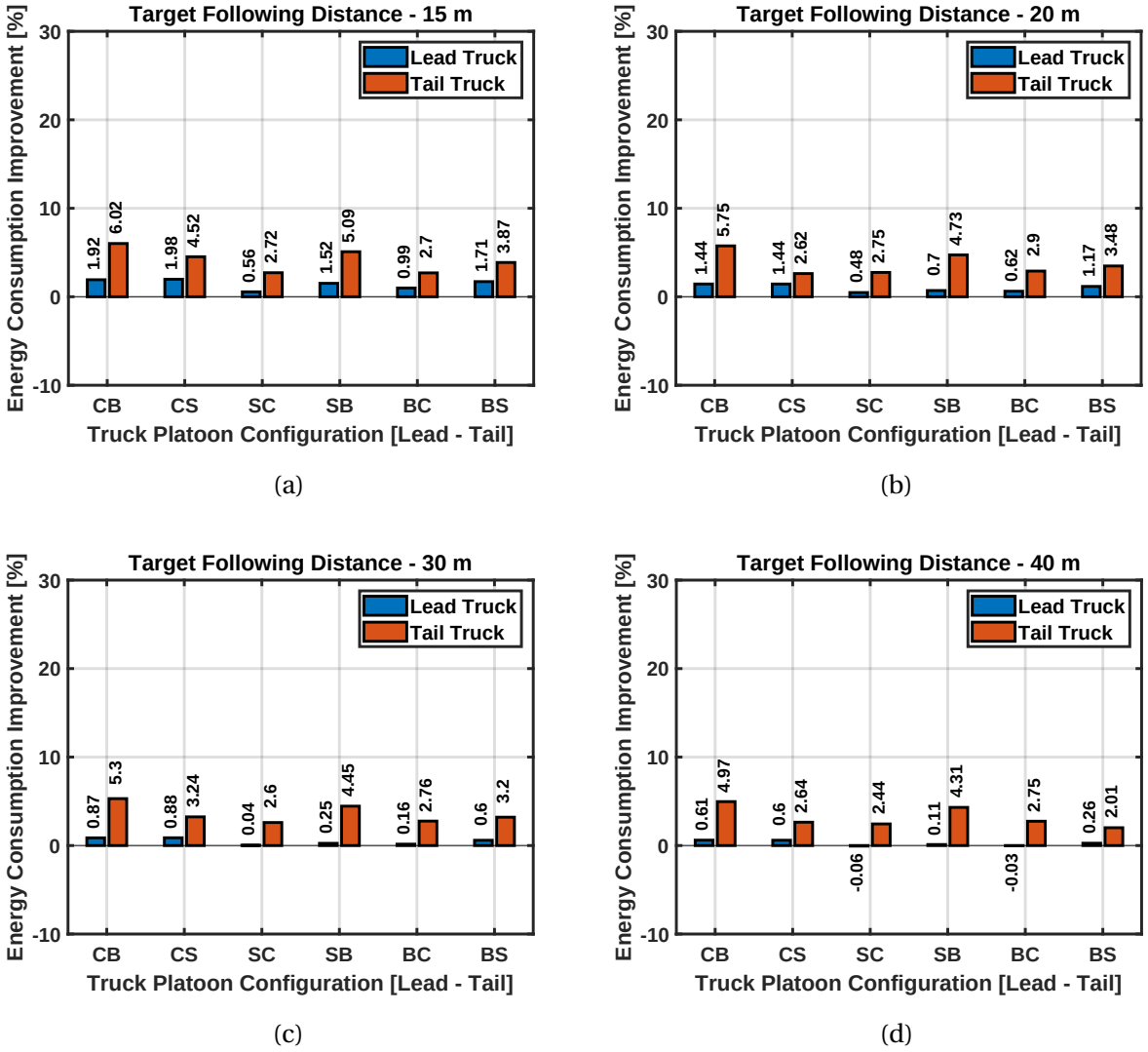


Figure 6.10: Energy Consumption improvement comparison between heterogeneous two truck platoon configurations simulated over PH Route for target separation distances of (a) 15m, (b) 20m, (c) 30m, and (d) 40m.

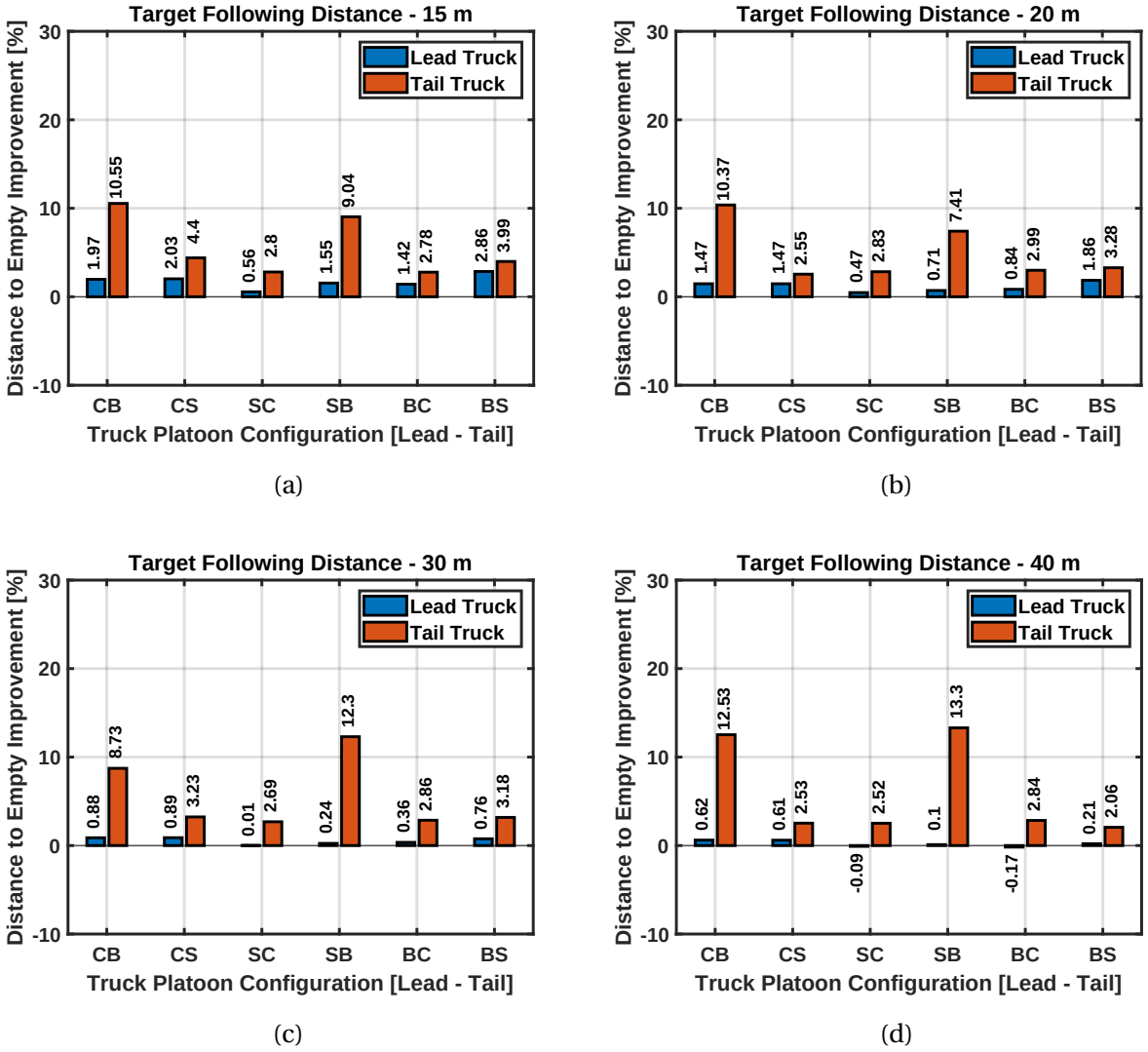


Figure 6.11: Distance to Empty improvement comparison between heterogeneous two truck platoon configurations simulated over PH Route for target separation distances of (a) 15m, (b) 20m, (c) 30m, and (d) 40m.

The energy consumption and distance to empty improvement results shown in Figure 6.10 and Figure 6.11 for the different target following distances are summarized as follows,

1. Evaluating the CB and CS configurations, where the lead truck has CT architecture, it is evident from the energy consumption improvement results as shown in Figure 6.10 that as inter-truck separation distance increases, the improvement for lead truck reduces. Among these two configurations, the energy consumption improvement for the lead truck is nearly the same irrespective of the target following distance. Thus, the reduction in energy consumption improvement can be solely attributed to reduction aerodynamic drag benefit.

2. For the same CB and CS configurations the BET architecture improvement reduces as inter-truck separation increases from 15m - 40m, a similar trend as observed with CT. Though, the SPHET architecture doesn't show a clear correlation in reduction similar to CT and BET. This trend can be attributed to two competing variables i.e. aerodynamic drag benefit and optimal gear command.

3. As for DTE values, the trends for lead truck follow the same as of energy consumption but for BET architecture as tail truck the improvement is higher at 40m. This is because the DTE is calculated based on the blended energy consumption value which takes into account short term average energy consumption. As for SPHET architecture, the DTE follows a similar trend as the it's energy consumption variation.

4. Now, for the SC and SB configurations, per expectations the key variable responsible for reduction in lead truck energy consumption improvement with increasing inter-truck separation distance is aerodynamic drag benefit change. Another observation is with reference to difference in energy consumption improvement be-

tween the two configurations for the lead truck. The relatively lower improvement in case of SC configuration can be attributed to the higher deviations by CT architecture truck from target distance in comparison to the BET architecture. Thus, leading to comparatively lower accumulated moments of aerodynamic drag benefit.

5. In the SC and SB configurations, with CT architecture as the tail truck, the energy consumption improvement can be observed to first rise and then fall due to the same two competing variables i.e. aerodynamic drag benefit and optimal gear command. In case of BET architecture, the reduction in energy consumption improvement is directly correlated with increase in inter-truck separation.

6. Finally, for the BC and BS configurations with BET architecture in the lead truck position, the lead truck energy consumption improvement shows the inverse correlation with inter-truck separation distance as expected. While for the tail trucks, CT architecture shows a similar behavior as in SC but SPHET architecture doesn't show a behavior similar to that in case of CS. In terms of DTE improvement, trends similar to energy consumption were observed.

On the basis of the observations listed above for the six different configurations, among possible permutations for CT and BET architecture, CB configuration shows a higher overall energy consumption improvement. Now for possible permutations for CT and SPHET architecture, CS shows the higher overall energy consumption improvement. Finally, for the possible permutations for BET and SPHET architecture, SB configuration shows the higher overall energy consumption. Hence, having CB, CS and SB configurations in two-truck platooning scenarios provides the best overall outcome.

6.4.3 Three-Truck Platoon Study

In case of three-truck platoon configuration evaluation, six feasible heterogeneous powertrain architecture configurations were evaluated. The evaluated configurations are identified as Lead Truck - Middle Truck - Tail Truck e.g. Conventional - Series Parallel Hybrid - Battery Electric (CSB) as shown in Table . These configurations were also evaluated on the PH Route Cycle with varying target following distance settings. The target following distance was simultaneously changed to the same value for both the middle and the tail truck. The results shown in Figure 6.12 and Figure 6.13 are computed as relative improvements over individual truck performances on the same route. A positive value shows improvement and a negative value shows degradation. Each bar group in the plot corresponds to the configuration code mentioned in the X-Axis label.

Table 6.6: Three-Truck Platoon configurations evaluated on PH Route Cycle.

Configuration	**Identifier**
Conventional - Series Parallel Hybrid - Battery Electric	CSB
Conventional - Battery Electric - Series Parallel Hybrid	CBS
Series Parallel Hybrid - Conventional - Battery Electric	SCB
Series Parallel Hybrid - Battery Electric - Conventional	SBC
Battery Electric - Conventional - Series Parallel Hybrid	BCS
Battery Electric - Series Parallel Hybrid - Conventional	BSC

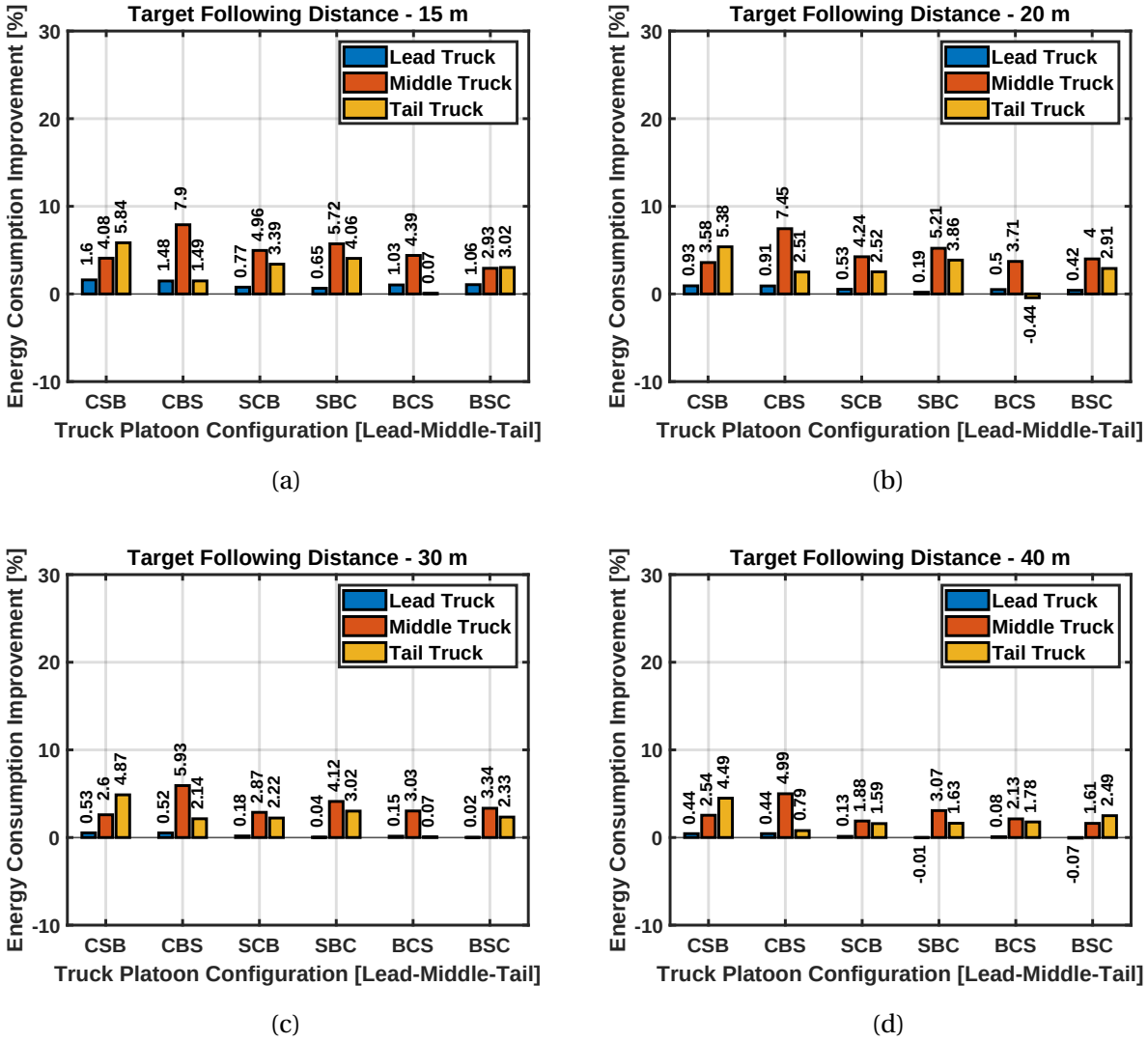


Figure 6.12: Energy Consumption improvement comparison between heterogeneous three truck platoon configurations simulated over PH Route for target separation distances of (a) 15m, (b) 20m, (c) 30m, and (d) 40m.

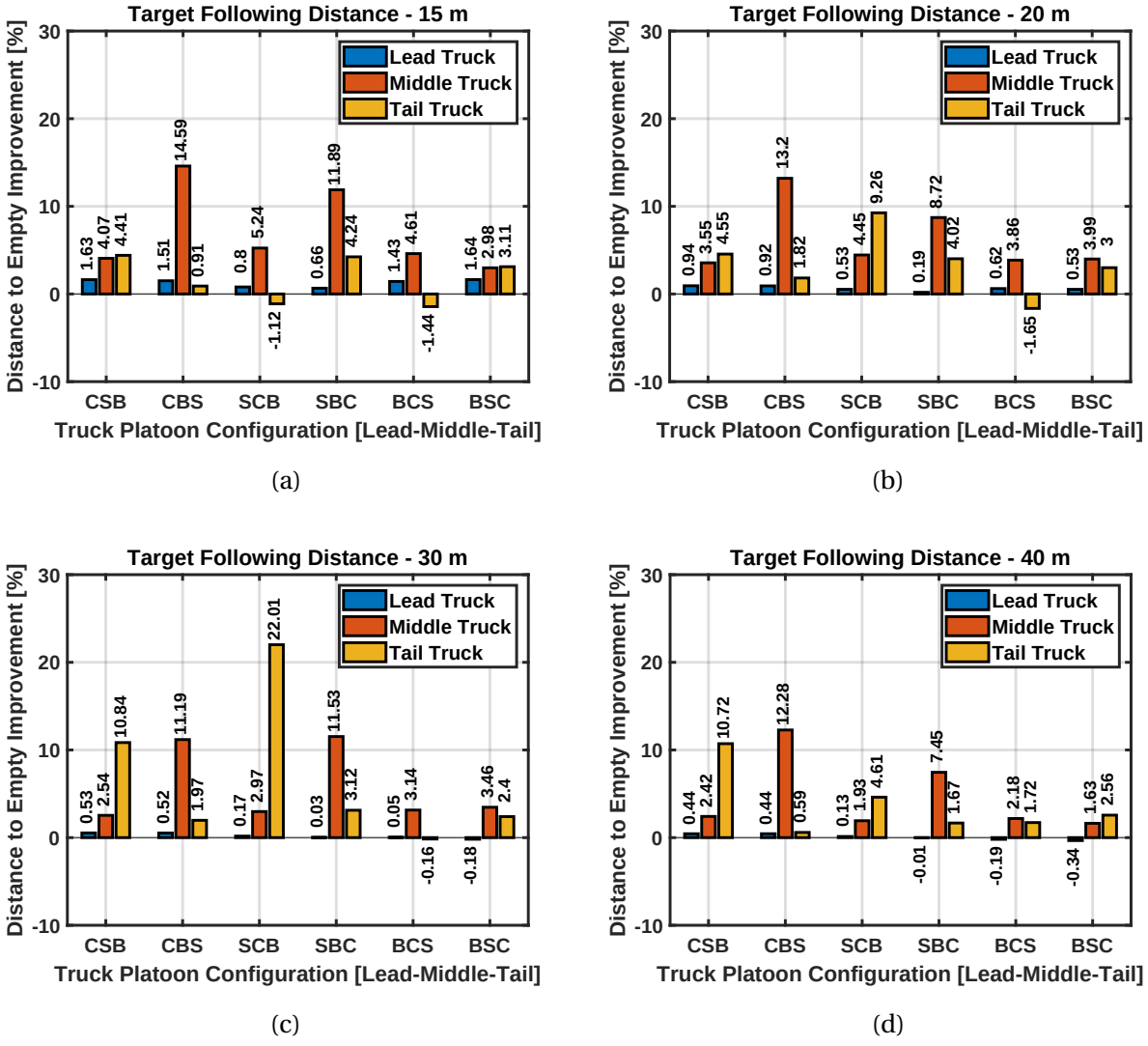


Figure 6.13: Distance to Empty improvement comparison between heterogeneous three truck platoon configurations simulated over PH Route for target separation distances of (a) 15m, (b) 20m, (c) 30m, and (d) 40m.

The energy consumption and distance to empty improvement results shown in Figure 6.12 and Figure 6.13 for the different target following distances are summarized as follows,

1. In the simulations conducted, all the configurations except BSC resulted in a vehicle crash at a target following distance at 15m. Among configurations CSB and CBS with CT architecture as the lead truck, the CT architecture energy consumption improvement is inversely proportional to inter-truck separation distance, as it was observed in case of two-truck platooning. Now among the mid truck and tail truck configurations i.e. SB and BS, the overall energy consumption improvement for SB is inversely proportional to the inter-truck distance. As for BS, the overall energy consumption improvement improves from 15m to 20m and then reduces towards 40m target following distance. This variation is specifically attributed to the SPHET architecture in the tail truck position and occurs due to the competing variables i.e. aerodynamic drag benefit and optimal gear demand.

2. For the same two configurations in terms of DTE improvement, for CT architecture in lead truck position the trend is similar to that of it's energy consumption. Among SB and BS sub-configurations, BET architecture benefits most from platooning in either of the positions. A consistent benefit is observed with BS configuration for BET architecture across all target following distance. In case of SB, the DTE is lower than BS for 15m and 20m but greater than BS from 30m and 40m.

3. In case of SCB and SBC configurations with the lead truck having an SPHET architecture, both the configurations follow a similar trend of inverse proportion to the inter-truck separation distance for all the architectures. Though among the sub-configurations CB and BC the combined overall energy consumption improvement is higher for BC than CB across all target following distances.

4. For DTE improvement of SCB and SBC configurations, the sub-configuration CB shows the expected trend for CT architecture but for BET architecture the variation is not highly correlated with inter-truck separation distance. This uncorrelated variation in BET architecture's distance to empty can be attributed to the extreme variations in inter-truck separation distance between lead truck and mid truck. As for the sub-configuration BC, the trend is inversely correlated as expected.

5. Among the BCS and BSC configurations with the lead truck having a BET architecture, lead truck energy consumption improvement trends are per expectations. The negative values for BET architecture in case of 40m target following distance is approximately zero. A variation of ± 0.5 in magnitude for the percent improvements being discussed is expected due to modeling errors. Energy consumption improvement trends for the SPHET architecture in both of the cases is uncorrelated as observed in other configurations. For the BCS architecture no significant improvement is observed for target following distances of 15m, 20m, 30m but for 40m there is some significant improvement observable. In case of BSC configuration, the SPHET architecture energy consumption improvement is observed to increase from 15m to 20m and then reduce as target following distance grows towards 40m. Trends similar to energy consumption improvement are observed for DTE improvements for these two configurations.

Therefore, based on these observations made for heterogeneous three-truck platoon evaluations, the CSB and CBS configurations show the largest combined improvement for all evaluated target distances. Among these two configurations the BET architecture experiences more improvement while being the middle truck in comparison to being the tail truck. Hence, this sub-configuration choice can be made based on the distance to empty target for the platoon.

6.4.4 Effect of SPHET Optimal Energy Management on Platoon Energy Consumption

Improvement in overall energy consumption for SPHET architecture was showed earlier with application of DC-ARTDP algorithm in Table 6.4. Based on that improvement the algorithm was implemented for two configurations each in the two-truck and three-truck platoon scenarios. These are CS and SC in case of two-truck platoon, CBS and SBC in case of three-truck platoon. These configurations were chosen to determine the impact of DC-ARTDP based on the SPHET position in the chosen configurations. The energy consumption improvement results for different target following distances of 15m, 20m, 30m, and 40m are shown in Figure 6.14 and Figure 6.15. A comparison of these results with those shown in Figure 6.10 and Figure 6.12 for the selected configurations are summarized as follows,

1. In the two-truck platoon scenario as a tail truck, the SPHET architecture shows an increase in energy consumption improvement in the range of 4.38 to 5.88 percent points. Whereas with SPHET as a lead truck, the improvement increase is in the range of 8.43 to 8.82 percent points.

2. In case of three-truck platoon scenario as a tail truck or a lead truck, the improvement increase is in the same range as observed in two-truck platoon scenario.

With the DC-ARTDP algorithm applied on SPHET architecture in a platoon configuration, the results showed that it further improved the platoon energy consumption. In case of two-truck platoon having the SPHET architecture in lead gives the most benefit till target following distance is less than or equal to 30m. In case of three-truck platoon the SPHET architecture in lead truck position provides the best overall improvement in comparison to tail truck. The middle truck position wasn't evaluated as the BET architecture was noted to benefit most at the middle position.

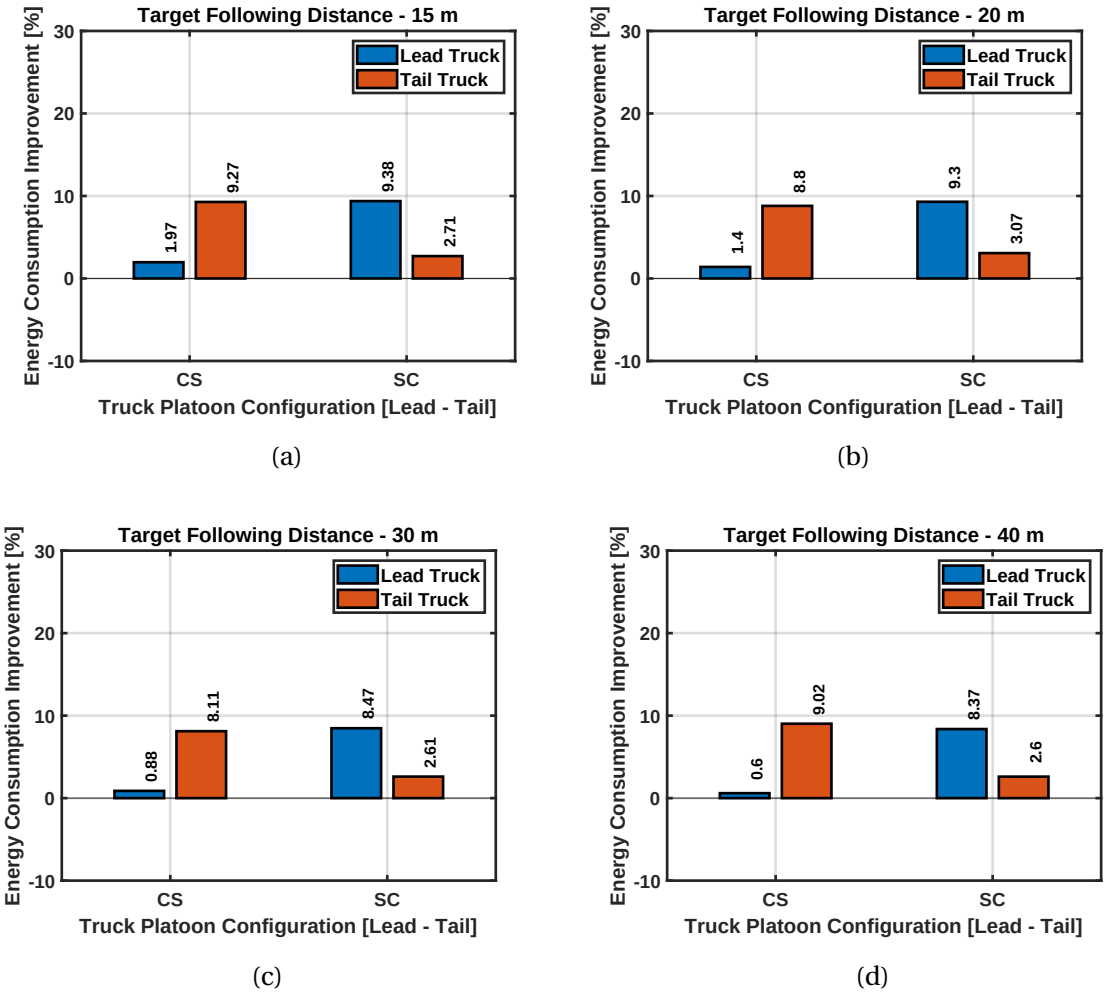


Figure 6.14: Energy consumption improvement comparison for two-truck platoon configurations CS and SC at (a) 15m, (b) 20m, (c) 30m and, (d) 40m target separation distance with DC-ARTDP optimal energy management.

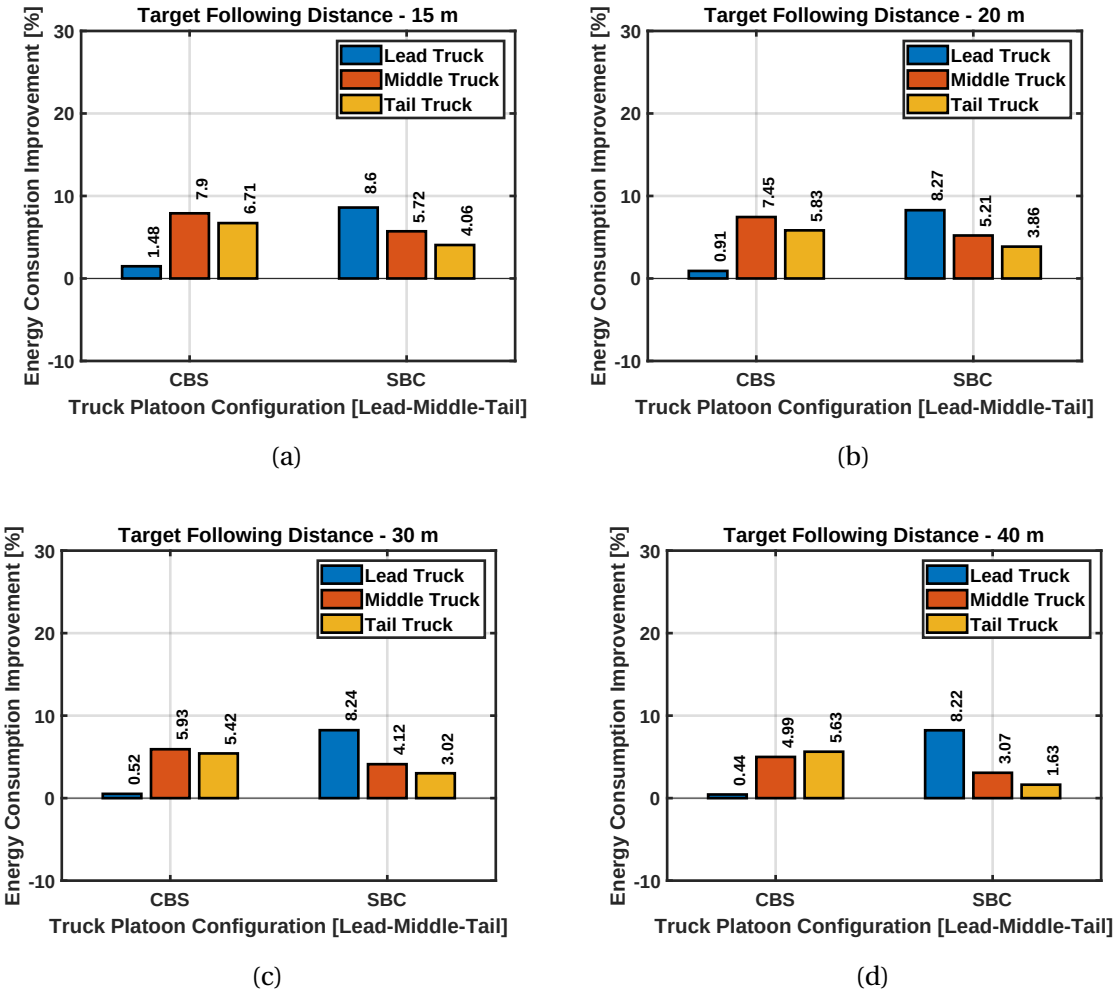


Figure 6.15: Energy consumption improvement comparison for two-truck platoon configurations CBS and SBC at (a) 15m, (b) 20m, (c) 30m and, (d) 40m target separation distance with DC-ARTDP optimal energy management.

6.5 Conclusion

Hybrid electric and Battery electric powertrain architectures are being introduced into the commercial vehicle industry to improve overall energy consumption and reduce operational costs. In this chapter, these alternative energy architectures were evaluated for their energy consumption and distance to empty performances over custom and real-world drive scenarios. The goal of this study was to understand these behaviors amongst different architectures and determine if platooning can help improve these behaviors further. We also aimed at determining if applying optimal control strategy like DC-ARTDP pushes the improvement even further.

Through this study we determined that distance to empty for BET architectures is a major concern, which is also supported by published research and industry evaluations. CT architectures as expected have the highest distance to empty values generally, owing to the energy dense fuel being used for propulsion. The SPHET architecture evaluated in this study is able to provide best compromise between overall energy consumption and distance to empty. Implementation of the DC-ARTDP algorithm adapted for the SPHET architecture further improved overall energy consumption of the SPHET architecture by $\approx 7.65\%$. The two-truck and three-truck platoon studies confirmed the initial premise of improving BET architecture distance to empty values. Evaluations showed the maximum DTE improvements are feasible only with CB and SB configurations with maximum feasible improvement of 12.53% and 13.3% at a target following distance of 40m. For the three-truck platooning scenario, the maximum improvement of 22.01% for SCB configuration was observed at a target following distance of 30m. In addition to this, application of DC-ARTDP optimal energy management algorithm to the SPHET architecture showed maximum benefit to the platoon, when in the lead truck position for most of the target following distances.

This research work and the semi-truck platooning simulation environment intro-

duces opportunities to further improve the overall energy consumption behavior by including vehicle-to-vehicle communication to share energy consumption information such as distance to empty and current energy levels. This information can assist in a combined energy plan for the platoon and change platoon configurations to maximize for distance to empty or minimize for overall energy consumption based on the desired goal.

Chapter 7

Conclusion

The doctoral research was motivated with the intent to develop a real-time implementable cost-effective energy management optimal control strategy for hybrid electric vehicles. As listed in the beginning, three key research goals were determined to guide this work to completion. These were,

1. Developing data driven power loss models for the experimental research vehicle to build the foundation for this research.

2. Formulate a novel energy management strategy that can be implemented real-time on a standard electronic control unit and provides the most efficient energy consumption while meeting the desired driving demand.

3. Evaluate the energy management optimization process for semi-trucks and evaluate it's combined benefit with heterogeneous powertrain platoons.

To achieve these goals, an initial model of the experimental research vehicle was formulated and evaluated for energy management optimization using the Multi-Objective Optimization methodology. This process provided a feasible and energy efficient on-board generator operating point, though the model used a lot of assumption which rendered the energy consumption predictions much lower in comparison to observations made in testing. The modeling approach was improved with a data-driven modeling

approach, that created a more accurate energy consumption representation of the experimental vehicle. This helped achieve the first goal of this research.

Using this model, the state-of-the-art energy management optimization algorithms were evaluated and compared. The outcomes allowed to develop a novel strategy termed as the Distance Constrained - Adaptive Real Time Dynamic Programming for the experimental research vehicle. The algorithm does not operate at every time step of the drive and is capable of being implemented on standard electronic control unit. The algorithm was thoroughly evaluated for it's adaptability to variations in driving requests, fault tolerance and disturbance rejection to changes in road grade. This completed our second research goal.

Finally, the hybrid electric commercial vehicle evaluation was conducted in platooning environment using a custom build platooning power loss model environment that supports individual truck simulations as well as two-truck and three-truck platoon simulations. This evaluation showed the potential of a series parallel hybrid electric truck over a conventional and and electric truck. The novel energy management algorithm was adapted for this new architecture and showed a maximum energy consumption improvement of $\approx 8\%$. The platooning scenarios revealed significant energy consumption and distance to empty improvements for battery electric trucks. Thus, the study showed the benefits of platooning for electrified commercial vehicles and additional improvement for the hybrid electric commercial vehicles. Hence, all of the research goals set out in the beginning were met.

Future extension of the presented work would focus on hardware implementation of the novel strategy and comparing MiL simulation outcomes with HiL simulation outcomes. Additionally, as the novel algorithm uses Dynamic Programming, longer duration routes increase the algorithm run-time. Temporal gradient variation can be applied to generate shorter time steps in the vicinity of current location of the vehicle which pro-

gressively increase to prevent excessive run-times and memory usage. This work also opens various avenues for future research in the domain of hybrid electric and electric vehicles, as well as energy management optimization. With intelligent transportation systems and vehicle autonomy research at rise, research areas expanding on vehicle path planning, leveraging vehicle-to-everything (V2X) communication, and improving vehicle platooning functionality are feasible future paths.

Bibliography

[1] Alternative Fuels Data Center: Maps and Data - Average Annual Vehicle Miles Traveled by Major Vehicle Category. https://afdc.energy.gov/data/10309. Accessed on 22 May 2020.

[2] The State of Fuel Economy in Trucking. https://www.geotab.com/truck-mpg-benchmark. Accessed on 22 May 2020.

[3] IEA (2019), Tracking Transport, IEA, Paris https://www.iea.org/reports/tracking-transport-2019. Accessed on 22 May 2020.

[4] NREL DriveCAT - Chassis Dynamometer Drive Cycles. (2019). National Renewable Energy Laboratory. www.nrel.gov/transportation/drive-cycle-tool. Accessed on 26 March 2020.

[5] Global ev outlook 2019 - analysis and key findings. a report by the international energy agency. Accessed on 06 December 2019.

[6] Ice tender. Accessed on 18 January 2020.

[7] U.s. hev sales by model. Accessed on 01 January 2017.

[8] Haroune Aouzellag, Kaci Ghedamsi, and Djamel Aouzellag. Energy management and fault tolerant control strategies for fuel cell/ultra-capacitor hybrid electric vehicles to enhance autonomy, efficiency and life time of the fuel cell system. *International Journal of Hydrogen Energy*, 40(22):7204–7213, Jun 2015.

[9] 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 *DSCC2010*, page 507–514, Sep 2010.

[10] Riccardo Biasini, Simona Onori, and Giorgio Rizzoni. A near-optimal rule-based energy management strategy for medium duty hybrid truck. *International Journal of Powertrains*, 2(2/3):232, 2013.

[11] Mogens Blanke, W. Christian Frei, Franta Kraus, J. Ron Patton, and Marcel Staroswiecki. What is fault-tolerant control? *4th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes 2000 (SAFEPROCESS 2000), Budapest, Hungary, 14-16 June 2000*, 33:41–52, Jun 2000.

[12] H. Ali Borhan, Ardalan Vahidi, Anthony M. Phillips, Ming L. Kuang, and Ilya V. Kolmanovsky. Predictive energy management of a power-split hybrid electric vehicle. In *2009 American Control Conference*, page 3970–3976, Jun 2009.

[13] Hoseinali Borhan, Ardalan Vahidi, Anthony M. Phillips, Ming L. Kuang, Ilya V. Kolmanovsky, and Stefano Di Cairano. Mpc-based energy management of a power-split hybrid electric vehicle. *IEEE Transactions on Control Systems Technology*, 20(3):593–603, May 2012.

[14] Christopher Borroni-Bird and Mohsen Shabana. Chevrolet sequel: Reinventing the automobile. *SAE International Journal of Engines*, 1(1):304–313, Apr 2008.

[15] K. R. Bouwman, T. H. Pham, S. Wilkins, and T. Hofman. Predictive energy management strategy including traffic flow data for hybrid electric vehicles. *IFAC-PapersOnLine*, 50(1):10046–10051, Jul 2017.

[16] A. Brahma, Y. Guezennec, and G. Rizzoni. Optimal energy management in series hybrid electric vehicles. In *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334)*, volume 1, page 60–64 vol.1, Jun 2000.

[17] P. Capaldi, A. Dannier, and I. Spina. A simple on-board electric generator for road electric vehicles based on single-cylinder engine and pm-brushless generator. In *Automation and Motion International Symposium on Power Electronics Power Electronics, Electrical Drives*, page 374–378, Jun 2012.

[18] 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, May 2015.

[19] 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.

[20] Mihael Cipek, Mirko Čorić, Branimir Škugor, Josip Kasać, and Joško Deur. Dynamic programming-based optimization of control variables of an extended range electric vehicle. SAE International, 2013.

[21] Scott Curran, Vitaly Prikhodko, Kukwon Cho, C. Scott Sluder, James Parks, Robert

Wagner, Sage Kokjohn, and Rolf D. Reitz. In-cylinder fuel blending of gasoline/diesel for improved efficiency and lowest possible emissions on a multi-cylinder light-duty diesel engine. Oct 2010.

[22] Ronald M. Dell, Patrick T. Moseley, and David A. J. Rand. *Chapter 5 - Progressive Electrification of Road Vehicles*, page 157–192. Academic Press, 2014.

[23] H. Christopher Frey, Alper Unal, Nagui M. Rouphail, and James D. Colyar. On-road measurement of vehicle tailpipe emissions using a portable instrument. *Journal of the Air & Waste Management Association*, 53(8):992–1002, Aug 2003.

[24] Jörg Gissing, Thomas Lichius, Sidney Baltzer, David Hemkemeyer, and Lutz Eckstein. Predictive energy management of range-extended electric vehicles considering cabin heat demand and acoustics. *IFAC-PapersOnLine*, 48(15):209–216, Jan 2015.

[25] Ge Guo and Qiong Wang. Fuel-efficient en route speed planning and tracking control of truck platoons. *IEEE Transactions on Intelligent Transportation Systems*, 20(8):3091–3103, Aug 2019.

[26] M. H. Hajimiri and F. R. 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*, page 1–5, Dec 2006.

[27] Jukka Halme and Jussi Suomela. Optimal efficiency based gen-set control for series hybrid work machine. In *2012 IEEE Vehicle Power and Propulsion Conference*, page 836–839, Oct 2012.

[28] Hongwen He, Jinquan Guo, and Chao Sun. Road grade prediction for predictive energy management in hybrid electric vehicles. *Energy Procedia*, 105:2438–2444, May 2017.

[29] Hongwen He, Rui Xiong, and Jinxin Fan. Evaluation of lithium-ion battery equivalent circuit models for state of charge estimation by an experimental approach. *Energies*, 4(4):582–598, Apr 2011.

[30] Xiaozheng He and Xinkai Wu. Eco-driving advisory strategies for a platoon of mixed gasoline and electric vehicles in a connected vehicle system. *Transportation Research Part D: Transport and Environment*, 63:907–922, Aug 2018.

[31] V. I. Herrera, A. Saez-de Ibarra, A. Milo, H. Gaztañaga, and H. Camblong. Optimal energy management of a hybrid electric bus with a battery-supercapacitor storage system using genetic algorithm. In *2015 International Conference on Electrical Sys-*

tems for Aircraft, Railway, Ship Propulsion and Road Vehicles (ESARS), page 1–6, Mar 2015.

[32] N. Jalil, N.A. Kheir, and M. Salman. A rule-based energy management strategy for a series hybrid vehicle. In *Proceedings of the 1997 American Control Conference (Cat. No.97CH36041)*, volume 1, page 689–693 vol.1, Jun 1997.

[33] Lars Johannesson, Nikolce Murgovski, Erik Jonasson, Jonas Hellgren, and Bo Egardt. Predictive energy management of hybrid long-haul trucks. *Control Engineering Practice*, 41:83–97, Aug 2015.

[34] Rajit Johri and Zoran Filipi. Optimal energy management of a series hybrid vehicle with combined fuel economy and low-emission objectives. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 228(12):1424–1439, Jun 2014.

[35] Aman V. Kalia and Brian C. Fabien. Development of optimal control strategy for a plug-in series hybrid electric vehicle with an on-board engine-generator system for overall fuel economy improvement and reduction in tail-pipe emissions. American Society of Mechanical Engineers Digital Collection, Nov 2017.

[36] Aman V. Kalia and Brian C. Fabien. Development of optimal control strategy for a plug-in series hybrid electric vehicle with an on-board engine-generator system for overall fuel economy improvement and reduction in tail-pipe emissions. American Society of Mechanical Engineers Digital Collection, Nov 2017.

[37] Aman V. Kalia and Brian C. Fabien. On implementing optimal energy management for erev using distance constrained adaptive real-time dynamic programming. *Electronics*, 9(2):228, Feb 2020.

[38] Elkhatib Kamal and Lounis Adouane. Robust energy management strategy based on the battery fault management for hydraulic-electric hybrid vehicle. In *ICINCO*, 2017.

[39] T. Kaneko, A. Nomura, W. Yang, Y. Daisho, Y. Kamiya, N. Sawada, M. Yasukawa, F. Takekoshi, and R. Tsushima. Optimization of engine control methods for range extender-type plug-in hybrid vehicles. In *2013 World Electric Vehicle Symposium and Exhibition (EVS27)*, page 1–8, Nov 2013.

[40] Florian Knorn. M-code latex package, 2020. (https://www.mathworks.com/matlabcentral/fileexchange/8015-m-code-latex-package), MATLAB Central File Exchange. Retrieved June 12, 2020.

[41] Mustafa Koç, Yakup Sekmen, Tolga Topgül, and Hüseyin Serdar Yücesu. The effects

of ethanol–unleaded gasoline blends on engine performance and exhaust emissions in a spark-ignition engine. *Renewable Energy*, 34(10):2101–2106, Oct 2009.

[42] Michael P. Lammert, Adam Duran, Jeremy Diez, Kevin Burton, and Alex Nicholson. Effect of platooning on fuel consumption of class 8 vehicles over a range of speeds, following distances, and mass. *SAE International Journal of Commercial Vehicles*, 7(22014-01–2438):626–639, Sep 2014.

[43] Thomas Leroy, Jérémy Malaizé, and Gilles Corde. Towards real-time optimal energy management of hev powertrains using stochastic dynamic programming. In *2012 IEEE Vehicle Power and Propulsion Conference*, page 383–388, Oct 2012.

[44] Chan-Chiao Lin, Huei Peng, and J.W. Grizzle. A stochastic control strategy for hybrid electric vehicles. In *Proceedings of the 2004 American Control Conference*, volume 5, page 4710–4715 vol.5, Jun 2004.

[45] Xue Lin, Yanzhi Wang, Paul Bogdan, Naehyuck Chang, and Massoud Pedram. Optimizing fuel economy of hybrid electric vehicles using a markov decision process model. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, page 718–723, Jun 2015.

[46] Fangwu Ma, Yu Yang, Jiawei Wang, Zhenze Liu, Jinhang Li, Jiahong Nie, Yucheng Shen, and Liang Wu. Predictive energy-saving optimization based on nonlinear model predictive control for cooperative connected vehicles platoon with v2v communication. *Energy*, 189:116120, Dec 2019.

[47] Brian McAuliffe, Michael Lammert, Xiao-Yun Lu, Steven Shladover, Marius-Dorin Surcel, and Aravind Kailas. *Influences on Energy Savings of Heavy Trucks Using Cooperative Adaptive Cruise Control*. Number 2018-01–1181. Apr 2018.

[48] Amor A. Menezes and Ilya V. Kolmanovsky. Energy and power management in a series hybrid electric vehicle using selective evolutionary generation. In *53rd IEEE Conference on Decision and Control*, page 3310–3315, Dec 2014.

[49] Richard T. Meyer, Scott C. Johnson, Raymond A. DeCarlo, Steve Pekarek, and Scott D. Sudhoff. Hybrid electric vehicle fault tolerant control. *Journal of Dynamic Systems, Measurement, and Control*, 140(2), Feb 2018.

[50] Scott Jason Moura, Hosam K. Fathy, Duncan S. Callaway, and Jeffrey L. Stein. A stochastic optimal control approach for power management in plug-in hybrid electric vehicles. *IEEE Transactions on Control Systems Technology*, 19(3):545–555, May 2011.

[51] Simona Onori, Lorenzo Serrao, and Giorgio Rizzoni. *Hybrid Electric Vehicles: En-*

ergy Management Strategies. SpringerBriefs in Control, Automation and Robotics. Springer-Verlag, 2016.

[52] Raouia Oubellil and Moussa Boukhnifer. Passive fault tolerant control design of energy management system for electric vehicle. In *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, page 1402–1408, Jun 2014.

[53] Zheng Pan and Chunyue Song. A predictive energy management strategy for hybrid electric bus based on greedy algorithm. In *Proceedings of the 10th World Congress on Intelligent Control and Automation*, page 2782–2787, Jul 2012.

[54] 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):244–254, Nov 2006.

[55] Christoph Romaus, Kai Gathmann, and Joachim Böcker. Optimal energy management for a hybrid energy storage system for electric vehicles based on stochastic dynamic programming. In *2010 IEEE Vehicle Power and Propulsion Conference*, page 1–6, Sep 2010.

[56] Kambiz Salari and Jason Ortega. Experimental investigation of the aerodynamic benefits of truck platooning. Apr 2018.

[57] Balaji Sampathnarayanan, Lorenzo Serrao, Simona Onori, Giorgio Rizzoni, and Steve Yurkovich. Model predictive control as an energy management strategy for hybrid electric vehicles. page 249–256. American Society of Mechanical Engineers Digital Collection, Sep 2010.

[58] S. Sasaki. Toyota's newly developed hybrid powertrain. In *Proceedings of the 10th International Symposium on Power Semiconductor Devices and ICs. ISPSD'98 (IEEE Cat. No.98CH36212)*, page 17–22, Jun 1998.

[59] Lorenzo Serrao, Simona Onori, and Giorgio Rizzoni. A comparative analysis of energy management strategies for hybrid electric vehicles. *Journal of Dynamic Systems, Measurement, and Control*, 133(3):031012–031012–9, Mar 2011.

[60] Daliang Shen, Liting Lu, and Steffen Müller. Utilization of predictive information to optimize driving and powertrain control of series hybrid vehicles. *Automotive and Engine Technology*, 2(1):39–47, Dec 2017.

[61] Hanho Son, Hyunhwa Kim, Sungho Hwang, and Hyunsoo Kim. Development of an advanced rule-based control strategy for a phev using machine learning. *Energies*, 11(1):89, Jan 2018.

[62] Marco Sorrentino, Gianfranco Rizzo, and Ivan Arsie. Analysis of a rule-based control strategy for on-board energy management of series hybrid vehicles. *Control Engineering Practice*, 19(12):1433–1441, Dec 2011.

[63] Alexander Styler, Andreas Sauer, Illah Nourbakhsh, and Hermann Rottengruber. Learned optimal control of a range extender in a series hybrid vehicle. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, page 2612–2618, Sep 2015.

[64] 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.

[65] B. Tabbache, A. Kheloui, M. E. H. Benbouzid, A. Mamoune, and D. Diallo. Research on fault analysis and fault-tolerant control of ev/hev powertrain. In *2014 First International Conference on Green Energy ICGE 2014*, page 284–289, Mar 2014.

[66] Kenji Tadakuma, Tomoyuki Doi, Mitsuhisa Shida, and Kazuhiro Maeda. Prediction formula of aerodynamic drag reduction in multiple-vehicle platooning based on wake analysis and on-road experiments. *SAE International Journal of Passenger Cars - Mechanical Systems*, 9(2):645–656, Apr 2016.

[67] Sadayuki Tsugawa. An overview on an automated truck platoon within the energy its project. *IFAC Proceedings Volumes*, 46(21):41–46, 2013.

[68] OAR US EPA. Greenhouse gas emissions model (gem) for medium- and heavy-duty vehicle compliance, Oct 2016. https://www.epa.gov/regulations-emissions-vehicles-and-engines/greenhouse-gas-emissions-model-gem-medium-and-heavy-duty. Accessed on 26 March 2020.

[69] Korosh Vatanparvar, Sina Faezi, Igor Burago, Marco Levorato, and Mohammad Abdullah Al Faruque. Extended range electric vehicle with driving behavior estimation in energy management. *IEEE Transactions on Smart Grid*, 10(3):2959–2968, May 2019.

[70] Hong Wang, Yanjun Huang, Amir Khajepour, and Qiang Song. Model predictive control-based energy management strategy for a series hybrid electric tracked vehicle. *Applied Energy*, 182:105–114, Nov 2016.

[71] Pengyue Wang, Yan Li, Shashi Shekhar, and William F. Northrop. A deep reinforcement learning framework for energy management of extended range electric delivery vehicles. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, page 1837–1842, Jun 2019.

[72] Johnny Wehbe and Nabil Karami. Battery equivalent circuits and brief summary of components value determination of lithium ion: A review. In *2015 Third International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE)*, page 45–49, Apr 2015.

[73] X. Wei, L. Guzzella, V. I. Utkin, and G. Rizzoni. Model-based fuel optimal control of hybrid electric vehicle using variable structure control systems. *Journal of Dynamic Systems, Measurement, and Control*, 129(1):13–19, Jan 2007.

[74] Jan Wittenbecher. *Contributions to the Analysis and Design of Mechanical Systems for a Series Hybrid Chevrolet Camaro*. PhD thesis, University of Washington, 2017.

[75] Dongxiao Wu, Jin Ren, Huw Davies, Jinlei Shang, and Olivier Haas. Intelligent hydrogen fuel cell range extender for battery electric vehicles. *World Electric Vehicle Journal*, 10(2):29, Jun 2019.

[76] Lihe Xi, Xin Zhang, Chuanyang Sun, Zexing Wang, Xiaosen Hou, and Jibao Zhang. Intelligent energy management control for extended range electric vehicles based on dynamic programming and neural network. *Energies*, 10(11):1871, Nov 2017.

[77] Liwei Xu, Weichao Zhuang, Guodong Yin, and Chentong Bian. Energy-oriented cruising strategy design of vehicle platoon considering communication delay and disturbance. *Transportation Research Part C: Emerging Technologies*, 107:34–53, Oct 2019.

[78] Xiaowei Xu, Hongxia Wang, Nan Zhang, Zhenxing Liu, and Xiaoqing Wang. Review of the fault mechanism and diagnostic techniques for the range extender hybrid electric vehicle. *IEEE Access*, 5:14234–14244, 2017. Conference Name: IEEE Access.

[79] Kaijiang Yu, Qing Liang, Junqi Yang, and Yanan Guo. Model predictive control for hybrid electric vehicle platooning using route information. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 230(9):1273–1285, Aug 2016.

[80] Michael Zabat, Nick Stabile, Stefano Frascaroli, and Fred Browand. Drag forces experienced by 2, 3 and 4-vehicle platoons at close spacings. *SAE Transactions*, 104:1173–1181, 1995.

[81] Xiangrui Zeng and Junmin 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.

[82] Xiangrui Zeng and Junmin Wang. A parallel hybrid electric vehicle energy manage-

ment strategy using stochastic model predictive control with road grade preview. *IEEE Transactions on Control Systems Technology*, 23(6):2416–2423, Nov 2015.

[83] Chen Zhang, Ardalan Vahidi, Pierluigi Pisu, Xiaopeng Li, and Keith Tennant. Role of terrain preview in energy management of hybrid electric vehicles. *IEEE Transactions on Vehicular Technology*, 59(3):1139–1147, Mar 2010.

[84] Shuwei Zhang, Yugong Luo, Junmin Wang, Xiao Wang, and Keqiang Li. Predictive energy management strategy for fully electric vehicles based on preceding vehicle movement. *IEEE Transactions on Intelligent Transportation Systems*, 18(11):3049–3060, Nov 2017.

[85] Hengbing Zhao and Andrew Burke. Modelling and analysis of plug-in series-parallel hybrid medium-duty vehicles. page 10.

[86] Andrzej Łebkowski. Steam and oxyhydrogen addition influence on energy usage by range extender—battery electric vehicles. *Energies*, 11(9):2403, Sep 2018.

[87] Branimir Škugor, Mihael Cipek, and Joško Deur. Control variables optimization and feedback control strategy design for the blended operating regime of an extended range electric vehicle. *SAE International Journal of Alternative Powertrains*, 3(1):152–162, 2014.

Appendix A

Algorithm Implementation

The implementation of algorithms discussed in this dissertation using MATLAB/Simulink platform is provided for reference. The code is also available online in the public Github repository: `https://github.com/avk4714/PhDDissertationCode_AVKalia`. The code is added as Matlab script using the m-code style for LaTeX, created by Florian Knorn [40].

A.1 SHEV/EREV DC-ARTDP Code

The provided function code is used to compute the optimal battery SOC trajectory using the DC-ARTDP alogrithm in context of the Series Hybrid Electric Vehicle. This function is available to be used along with the Passenger Vehicle Model in the Github repository.

```
function [optResults] = optimize_SHEV_DP(driveParams,vehParams)
%OPTIMIZE_SHEV_DP Generates optimal operating points for SHEV using DP.
%   The function is based on the SHEV_DP_FwdProp.m script to assist in an
%   easier integration with Simulink for RT evaluation. The input argument
%   is form of structure.

%   -> driveParams : Includes drive trace parameters
%   -->driveParams.time_s
```

```
%    -->driveParams.spd_mph
%    -->driveParams.grade_pct
%    -->driveParams.startIdx
%    -->driveParams.tripTypeChoice

%    -> vehParams : Includes vehicle initialization parameters
%    -->vehParams.SOC_Max
%    -->vehParams.SOC_Min
%    -->vehParams.SOC_Begin
%    -->vehParams.SOC_Final
%    -->vehParams.Fuel_init
%    -->vehParams.VehMass
%    -->vehParams.rl_a
%    -->vehParams.rl_b
%    -->vehParams.rl_c
%    -->vehParams.L_aux
%    -->vehParams.SOC_PRCSN
%    -->vehParams.minFsblRng
%    -->vehParams.SOC_ReOpt
%    -->vehParams.P_gen_max

%    # Output structure is as follows:
%    -> optResults : Includes optimization results for the model
%    -->optResults.optSOC_pct
%    -->optResults.optPgen_W
%    -->optResults.optPbatt_W
%    -->optResults.optPfuel_W
%    -->optResults.optPregen_W
%    -->optResults.optNetEC
%    -->optResults.fsblRange_m
%    -->optResults.estRange_m
%    -->optResults.EC_Wh_m
%    -->optResults.minFsblRng_m
```

```
42 %   -->optResults.Veh_Load_W
43 %   -->optResults.Veh_Dem_W
44
45 %% Main Body
46 %-- SHEV State Vector
47 SOC_Max = vehParams.SOC_Max;
48 SOC_Min = vehParams.SOC_Min;
49 SOC_Begin = vehParams.SOC_Begin;
50 SOC_ReOpt = vehParams.SOC_ReOpt;
51 SOC_Final = vehParams.SOC_Final;
52 SOC_Range = [];                    % Initializing SOC_range vector as an
53                                    % empty vector
54 MAX_FUEL = vehParams.Fuel_init;    % Liters. 1 Liter = 0.264172.
55                                    % Max is 26.49
56 LHV_FUEL = 30000;                  % kJ/kg or J/g
57 RHO_FUEL = 783;                    % g/L
58 J2WH_CONV = 0.00027778;            % Conversion multiplier
59 N = length(driveParams.time_s);
60 STRTIDX = driveParams.startIdx;    % To assist with re-optimization
61
62 %-- Vehicle Parameters
63 VehMass = vehParams.VehMass;       % Vehicle + Driver
64 rl_a = vehParams.rl_a;
65 rl_b = vehParams.rl_b;
66 rl_c = vehParams.rl_c;
67 L_aux = vehParams.L_aux;           % Auxiliary load due to electronics [W]
68
69 E_batt = 18900;                        % Wh
70 P_gen_max = vehParams.P_gen_max;       %-15000; %-11500;    % W
71 P_batt_dchg_max = 208000;              % W
72 P_batt_ch_max = -102000;               % W
73 P_regen_max = -15000;                  % W
74
```

```
75 for ii = 1:N
76     if ii == 1
77         L_drv(ii,1) = 0;
78         acc_drv(ii,1) = 0;
79     else                                % Power = mass * acc * velocity [W]
80         acc_drv(ii,1) = (((driveParams.spd_mph(ii) -...
81             driveParams.spd_mph(ii-1))*0.447)/(driveParams.time_s(ii)...
82             - driveParams.time_s(ii-1)));
83         L_drv(ii,1) = VehMass * acc_drv(ii,1) *...
84             abs((driveParams.spd_mph(ii))*0.447);%driveParams.spd_mph(ii-1)
85     end
86     L_load(ii,1) = ((((rl_c * (driveParams.spd_mph(ii))^2)) +...
87         (rl_b * driveParams.spd_mph(ii)) + (rl_a)) * 4.448) *...
88         (driveParams.spd_mph(ii) * 0.447);
89     L_total(ii,1) = calcMotElecPwr((L_drv(ii) +...
90         L_load(ii)),driveParams.spd_mph(ii));
91     if L_total(ii,1) ≤ 0
92         L_total(ii,1) = max(P_batt_ch_max,L_total(ii,1));
93     else
94         L_total(ii,1) = min(P_batt_dchg_max,L_total(ii,1));
95     end
96 end
97
98 optResults.Veh_Dem_W = L_drv + L_load;
99
100 %-- Adaptive DP
101 % 1. Adjust power demand - Negative power demand values imply
102 % braking/stoppage. For ease of calculation, any power value below or equal
103 % to zero will be treated as a zero.
104
105 L_total = L_total + L_aux;
106
107
```

```
% 2. Total Energy Demand Associated with Drive Cycle and Distance
% Accmulated
for j = 1:N
    if j == 1
        E_dem_total(j,1) = 0;
        DistTrvld_m(j,1) = 0;
    else
        E_dem_total(j,1) = E_dem_total(j-1,1) + (L_total(j,1)/3600);   % Wh
        DistTrvld_m(j,1) = DistTrvld_m(j-1,1) + ...
            abs((driveParams.spd_mph(j-1)) * 0.44704);             % meters
    end
end

% 3. Delta Power - Calculate change in power demand between steps to
% determine discharge and charge scenarios. This helps compute feasible SOC
% limits.

delPwr_W = zeros(size(L_total));        % First element is zero by default.
for i = 2:N
    delPwr_W(i,1) = L_total(i,1) - L_total(i-1,1);
end

% 4. Initialize minimum and maximum SOC values - For each time instance, a
% min and max SOC value can be obtained based on drive cycle data as well
% as power and energy limits.

X_lim = zeros([N 2]);         % X represents state (SOC) here. First column
                              % is min value and second is max value.

X_lim(STRTIDX,1) = SOC_Begin;% Initializing limits at initial time equal to
X_lim(STRTIDX,2) = SOC_Begin;% the final SOC value.

X_PRCSN = vehParams.SOC_PRCSN;
```

```
141 % Precision for the SOC. 0.0001 corresponds to 0.01% SOC change.
142
143 % 5. Initialize additional variables and parameters
144
145 L_xN = 0;
146 X_opt = zeros([N 1]);                          % State (SOC)
147 P_gen_opt = zeros([N 1]);
148 P_batt_opt = zeros([N 1]);
149 P_battopt_total = zeros([N 1]);
150 P_regen_opt = zeros([N 1]);
151 P_fuel_opt = zeros([N 1]);
152 P_fuelopt_total = zeros([N 1]);
153 FuelUsed_L = zeros([N 1]);
154 fuelRate_opt = zeros([N 1]);
155
156 P_regen_LUT = [0 0;
157                2.6 0;
158                5.2 -1250;
159                7.8 -3000;
160                10.4 -5500;
161                13 -8000;
162                15.6 -10250;
163                18.2 -12500;
164                20.8 -13750;
165                23.4 -14000;
166                26 -14500;
167                28.6 -15000;
168                31.2 -15000;
169                33.8 -15000;
170                38 -15000];          % Regen power Lookup Table
171 eta_genmot = 0.86;             % mean value obtained based on the Bosch ICD
172 %% Loop section - DP Forward Propagation
173 X_opt(1,1) = SOC_Begin;
```

```
174 X_opt(STRTIDX,1) = SOC_ReOpt;
175 l = STRTIDX + 1;
176 switch driveParams.tripTypeChoice
177     case 'one-way'
178         minFsblRng = 0;
179     case 'return'
180         minFsblRng = DistTrvld_m(end);          % meters
181     case 'custom'
182         minFsblRng = vehParams.minFsblRng;      % meters
183     otherwise
184         minFsblRng = 0;
185 end
186 FUEL_FLAG = 0;
187 while(l ≤ N)
188     % disp(l)
189     P_regen_avail = interp1(P_regen_LUT(:,1),P_regen_LUT(:,2),...
190         (driveParams.spd_mph(l,1)*0.447),'linear');
191     if L_total(l) > 0                               % Discharging condition
192         % Step 1: Determine SOC Limits
193         X_lim(l,2) = X_opt(l-1,1) -...
194             ((L_total(l) + P_gen_max)/(E_batt * 3600));  % Max feasible SOC
195         X_lim(l,1) = X_opt(l-1,1) -...
196             ((L_total(l))/(E_batt * 3600));              % Min feasible SOC
197
198         % NOTE: Here Max feasible SOC corresponds to with generator and Min
199         % is without generator. Because of Forward Propagation.
200
201         % Step 2a: Form SOC Query Range and form ∆ SOC vector
202         X_range(:,1) = X_lim(l,1):X_PRCSN:X_lim(l,2);
203         P_gen_range(:,1) = ((X_opt(l-1,1) - X_range) *...
204             (E_batt * 3600)) - L_total(l);
205         P_batt_range(:,1) = L_total(l) + P_gen_range(:,1);
206         len_x_rng = length(X_range);
```

```
207         fuelRate = zeros([len_x_rng 1]);
208
209         for k = 1:len_x_rng
210             if P_gen_range(k) < -500
211                 [engEff,fuelRate(k,1)] = calcEngEff(abs(P_gen_range(k)...
212                     /eta_genmot));
213                 P_fuel_range(k,1) = (P_gen_range(k)/eta_genmot)*(1/engEff);
214             else
215                 fuelRate(k,1) = 0;
216                 P_fuel_range(k,1) = 0;
217             end
218         end
219
220         % Step 2b: Determine feasible SOC values based on range estimation
221         ECvector_Wh_m(:,1) = ((X_opt(1,1) - X_range(:,1)) * E_batt)/...
222            (DistTrvld_m(l,1) - DistTrvld_m(1,1));
223
224         % Step 3a: Compute control candidate values corresponding to all
225         % Δ SOC
226         for k = 1:len_x_rng
227             if ECvector_Wh_m(k,1) > 0
228                 estRngVec_m(k,1) = ((X_range(k,1) - SOC_Min) * E_batt)...
229                     /ECvector_Wh_m(k,1);
230                 fsblRngVec(k,1) = estRngVec_m(k,1) - (DistTrvld_m(end,1)...
231                     - DistTrvld_m(l,1));
232             else
233                 estRngVec_m(k,1) = Inf;
234                 fsblRngVec(k,1) = estRngVec_m(k,1) - (DistTrvld_m(end,1)...
235                     - DistTrvld_m(l,1));
236             end
237
238             P_gen(k) = P_gen_max *...
239                 ((X_range(k) - X_range(1))/(X_range(end) - X_range(1)));
```

```
240             if P_gen(k) ≠ 0 && ¬isnan(P_gen(k))
241                 [engEff,¬] = calcEngEff(abs(P_gen(k)/eta_genmot));
242                 P_fuel(k) = (P_gen(k)/eta_genmot) * (1/engEff);
243             else
244                 P_fuel(k) = 0;
245             end
246             if (fsblRngVec(k,1) ≥ minFsblRng) && FUEL_FLAG == 0
247                 P_batt(k) = L_total(l) + P_gen(k);
248             else
249                 P_batt(k) = Inf;  % Maximum penalty added for infeasibility
250             end
251             P_regen(k) = 0;
252         end
253
254         % Step 3b: If all solution values result in infinity
255         if isinf(P_batt)
256             if FUEL_FLAG == 0
257                 P_batt(end) = L_total(l) + P_gen(end);
258             else
259                 P_batt(1) = L_total(l);
260             end
261         end
262
263         % Step 4: Calculate Cost-to-go for each possible path.
264         Y_x = (P_batt + abs(P_fuel))/3600;
265
266         % Step 5: Determine optimal path as per cost function.
267         opt_idx = find(Y_x == min(Y_x));
268
269         % 4200 W is minimum power engine map can resolve
270         if P_gen_max ≤ -500
271             % Step 6: Assign optimal SOC value and repeat.
272             P_gen_opt(l,1) = P_gen(opt_idx);
```

```
273             P_fuel_opt(l,1) = abs(P_fuel(opt_idx));
274             P_batt_opt(l,1) = P_batt(opt_idx);
275             P_regen_opt(l,1) = P_regen(opt_idx);
276             X_opt(l,1) = X_range(opt_idx);
277             fuelRate_opt(l,1) = fuelRate(opt_idx);
278         else
279             % Step 6: Assign optimal SOC value and repeat.
280             P_gen_opt(l,1) = P_gen_max;
281             P_fuel_opt(l,1) = P_fuel;
282             P_batt_opt(l,1) = P_batt_range;
283             P_regen_opt(l,1) = P_regen;
284             X_opt(l,1) = X_range;
285             fuelRate_opt(l,1) = fuelRate;
286         end
287
288     elseif L_total(l) ≤ 0           % Regen-braking condition
289         % Step 1: Determine SOC Limits
290         if L_total(l) > P_regen_avail
291             L_final = L_total(l);
292             X_lim(l,2) = X_opt(l-1,1) -...
293                 ((L_final + P_gen_max)/(E_batt * 3600)); % Max feasible SOC
294             X_lim(l,1) = X_opt(l-1,1) -...
295                 ((L_final)/(E_batt * 3600));             % Min feasible SOC
296         else
297             L_final = P_regen_avail;
298             X_lim(l,2) = X_opt(l-1,1) -...
299                 ((L_final + P_gen_max)/(E_batt * 3600)); % Max feasible SOC
300             X_lim(l,1) = X_opt(l-1,1) -...
301                 ((L_final)/(E_batt * 3600));             % Min feasible SOC
302         end
303
304         % Step 2a: Form SOC Query Range and form ∆ SOC vector
305         X_range(:,1) = X_lim(l,1):X_PRCSN:X_lim(l,2);
```

```
306         P_gen_range(:,1) = ((X_opt(l-1,1) - X_range) *...
307             (E_batt * 3600)) - L_total(l);
308         P_batt_range(:,1) = L_total(l) + P_gen_range(:,1);
309         len_x_rng = length(X_range);
310         fuelRate = zeros([len_x_rng 1]);
311
312
313         for k = 1:len_x_rng
314             if P_gen_range(k) < -500
315                 [engEff,fuelRate(k,1)] = calcEngEff(abs(P_gen_range(k)...
316                     /eta_genmot));
317                 P_fuel_range(k,1) = (P_gen_range(k)/eta_genmot) *...
318                     (1/engEff);
319             else
320                 fuelRate(k,1) = 0;
321                 P_fuel_range(k,1) = 0;
322             end
323         end
324
325         ECvector_Wh_m(:,1) = ((X_opt(1,1) - X_range(:,1)) * E_batt)/...
326            (DistTrvld_m(l,1) - DistTrvld_m(1,1));
327
328         % Step 3a: Compute control candidate values corresponding to all
329         % Δ SOC
330         for k = 1:len_x_rng
331             if ECvector_Wh_m(k,1) > 0
332                 estRngVec_m(k,1) = ((X_range(k,1) - SOC_Min) * E_batt)...
333                     /ECvector_Wh_m(k,1);
334                 fsblRngVec(k,1) = estRngVec_m(k,1) - (DistTrvld_m(end,1)...
335                     - DistTrvld_m(l,1));
336             else
337                 estRngVec_m(k,1) = Inf;
338                 fsblRngVec(k,1) = estRngVec_m(k,1) - (DistTrvld_m(end,1)...
```

```
339                     - DistTrvld_m(l,1));
340             end
341             % ---
342             P_gen(k) = P_gen_max *...
343                 ((X_range(k) - X_range(1))/(X_range(end) - X_range(1)));
344             if P_gen(k) ≠ 0 && ¬isnan(P_gen(k))
345                 [engEff,¬] = calcEngEff(abs(P_gen(k)/eta_genmot));
346                 P_fuel(k) = (P_gen(k)/eta_genmot) * (1/engEff);
347             else
348                 P_fuel(k) = 0;
349             end
350             P_regen(k) = L_final;
351             % ---
352             if (fsblRngVec(k,1) ≥ minFsblRng) && FUEL_FLAG == 0
353                 P_batt(k) = P_regen(k) + P_gen(k);
354             else
355                 P_batt(k) = Inf;  % Maximum penalty added for infeasibility
356             end
357         end
358
359         % Step 3b: If all solution values result in infinity
360         if isinf(P_batt)
361             if FUEL_FLAG == 0
362                 P_batt(end) = L_total(l) + P_gen(end);
363             else
364                 P_batt(1) = L_total(l);
365             end
366         end
367
368         % Step 4: Calculate Cost-to-go for each possible path.
369         Y_x = (P_batt + abs(P_fuel))/3600;
370
371         % Step 5: Determine optimal path as per cost function.
```

```
372         opt_idx = find(Y_x == min(Y_x));
373
374         % 4200 W is minimum power engine map can resolve
375         if P_gen_max ≤ -500
376             % Step 6: Assign optimal SOC value and repeat.
377             P_gen_opt(l,1) = P_gen(opt_idx);
378             P_fuel_opt(l,1) = abs(P_fuel(opt_idx));
379             P_batt_opt(l,1) = P_batt(opt_idx);
380             P_regen_opt(l,1) = P_regen(opt_idx);
381             X_opt(l,1) = X_range(opt_idx);
382             fuelRate_opt(l,1) = fuelRate(opt_idx);
383         else
384             % Step 6: Assign optimal SOC value and repeat.
385             P_gen_opt(l,1) = P_gen_max;
386             P_fuel_opt(l,1) = P_fuel;
387             P_batt_opt(l,1) = P_batt_range;
388             P_regen_opt(l,1) = P_regen;
389             X_opt(l,1) = X_range;
390             fuelRate_opt(l,1) = fuelRate;
391         end
392
393     end
394     % Optimal solution addition
395     P_battopt_total(l,1) = P_battopt_total(l-1,1) + P_batt_opt(l,1);
396     P_fuelopt_total(l,1) = P_fuelopt_total(l-1,1) + P_fuel_opt(l,1);
397     FuelUsed_L(l,1) = FuelUsed_L(l-1,1) + fuelRate_opt(l,1);
398
399     % Fuel completion check
400     if (MAX_FUEL - FuelUsed_L(l,1)) ≥ 0
401         FUEL_FLAG = 0;
402     else
403         FUEL_FLAG = 1;
404     end
```

```
405
406     EC_Wh_m(l,1) = ((X_opt(1,1) - X_opt(l,1)) * E_batt)/(DistTrvld_m(l,1));
407
408     if EC_Wh_m(l,1) > 0
409         estRange_m(l,1) = ((X_opt(l,1) - SOC_Min) * E_batt)/EC_Wh_m(l,1);
410         fsblRange(l,1) = estRange_m(l,1) - (DistTrvld_m(end,1)...
411             - DistTrvld_m(l,1));
412     else
413         estRange_m(l,1) = Inf;
414         fsblRange(l,1) = estRange_m(l,1) - (DistTrvld_m(end,1)...
415             - DistTrvld_m(l,1));
416     end
417     l = l + 1;
418 end
419
420 if any(X_opt < SOC_Min)
421     disp('## Infeasible Solution due to SOC violation ##')
422     OPT_KEYWORD = 'UNSUCCESSFUL';
423     idx = find(X_opt < SOC_Min, 1);
424     NetEC_opt = ((sum(P_batt_opt(1:idx-1)) + sum(P_fuel_opt(1:idx-1)))...
425         /3600)/DistTrvld_m(idx-1);     % Wh/m
426 else
427     disp('## Optimization was successful for SOC ##')
428     OPT_KEYWORD = 'SUCCESSFUL';
429     NetEC_opt = ((sum(P_batt_opt) + sum(P_fuel_opt))/3600)...
430         /DistTrvld_m(end);     % Wh/m
431 end
432
433 % Range based Feasibility
434 if fsblRange(end) < minFsblRng
435     disp('## Infeasible solution due to range violation ##')
436     OPT_KEYWORD_2 = 'UNSUCCESSFUL';
437 else
```

```
438     disp('## Optimization was successful for Range ##')
439     OPT_KEYWORD_2 = 'SUCCESSFUL';
440 end
441 
442 %% Output Data Structure
443 optResults.optSOC_pct = X_opt;
444 optResults.optPgen_W = P_gen_opt;
445 optResults.optPbatt_W = P_batt_opt;
446 optResults.optPfuel_W = P_fuel_opt;
447 optResults.optPregen_W = P_regen_opt;
448 optResults.optNetEC = NetEC_opt;
449 optResults.fsblRange_m = fsblRange;
450 optResults.estRange_m = estRange_m;
451 optResults.distTrvld_m = DistTrvld_m;
452 optResults.EC_Wh_m = EC_Wh_m;
453 optResults.Veh_Load_W = L_load;
454 optResults.FuelUsed_L = FuelUsed_L;
455 optResults.minFsblRng_m = minFsblRng;
456 optResults.STATUS_KEY = OPT_KEYWORD;
457 optResults.STATUS_KEY_2 = OPT_KEYWORD_2;
458 end
```

A.2 SHEV/EREV MPC Code

The provided function code is used to compute the receding horizon model predictive control output for Series Hybrid Electric Vehicle generator system set point. This function is available to be used along with the Passenger Vehicle Model in the Github repository.

```
function [x_soc_est_k_1,P_gen_opt,SOC_opt,status,p] =...
    soc_mpc(x_soc_k,u_soc_ref_vec,u_Paux_k_1,u_P_gen_k_1,u_est_spd_vec,...
    u_est_acc_vec,md_grade_vec,estHorizon,ctlHorizon,Ts)
% ,b,H,f,A
%% Linearized Vehicle Power Model - UW EREV
m = 2012;
g = 9.81;
% Aerodynamic Drag Resistance Power Coefficient
Af = 0.4;
Cd = 0.34;
rho = 1.225;
A1 = 0.5 * Af * Cd * rho;
% Rolling Resistance Power Coefficient
Cr = 0.00502;
B1 = m * g * Cr;
% Climbing Resistance Power Coefficient
C1 = m * g;
% Battery information
Ebatt = 18900;

%% Estimation vector
persistent soc_est_vec
if isempty(soc_est_vec)
    soc_est_vec = ones([estHorizon 1]);
end
```

```
26
27 %% Model based estimation
28 Pbatt = zeros([estHorizon 1]);
29 for k = 1:estHorizon
30     Pbatt(k,1) = u_Paux_k_1 - u_P_gen_k_1 + (m * u_est_spd_vec(k,1) *...
31         u_est_acc_vec(k,1)) + (A1 * (u_est_spd_vec(k,1)^3)) +...
32         (B1 * u_est_spd_vec(k,1)) + (C1 * (md_grade_vec(k,1)/100) *...
33         u_est_spd_vec(k,1));
34     if Pbatt(k,1) < -15000
35         Pbatt(k,1) = -15000;
36     end
37     if k == 1
38         soc_est_vec(k,1) = x_soc_k - ((Ts/(Ebatt * 3600)) * Pbatt(k,1));
39     else
40         soc_est_vec(k,1) = soc_est_vec(k-1,1) - ((Ts/(Ebatt * 3600)) *...
41             Pbatt(k,1));
42     end
43 end
44 x_soc_est_k_1 = soc_est_vec;
45
46 %% Optimal Generator Power - mpcqpActiveSetSolver
47 p1 = 1;
48 p2 = 1e8;
49 k1 = 0.000141 * Ts;  % g/W; %Ts/(3600*18900);
50 k2 = 1;
51 H = eye(ctlHorizon * 3);
52 for j = 1:(ctlHorizon * 3)
53     if mod(j,3) == 1
54         H(j,:) = H(j,:) * (p1 * k1 * k1);
55     elseif mod (j,3) == 2
56         H(j,:) = H(j,:) * (p2 * k2 * k2);
57     else
58         H(j,:) = H(j,:) * p2 * (u_soc_ref_vec(j/3)^2);
```

```
    end
end
H = 2 * H;
[L,p] = chol(H,'lower');
Linv = inv(L);

f = zeros([ctlHorizon*3 1]);
f(2:3:end) = -2 * p2 * k2 * k2 * u_soc_ref_vec(1:ctlHorizon);

ineqVec_1 = [1 0 0;
             -1 0 0;
              0 1 0;
              0 -1 0];
ineqVec_3 = [1 0 0 1 0 0 1 0 0;
             -1 0 0 -1 0 0 -1 0 0;
              0 1 0 0 1 0 0 1 0;
              0 -1 0 0 -1 0 0 -1 0];
ineqVec_5 = [1 0 0 1 0 0 1 0 0 1 0 0 1 0 0;
             -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0;
              0 1 0 0 1 0 0 1 0 0 1 0 0 1 0;
              0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0];
ineqVec_10 = [1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0;
      -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0;
      0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0;
     0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0];
ineqVec_15 = [ineqVec_5 ineqVec_10];
ineqVec_20 = [ineqVec_10 ineqVec_10];
ineqVec_25 = [ineqVec_5 ineqVec_20];
ineqVec_30 = [ineqVec_15 ineqVec_15];
switch ctlHorizon
    case 1
        mult = 1;
        A = ineqVec_1;
```

```
92          b = [0 * mult; -15000 * mult; sum(x_soc_est_k_1(1:mult,1));
93              -(sum(x_soc_est_k_1(1:mult,1)) +...
94              (((15000 * Ts)/(18900 * 3600))*mult))];
95          Aeq = [-(Ts/(18900 * 3600)) 1 0];
96          beq = sum(x_soc_est_k_1(1:mult,1));
97      case 3
98          mult = 3;
99          A = ineqVec_3;
100         b = [0 * mult; -15000 * mult; sum(x_soc_est_k_1(1:mult,1));
101             -(sum(x_soc_est_k_1(1:mult,1)) +...
102             (((15000 * Ts)/(18900 * 3600))*mult))];
103         tmp_Aeq = [-(Ts/(18900 * 3600)) 1 0];
104         Aeq = [tmp_Aeq tmp_Aeq tmp_Aeq];
105         beq = sum(x_soc_est_k_1(1:mult,1));
106     case 5
107         mult = 5;
108         A = ineqVec_5;
109         b = [0 * mult; -15000 * mult; sum(x_soc_est_k_1(1:mult,1));
110             -(sum(x_soc_est_k_1(1:mult,1)) +...
111             (((15000 * Ts)/(18900 * 3600))*mult))];
112         tmp_Aeq = [-(Ts/(18900 * 3600)) 1 0];
113         Aeq = [tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq];
114         beq = sum(x_soc_est_k_1(1:mult,1));
115     case 10
116         mult = 10;
117         A = ineqVec_10;
118         b = [0 * mult; -15000 * mult; sum(x_soc_est_k_1(1:mult,1));
119             -(sum(x_soc_est_k_1(1:mult,1)) +...
120             (((15000 * Ts)/(18900 * 3600))*mult))];
121         tmp_Aeq = [-(Ts/(18900 * 3600)) 1 0];
122         Aeq = [tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
123             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq];
124         beq = sum(x_soc_est_k_1(1:mult,1));
```

```
125     case 15
126         mult = 15;
127         A = ineqVec_15;
128         b = [0 * mult; -15000 * mult; sum(x_soc_est_k_1(1:mult,1));
129             -(sum(x_soc_est_k_1(1:mult,1)) +...
130             (((15000 * Ts)/(18900 * 3600))*mult))];
131         tmp_Aeq = [-(Ts/(18900 * 3600)) 1 0];
132         Aeq = [tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
133             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
134             tmp_Aeq tmp_Aeq];
135         beq = sum(x_soc_est_k_1(1:mult,1));
136     case 20
137         mult = 20;
138         A = ineqVec_20;
139         b = [0 * mult; -15000 * mult; sum(x_soc_est_k_1(1:mult,1));
140             -(sum(x_soc_est_k_1(1:mult,1)) +...
141             (((15000 * Ts)/(18900 * 3600))*mult))];
142         tmp_Aeq = [-(Ts/(18900 * 3600)) 1 0];
143         Aeq = [tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
144             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
145             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq];
146         beq = sum(x_soc_est_k_1(1:mult,1));
147     case 25
148         mult = 25;
149         A = ineqVec_25;
150         b = [0 * mult; -15000 * mult; sum(x_soc_est_k_1(1:mult,1));
151             -(sum(x_soc_est_k_1(1:mult,1)) +...
152             (((15000 * Ts)/(18900 * 3600))*mult))];
153         tmp_Aeq = [-(Ts/(18900 * 3600)) 1 0];
154         Aeq = [tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
155             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
156             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
157             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq];
```

```
158         beq = sum(x_soc_est_k_1(1:mult,1));
159     case 30
160         mult = 30;
161         A = ineqVec_30;
162         b = [0 * mult; -15000 * mult; sum(x_soc_est_k_1(1:mult,1));
163             -(sum(x_soc_est_k_1(1:mult,1)) +...
164             (((15000 * Ts)/(18900 * 3600))*mult))];
165         tmp_Aeq = [-(Ts/(18900 * 3600)) 1 0];
166         Aeq = [tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
167             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
168             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
169             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
170             tmp_Aeq tmp_Aeq tmp_Aeq];
171         beq = sum(x_soc_est_k_1(1:mult,1));
172     otherwise
173         mult = 30;
174         A = ineqVec_30;
175         b = [0 * mult; -15000 * mult; sum(x_soc_est_k_1(1:mult,1));
176             -(sum(x_soc_est_k_1(1:mult,1)) +...
177             (((15000 * Ts)/(18900 * 3600))*mult))];
178         tmp_Aeq = [-(Ts/(18900 * 3600)) 1 0];
179         Aeq = [tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
180             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
181             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
182             tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq tmp_Aeq...
183             tmp_Aeq tmp_Aeq];
184         beq = sum(x_soc_est_k_1(1:mult,1));
185 end
186
187 opt = mpcqpsolverOptions;
188
189 iA0 = false(size(b));
190
```

```
191 [x,status] = mpcqpsolver(Linv,f,A,b,Aeq,beq,iA0,opt);
192 P_gen_opt = x(1:3:end);
193 SOC_opt = x(2:3:end);
194 end
```

A.3 SPHET DC-ARTDP Code

The provided function code is used to compute the optimal battery SOC trajectory using the DC-ARTDP alogrithm in context of the Series Parallel Hybrid Electric Truck. This function is available to be used along with the Commercial Vehicle Platoon Model in the Github repository.

```
function [optResults] = opt_SPHET_DP(driveParams,vehParams)
%OPTIMIZE_SPHET_DP Generates optimal operation for SPHET using DC-ARTDP.
%   The input argument is form of structure.

%   -->driveParams : Includes drive trace parameters
%   -->driveParams.time_s
%   -->driveParams.spd_mph
%   -->driveParams.grade_pct
%   -->driveParams.startIdx
%   -->driveParams.tripTypeChoice

%   -> vehParams : Includes vehicle initialization parameters
%   -->vehParams.SOC_Max
%   -->vehParams.SOC_Min
%   -->vehParams.SOC_Begin
%   -->vehParams.SOC_Final
%   -->vehParams.Fuel_init
%   -->vehParams.VehMass
%   -->vehParams.rl_a
%   -->vehParams.rl_b
%   -->vehParams.rl_c
%   -->vehParams.L_aux
%   -->vehParams.SOC_PRCSN
%   -->vehParams.minFsblRng
%   -->vehParams.SOC_ReOpt
```

```
26 %   -->vehParams.P_gen_max
27
28 %   # Output structure is as follows:
29 %   -> optResults : Includes optimization results for the model
30 %   -->optResults.optSOC_pct
31 %   -->optResults.optPgen_W
32 %   -->optResults.optPbatt_W
33 %   -->optResults.optPfuel_W
34 %   -->optResults.optPregen_W
35 %   -->optResults.optNetEC
36 %   -->optResults.fsblRange_m
37 %   -->optResults.estRange_m
38 %   -->optResults.EC_Wh_m
39 %   -->optResults.minFsblRng_m
40 %   -->optResults.Veh_Load_W
41 %   -->optResults.Veh_Dem_W
42
43 %% Main Body
44 %-- SHEV State Vector
45 SOC_Max = vehParams.SOC_Max;
46 SOC_Min = vehParams.SOC_Min;
47 SOC_Begin = vehParams.SOC_Begin;
48 SOC_ReOpt = vehParams.SOC_ReOpt;
49 SOC_Final = vehParams.SOC_Final;
50 SOC_Range = [];                          % Initializing SOC_range vector as
51                                          % an empty vector
52 MAX_FUEL = vehParams.Fuel_init;          % Liters. 1 Liter = 0.264172. Max
53                                          % is 26.49
54 LHV_FUEL = 42600;                       % kJ/kg or J/g - Diesel
55 RHO_FUEL = 846;                         % g/L - Diesel
56 J2WH_CONV = 0.00027778;                 % Conversion multiplier
57 N = length(driveParams.time_s);
58 STRTIDX = driveParams.startIdx;         % To assist with re-optimization
```

```
59
60 %-- Vehicle Parameters
61 VehMass = vehParams.VehMass;            % Vehicle + Driver
62 rl_a = vehParams.rl_a;
63 rl_b = vehParams.rl_b;
64 rl_c = vehParams.rl_c;
65 L_aux = vehParams.L_aux;                % Auxiliary load due
66                                         % to electronics [W]
67 g = 9.81;
68
69 E_batt = 145920;                        % Wh
70 P_gen_max = vehParams.P_gen_max;        % W
71 P_batt_dchg_max = 432000;               % W
72 P_batt_ch_max = -144000;                % W
73 P_regen_max = -109440;                  % W
74 P_mot_max = vehParams.P_mot_max;        % W
75
76 % -- Initialization --
77 L_drv = zeros([N 1]);
78 acc_drv = zeros([N 1]);
79 L_load = zeros([N 1]);
80 L_total = zeros([N 1]);
81 HybModeGrp = zeros([N 1]);      % This variable stores the value 1,2.
82                                 % -1: Regen
83                                 % 1: EV, Series
84                                 % 2: Engine, Parallel
85
86 % N by 2 matrix storing min and max value of component power.
87 P_mot_lim = zeros([N 2]);
88 P_gen_lim = zeros([N 2]);
89 P_eng_lim = zeros([N 2]);
90
91 %%
```

```
92 for ii = 1:N
93     if ii == 1
94         L_drv(ii,1) = 0;
95         acc_drv(ii,1) = 0;
96     else
97         acc_drv(ii,1) = (((driveParams.spd_mph(ii) -...
98             driveParams.spd_mph(ii-1))*0.447)/(driveParams.time_s(ii)...
99             - driveParams.time_s(ii-1)));
100        L_drv(ii,1) = VehMass * acc_drv(ii,1) *...
101            abs((driveParams.spd_mph(ii))*0.447);
102    end
103    % Calculates total power load due to resistances at wheels
104    L_load(ii,1) = (((((rl_c * (driveParams.spd_mph(ii))^2)) +...
105        (rl_b * driveParams.spd_mph(ii)) +...
106        (rl_a)) * 4.448) + (VehMass * g *...
107        sin(driveParams.grade_pct(ii)/100))) *...
108        (driveParams.spd_mph(ii) * 0.447);
109
110    % Determine if the load demand can be met by motor alone or requires
111    % engine too.
112    L_total(ii,1) = L_drv(ii,1) + L_load(ii,1);  % Total power demand at
113                                                 % wheels.
114
115    % Sub-Problem. Group 1 is EV or SHEV, Group 2 is Parallel or Engine
116    % Only.
117    if (L_total(ii,1)) > 0
118        if L_total(ii,1) ≤ P_mot_max
119            HybModeGrp(ii,1) = 1;
120            P_eng_lim(ii,1) = 0;
121            P_eng_lim(ii,2) = 0;
122            P_gen_lim(ii,1) = P_gen_max;
123            P_gen_lim(ii,2) = 0;
124            P_mot_lim(ii,1) = 0;
```

```
125             P_mot_lim(ii,2) = (L_total(ii,1));
126         else
127             HybModeGrp(ii,1) = 2;
128             P_eng_lim(ii,1) = (L_total(ii,1)) - P_mot_max;
129             P_eng_lim(ii,2) = (L_total(ii,1));
130             P_gen_lim(ii,1) = 0;
131             P_gen_lim(ii,2) = 0;
132             P_mot_lim(ii,1) = 0;
133             P_mot_lim(ii,2) = P_mot_max;
134         end
135     else
136         HybModeGrp(ii,1) = -1;
137         P_eng_lim(ii,1) = 0;
138         P_eng_lim(ii,2) = 0;
139         P_gen_lim(ii,1) = max(P_gen_max,P_batt_ch_max - P_regen_max);
140         P_gen_lim(ii,2) = 0;
141         P_mot_lim(ii,1) = P_regen_max;
142         P_mot_lim(ii,2) = 0;
143     end
144
145 end
146
147 optResults.Veh_Dem_W = L_total;
148
149 %-- Adaptive DP
150 % 1. Adjust power demand - Negative power demand values imply
151 % braking/stoppage. For ease of calculation, any power value below or equal
152 % to zero will be treated as a zero.
153
154 % 2. Total Energy Demand Associated with Drive Cycle and Distance
155 % Accmulated. This energy is at the wheels and not at the battery or
156 % engine.
157 for j = 1:N
```

```
158     if j == 1
159         E_dem_total(j,1) = 0;
160         DistTrvld_m(j,1) = 0;
161     else
162         E_dem_total(j,1) = E_dem_total(j-1,1) + (L_total(j,1)/3600);  % Wh
163         DistTrvld_m(j,1) = DistTrvld_m(j-1,1) + ...
164             abs((driveParams.spd_mph(j-1)) * 0.44704);            % meters
165     end
166 end
167
168 % 3. Delta Power - Calculate change in power demand between steps to
169 % determine discharge and charge scenarios. This helps compute feasible SOC
170 % limits.
171
172 delPwr_W = zeros(size(L_total));    % First element is zero by default.
173 for i = 2:N
174     delPwr_W(i,1) = L_total(i,1) - L_total(i-1,1);
175 end
176
177 % 4. Initialize minimum and maximum SOC values - For each time instance, a
178 % min and max SOC value can be obtained based on drive cycle data as well
179 % as power and energy limits.
180
181 X_lim = zeros([N 2]);         % X represents state (SOC) here. First column
182                                 % is min value and second is max value.
183
184 X_lim(STRTIDX,1) = SOC_Begin;% Initializing limits at initial time equal to
185 X_lim(STRTIDX,2) = SOC_Begin;       % the final SOC value.
186
187 X_PRCSN = vehParams.SOC_PRCSN;  % Precision for SOC. 0.0001 corresponds to
188                                 % 0.01% SOC change.
189
190 % 5. Initialize additional variables and parameters
```

```
191
192 L_xN = 0;
193 X_opt = zeros([N 1]);            % State (SOC)
194 P_gen_opt = zeros([N 1]);
195 P_eng_opt = zeros([N 1]);
196 P_mot_opt = zeros([N 1]);
197 P_batt_opt = zeros([N 1]);
198 P_regen_opt = zeros([N 1]);
199 P_fuel_opt = zeros([N 1]);
200 EC_fuel_opt = zeros([N 1]);
201 FuelUsed_L = zeros([N 1]);
202 fuelRate_opt = zeros([N 1]);
203
204 P_regen_LUT = [0 0;
205                2 -3648;
206                4 -10944;
207                8 -21888;
208                12 -43776;
209                16 -65664;
210                20 -87552;
211                25 -98496;
212                30 -105792;
213                35 -109440];    % Regen power Lookup Table
214 eta_genmot = 0.92;             % mean value obtained based on the Bosch ICD
215 eta_mot = 0.931;
216 %% Loop section - DP Forward Propagation
217 X_opt(1,1) = SOC_Begin;
218 X_opt(STRTIDX,1) = SOC_ReOpt;
219 l = STRTIDX + 1;
220 switch driveParams.tripTypeChoice
221     case 'one-way'
222         minFsblRng = 0;
223     case 'return'
```

```
224         minFsblRng = DistTrvld_m(end);          % meters
225     case 'custom'
226         minFsblRng = vehParams.minFsblRng;      % meters
227     otherwise
228         minFsblRng = 0;
229 end
230 FUEL_FLAG = 0;
231 while(l ≤ N)
232     P_regen_avail = interp1(P_regen_LUT(:,1),P_regen_LUT(:,2),...
233         (driveParams.spd_mph(l,1)*0.447),'linear');
234     if L_total(l) > 0                               % Discharging condition
235
236         % Step 1: Determine SOC Limits based on the Hybrid Mode
237         % EV and Series Mode
238         if HybModeGrp(l) == 1 && (X_opt(l-1,1) ≥ SOC_Final)
239             X_lim(l,2) = X_opt(l-1,1) -...
240                 (((L_total(l,1)/eta_mot) +...
241                 L_aux + P_gen_lim(l,1))/(E_batt * 3600));% Max feasible SOC
242             X_lim(l,1) = X_opt(l-1,1) -...
243                 (((L_total(l,1)/eta_mot) +...
244                 L_aux + P_gen_lim(l,2))/(E_batt * 3600));% Min feasible SOC
245         % Engine Only and Parallel Mode
246         elseif HybModeGrp(l) == 2 || (X_opt(l-1,1) < SOC_Final)
247             X_lim(l,2) = X_opt(l-1,1) -...
248                 (((P_mot_lim(l,1)/eta_mot) +...
249                 L_aux)/(E_batt * 3600));     % Max feasible SOC
250             X_lim(l,1) = X_opt(l-1,1) -...
251                 (((P_mot_lim(l,2)/eta_mot) +...
252                 L_aux)/(E_batt * 3600));     % Min feasible SOC
253             HybModeGrp(l) = 2;
254         elseif HybModeGrp(l) == -1      % Regen Braking Mode
255             X_lim(l,2) = X_opt(l-1,1) -...
256                 (((P_regen_avail/eta_mot) + L_aux +...
```

```
257                 max(P_gen_max,P_batt_ch_max -...
258                 (P_regen_avail/eta_mot)))/...
259                 (E_batt * 3600));% Max feasible SOC
260             X_lim(l,1) = X_opt(l-1,1) -...
261                 (((P_mot_lim(l,2)/eta_mot) + L_aux +...
262                 P_gen_lim(l,2))/(E_batt * 3600));  % Min feasible SOC
263         end
264
265         % NOTE: Here Max feasible SOC corresponds to operation with
266         % generator and Min is without generator. Because of Forward
267         % Propagation.
268
269         % Step 2a: Form SOC Query Range and form Δ SOC vector
270         temp = X_lim(l,1):X_PRCSN:X_lim(l,2);                    % SOC Range
271         X_range = zeros(size(temp))';
272         P_batt_range = zeros(size(temp))';
273         P_mot_range = zeros(size(temp))';
274         P_gen_range = zeros(size(temp))';
275         P_eng_range = zeros(size(temp))';
276         P_fuel_range = zeros(size(temp))';
277         ECvector_Wh_m = zeros(size(temp))';
278         X_range(:,1) = X_lim(l,1):X_PRCSN:X_lim(l,2);            % SOC Range
279
280         if HybModeGrp(l) == 1
281             P_batt_range(:,1) = (X_opt(l-1,1) - X_range) * (E_batt * 3600);
282             P_mot_range(:,1) = L_total(l);
283             P_gen_range(:,1) = P_batt_range(:,1) - L_aux -...
284                 (L_total(l)/eta_mot);
285             P_eng_range(:,1) = zeros(size(P_batt_range(:,1)));
286         elseif HybModeGrp(l) == 2
287             P_batt_range(:,1) = (X_opt(l-1,1) - X_range) * (E_batt * 3600);
288             P_mot_range(:,1) = (P_batt_range(:,1) - L_aux) * eta_mot;
289             P_gen_range(:,1) = zeros(size(P_batt_range(:,1)));
```

```
290             P_eng_range(:,1) = L_total(l) - P_mot_range(:,1);
291         elseif HybModeGrp(l) == -1
292             P_batt_range(:,1) = (X_opt(l-1,1) - X_range) * (E_batt * 3600);
293             P_mot_range(:,1) = P_regen_avail;
294             P_gen_range(:,1) = P_batt_range(:,1) - P_mot_range(:,1);
295             P_eng_range(:,1) = zeros(size(P_batt_range(:,1)));
296         end
297
298         len_x_rng = length(X_range);
299         fuelRate = zeros([len_x_rng 1]);
300
301         for k = 1:len_x_rng
302             if abs(P_gen_range(k,1)) + P_eng_range(k,1) > 3000
303                 [engEff,fuelRate(k,1)] =...
304                     calcEngFuelRate(abs(P_gen_range(k,1)/eta_genmot) +...
305                     P_eng_range(k,1));
306                 P_fuel_range(k,1) = (abs(P_gen_range(k,1)/eta_genmot) +...
307                     P_eng_range(k,1)) * (1/engEff);
308             else
309                 fuelRate(k,1) = 0;
310                 P_fuel_range(k,1) = 0;
311             end
312         end
313
314         % Step 2b: Determine feasible SOC values based on range estimation
315         ECvector_Wh_m(:,1) = (((X_opt(1,1) - X_range(:,1)) * E_batt) +...
316             (EC_fuel_opt(l,1) + (P_fuel_range(:,1)/3600)))/...
317            (DistTrvld_m(l,1) - DistTrvld_m(1,1));
318
319         % Step 3a: Compute control candidate values corresponding to all
320         % Δ SOC
321         for k = 1:len_x_rng
322             if ECvector_Wh_m(k,1) > 0
```

```
323                 estRngVec_m(k,1) = (((X_range(k,1) - SOC_Min) * E_batt)...
324                     + (MAX_FUEL - (FuelUsed_L(l,1) +...
325                     (fuelRate(k,1)/RHO_FUEL))) *...
326                     (LHV_FUEL * RHO_FUEL * J2WH_CONV))/ECvector_Wh_m(k,1);
327                 fsblRngVec(k,1) = estRngVec_m(k,1) -...
328                     (DistTrvld_m(end,1) - DistTrvld_m(l,1));
329             else
330                 estRngVec_m(k,1) = Inf;
331                 fsblRngVec(k,1) = estRngVec_m(k,1) -...
332                     (DistTrvld_m(end,1) - DistTrvld_m(l,1));
333             end
334             temp_P_batt_range = P_batt_range;
335             if (fsblRngVec(k,1) < minFsblRng) || FUEL_FLAG == 1 ||...
336                     (X_range(k,1) < SOC_Min)
337                 P_batt_range(k,1) = Inf;
338             end
339         end
340
341         % Step 3b: If all solution values result in infinity
342         if isinf(P_batt_range)
343             if FUEL_FLAG == 0
344                 P_batt_range(end) = temp_P_batt_range(end);
345             else
346                 P_batt_range(1) = temp_P_batt_range(l);
347             end
348         end
349
350         % Step 4: Calculate Cost-to-go for each possible path.
351         Y_x = (P_batt_range + abs(P_fuel_range))/3600;
352
353         % Step 5: Determine optimal path as per cost function.
354         opt_idx = find(Y_x == min(Y_x));
355
```

```
356         % 4200 W is minimum power engine map can resolve
357         if abs(P_gen_range(k,1)) + P_eng_range(k,1) > 100
358             % Step 6: Assign optimal SOC value and repeat.
359             P_gen_opt(l,1) = P_gen_range(opt_idx);
360             P_fuel_opt(l,1) = P_fuel_range(opt_idx);
361             P_eng_opt(l,1) = P_eng_range(opt_idx);
362             P_batt_opt(l,1) = P_batt_range(opt_idx);
363             P_mot_opt(l,1) = P_mot_range(opt_idx);
364             X_opt(l,1) = X_range(opt_idx);
365             fuelRate_opt(l,1) = fuelRate(opt_idx);
366         else
367             % Step 6: Assign optimal SOC value and repeat.
368             P_gen_opt(l,1) = P_gen_max;
369             P_fuel_opt(l,1) = P_fuel_range;
370             P_eng_opt(l,1) = P_eng_range;
371             P_batt_opt(l,1) = P_batt_range;
372             P_mot_opt(l,1) = P_mot_range;
373             X_opt(l,1) = X_range;
374             fuelRate_opt(l,1) = fuelRate;
375         end
376
377     elseif L_total(l) ≤ 0       % Regen-braking condition
378         % Step 1: Determine SOC Limits based on the Hybrid Mode
379         % EV and Series Mode
380         if HybModeGrp(l) == 1 && (X_opt(l-1,1) ≥ SOC_Final)
381             X_lim(l,2) = X_opt(l-1,1) -...
382                 (((L_total(l,1)/eta_mot) +...
383                 L_aux + P_gen_lim(l,1))/(E_batt * 3600));% Max feasible SOC
384             X_lim(l,1) = X_opt(l-1,1) -...
385                 (((L_total(l,1)/eta_mot) +...
386                 L_aux + P_gen_lim(l,2))/(E_batt * 3600));% Min feasible SOC
387         % Engine Only and Parallel Mode
388         elseif HybModeGrp(l) == 2
```

```
389             X_lim(l,2) = X_opt(l-1,1) -...
390                 (((P_mot_lim(l,1)/eta_mot) + L_aux)/...
391                 (E_batt * 3600));     % Max feasible SOC
392             X_lim(l,1) = X_opt(l-1,1) -...
393                 (((P_mot_lim(l,2)/eta_mot) + L_aux)/...
394                 (E_batt * 3600));     % Min feasible SOC
395             HybModeGrp(l) = 2;
396         elseif HybModeGrp(l) == -1      % Regen Braking Mode
397             X_lim(l,2) = X_opt(l-1,1) -...
398                 (((P_regen_avail/eta_mot) + L_aux +...
399                 max(P_gen_max,P_batt_ch_max -...
400                 (P_regen_avail/eta_mot)))/...
401                 (E_batt * 3600));       % Max feasible SOC
402             X_lim(l,1) = X_opt(l-1,1) -...
403                 (((P_mot_lim(l,2)/eta_mot) + L_aux + P_gen_lim(l,2))/...
404                 (E_batt * 3600));  % Min feasible SOC
405         end
406
407         % NOTE: Here Max feasible SOC corresponds to operation with
408         % generator and Min is without generator. Because of Forward
409         % Propagation.
410
411         % Step 2a: Form SOC Query Range and form ? SOC vector
412         temp = X_lim(l,1):X_PRCSN:X_lim(l,2);                    % SOC Range
413         X_range = zeros(size(temp))';
414         P_batt_range = zeros(size(temp))';
415         P_mot_range = zeros(size(temp))';
416         P_gen_range = zeros(size(temp))';
417         P_eng_range = zeros(size(temp))';
418         P_fuel_range = zeros(size(temp))';
419         ECvector_Wh_m = zeros(size(temp))';
420         X_range(:,1) = X_lim(l,1):X_PRCSN:X_lim(l,2);            % SOC Range
421
```

```
422         if HybModeGrp(l) == 1
423             P_batt_range(:,1) = (X_opt(l-1,1) - X_range) * (E_batt * 3600);
424             P_mot_range(:,1) = L_total(l);
425             P_gen_range(:,1) = P_batt_range(:,1) - L_aux -...
426                 (L_total(l)/eta_mot);
427             P_eng_range(:,1) = zeros(size(P_batt_range(:,1)));
428         elseif HybModeGrp(l) == 2
429             P_batt_range(:,1) = (X_opt(l-1,1) - X_range) * (E_batt * 3600);
430             P_mot_range(:,1) = (P_batt_range(:,1) - L_aux) * eta_mot;
431             P_gen_range(:,1) = zeros(size(P_batt_range(:,1)));
432             P_eng_range(:,1) = L_total(l) - P_mot_range(:,1);
433         elseif HybModeGrp(l) == -1
434             P_batt_range(:,1) = (X_opt(l-1,1) - X_range) * (E_batt * 3600);
435             P_mot_range(:,1) = P_regen_avail;
436             P_gen_range(:,1) = P_batt_range(:,1) - P_mot_range(:,1);
437             P_eng_range(:,1) = zeros(size(P_batt_range(:,1)));
438         end
439
440         len_x_rng = length(X_range);
441         fuelRate = zeros([len_x_rng 1]);
442
443         for k = 1:len_x_rng
444             if abs(P_gen_range(k,1)) + P_eng_range(k,1) > 3000
445                 [engEff,fuelRate(k,1)] =...
446                     calcEngFuelRate(abs(P_gen_range(k,1)/eta_genmot) +...
447                     P_eng_range(k,1));
448                 P_fuel_range(k,1) = (abs(P_gen_range(k,1)/eta_genmot) +...
449                     P_eng_range(k,1)) * (1/engEff);
450             else
451                 fuelRate(k,1) = 0;
452                 P_fuel_range(k,1) = 0;
453             end
454         end
```

```
455
456         % Step 2b: Determine feasible SOC values based on range estimation
457         % -- Old Version
458         ECvector_Wh_m(:,1) = (((X_opt(1,1) - X_range(:,1)) * E_batt) +...
459             (EC_fuel_opt(l,1) + (P_fuel_range(:,1)/3600)))/...
460            (DistTrvld_m(l,1) - DistTrvld_m(1,1));
461
462         % Step 3a: Compute control candidate values corresponding to all
463         % Δ SOC
464         for k = 1:len_x_rng
465             if ECvector_Wh_m(k,1) > 0
466                 estRngVec_m(k,1) = (((X_range(k,1) - SOC_Min) *...
467                     E_batt) + (MAX_FUEL - (FuelUsed_L(l,1) +...
468                     (fuelRate(k,1)/RHO_FUEL))) *...
469                     (LHV_FUEL * RHO_FUEL * J2WH_CONV))/ECvector_Wh_m(k,1);
470                 fsblRngVec(k,1) = estRngVec_m(k,1) -...
471                     (DistTrvld_m(end,1) - DistTrvld_m(l,1));
472             else
473                 estRngVec_m(k,1) = Inf;
474                 fsblRngVec(k,1) = estRngVec_m(k,1) -...
475                     (DistTrvld_m(end,1) - DistTrvld_m(l,1));
476             end
477
478             temp_P_batt_range = P_batt_range;
479             if (fsblRngVec(k,1) < minFsblRng) || FUEL_FLAG == 1 ||...
480                     (X_range(k,1) < SOC_Min)
481                 P_batt_range(k,1) = Inf;
482             end
483         end
484
485         % Step 3b: If all solution values result in infinity
486         if isinf(P_batt_range)
487             if FUEL_FLAG == 0
```

```
488                 P_batt_range(end) = temp_P_batt_range(end);
489             else
490                 P_batt_range(1) = temp_P_batt_range(l);
491             end
492         end
493 
494         % Step 4: Calculate Cost-to-go for each possible path.
495         Y_x = (P_batt_range + abs(P_fuel_range))/3600;
496 
497         % Step 5: Determine optimal path as per cost function.
498         opt_idx = find(Y_x == min(Y_x));
499 
500         % 4200 W is minimum power engine map can resolve
501         if abs(P_gen_range(k,1)) + P_eng_range(k,1) > 100
502             % Step 6: Assign optimal SOC value and repeat.
503             P_gen_opt(l,1) = P_gen_range(opt_idx);
504             P_fuel_opt(l,1) = P_fuel_range(opt_idx);
505             P_eng_opt(l,1) = P_eng_range(opt_idx);
506             P_batt_opt(l,1) = P_batt_range(opt_idx);
507             P_mot_opt(l,1) = P_mot_range(opt_idx);
508             X_opt(l,1) = X_range(opt_idx);
509             fuelRate_opt(l,1) = fuelRate(opt_idx);
510         else
511             % Step 6: Assign optimal SOC value and repeat.
512             P_gen_opt(l,1) = P_gen_max;
513             P_fuel_opt(l,1) = P_fuel_range;
514             P_eng_opt(l,1) = P_eng_range;
515             P_batt_opt(l,1) = P_batt_range;
516             P_mot_opt(l,1) = P_mot_range;
517             X_opt(l,1) = X_range;
518             fuelRate_opt(l,1) = fuelRate;
519         end
520     end
```

```
521
522     % Optimal solution addition
523     FuelUsed_L(l,1) = FuelUsed_L(l-1,1) + (fuelRate_opt(l,1) / RHO_FUEL);
524
525     % Fuel completion check
526     if (MAX_FUEL - FuelUsed_L(l,1)) ≥ 0
527         FUEL_FLAG = 0;
528     else
529         FUEL_FLAG = 1;
530     end
531
532     % Mileage computation and range estimation
533     EC_Wh_m(l,1) = (((X_opt(1) - X_opt(l,1)) * E_batt) +...
534             (FuelUsed_L(l,1) * RHO_FUEL * LHV_FUEL * J2WH_CONV))/...
535            (DistTrvld_m(l,1));
536
537     if EC_Wh_m(l,1) > 0
538         estRange_m(l,1) = (((X_opt(l,1) - SOC_Min) * E_batt) +...
539                     (MAX_FUEL - FuelUsed_L(l,1)) *...
540                     (LHV_FUEL * RHO_FUEL * J2WH_CONV))/EC_Wh_m(l,1);
541         fsblRange(l,1) = estRange_m(l,1) - (DistTrvld_m(end,1)...
542             - DistTrvld_m(l,1));
543     else
544         estRange_m(l,1) = Inf;
545         fsblRange(l,1) = estRange_m(l,1) - (DistTrvld_m(end,1)...
546             - DistTrvld_m(l,1));
547     end
548     %clear estRngVec_m fsblRngVec ECvector_Wh_m
549     l = l + 1;
550 end
551
552 if any(X_opt < SOC_Min)
553     disp('## Infeasible Solution due to SOC violation ##')
```

```
554     OPT_KEYWORD = 'UNSUCCESSFUL';
555     idx = find(X_opt < SOC_Min, 1);
556     NetEC_opt = ((sum(P_batt_opt(1:idx-1)) + sum(P_fuel_opt(1:idx-1)))...
557         /3600)/DistTrvld_m(idx-1);      % Wh/m
558 else
559     disp('## Optimization was successful for SOC ##')
560     OPT_KEYWORD = 'SUCCESSFUL';
561     NetEC_opt = ((sum(P_batt_opt) + sum(P_fuel_opt))/3600)...
562         /DistTrvld_m(end);              % Wh/m
563 end
564
565 % Range based Feasibility
566 if fsblRange(end) < minFsblRng
567     disp('## Infeasible solution due to range violation ##')
568     OPT_KEYWORD_2 = 'UNSUCCESSFUL';
569 else
570     disp('## Optimization was successful for Range ##')
571     OPT_KEYWORD_2 = 'SUCCESSFUL';
572 end
573
574 %% Output Data Structure
575 optResults.optSOC_pct = X_opt;
576 optResults.optPgen_W = P_gen_opt;
577 optResults.optPbatt_W = P_batt_opt;
578 optResults.optPfuel_W = P_fuel_opt;
579 optResults.optPeng_W = P_eng_opt;
580 optResults.optPmot_W = P_mot_opt;
581 optResults.optNetEC = NetEC_opt;
582 optResults.fsblRange_m = fsblRange;
583 optResults.estRange_m = estRange_m;
584 optResults.distTrvld_m = DistTrvld_m;
585 optResults.EC_Wh_m = EC_Wh_m;
586 optResults.Veh_Load_W = L_load;
```

```
587 optResults.FuelUsed_L = FuelUsed_L;
588 optResults.minFsblRng_m = minFsblRng;
589 optResults.HybModeGrp = HybModeGrp;
590 optResults.STATUS_KEY = OPT_KEYWORD;
591 optResults.STATUS_KEY_2 = OPT_KEYWORD_2;
592 end
```

Appendix B

Modeled System Parameters

The propulsion system component parameters, modeled for Series Hybrid Electric/ Extended Range Electric Chevrolet Camaro experimental research vehicle mentioned in Chapters 2 - 5 are provided in this Appendix section. Component parameters associated with Conventional, Series Parallel Hybrid Electric and Battery Electric long-haul semi-truck architectures mentioned in Chapter 6 are also provided. These parameters can be used to develop new models and provide necessary information for comparative studies. The component parameter information is provided in Tables B.1 - B.2.

Table B.1: Series Hybrid Electric Vehicle Power Loss Model Parameters

System	Parameter	Value	Units
Vehicle	Mass	2012	kg
	Road Load Coefficients: [a, b, c]	[23, 0.1, 0.002]	$[lbf, \frac{lbf}{mph}, \frac{lbf}{mph^2}]$
	Auxiliary Load	520	W
	Tire Radius	0.346	m
Motor	Peak Torque, Power	500, 200	Nm, kW
	Continuous Torque, Power	250, 91	Nm, kW
	Gearbox Ratio	4.2	-
	Efficiency [1]	read below	-
Battery	Nominal Voltage	340	V
	Minimum Capacity	18.9	kWh
	Peak Discharge, Charge Current	612, 300	A
	Continuous Discharge, Charge Current	180, 60	A
Engine	Peak Power	78	kW
	Available Peak Power	20	kW
	Efficiency [2]	read below	-
Generator	Peak Torque	200	Nm
	Continuous Torque	95	Nm
	Efficiency [3]	83-95	%
Fuel	Maximum Capacity	26.5	Liters
	Density	0.783	kg/L
	Lower Heating Value	30000	kJ/kg

[1] Part of the motor system combined efficiency map in Chapter 3.

Table B.2: Conventional Truck Power Loss Model Parameters

System	Parameter	Value	Units
Vehicle [4]	Mass	35,380	kg
	Frontal Area	10.4	m^2
	Average C_d	0.52	-
	Road Load Coefficient, [a, b, c]	[319.9, 3.2, 0.15]	$[lbf, \frac{lbf}{mph}, \frac{lbf}{mph^2}]$
	Auxiliary Load	3500	W
	Tire Radius	0.489	m
Engine	Peak Torque, Power	2100, 360	Nm, kW
	Efficiency [5]	read below	-
Transmission	Number of Gears	10	-
	Maximum, Minimum Ratio [5]	12.8, 0.73	-
	Efficiency[5]	read below	-
	Final Drive Ratio	2.64	-
Fuel	Maximum Capacity	1135.5	Liters
	Density	0.846	kg/L
	Lower Heating Value	42600	kJ/kg

[2]Part of the engine BSFC map in Chapter 3.

[3]Obtained from component datasheet.

[4]All trucks modeled in Chapter 6 use the same parameters. Accept for Battery Electric Truck for Auxiliary Load.

[5]Based on fuel map from EPA GEM v2 Simulator

Table B.3: Series Parallel Hybrid Electric Truck Power Loss Model Parameters

System	Parameter	Value	Units
Engine	Peak Torque, Power	2100, 360	Nm, kW
	Efficiency [5]	read below	-
Transmission	Number of Gears	10	-
	Maximum, Minimum Ratio [5]	12.8, 0.73	-
	Efficiency[5]	read below	-
	Final Drive Ratio	2.64	-
Fuel	Maximum Capacity	567.75	Liters
	Density	0.846	kg/L
	Lower Heating Value	42600	kJ/kg
Motor	Peak Torque, Power	3400, 350	Nm, kW
	Continuous Torque, Power	2380, 245	Nm, kW
	Efficiency[6]	95	%
Battery	Nominal Voltage	600	V
	Minimum Capacity	145.9	kWh
	Peak Discharge, Charge Current	2448, 1200	A
	Continuous Discharge, Charge Current	720, 240	A
Generator	Peak Torque	373	Nm
	Continuous Torque	191	Nm
	Efficiency[7]	read below	-

[6]Estimated from datasheet.

[7]Obtained from datasheet.

Table B.4: Battery Electric Truck Power Loss Model Parameters

System	**Parameter**	**Value**	**Units**
Vehicle	Auxiliary Load	2200	W
Motor	Peak Torque, Power	3400, 350	Nm, kW
	Continuous Torque, Power	2380, 245	Nm, kW
	Efficiency[8]	95	%
	Gearbox Ratio	5.2	-
Battery	Nominal Voltage	600	V
	Minimum Capacity	462.1	kWh
	Peak Discharge, Charge Current	7752, 3800	A
	Continuous Discharge, Charge Current	2280, 760	A

[8]Estimated from datasheet.