

Mission-based Design Space Exploration and Traffic-in-the-Loop Simulation for a  
Range-Extended Plug-in Hybrid Delivery Vehicle

Thesis

Presented in Partial Fulfillment of the Requirements for the Degree Master of Science in  
the Graduate School of The Ohio State University

By

Vijay Sankar Anil

Graduate Program in Mechanical Engineering

The Ohio State University

2020

Thesis Committee

Prof. Giorgio Rizzoni, Advisor

Prof. Qadeer Ahmed, Committee Member

© Copyright by  
Vijay Sankar Anil  
2020

## Abstract

With the on-going electrification and data-intelligence trends in logistics industries, enabled by the advances in powertrain electrification, and connected and autonomous vehicle technologies, the traditional ways vehicles are designed by engineering experience and sales data are to be updated with a *design for operation* notion that relies intensively on operational data collection and large scale simulations. In this work, this design for operation notion is revisited with a specific combination of optimization and control techniques that promises accurate results with relatively fast computational time. The specific application that is explored here is a Class 6 pick-up and delivery truck that is limited to a given driving mission. A Gaussian Process (GP) based statistical learning approach is used to refine the search for the most accurate, optimal designs. Five hybrid powertrain architectures are explored, and a set of Pareto-optimal designs are found for a specific driving mission that represents the variations in a hypothetical operational scenario. A cross-architecture performance and cost comparison is performed and the selected architecture is developed further in the form of a forward simulator with a dedicated ECMS controller. In the end, a traffic-in-the-loop simulation is performed by integrating the selected powertrain architecture with a SUMO traffic simulator to evaluate the performance of the developed controller against varying driving conditions.

*Dedicated to a sustainable and peaceful future for our planet.*

## **Acknowledgments**

First and foremost, I wish to acknowledge Prof. Giorgio Rizzoni, Director, Center for Automotive Research, OSU, and my graduate advisor for his trust and inspired guidance throughout this project, and for his admirable work ethic I tried to emulate over the past two years. A word of gratitude to Research Associate Prof. Qadeer Ahmed for being a guiding beacon through the course of this project. In equal measure, I express my gratefulness to my diligent research partner and PhD student, Tong Zhao, who has worked alongside me with the utmost patience, while imparting vital knowledge in the domains of controls and systems engineering.

I am indebted to Associate Professor Marcello Canova for his insightful courses in system dynamics, modeling and control, along with those by my advisor, which taught me the essence of automotive powertrain modeling and control in the most elegant of ways. Many thanks to the project sponsors, Department of Energy (DOE), USA, technical partners, Cummins Inc. and FCCC, and to the entire CERC TRUCK team.

None of this could have been achieved without the unconditional love and decades-long sacrifice of my family, who keep inspiring me to dream bigger and follow my ambitions. Thanks to my friends at OSU EcoCAR, the technical, infrastructural and financial support from my esteemed colleagues at the Center for Automotive Research, and the Department

of Mechanical and Aerospace Engineering, OSU. The success of my graduate research is down to the highly stimulating research environment rendered by CAR and all its members.

My budding interest in engineering research was nurtured by my undergraduate advisor Prof. M S Dasgupta and research guide Prof. Tufan Chandra Bera at BITS Pilani. I wish to express my heartfelt appreciation for their mentorship and encouragement throughout my formative days. In doing so, I must also mark my affection and gratefulness to the Inspired Karters Racing Baja team of BITS Pilani, which was a true test of fire during my undergraduate days, cementing my passion for motorsports, and finally culminating in my research focus in powertrain electrification. Thanks to motorsports. General Motors and Mahindra offered me my first steps into the fast-paced R&D environment of automotive product development, and I am beholden to them for the valuable exposure and many professional skills, which have served me exceedingly well in every step since.

Finally, this work is built on the unfailing kindness of my kindred soul, whose constant love and creativity helped keep me sane through thick and thin.

## Vita

2012-2016.....	B.E. Mechanical Engineering Birla Institute of Technology and Science, Pilani Rajasthan, India
2016-2017.....	Graduate Engineer Trainee General Motors Bangalore, India
2017-2018.....	Engine Calibration Engineer Mahindra Research Valley Chennai, India
2019-present.....	Graduate Research Associate Center for Automotive Research The Ohio State University Columbus, Ohio

## Publications

V. Anil, T. Zhao, Q. Ahmed and G. Rizzoni “Design Space Exploration for a Series Plug-in Hybrid Pickup and Delivery Truck Using Gaussian Process Optimization” *SAE International Journal of Commercial Vehicles*, September 1, 2020 (accepted)

V. Anil, A. Pathak and S. Anil “Electric Autonomous Shuttles as a Sustainable Solution to the Last Mile Problem: A Leapfrog Opportunity for Mahindra World City” *FISITA World Automotive Congress*, October 2018, Chennai, India

## Fields of Study

Major Field: Mechanical Engineering

Graduate Specialization: Automotive Systems and Mobility

## Table of Contents

Abstract .....	ii
Dedication .....	iii
Acknowledgments .....	iv
Vita .....	vi
List of Tables.....	x
List of Figures .....	xii
List of Acronyms.....	xvi
Chapter 1. Introduction to Powertrain Electrification .....	1
1.1 Motivation for Powertrain Electrification .....	1
1.2 Markets and Opportunities for US and China Electrified Powertrains .....	5
1.2.1 CERC TRUCK Research .....	8
1.2.2 Classification .....	11
1.2.3 United States Market Study.....	13
1.2.4 China Market Study .....	19
1.3 Range-extended HEVs for Commercial Vehicle Applications .....	28
1.4 Hybrid Electric Vehicle Optimal Energy Management .....	30
1.4.1 Global Optimal Energy Management Strategy: DP .....	32
1.4.2 Online Energy Management Strategies .....	36
1.4.3 Adaptive Optimal Energy Management Strategies .....	45
1.4.4 Model Predictive Control (MPC) .....	51
Chapter 2. Design Space Exploration.....	54
2.1 Design Space Exploration for Delivery Trucks .....	56
2.2 Framework, Methodology and Problem Formulation .....	58
2.2.1 Vehicle Road Load Model .....	60
2.2.2 Powertrain Dynamics Model.....	61
2.2.3 Library of Components .....	69
2.2.4 Design Space Exploration .....	70
2.2.5 Search Algorithms Used in DSE .....	74

2.2.6 Optimal Energy Management Strategy .....	83
2.3 Simulation Results and Problem Solution .....	85
2.3.1 Duty Cycles for the Delivery Truck .....	85
2.3.2 Constraints in the Optimization Process .....	87
2.3.3 Conventional Powertrain.....	91
2.3.4 Hybrid Powertrains .....	94
2.3.5 Cross-Architecture Performance and Cost Evaluation .....	122
2.3.6 Optimal Architecture and Conclusions .....	129
2.4 Summary of Results .....	130
Chapter 3. Performance Metrics to Enhance Power Management in HEVs .....	132
3.1 Performance Metrics for Delivery Trucks.....	132
3.1.1 Fuel Economy.....	134
3.1.2 Engine Start/Stop Events.....	136
3.1.3 Emissions .....	139
3.1.4 Introduction to the Forward Simulator .....	141
3.1.5 Model Setup with MATLAB .....	143
3.1.6 Powertrain Controller: A-ECMS .....	146
3.1.7 Cross-validation between Forward and Backward Models .....	153
3.2 Literature Review: Look-Ahead Information in Delivery Trucks .....	158
3.3 Look-Ahead Energy Management .....	162
3.3.1 Feasible Look-Ahead Technologies .....	164
3.3.2 Quantifying the Benefits of Look-Ahead Information .....	170
3.3.3 Incorporating Look-Ahead Information in the Control Problem ..	175
3.4 Summary of Results .....	182
Chapter 4. Traffic-in-the-Loop Simulation for Controller Evaluation .....	184
4.1 Introduction to Traffic-in-the-Loop Simulation.....	184
4.2 Literature Review: Traffic Simulators for Controller Evaluation.....	189
4.3 Cross-Platform Simulation with Urban Drive Cycles.....	190
4.4 SUMO – Simulation of Urban Mobility.....	193
4.4.1 Forward Simulator .....	196
4.4.2 Open Street Maps for Route Generation.....	196
4.4.3 Inserting the Host Vehicle into the Co-simulation .....	200
4.4.4 Intelligent Driver Model (IDM) for Velocity Profile Generation ..	201
4.4.5 Co-simulation Framework and Sensor Emulation.....	204
4.5 Simulation Results and Discussion .....	211
4.5.1 Drive Cycle Characterization .....	211
4.5.2 Analysis of the ECMS Controller Performance .....	212

4.5.3 Effect on Trip Fuel Economy and Start-Stops .....	212
4.5.4 Effect on Trip Emissions.....	216
4.5.5 Advantage of Traffic Co-simulation over Deterministic Cycles....	216
4.6 Summary of Results .....	222
Chapter 5. Conclusion and Future Scope.....	223
5.1 Summary of the Study .....	223
5.2 Future Scope .....	225
Bibliography.....	227
Appendix A. Further Reading .....	241
Appendix B. Contact Information .....	242

## List of Tables

Table 1: Vehicle weight classes in China .....	11
Table 2: Comparison of vehicle weight ratings in the US and China .....	15
Table 3: Specifications of Cummins EDI PowerDrive 6000.....	18
Table 4: Specifications of TEVVA MOTORS and Cummins ETREE.....	19
Table 5: Specifications of the Chanje V8070 .....	20
Table 6: Specifications of the BYD 6D.....	21
Table 7: Specifications of the BYD Class 6 Truck .....	22
Table 8: Specifications of the BYD Class 5 Truck .....	23
Table 9: Specifications of FAW Jiefang J6F direct drive truck.....	24
Table 10: Specifications of BYD T7 Direct BEV Truck.....	25
Table 11: Specifications of Daimler Atego BlueTec Hybrid.....	26
Table 12: Specifications of HINO 195h diesel-electric hybrid system .....	27
Table 13: Default vehicle parameters.....	61
Table 14: Genset class definition .....	63
Table 15: Electric machine class definition .....	66
Table 16: Battery pack parameters.....	66
Table 17: Variables required to setup the dpm function.....	69
Table 18: Specification for the vehicle parameters and range for components .....	70
Table 19: Design space variables, range and discretization .....	73
Table 20: Driving cycle characteristics .....	86
Table 21: Weight and cost of the REEV powertrain:.....	87
Table 22: Calculation of the cost and weight of the REEV powertrain: .....	88
Table 23: Powertrain performance benchmark.....	89
Table 24: Gradeability benchmarks for the powertrain.....	90
Table 25: Conventional powertrain specification range.....	92
Table 26: Conventional powertrain control parameters and cost function.....	92
Table 27: Conventional IC engine powertrain specifications.....	94
Table 28: Two-speed e-Axle powertrain parameters .....	95
Table 29: Two-speed e-Axle powertrain control parameters and cost function .....	95
Table 30: Two-speed e-axle – best candidate; genset penalty analysis.....	98
Table 31: Electric machine operating time against efficiency .....	99
Table 32: Three-speed AMT+EM powertrain parameters .....	101
Table 33: Three-speed AMT+EM powertrain control parameters and cost function .....	102
Table 34: Electric machine operating time against efficiency .....	106
Table 35: Four-speed AMT+EM powertrain parameters .....	108

Table 36: Four-speed AMT+EM powertrain control parameters and cost function.....	108
Table 37: Electric machine operating time against efficiency .....	112
Table 38: Direct drive powertrain parameters .....	114
Table 39: Direct drive powertrain control parameters and cost function .....	114
Table 40: Electric machine operating time against efficiency .....	116
Table 41: Dual motor powertrain parameters .....	118
Table 42: Dual motor powertrain control parameters and cost function.....	118
Table 43: Electric machine operating time against efficiency .....	121
Table 44: Comparison of fuel economy results between DP and ECMS.....	155
Table 45: Comparison of the benchmark ECMS controller against adaptive ECMS .....	179
Table 46: Intelligent Driver Model (IDM) parameters.....	203
Table 47: TraCI commands to extract sensor information .....	210
Table 48: Statistical data: dependence of powertrain performance on IDM behavior....	213

## List of Figures

Figure 1: Comparative life-cycle GHG emissions of a mid-size global average car by powertrain [2].....	4
Figure 2: Projected CO2 emissions – heavy-duty vehicles in the Sustainable Development Scenario [7] .....	6
Figure 3: IEA analysis based on country submissions [2].....	10
Figure 4: Passenger vehicles of category M1, M2 and M3 respectively (data from: Chinese Classification of power-driven vehicles and trailers, GB/T 15089-2001) .....	12
Figure 5: Freight vehicles of category N1, N2 and N3 respectively (data from: Chinese Classification of power-driven vehicles and trailers, GB/T 15089-2001).....	12
Figure 6: Load capacity: Electrified vehicles (Chinese Ministry of Transportation).....	13
Figure 7: Weight class of vehicles in the US (data from: US Department of Energy, <a href="http://www.afdc.energy.gov/data/">http://www.afdc.energy.gov/data/</a> ).....	14
Figure 8: A Class 6 delivery truck (image: US Department of Energy) .....	15
Figure 9: Class 3-8 truck sales in the US from 2001-2018 (US Department of Energy; <a href="http://www.afdc.energy.gov/data/">http://www.afdc.energy.gov/data/</a> ).....	16
Figure 10: Transportation energy use by mode and fuel type in the US in 2018 (US Department of Energy; <a href="http://www.afdc.energy.gov/data/">http://www.afdc.energy.gov/data/</a> ) .....	17
Figure 11: The Kenworth T370 Utility Truck (images from Cummins Inc.).....	18
Figure 12: Cummins ETREE and TEVVA MOTORS (images from Cummins Inc.) .....	19
Figure 13: Chanje V8070 electrified truck (images from Chanje product portfolio) .....	20
Figure 14: BYD 6D Class 6 EV Step Van (image from BYD Auto).....	22
Figure 15: BYD Class 6 Truck (image from BYD Auto) .....	22
Figure 16: BYD Class 5 Truck (image from BYD Auto) .....	23
Figure 17: FAW Jiefang J6F Direct Drive Truck (images from FAW Jiefang) .....	24
Figure 18: BYD T7 Direct BEV Truck (images from BYD Auto).....	25
Figure 19: Daimler Atego BlueTec Hybrid (images from Daimler).....	26
Figure 20: HINO 195h diesel-electric hybrid system (images from HINO Trucks).....	27
Figure 21: Range-extended Hybrid Electric Vehicle Architecture .....	28
Figure 22: Objectives of an Energy Management Strategy [10].....	31
Figure 23: Dynamic Programming schematic.....	33
Figure 24: Classification of Power Management Strategies [15] .....	37
Figure 25: Schematic of the Pontryagin’s Minimum Principle [25].....	40
Figure 26: Open-loop PMP-based energy management control scheme [25] .....	40
Figure 27: Energy path during discharge (a) and charge (b) in a parallel HEV [25] .....	42
Figure 28: Schematic of adaptive energy management strategies [25] .....	47
Figure 29: Components of Design Space Exploration .....	57
Figure 30: A holistic schematic of DSE .....	58

Figure 31: REEV architecture with representation of energy flow.....	59
Figure 32: Original and scaled engine efficiency maps .....	62
Figure 33: Scaled engine optimal operating line (OOL) .....	63
Figure 34: The original motor map (above) and the scaled-up map (below) .....	64
Figure 35: Battery internal resistance maps.....	67
Figure 36: Backward powertrain model for a conventional vehicle .....	68
Figure 37: The progressive list of stages and outcomes of Design Space Exploration.....	71
Figure 38: Design and operational objectives of Hybrid Electric Vehicles [15].....	72
Figure 39: Arriving at the optimal candidate using the search algorithm in [44] .....	82
Figure 40: Backward simulator representation for a hybrid vehicle .....	84
Figure 41: The selected pick-up and delivery drive cycle .....	86
Figure 42: Electric motor map with operating points for the selected drive cycle .....	87
Figure 43: Preliminary matching for the axle reduction ratios with speed and grade.....	91
Figure 44: Conventional powertrain architecture.....	92
Figure 45: Engine efficiency and fuel consumption for the conventional powertrain .....	93
Figure 46: Fuel consumption for the conventional powertrain.....	93
Figure 47: Two-speed e-Axle powertrain architecture.....	94
Figure 48: Pareto front generation.....	95
Figure 49: Pareto front generation for two-speed e-axle candidates .....	97
Figure 50: Optimal candidate selection for two-speed e-axle candidates .....	98
Figure 51: EM optimal operating points and gear shifts.....	100
Figure 52: Genset power, engine state, battery power for the speed profile .....	100
Figure 53: Three-speed AMT+EM powertrain architecture .....	101
Figure 54: Pareto front generation.....	102
Figure 55: Candidate feasibility study for 3-speed AMT+EM.....	104
Figure 56: Optimal component sizes .....	104
Figure 57: EM optimal operating points and gear shifts .....	106
Figure 58: Genset power, engine state, battery power for the speed profile .....	107
Figure 59: Four-speed AMT+EM architecture .....	108
Figure 60: Generation of pareto-optimal candidates .....	109
Figure 61: Four-speed AMT+EM Pareto front .....	110
Figure 62: Optimal component sizes .....	111
Figure 63: EM optimal operating points and gear shifts .....	112
Figure 64: Genset power, engine state, battery power for the speed profile .....	113
Figure 65: Direct drive powertrain architecture.....	114
Figure 66: Direct drive pareto front.....	115
Figure 67: EM optimal operating points.....	116
Figure 68: Genset power, engine state, battery power for the speed profile .....	117
Figure 69: Dual motor powertrain architecture.....	118
Figure 70: Dual motor pareto front .....	119
Figure 71: Dual motor architecture component sizes.....	120
Figure 72: Dual motor operating maps.....	121
Figure 73: Genset power, engine state, battery power for the speed profile .....	122
Figure 74: Comparison of powertrain cost vs weight vs freight ton efficiency.....	124

Figure 75: Comparison of powertrain cost vs weight vs freight ton efficiency.....	124
Figure 76: Comparison against the conventional powertrain .....	125
Figure 77: Powertrain cost comparison over the lifetime of operation .....	125
Figure 78: Freight ton efficiency comparison.....	127
Figure 79: Motor size comparison.....	128
Figure 80: Powertrain weight comparison.....	129
Figure 81: Powertrain cost comparison .....	129
Figure 82: Forward simulator for the selected 2-speed e-axle architecture.....	144
Figure 83: Powertrain module inside the forward simulator .....	144
Figure 84: Object oriented design of the simulator environment .....	145
Figure 85: Sample results from the forward simulator with an ECMS controller .....	153
Figure 86: Sample results from the backward simulator using Dynamic Programming.....	154
Figure 87: Powertrain states observed during the P&D drive cycle .....	154
Figure 88: Sankey diagram showing the energy flow in the forward simulator.....	156
Figure 89: Energy split at component level for forward and DP simulations – the comparison is performed at four levels of driver behavior matching .....	157
Figure 90: Sankey diagram with normalized comparison of energy flow between the forward simulator (black) and DP (red); Note: the driver inputs are matched between the two simulations in order to equate the power demand from the duty cycle .....	157
Figure 91: Illustration of a look-ahead control methodology [61].....	163
Figure 92: Determination of the horizon for predictive control based on various inputs.....	163
Figure 93: Driving profile estimation from different levels of look-ahead data [96] .....	166
Figure 94: SOC profile and fuel usage resulting from the baseline ECMS controller....	174
Figure 95: Distance-based reference SOC profile for the ECMS controller .....	174
Figure 96: Simulink block with the Adaption Method for the Equivalence Factor .....	176
Figure 97: Velocity prediction block using the CERC P&D drive cycle.....	177
Figure 98: Reference SOC profile generation.....	178
Figure 99: SOC trajectory, fuel consumption and s-factor adaption with A-ECMS .....	179
Figure 100: SOC trajectory, fuel consumption and equivalence factor adaption with the A-ECMS controller (horizon length: 20 seconds).....	180
Figure 101: SOC trajectory, fuel consumption and equivalence factor adaption with the A-ECMS controller (horizon length: 300 seconds).....	180
Figure 102: A-ECMS controller (initial equivalence factor = 3).....	181
Figure 103: SOC trajectory, fuel consumption and equivalence factor adaption with the A-ECMS controller (initial equivalence factor = 1).....	182
Figure 104: Cross-platform simulation.....	192
Figure 105: Vehicle mobility simulation process in SUMO .....	195
Figure 106: Examples of abstract road networks as built using “netgenerate”; from left to right: grid (“Manhattan”, spider, and random network) .....	198
Figure 107: A sample route for CERC P&D application on an OSM map .....	199
Figure 108: SUMO framework – a high level overview .....	200
Figure 109: Simulink implementation of the Intelligent Driver Model (IMD) which is the interface for information exchange between MATLAB and SUMO .....	202
Figure 110: Paradigm of the Intelligent Driver Model (IDM) within SUMO .....	203

Figure 111: Flowchart depicting the SUMO-MATLAB integration via the IDM [96] ..	204
Figure 112: Information flow between Simulink and SUMO .....	206
Figure 113: Information flow in closed-loop simulation.....	207
Figure 114: Bird's-eye view of the simulation framework.....	209
Figure 115: The complete layout of the powertrain-traffic co-simulation environment.	211
Figure 116: Impact of driver behavior on fuel economy and engine start-stops .....	213
Figure 117: Statistical data: dependence of powertrain efficiency on IDM behavior.....	214
Figure 118: Statistical data on the dependence of genset start-stops on IDM behavior..	215
Figure 119: Energy dynamics in the generator set system .....	217
Figure 120: Thermal dynamics of the two-way catalyst .....	217
Figure 121: Temperature dynamics of the catalyst based on the exhaust gas flow .....	219
Figure 122: Study of the catalyst temperature dynamics over the drive cycle .....	220

## List of Acronyms

A-ECMS Adaptive ECMS  
AMT Automated Manual Transmission  
APP Accelerator Pedal Position  
BPP Brake Pedal Position  
CAR Center for Automotive Research  
CERC Clean Energy Research Center  
DOE Department of Energy  
DP Dynamic Programming  
DSE Design Space Exploration  
ECMS Equivalent Consumption Minimization Strategy  
EM Electric Machine  
EMS Energy Management Strategy  
EPA Environmental Protection Agency  
EV Electric Vehicle  
GP Gaussian Process  
GPS Global Positioning System  
HEV Hybrid Electric Vehicle  
ICE Internal Combustion Engine  
MD/HD Medium-duty/Heavy-duty  
MPC Model Predictive Control  
OOL Optimal Operation Line  
PHEV Plug-in Hybrid Electric Vehicle  
PID Proportional Integral Derivative controller  
PMP Pontryagin's Minimum Principle  
P&D Pick-up and Delivery  
REEV Range Extender Electric Vehicle  
SOC State of Charge  
SUMO Simulation of Urban Mobility  
TIL Traffic-in-the-loop Simulation  
V2I Vehicle to Infrastructure communication  
V2V Vehicle to Vehicle communication  
V2X Vehicle to X communication

## **Chapter 1. Introduction to Powertrain Electrification**

The aim of the research presented in this thesis is to: First, formulate a suitable design space exploration search scheme that is capable of using realistic drive cycle data and a collection of powertrain component information to arrive at an optimal architecture for a range-extended plugin hybrid electric delivery truck. Second, design an online implementable energy management strategy to maximize the selected performance metrics of the vehicle over a driving mission; and finally, to test the robustness of the controller with an integrated powertrain-cum-traffic-in-the-loop simulation environment. This entire exercise will culminate in conclusions that will aid in making an informed decision about the final implementation of the specified real-world product.

### **1.1 Motivation for Powertrain Electrification**

Hybrid electric vehicles (HEVs) are essential to meet the targets imposed by recent regulations mandating reduced fuel consumption and emissions in the automotive industry. The motivation for this thesis comes from the US-China Clean Energy Research Center program (CERC TRUCK), which investigates technologies with the potential to reduce dependence of commercial trucks on oil and reduce GHG emissions. The aim is to demonstrate an overarching goal of a 50% freight ton efficiency improvement for a pickup and delivery truck over a 2016 baseline conventional vehicle [1].

Electrified vehicle powertrains, viz. HEVs and EVs are relevant alternatives to the vast majority of the current market offerings with conventional, internal combustion engine (ICE) powered vehicles. In addition to the ICE, hybrid powertrains consist of electric motors and power electronics as well as energy storage systems such as batteries and supercapacitors, depending on the architecture and application. The efficient operation of such an electrified powertrain requires careful selection of the architecture, component sizes and energy management strategy. According to the 2019 annual report published by Global EV Outlook, these environmentally friendly alternatives to IC engine vehicles grew to represent 4% of the global vehicle market share from 2013 to 2018 and are expected to go beyond 40% by 2030 [2].

The same research points out that the number of charging points worldwide was estimated to be approximately 5.2 million at the end of 2018, growing by 44% from the previous year. Most of this increase was in private charging points, accounting for more than 90% of the 1.6 million installations last year. It is indispensable that a commensurate growth in the infrastructure for electrified powertrain happens with a growth in their market share in order to achieve desired lifecycle emissions reduction while meeting commercial targets.

On a well-to-wheel basis, greenhouse gas (GHG) projected emissions from HEVs and EVs will continue to be lower than for conventional internal combustion engine (ICE) vehicles. According to the proposed New Policies Scenario, GHG emissions of the electrified fleets will reach almost 230 million tons of carbon-dioxide equivalent (metric ton CO<sub>2</sub>-eq) in 2030, offsetting about 220 metric ton CO<sub>2</sub>-eq emissions. The International

Energy Agency proposes that the assumed trajectory for power grid decarbonization is consistent its Sustainable Development Scenario and further strengthens GHG emission savings from HEVs and EVs.

It is widely understood that the true emissions reduction that can be achieved from powertrain electrification needs a holistic analysis of the well-to-wheel energy flow. Whether or not HEVs and EVs deliver net benefits in terms of GHG emissions savings ultimately depends on the emissions that occur throughout the entire value chain, i.e. over the life cycle compared with other options. Going by the current global average carbon intensity (518 grams of carbon-dioxide equivalent per kilowatt-hour [g CO<sub>2</sub>-eq/kWh]) [2], an average passenger EV and plug-in hybrid electric car (PHEV) emits less GHGs than a global average ICE vehicle using gasoline over their life cycle. This is, however a global outlook, and the extent of emissions reduction ultimately depends on the power source: CO<sub>2</sub> emissions savings are significantly higher for electric cars used in countries where the power generation demographic is dominated by renewable sources. In countries where the power generation mix is dominated by coal – such as India, with two-thirds of its electricity coming from fossil fuels – it has been found that hybrid vehicles exhibit lower emissions than EVs. The extent of the impact differs depending on the size of the ICE vehicle.

In the freight transport segment around the world, electrified vehicles were mostly deployed as light-commercial vehicles (LCVs), which reached 250,000 units in 2018, while medium electric truck sales were in the range of 1000-2000 in 2018. In addition to this, the International Energy Agency reports that the global EV stock in 2018 was served

by 5.2 million light-duty vehicle chargers, (540,000 of which are publicly accessible), along with 157,000 fast chargers for buses.

Figure 1 sourced from IEA 2019 report depicts the comparative life-cycle GHG emissions of a mid-size global average car by powertrain in 2018. The data suggests that the GHG emissions of BEVs using electricity characterized by the aforementioned global average carbon intensity are similar to those of fuel cell electric vehicles (FCEVs) using hydrogen generated from steam methane reforming and to those of HEVs using gasoline. On average, the capacity of BEV cars to deliver net GHG emission savings in comparison with PHEV cars depends on the size of the battery pack. In the large vehicle segment, EVs save more GHG emissions compared to ICE vehicles having similar characteristics. This is due to the higher fuel economy penalty from the heavier ICE vehicles in comparison with EVs.

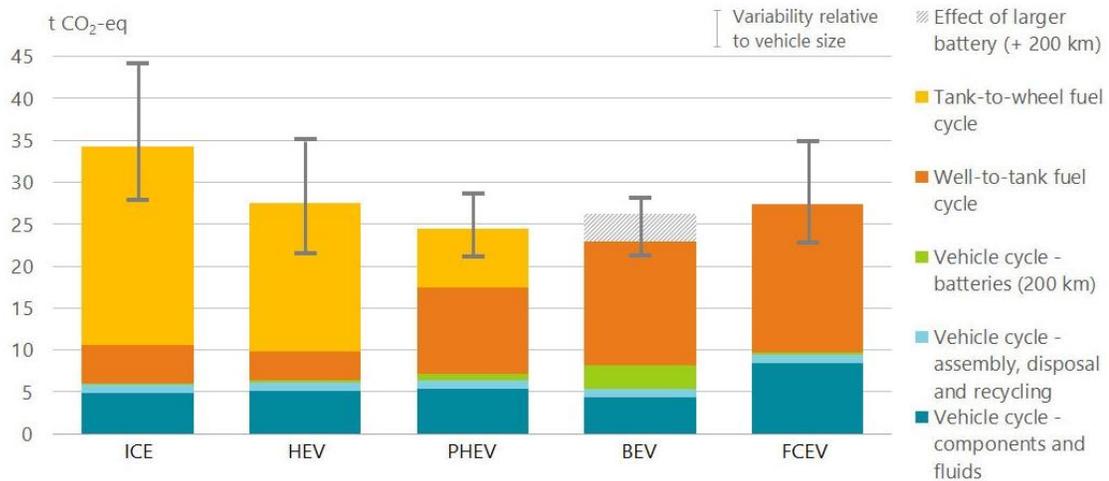


Figure 1: Comparative life-cycle GHG emissions of a mid-size global average car by powertrain [2]

**Notes:** The BEV refers to a vehicle with 200 km range, the addition of the shaded area refers to a vehicle with 400 km range. The ranges suggested by the sensitivity bars represent the case of small cars (lower bound) and of large cars (upper bound) – for BEVs, the lower bound of the sensitivity bar represents a small car with a 200 km range, and the upper bound represents a large car with a 400 km range.

## **1.2 Markets and Opportunities for US and China Electrified Powertrains**

Since the year 2000, data from IEA (Figure 2) shows that global emissions from trucks and buses (heavy-duty vehicles) have annually risen at a rate of 2.2%. With the pressing need to curtail CO<sub>2</sub> emissions, more countries around the world must adopt policies and standards, and existing ones must become more comprehensive and stringent. This applies to passenger as well as commercial vehicles. In urban settings, rapid electrification – especially of buses, but increasingly of light commercial and medium-duty trucks – will help in staying abreast of emissions goals. US and China being the two biggest economies in the world today, the impact of electrification of light commercial and medium-duty trucks, together with coordinated efforts by multiple stakeholders to improve logistics and operational efficiency will be highly beneficial.

The rapid growth of commerce in China and strong demand for delivery vehicles in the US driven by a steadily growing economy and increased demand for goods translates into more delivery and more trucking activity.

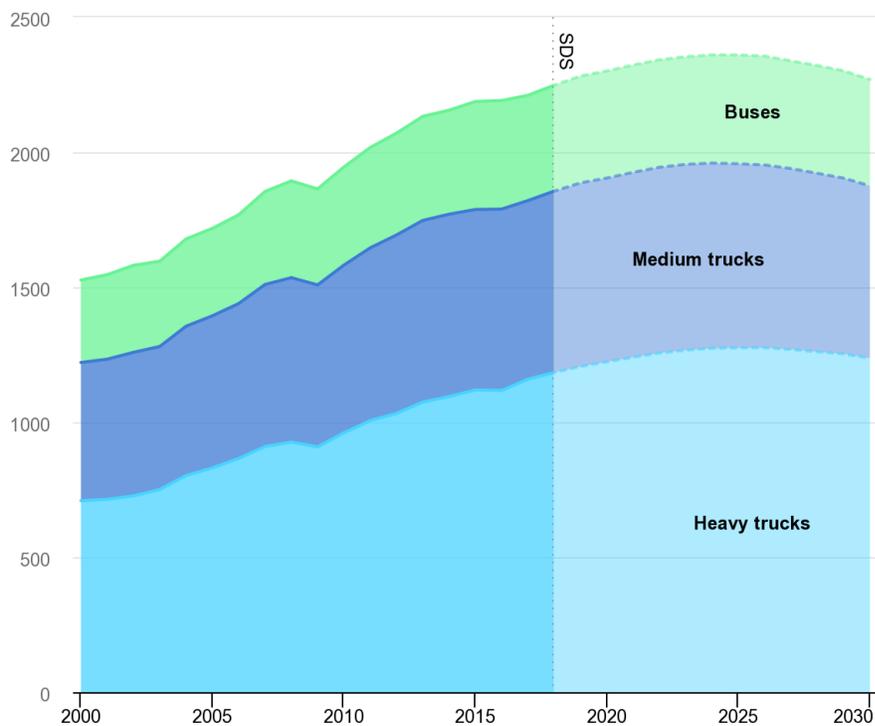


Figure 2: Projected CO2 emissions – heavy-duty vehicles in the Sustainable Development Scenario [7]

Building upon recent momentum, rapid adoption of battery electric buses and trucks in these cities will not only reduce energy consumption but also cut local pollutant and CO<sub>2</sub> emissions. In addition to that, medium-duty trucks (MDV) with regional and long-haul missions will need to transition to low-carbon alternative fuels and efficient powertrains. Options include hybridization and hydrogen fuel-cell electric vehicles, advanced biofuels and electro-fuels [2].

In terms of market adoption in US and China, urban buses are one of the great success stories of rapid electrified vehicle market uptake. Taking advantage of the suitability of

their fixed routes and schedules, their frequent start-stops and cities' targets to reduce local air pollution, a market for electric buses has emerged quite rapidly. The next phase of growth in urban electrification of transport pertains to pickup and delivery vehicles with similar driving missions to those of urban road transport.

China has led the way for the world in injecting locally made electrified buses to their market, first to Chinese cities – Shenzhen became the global model [3] when it transitioned entirely to electric city buses within only a few years [4] – and increasingly to Europe and North America. Commercial trucks operating in urban environments may be the next to heavily electrify. The consensus is that long-term payback of initial capital investments in electrified medium-duty trucks, especially those belonging to large, well-coordinated fleets and logistics services, may incite these fleets to electrify even more quickly than private passenger cars to push for sustainability.

City-level efforts to contain air pollution, including targets for phasing out diesel and internal combustion engines, together with corporate efforts to anticipate and take the lead on pressing public issues, will further spur HEV and EV adoption for light-duty commercial fleets. For a GVW of less than 16 tons, an increasingly wide selection of electrified trucks is reaching the market. In fact, major postal and package delivery companies, including DHL, UPS and FedEx, are expanding their fleets, and the Swiss and Austrian postal services have pledged to transition to all-electric fleets by 2030 or earlier. Meanwhile, momentum continues in the demonstration and commercialization of zero-

emission trucks – the majority of which are also electric, as cited by the International Council on Clean Transportation [5].

As battery technology improves steadily along with cheaper price per kWh of Li-ion batteries, the weight and range thresholds for electric vehicles continue to rise. For trucks operating on regional delivery segments, the suitability of electrification will depend upon continuing energy density improvements and cost reductions in lithium-based batteries. In addition to leading in electrifying buses, light commercial vehicles, and even medium-duty trucks, China has introduced fleets of hydrogen fuel-cell electric trucks and buses, most of which operate on reliable routes and refuel centrally at a single station [6]. In this way, China has demonstrated that they have moved ahead of the rest of the world into uncharted territory.

### **1.2.1 CERC TRUCK Research**

As mentioned in section 1.1, the motivation for this thesis comes from the US-China Clean Energy Research Center’s Truck Research Utilizing Collaborative Knowledge program (CERC TRUCK), which investigates technologies with the potential to reduce dependence of commercial trucks on oil and reduce GHG emissions. In November 2009, the United States and China established the bilateral Clean Energy Research Center (CERC), the primary purpose of which is to facilitate joint research and development on clean energy technology by teams of scientists and engineers from both countries [1].

As the world's top energy consumers, energy producers and greenhouse gas emitters, the United States and China will play leading roles in the world's transition to a clean energy economy in the years ahead. Medium- and heavy-duty truck transport systems account for a large share of petroleum-based fuel consumption in both countries. With the Chinese and the U.S. markets representing the two biggest markets for commercial vehicles in the world, one country alone cannot achieve global goals for the reduction of petroleum consumption and greenhouse gas (GHG) emissions.

The CERC Truck Research Utilizing Collaborative Knowledge Consortium will:

- Contribute to dramatic improvements in technologies with the potential to reduce dependence of commercial trucks on oil and reduce GHG emissions,
- Build a foundation of knowledge, technologies, human capabilities, and relationships in mutually beneficial areas that will position the United States and China for a future with highly efficient clean commercial trucks that have very low environmental impacts,
- Leverage the complementary strengths of each country's intellectual and research capacities, and
- Accelerate the advancement of technologies for clean trucks through joint research and development.

China is the largest market for electric vehicle sales worldwide today, followed by Europe and the United States. Figure 3 obtained from the IEA Global EV Outlook 2019 report, the global electric car sales and market share from the years 2013-18 can be seen in terms of

the market share and number of vehicles sold. The rapid growth in the Chinese market share from well below 1% to over 4% is in stark contrast to the slow rise in market uptake for Europe and the US. This difference can be attributed largely to the aggressive policy adoption by the Chinese government favoring zero emission vehicles. The share of pure electric vehicles in the Chinese market is far greater than those in its two counterparts, where hybridized options have a more significant share of the electrified vehicle market. Nevertheless, the market shift towards adoption of the electrified powertrain is clear, and the stage is set for substantial overhaul of the market share over the next decade in China and the United States.

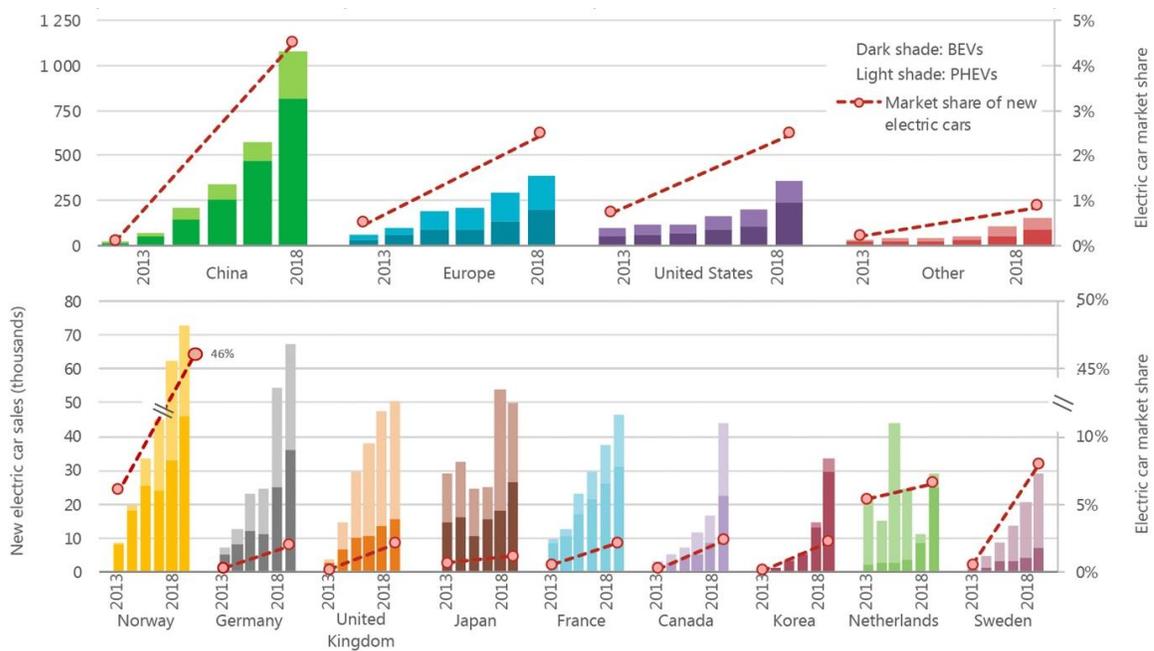


Figure 3: IEA analysis based on country submissions [2]

**Note:** IEA analysis based on country submissions, complemented by ACEA (2019); EAFO (2019); EV Volumes (2019); Marklines (2019); OICA (2019). Europe includes Austria,

Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Liechtenstein, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom. Other includes Australia, Brazil, Chile, India, Japan, Korea, Malaysia, Mexico, New Zealand, South Africa and Thailand.

### 1.2.2 Classification

Vehicles are generally classified based on their weight category, especially in the commercial and freight segment. Table 1 lists a broad classification of passenger and freight vehicles in the Chinese market on the basis of gross vehicle weight rating, based on data obtained from the classification of power-driven vehicles and trailers.

Table 1: Vehicle weight classes in China

Basic Application	Vehicle Class	Gross Vehicle Weight Rating	Number of Seats
Passenger	M1	-	≤ 9
	M2	< 5,000 kg	> 9
	M3	> 5,000 kg	> 9
Load	N1	< 3,500 kg	-
	N2	3,500 ~ 12,000 kg	-
	N3	> 12,000 kg	-

The corresponding vehicle types are depicted in Figures 4 and 5, starting with M1 class passenger vehicles and going up to N3 freight vehicles.

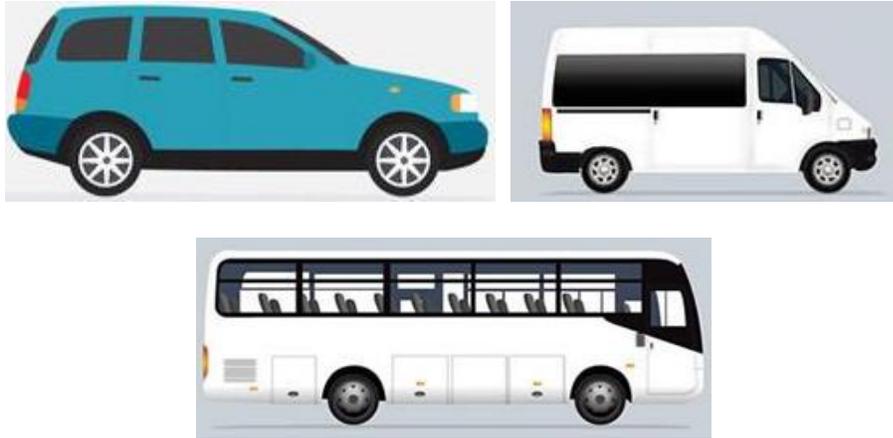


Figure 4: Passenger vehicles of category M1, M2 and M3 respectively (data from: Chinese Classification of power-driven vehicles and trailers, GB/T 15089-2001)



Figure 5: Freight vehicles of category N1, N2 and N3 respectively (data from: Chinese Classification of power-driven vehicles and trailers, GB/T 15089-2001)

According to data from the Chinese Ministry of Transportation, until September 2018, the number of electric vehicles has reached 2.21 million, of which 1.78 million are pure electric vehicles, accounting for 80.53%, and 254,000 are electric trucks, accounting for 11.46%.

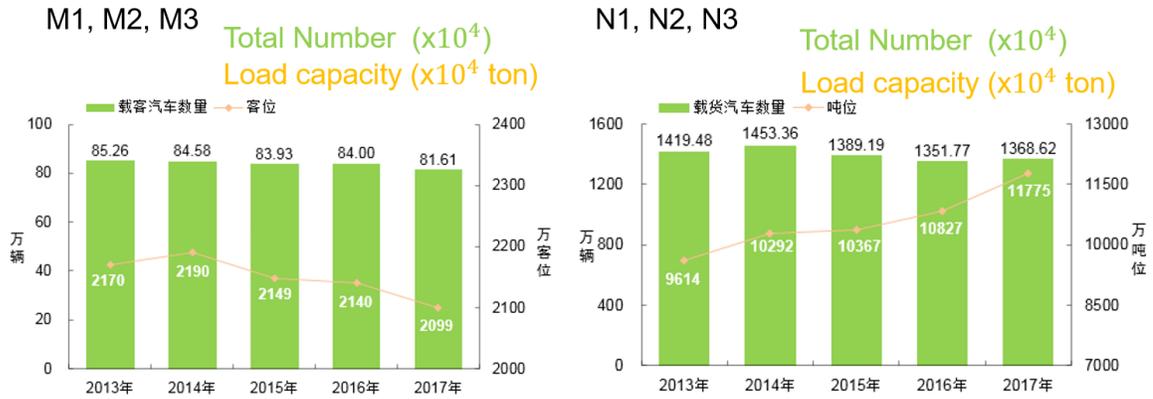


Figure 6: Load capacity: Electrified vehicles (Chinese Ministry of Transportation)

As depicted in Figure 6, the load capacity of passenger segment electrified vehicles has fallen, while that of the freight vehicles have increased steadily from 2013-2018. Meanwhile, the total number of vehicles in both the applications has remained fairly steady on a year-on-year basis.

### 1.2.3 United States Market Study

In the United States, both trucks and buses are classified by weight. The US Department of Energy classifies them as depicted in Figure 7, ranging from Class 1 passenger pick-up trucks weighting in at 6000 lbs. or less, all the way to Class 8 long-haul semi-trucks tipping the scales at 33,000 lbs. and above. This study focuses specifically on urban pick-up and delivery trucks in the Class 6 range weighing 19,600 lbs. – these are picked out with red outlines inset in Figure 7. In the United States, they are commonly seen in applications like FedEx, UPS and similar postal delivery applications with a frequent start-stop driving mission in low to medium speed environments. A series hybrid powertrain makes a good case for improved efficiency in these trucks due to this very driving mission.

Despite their potential benefits, the application of hybrid technologies into the medium- and heavy duty (MD/HD) markets is still at a relatively early stage. The size and diversity of the market prevents the application of a single hybrid solution that works equally well for the diverse vocational population. The MD/HD vehicle market encompasses a much wider variety of vehicle classes and thus Gross Vehicle Weights (GVWs) as compared to the LD passenger vehicle industry. Light heavy-duty trucks are classified as vehicles between 8,500lbs GVW and less than 19,500lbs GVW (Class 2b through Class 5). Medium duty trucks are classified as vehicles between 19,500lbs and 33,000lbs GVW (Class 6 and Class 7). Heavy duty trucks are classified as vehicles over 33,000lbs GVW (Class 8).

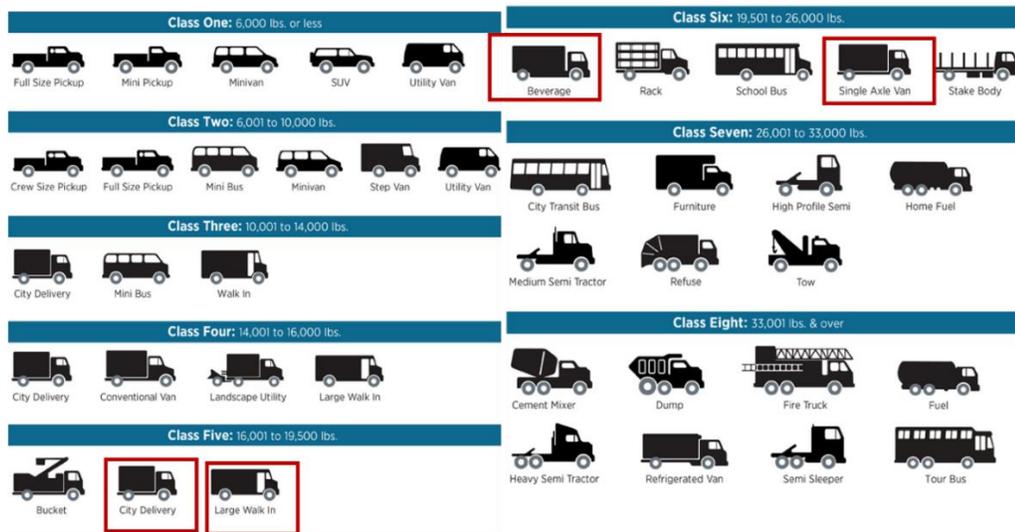


Figure 7: Weight class of vehicles in the US (data from: US Department of Energy, <http://www.afdc.energy.gov/data/>)

A typical Class 6 delivery vehicle is depicted in Figure 8. This exercise in efficiency improvement assumes the baseline vehicle to be powered by a gasoline IC engine, with a daily duty cycle lasting approximately 8 hours in an urban setting. The fuel economy of this

vehicle is evaluated in a chassis dyno with multiple loads over different days to obtain a benchmark to improve upon while performing the design space exploration detailed in the next chapter.



Figure 8: A Class 6 delivery truck (image: US Department of Energy)

A ballpark equivalence between the truck weight classes between the United States and China is indicated in Table 2 with correlations drawn between the eight different classes in the US and the six categories in China – M1, M2, M3, N1, N2 and N3. This is useful in cross-market comparison of available products in each segment.

Table 2: Comparison of vehicle weight ratings in the US and China

Gross Vehicle Weight Rating (lbs.)	Federal Highway Administration		US Census Bureau	China Equivalent
	Vehicle Class	GVWR Category	VIUS Classes	
<6000	Class 1: <6000 lbs.	Light Duty <10,000 lbs.	Light Duty <10,000 lbs.	M1, M2, N1
10,000	Class 2: 6001 – 10,000 lbs.			
14,000	Class 3: 10,001 – 14,000 lbs.	Medium Duty 10,001 – 26,000 lbs.	Medium Duty 10,001 – 19,500 lbs. Light Heavy Duty 19,001 – 26,000 lbs.	M3, N2
16,000	Class 4: 14,001 – 16,000 lbs.			
19,500	Class 5: 16,001 – 19,500 lbs.			
26,000	Class 6: 19,501 – 26,000 lbs.	Heavy Duty >26,001 lbs.	Heavy Duty >26,001 lbs.	M3, N3
33,000	Class 7: 26,001 – 33,000 lbs.			
>33,000	Class 8: >33,000 lbs.			

An insight into the truck sales in the United States for Class 3 to Class 8 is provided in Figure 9 sourced from the US Department of Energy and Ward’s Auto. The general trend is that of increased sales number across the board with a conducive economic growth since 2009. Class 5 and Class 6 trucks, which are of particular interest in this study have witnessed sustained growth in this period. With their growing numbers, and tightening global demand to curtail emissions drastically by 2030, marginal improvements in fuel economy are not sufficient.

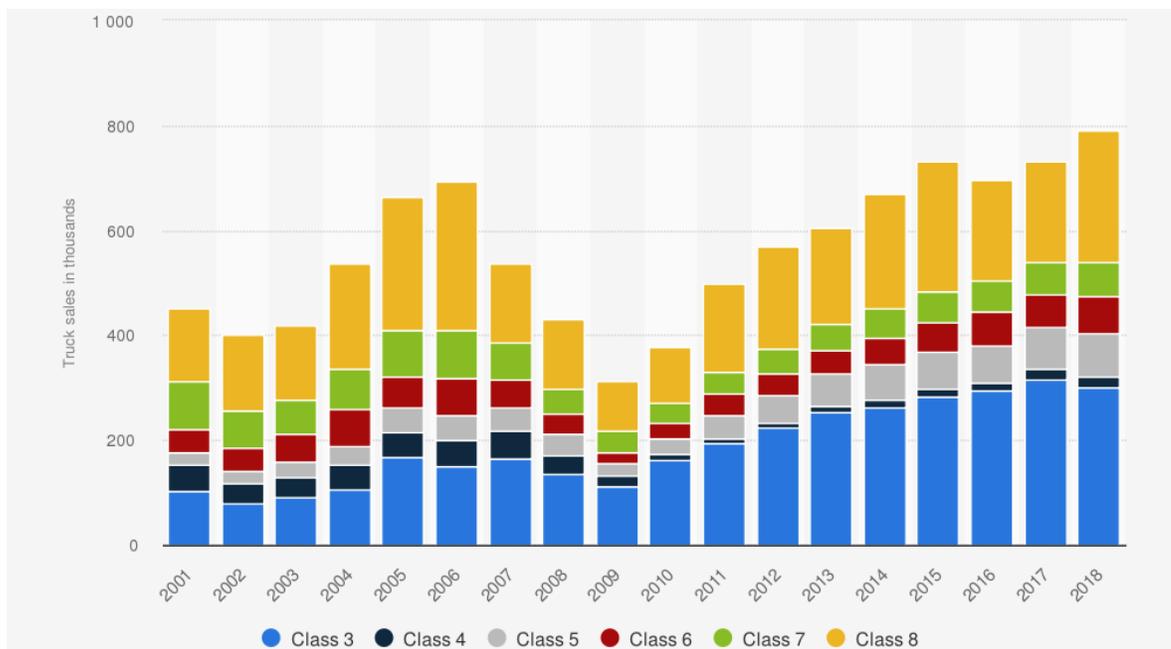


Figure 9: Class 3-8 truck sales in the US from 2001-2018 (US Department of Energy; <http://www.afdc.energy.gov/data/>)

Large-scale improvement in fuel economy standards is the most effective way to get on track to meet 2030 global targets. The transportation energy expenditure for the US in the year 2018 is shown in Figure 10 obtained from the US DoE. The share of fuel consumption

for medium and heavy trucking industry far outweighs the demand from air, water and rail combined. This calls for manifold increase in fuel economy standards in this industry, which can be driven by large electrification, connectivity, platooning and data analytics among other means. While the current state of the art in Li-ion battery technology does not permit a viable application for long-haul trucks, medium duty trucks are well within the scope of large-scale electrification, especially with range-extended EV technology.

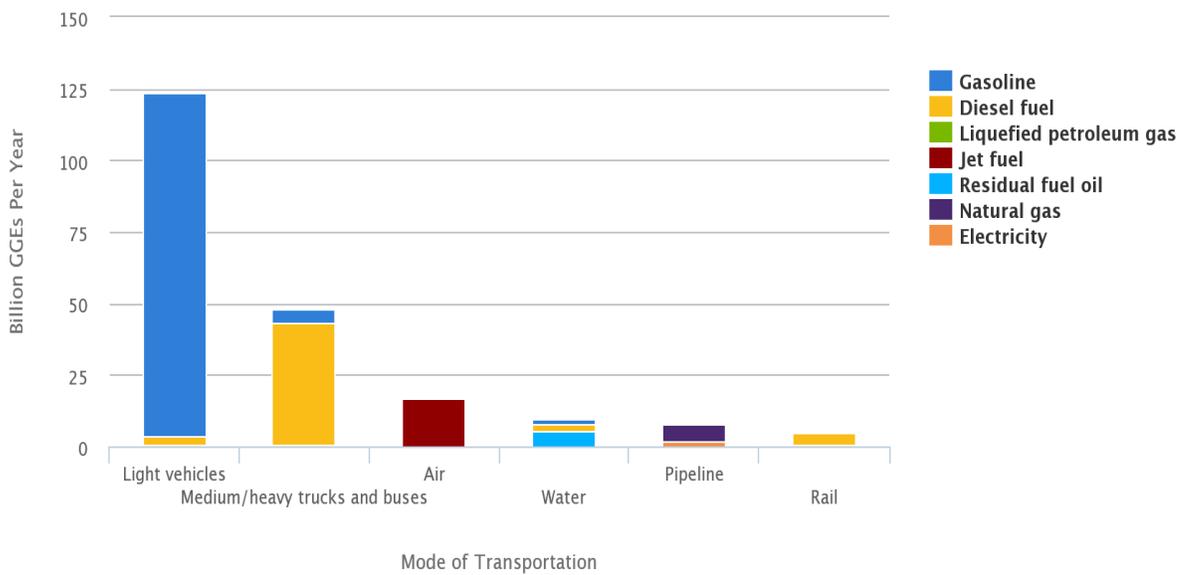


Figure 10: Transportation energy use by mode and fuel type in the US in 2018 (US

Department of Energy; <http://www.afdc.energy.gov/data/>)

A few of the current products and components in the US market in the HEV space are highlighted in the following paragraphs along with their key technical specifications.

1. Cummins EDI PowerDrive 6000 in Kenworth T370 Utility Truck

Table 3: Specifications of Cummins EDI PowerDrive 6000

Items	Parameters	Value
<b>Architecture</b>	PHEV	-
<b>Vehicle</b>	Gross Weight	33,000 lbs.
	Shape Size	-
	Max Velocity	-
	Distance Range	Pure electric mode: 50 miles; hybrid mode: > 300 miles
<b>Motor</b>	Type	-
	Peak Torque	-
<b>Engine</b>	Cummins B6.7	-
	Type	Diesel

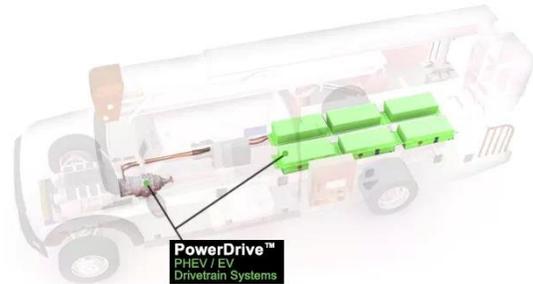


Figure 11: The Kenworth T370 Utility Truck (images from Cummins Inc.)

## 2. TEVVA MOTORS and Cummins ETREE

Table 4: Specifications of TEVVA MOTORS and Cummins ETREE

Component	ETREE	TEVVA
Vehicle	Class 6 Pick-Up & Delivery Truck Curb Weight: 19706.9 lbs.	Pick up & delivery 7.5 ton /16500 lbs.
Engine (REEV)	Cummins 4.5L 180 HP Diesel	1.6L Diesel, 100 HP (74 kW)
Battery	100-150 kWh	66 kWh, 350 V
Generator	135 kW (continuous)	unknown
Traction Motor	165 kW (continuous) AC PM Motor	120 kW
Gearbox	4 speed AT	unknown



Figure 12: Cummins ETREE and TEVVA MOTORS (images from Cummins Inc.)

### 1.2.4 China Market Study

According to the 2019 annual report of the Global EV Outlook [2], EV deployment has been growing rapidly over the past ten years, with the global stock of electric passenger cars passing 5 million in 2018, an increase of 63% from the previous year. Around 45% of electric cars on the road in 2018 were in China – a total of 2.3 million – compared to 39% in 2017. In comparison, Europe accounted for 24% of the global fleet, and the United States

22%. China has been on the forefront of the aforementioned market overhaul with aggressive policy mandates from the government along with fiscal incentives to businesses.

A few of the current products and components in the China market in the HEV commercial truck space are highlighted here along with their key technical specifications.

1. Chanje V8070

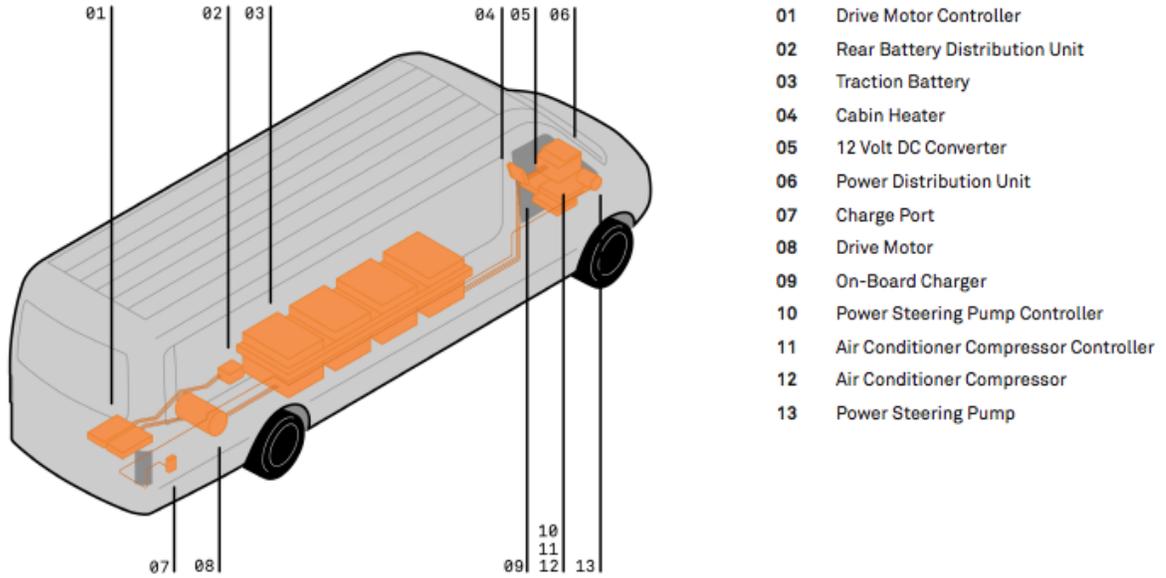
Table 5: Specifications of the Chanje V8070

Items	Parameters	Value
<b>Architecture</b>	Dual Rear Motors	-
<b>Vehicle</b>	Gross Weight	16,535 lbs.
	Shape Size	318.8*86.4*107.7 in
	Max Velocity	80 mph
	Distance Range	150 miles
<b>Motor</b>	Type	PMSM*2
	Peak Torque	564 ft-lbs.
	Max Power	198 hp
	Rated Power	-
<b>Chassis</b>	Wheelbase	194.3 in



Figure 13: Chanje V8070 electrified truck (images from Chanje product portfolio)

Figure 13 continued



2. BYD 6D Class 6 Step Van

Table 6: Specifications of the BYD 6D

Items	Parameters	Value
<b>Architecture</b>	EV, Direct Drive	-
<b>Vehicle</b>	Gross Weight	23000 lbs.
	Shape Size	325.6*96.9*121.5 in
	Max Velocity	70 mph
	Distance Range	125 miles
<b>Motor</b>	Type	PMSM
	Peak Torque	1328 ft-lbs.
	Max Power	335 hp
	Rated Power	-
<b>Chassis</b>	Wheelbase	178.0 in



Figure 14: BYD 6D Class 6 EV Step Van (image from BYD Auto)

### 3. BYD Class 6 Truck

Table 7: Specifications of the BYD Class 6 Truck

Items	Parameters	Value
<b>Architecture</b>	EV, Direct Drive	-
<b>Vehicle</b>	Gross Weight	26000 lbs.
	Shape Size	277.0*86.6*93.9 in
	Max Velocity	65 mph
	Distance Range	125 miles
<b>Motor</b>	Type	PMSM
	Peak Torque	1328 ft-lbs.
	Max Power	335 hp
	Rated Power	-
<b>Chassis</b>	Wheelbase	153.5 in



Figure 15: BYD Class 6 Truck (image from BYD Auto)

#### 4. BYD Class 5 Truck

Table 8: Specifications of the BYD Class 5 Truck

<b>Items</b>	<b>Parameters</b>	<b>Value</b>
<b>Architecture</b>	EV, Direct Drive	-
<b>Vehicle</b>	Gross Weight	16141 lbs.
	Shape Size	236.0*80.5*86.8 in
	Max Velocity	62 mph
	Distance Range	155 miles
<b>Motor</b>	Type	PMSM
	Peak Torque	406 ft-lbs.
	Max Power	201 hp
	Rated Power	-
<b>Chassis</b>	Wheelbase	132.3 in



Figure 16: BYD Class 5 Truck (image from BYD Auto)

5. FAW Jiefang J6F CA5042XXYP40LEVA84-3

Table 9: Specifications of FAW Jiefang J6F direct drive truck

Items	Parameters	Value
<b>Architecture</b>	EV, Direct Drive	-
<b>Vehicle</b>	Gross Weight	12.49 t
	Shape Size	7000*2364*2805 mm
	Max Velocity	90 km/h
	Distance Range	260 km
<b>Motor</b>	Type	PMSM
	Peak Torque	700 Nm
	Max Power	125 kW
	Rated Power	85 kW
<b>Chassis</b>	Wheelbase	3800 mm
	Axle load	4740/7750 kg



Figure 17: FAW Jiefang J6F Direct Drive Truck (images from FAW Jiefang)

6. BYD T7 BYD5110XXYBEV

Table 10: Specifications of BYD T7 Direct BEV Truck

Items	Parameters	Value
<b>Architecture</b>	EV, Direct Drive	-
<b>Vehicle</b>	Gross Weight	10.695 ton
	Shape Size	7450*2250*3300 mm
	Max Velocity	100 km/h
	Distance Range	240 km
<b>Motor</b>	Type	PMSM
	Peak Torque	550 m
	Max Power	150 kW
	Rated Power	110 kW
<b>Chassis</b>	Wheelbase	4250 mm
	Axle load	3600/7095 kg



Figure 18: BYD T7 Direct BEV Truck (images from BYD Auto)

## 7. Daimler Atego BlueTec Hybrid 12T Series Production Prototype

Table 11: Specifications of Daimler Atego BlueTec Hybrid

Items	Parameters	Value
<b>Architecture</b>	parallel hybrid P2	-
<b>Vehicle</b>	Gross Weight	11.99 ton
	Shape Size	-
	Max Velocity	> 70 mph
<b>Engine</b>	Type	four-cylinder OM 924 LA diesel engine
	Peak Torque	810Nm@ 1200-1600rpm
	Max Power	44 kW
<b>Motor</b>	Peak Torque	420 Nm
	Max Power	36 kW
<b>Transmission</b>	Type	Mercedes-Benz G 85-6
	Ratios	-



Figure 19: Daimler Atego BlueTec Hybrid (images from Daimler)

8. HINO 195h 6<sup>th</sup> Generation diesel-electric hybrid system

Table 12: Specifications of HINO 195h diesel-electric hybrid system

Items	Parameters	Value
<b>Architecture</b>	hybrid+6AT	-
<b>Vehicle</b>	Gross Weight	19500 lbs.
	Shape Size	318.8*86.4*107.7 in
	Max Velocity	> 70 mph
<b>Engine</b>	Type	Diesel-Electric Hybrid 4-cycle
	Peak Torque	440 ft-lbs. @1500rpm
	Max Power	210 hp@2500rpm
<b>Motor</b>	Peak Torque	258 ft-lbs.
	Max Power	36 kW
<b>Transmission</b>	Type	6-AT
	Ratios	3.742/2.003/1.343/1/0.773/0.634



Figure 20: HINO 195h diesel-electric hybrid system (images from HINO Trucks)

### 1.3 Range-extended HEVs for Commercial Vehicle Applications

Range extended electric vehicles (REEVs) have the ability to operate both as a pure electric vehicle as well as a hybrid electric vehicle based on the state of charge (SoC) of the energy store, which is usually a battery. The state of charge of the energy storage represents the amount of usable energy that remains in it. As a hybrid, the on-board range extender or generator set provides additional energy to increase the feasible driving range by recharging the electrical energy store. HEVs have more than one source of propulsive energy, which makes energy management imperative and challenging. Hybrid vehicles can have varying architectures, including parallel, series and power-split layouts each of which demand different power management strategies to be operated efficiently. While operating as an electric vehicle, the REEV utilizes only the battery to meet its energy demands and operates in a charge depleting (CD) mode to maximize the large energy store on board [8].

The generic schematic for a range-extended hybrid is depicted in Figure 21.

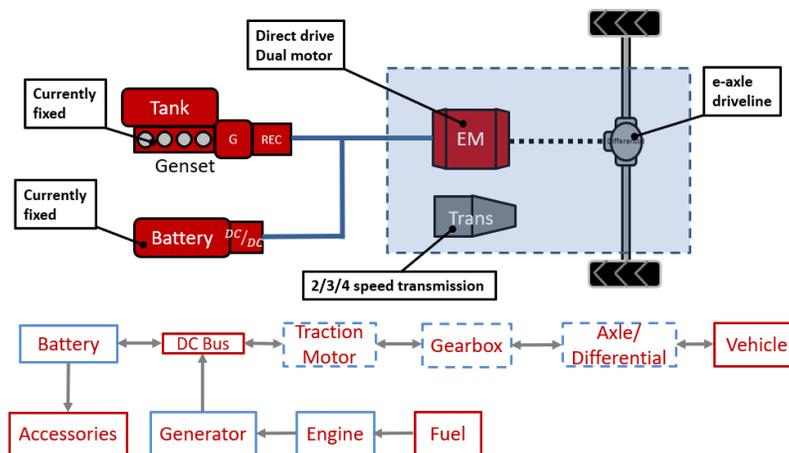


Figure 21: Range-extended Hybrid Electric Vehicle Architecture

The primary advantages of pursuing a range-extended architecture for a medium-duty urban delivery truck application optimized traction driveline, modular powerplant possibilities (engine downsizing), packaging benefits, excellent transient response, long operation life and possible zero emission operation. The main disadvantage lies in the fact that there are multiple energy conversion losses involved compared to a parallel architecture. Hence, such an architecture is not ideal for a long-haul highway application.

However, the diversity and physical size of medium-duty vehicles also allows manufacturers to consider the use of multiple high voltage battery packs to increase range in either a charge sustaining or charge depleting configuration. This provides design benefits to extend range from 50 miles versus 100 miles of charge depleting range capability for an inner city or urban PHEV pick-up and delivery vehicle versus a residential mixed urban and rural application. This approach can also be used to target specific performance. For example, a PHEV energy storage system could consist of a charge depleting high voltage battery and a high voltage charge sustaining battery – a hybrid energy storage system (HESS) – each of a specific chemistry suited to the duty cycle. This could reduce the premium price to a fleet operator by allowing the operator to buy the energy storage (or range) and performance they require [9].

Potential candidates for electrified powertrain truck technologies range from electric vehicles to range-extended electric vehicles to plug-in hybrid electric vehicles. The CERC-TRUCK team studies powertrain architectures to create a modular approach that considers

mission requirements, flexibility of powertrain design for second life, and regional and vocational needs, taking into account the tradeoffs in subsystem sizing, cost, and matching.

Longer-term goals include development of a novel battery pack design that hinges on the concepts of flexibility and modularity, and creation of a completely reconfigurable energy storage system for medium-duty and heavy-duty hybrid trucks, whose characteristics (power density, energy density, thermal performance, and degradation) can be optimized directly in the field. Later, the team hopes to design a flexible and reconfigurable drivetrain architecture (power electronics + traction motor + engine + mechanical coupling and transmission) and quantify tradeoffs as a function of cost, weight, and performance.

#### **1.4 Hybrid Electric Vehicle Optimal Energy Management**

In comparison to conventional IC engine vehicles, a relatively low energy-to-weight ratio amongst HEVs and EVs makes the efficient and optimal utilization of their energy very important. Energy management algorithms enable these vehicles to perform efficiently and at optimal operation points within the constraints of the powertrain components and the specific application. Energy management in HEVs involves deciding the amount of power delivered at each instant by the energy sources present in the vehicle while meeting required physical constraints. The primary objectives of an Energy Management System (EMS) are detailed in Figure 22.

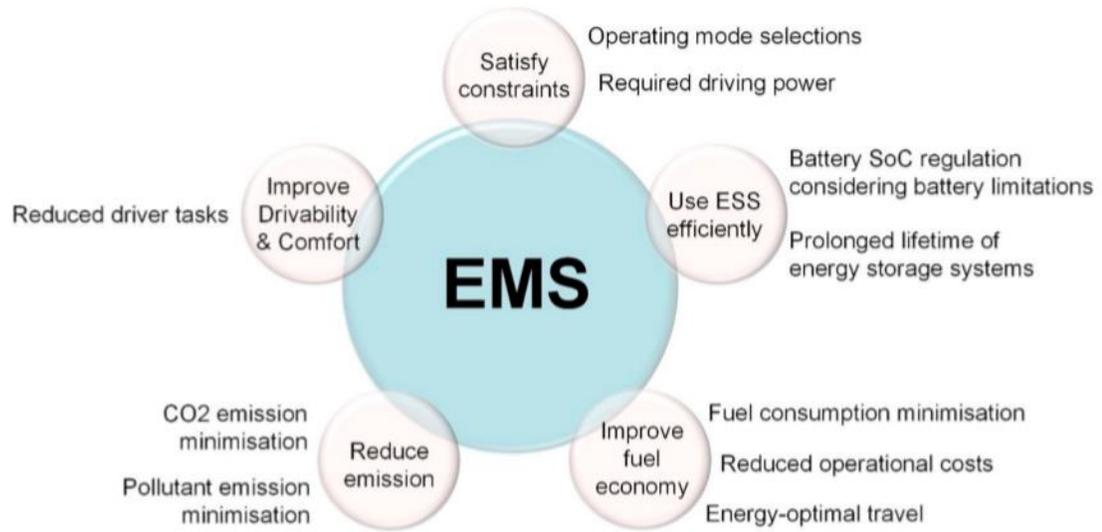


Figure 22: Objectives of an Energy Management Strategy [10]

CERC-TRUCK focuses on advanced energy management at the vehicle, multi-vehicle and fleet levels, with an overall target of improving vehicle energy consumption at the individual vehicle level by 15% and the freight efficiency of fleets at the regional level by 30%. The overarching goal is to maximize fleet efficiency through advanced energy management at the vehicle, multi-vehicle, and fleet levels.

Initially, partners in the United States and China collect existing on-road vehicle trip profiles for both countries. Researchers will use data collected to develop and optimize specific algorithms for medium-duty and heavy-duty applications for both U.S. and China markets. Analysis will quantify and understand the market differences and the impact of predictive energy management together with engine-in-the-loop assessments to validate the proposed algorithms.

In the longer term, research teams will develop a framework and specific use cases for both countries that automate the evaluation of multiple vehicle-level controls and study the impacts of a variety of driving scenarios and powertrain technologies. Ultimately, this team seeks to implement an algorithm to schedule the departure times of individual trucks to ensure that they can form a platoon.

The CERC-TRUCK team uses vehicle fuel economy modeling to identify subsystems that might yield efficiency improvements. After selecting candidate technologies for improving freight efficiency, research teams will prototype and fabricate or procure and evaluate potential alternatives, with longer-term goals of implementing and validating those showing greatest promise in a demonstration vehicle.

What follows is not meant to be an exhaustive review of contributions to the topic of adaptive strategies. Only the pivotal works on the subject have been reviewed.

#### **1.4.1 Global Optimal Energy Management Strategy: Dynamic Programming**

When computational capabilities are not a limiting factor, and assuming availability of a priori information about the entire optimization horizon (driving cycle), dynamic programming (DP) can be a powerful tool in benchmarking all online strategies. However, online optimal energy management techniques are chosen for practical application in hybrid vehicles partly due to their reduced computational demands in a real-time scenario and the causal nature.

The DP algorithm was proposed by Richard Bellman in the 1950s as a powerful tool to solve multi-stage decision problems with easy computational implementation. The essential idea behind the algorithm is the Bellman's Principle of Optimality, which states: An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision [Bellman, 1957].

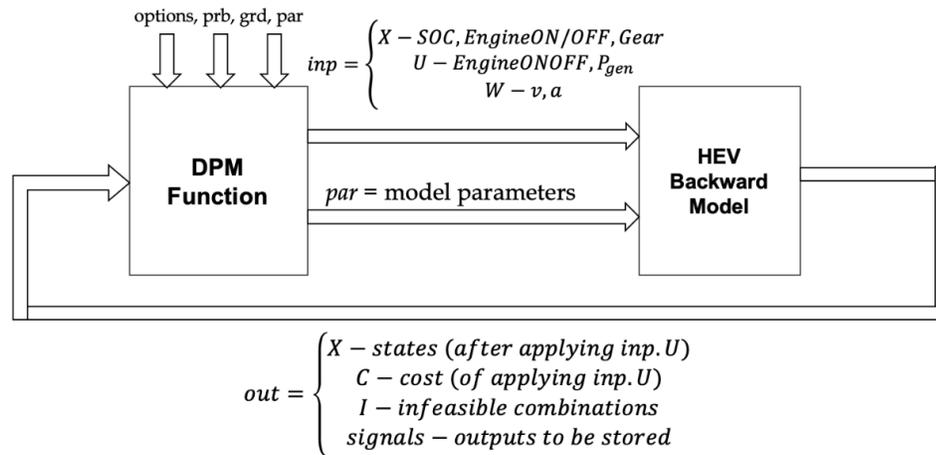


Figure 23: Dynamic Programming schematic

It is capable of providing optimal solution to problems of high complexity. However, it is noncausal and is only implementable in simulation – i.e., in a backward sense, starting from the final solution, tracking back through the control inputs needed at each stage to get there with the minimal cost. To determine the first control action, DP needs the backward solution of the entire problem, which explains why it is not practical for a real-time application. A simplified schematic of dynamic programming is illustrated in Figure 23.

This simple yet powerful principle turned into an iterative algorithm that efficiently searched the entire control space for the optimal policy. A general software toolbox called

‘**dpm**’ was developed at ETH Zurich [11] that allows simulation of HEV energy usage in a specified drive cycle.

For a hybrid electric vehicle with multiple energy sources, DP provides the sequence of controls that dictate the power split between the internal combustion engine and the energy storage system at every time step of the powertrain operation. Associated with control input to the powertrain is the cost that corresponds to fuel consumption, emissions, battery energy depletion, battery ageing or those based on other objectives. As with any optimal control strategy, the control input(s) at any stage is determined by in order to meet the demands of the system, while respecting the states of the system. The number of solution candidates that can be evaluated is a compromise between the computational power and the accuracy of the result: for instance, the minimum cost candidate may not always coincide with chosen candidate due to physical limits of the powertrain, but the closer they are to each other, the better the approximation of the optimal solution [12].

Once the grid of possible power splits, or solution candidates, is created, the procedure outlined earlier can be used, associating a cost to each of the solution candidates. Proceeding backwards (i.e., from the end of the driving cycle), the optimal cost-to-go is calculated for each grid point, and stored in a matrix of costs. When the entire cycle has been examined, the path with the lowest total cost represents the optimal solution [12]. It is to be noted that the accuracy of the solution offered by DP is limited by the discretization of the candidates in the problem.

Hence, DP offers the closest approximation to the optimal solution of the energy management problem for HEVs, and is therefore used to determine the maximum potential of any given architecture, thus serving as an essential benchmark for all online control strategies, in general.

The implementation of DP is a process with three steps: determination of arc costs, minimization of cost-to-go to determine the optimal control policy (proceeding backward in time), and application of the optimal control policy to the system (proceeding forward in time). While the last step can be performed on a standard simulator, the first two steps require a specific coding able to separate clearly the definition of the cost at each time from the integration of the dynamic state equations. In other words, a non-dynamic representation of the cost function  $L_k = L(x_k, u_k, \omega_k)$  and the system function  $x_{k+1} = f_k(x_k, u_k, \omega_k)$  are needed, where the external disturbance  $\omega_k$  may represent the driving cycle, the road grade etc. These functions are then called by the DP algorithm to compute the cost-to-go starting from the last point of the driving cycle, and to determine the optimal policy [12].

More recently, there have been successful implementation of short horizon DP in tandem with model predictive control (MPC), by leveraging look-ahead information to predict the future drive cycle encountered by the vehicle. The DOE ARPA-E NEXTCAR program undertaken by the Ohio State University Center for Automotive Research [13] and studies by Arizona State University researchers on look-ahead optimization strategies for HEVs [14] are prime examples.

### **1.4.2 Online Energy Management Strategies**

This CERC-TRUCK team focuses on integrating the key technologies identified across multiple research areas to demonstrate the efficiency improvement at the vehicle level. This effort assesses and characterizes a baseline demonstration vehicle. After modifying the demonstration vehicle with promising technology candidates, research teams will assess those technologies in order to quantify their respective impacts on vehicle efficiency.

After evaluating individual technologies on the demonstration vehicle, researchers will assess the aggregate improvement of all technologies to quantify overall freight efficiency improvements. This will require an online implementable energy management strategy to be integrated with the selected hybrid architecture. At present, most hybrid vehicles in production implement rule-based or heuristic strategies to split the power demand between the two forms of available energy stores in the vehicle – fuel and battery.

HEV energy management strategies have been classified over the years under several different criteria, focusing on the characteristics, approach and implementation adopted by them. One such classification that covers the majority of the state-of-the-art, proposed by Ahmed M. Ali et al [[15](#)], is depicted in Figure 24. The two major categories in this taxonomy are Heuristic or Rule-based and Optimization-based or Analytical. This section throws light on a few of their sub-categories.

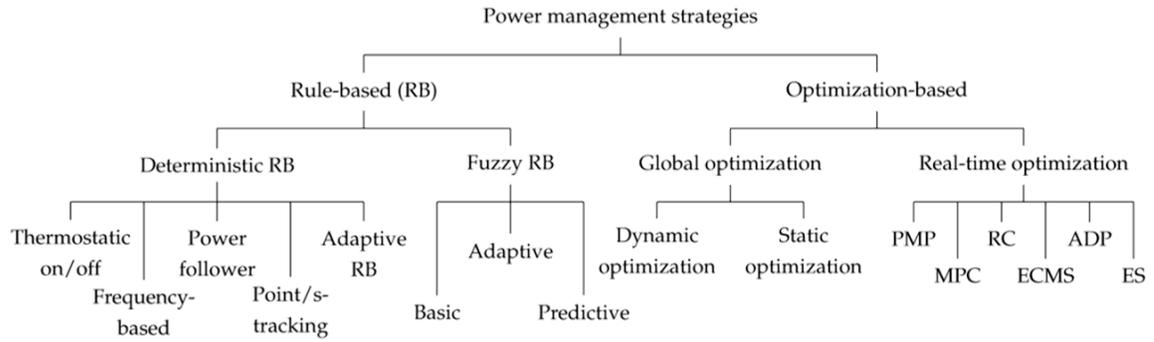


Figure 24: Classification of Power Management Strategies [15]

While rule-based strategies are easy to implement and robust, they are by and large sub-optimal in the grand scheme of HEV energy management as pointed out by D. Tran et al [10]. As a result, research over the past two decades have sought to arrive at real-time control strategies that are driven by optimization techniques. The most popular among these optimal control methods is the Equivalent Consumption Minimization Strategy (ECMS), which is a realization of offline PMP [16].

### (i) Pontryagin's Minimum Principle

The most well-known algorithm for solving the optimal control problem is Pontryagin's Minimum Principle (PMP), which is an extension of the calculus of variations with regard to the Euler–Lagrange equation [17]. The PMP was derived by Russian mathematician Lev Pontryagin in 1956 to solve the constrained global optimization problem; some regard this theorem as the beginning of modern optimal control theory. There are several formulations of PMP, depending on the way the optimal control problem is specified.

Each HEV architecture has one or more degrees of freedom to fulfill a prescribed driving cycle. For parallel HEV, for example, the road load is imposed by the drive cycle, however, one among the torque from the engine and that from the electric machine can be selected freely. In series or power-split HEV there is one degree of freedom more to be chosen, a speed value (e.g., the engine speed). A further degree of freedom to be optimized can be the transmission mode or gear number.

L. Serrao et al. [18] applied the same concept to find the optimal power split strategy for a hybrid electric refuse vehicle in 2005. PMP method does offer optimal solutions close to the DP results, however, the initial costate has a significant impact on the SoC variation, according to the work by S. Delprat et al [19]. Considerable amount of research has been published with methods that can be used to estimate the initial costate value, as seen in [20], [21], [22], [23] One major limitation of pursuing this route of obtaining the costate is that the optimal costate value can be computed only if the future driving cycle is known in advance. Hence, the driving cycle prediction with historical data, statistical methods and driving pattern recognition based on the vehicle location have been integrated with PMP to handle the dependence of the costate on the SOC.

Finding the optimal control strategy thus means finding the optimal value of each degree of freedom at each time step of the cycle. The PMP states that at each time step, the optimal values of the degrees of freedom are found by minimizing a function called Hamiltonian. The Hamiltonian is characterized by a 'costate', which is interpreted as a penalty factor for the battery energy utilization [24]. The optimal value of the initial costate can be found

through an iterative process, known as the shooting method, given full knowledge of the driving cycle is available. If different driving cycles are utilized, the initial costate obtained may have different values.

As formulated by G. Rizzoni et al. in [25], the Hamiltonian function for the HEV energy management problem is:

$$H(SOC(t), P_{batt}(t), \lambda(t), P_{req}(t)) = \dot{m}_f(P_{batt}(t), P_{req}(t)) + (\lambda(t) + \omega(SOC)) \cdot \dot{SOC}(t),$$

and the necessary conditions are:

$$P_{batt}^*(t) = \arg \min_{P_{batt}(t) \in U_{P_{batt}}} H(P_{batt}(t), SOC(t), \lambda(t), P_{req}(t))$$

$$\dot{SOC}^*(t) = f(SOC^*(t), P_{batt}^*(t))$$

$$\dot{\lambda}^*(t) = -(\lambda^*(t) + \omega(SOC)) \frac{\partial f}{\partial SOC}(SOC^*, P_{batt}^*) = h(SOC^*(t), P_{batt}^*(t), \lambda^*(t))$$

$$SOC^*(t_0) = SOC_0$$

$$SOC^*(t_f) = SOC_{target}$$

$$SOC_{min} \leq SOC^*(t) \leq SOC_{max}$$

The solution of the PMP's necessary conditions is obtained via shooting method according to the scheme provided in Figure 25, described by G. Rizzoni et al [25].

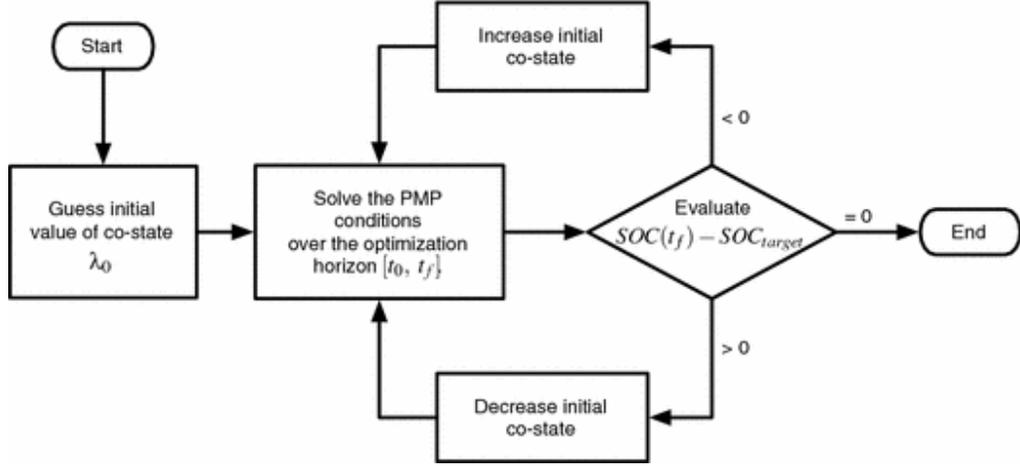


Figure 25: Schematic of the Pontryagin's Minimum Principle [25]

The implementation of the PMP's necessary conditions is shown in the schematic of Figure 25. At each instant of time over the optimization horizon  $[t_0, t_f]$ , given a request of power,  $P_{req}$ , the Hamiltonian is built and minimized. This generates the optimal control,  $P_{batt}^*(t)$  that is applied to the state and co-state dynamic block to compute the state of charge and co-state variation at the next step.

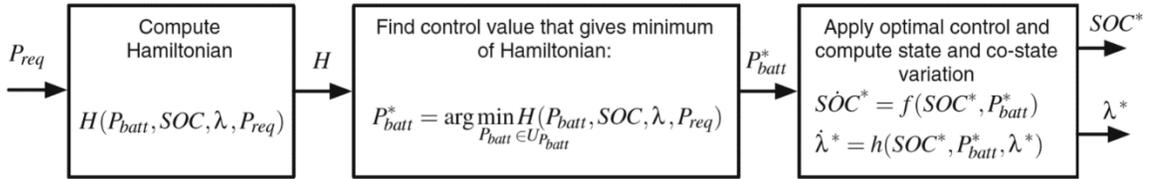


Figure 26: Open-loop PMP-based energy management control scheme [25]

The additive penalty function  $\omega(\text{SOC})$  is the piecewise function given by:

$$\omega(\text{SOC}) = \begin{cases} 0 & \text{if } \text{SOC}_{max} < \text{SOC} < \text{SOC}_{min} \\ K & \text{if } \text{SOC} < \text{SOC}_{min} \\ -K & \text{if } \text{SOC} > \text{SOC}_{max} \end{cases}$$

The constant  $K$  is determined in simulation iteratively by a hit and trial method to ensure that whenever the SOC hits the lower bound  $SOC_{min}$  and less whenever the SOC hits the upper bound  $SOC_{max}$ . When SOC is within bounds, the penalty function is inactive, i.e., a zero term is added to the instantaneous cost.

## **(ii) Equivalent Consumption Minimization Strategy**

The equivalent consumption minimization strategy (ECMS) was introduced by Paganelli et al. in 1999 as a method to address the optimal control problem, and has been shown to provide an effective solution to the HEV energy management problem. ECMS reduces the global optimal problem of minimization of fuel consumption to an instantaneous minimization problem to be solved at each time step, without use of information about the future operation of the powertrain.

According to G. Rizzoni et al., [25] ECMS is based on the idea that, in charge-sustaining (CS) HEVs, the difference between the initial and final battery SOC is negligible in comparison to the total energy used. This implies that the electrical energy storage system is used only as an energy buffer, and ultimately, all energy comes from fuel, and the battery can be seen as an auxiliary, reversible fuel tank. Any stored electrical energy used during a battery discharge phase must be replenished at a later stage using fuel from the engine, or through regenerative braking.

Two cases are possible at a given operating point:

1. the battery power is positive (discharging): a recharge with the engine will require some additional fuel consumption in the future;
2. the battery power is negative (charging): the stored electrical energy will be used to reduce the engine load, which implies a fuel saving.

The principle underlying the ECMS approach is that a cost is assigned to the electrical energy, so that the use of electrical stored energy is made equivalent to using (or saving) a certain quantity of fuel. This cost is obviously unknown, as it depends on future vehicle behavior, but it has been shown that the cost can be related to driving conditions in a broad sense (for example, urban versus highway driving). The concept implemented by the ECMS is illustrated in Figure 27, which refers to a parallel HEV, but the concept can be applied to a series HEV – the only difference is the location of the power summation node.

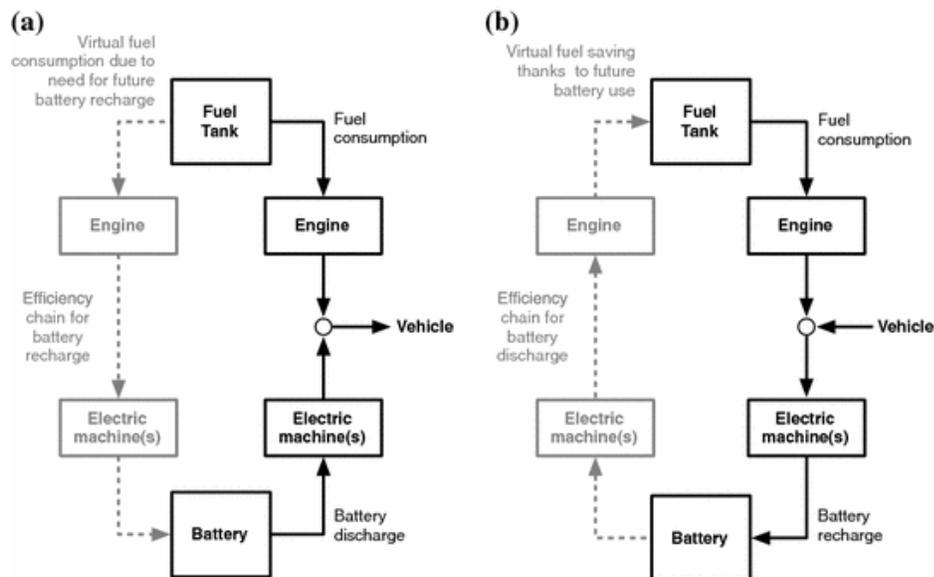


Figure 27: Energy path during discharge (a) and charge (b) in a parallel HEV [25]

*Note: When implementing ECMS, we account for the use of stored electrical energy, in units of chemical fuel use (g/s), such that an equivalent fuel consumption taking into account the cost of electricity*

The key idea of ECMS is that in both charge and discharge, an equivalent fuel consumption can be associated with the use of electrical energy; the equivalent future (or past) fuel consumption,  $\dot{m}_{ress}(t)$  (g/s), can be summed to the present real fuel consumption – fuel mass flow rate  $\dot{m}_f(t)$  (g/s) – to obtain the instantaneous equivalent fuel consumption,  $\dot{m}_{f,eqv}(t)$ :

$$\dot{m}_{f,eqv}(t) = \dot{m}_f(t) + \dot{m}_{ress}(t) \quad (1)$$

By analogy to an engine which consumes *real* fuel and for which the instantaneous fuel consumption is given as

$$\dot{m}_f(t) = \frac{P_{eng}(t)}{\eta_{eng}(t)Q_{LHV}}$$

where  $Q_{LHV}$  (MJ/kg) is the fuel lower heating value (energy content per unit of mass),  $\eta_{eng}(t)$  is the engine efficiency, and  $P_{eng}(t)$  is the power produced by the engine when it operates at a certain efficiency, the electric machine consumes *virtual* fuel

$$\dot{m}_{ress}(t) = \frac{s(t)}{Q_{LHV}} P_{batt}(t).$$

The equivalence factor  $s(t)$  is a vector of values, one for charge and one for discharge,  $s(t) = [s_{chg}(t), s_{dis}(t)]$ . Its task is to assign a cost to the use of electricity, converting electrical power into equivalent fuel consumption. Although, C. Musardo, G. Rizzoni et al. [26] showed that a single value of  $s$  can be used with little sacrifice in fuel economy.

By using ECMS the global problem of minimizing the total cost is reduced to the local (instantaneous) problem of minimizing  $\dot{m}_{f,eqv}(t)$ :

$$Global = \begin{cases} \min_{P_{batt}(t) \in U_{P_{batt}}} \int_{t_0}^{t_f} \dot{m}_f(t) dt \\ SOC_{min} \leq SOC \leq SOC_{max} \end{cases}$$

$$Local = \begin{cases} \int_{t_0}^{t_f} \min_{P_{batt}(t) \in U_{P_{batt}}} \dot{m}_{f,eqv}(t) dt \\ SOC_{min} \leq SOC \leq SOC_{max} \end{cases}$$

At each time instant (and over the entire duration of the driving cycle, the equivalent fuel consumption is calculated using (1) for several candidate values of the control variable  $P_{batt}(t)$ ; the value that gives the lowest equivalent fuel consumption is selected.

This approach has been shown to closely approximate the global optimal solution. In addition, the instantaneous minimization problem is computationally less demanding than the global problem solved with dynamic programming, and applicable to real-world situations since it does not rely (explicitly) on information about future driving conditions.

A constant value of the equivalence factor in charge,  $s_{chg}$ , and in discharge,  $s_{dis}$ , must be selected beforehand. Further research by C. Musardo et al [26] discuss that a real-time

energy management for HEV is obtained by adding to the ECMS framework an on-the-fly algorithm for the estimation of the equivalence factor according to the driving conditions. The main idea is to periodically refresh the control parameter according to the current road load, so that the battery state of charge is maintained within the boundaries and the fuel consumption is minimized. Pei et al [27] presented a direct mathematical approach for determining the state of charge (SOC)-dependent equivalence factor in the HEV supervisory control problem using dynamic programming. It provides a rational basis for designing ECMS which achieve near optimal fuel economy compared to DP.

### **1.4.3 Adaptive Optimal Energy Management Strategies**

If a priori knowledge of the drive cycle is not available, PMP (and ECMS) offers suboptimal online implementable control because the co-state  $\lambda$  or the equivalence factor 's' needs to be estimated as driving demands change. The hypothesis of perfect knowledge of the driving conditions clashes with the characteristics of general-purpose automotive applications and is not suitable for real-time applications [26]. The task of updating  $\lambda$  or the 's' online as driving scenarios vary results in a general supervisory controller that is referred to as adaptive optimal supervisory controller. Methods falling into this category have been indicated in the literature as Adaptive PMP (A-PMP) or Adaptive-ECMS (A-ECMS) strategies. In particular, three main categories of adaptation techniques to design A-ECMS can be identified:

- adaptation based on driving cycle prediction;
- adaptation based on driving pattern recognition;

- adaptation based on feedback from SOC.

### **Adaptation Based on Driving Cycle Prediction**

As in most realistic scenarios, when no information on the future drive cycle demand is available, optimal fuel economy cannot be guaranteed for any mission. The optimal value of the equivalence factor can be found through a systematic optimization only if the driving cycle is known. In such cases, better controller performance can be achieved by updating the ECMS equivalence factor as the driving cycle varies over time. These variation in the future demand can be predicted to varying levels and degrees of accuracy utilizing the technologies described in Chapter 3.

In the early methods, described by C. Musardo et al. in [26], [28] and [29]), a real-time energy management strategy was proposed by adding a real-time algorithm to the ECMS module for the periodic estimation and updating of the equivalence factor. The main idea is to periodically refresh the control parameter according to the current road load, so that the battery state of charge is maintained within the boundaries and the fuel consumption is minimized. For instance, this can be achieved in a charge-depleting hybrid by means of a delta energy or delta SOC feedback to constrain the SOC to follow a reference line.

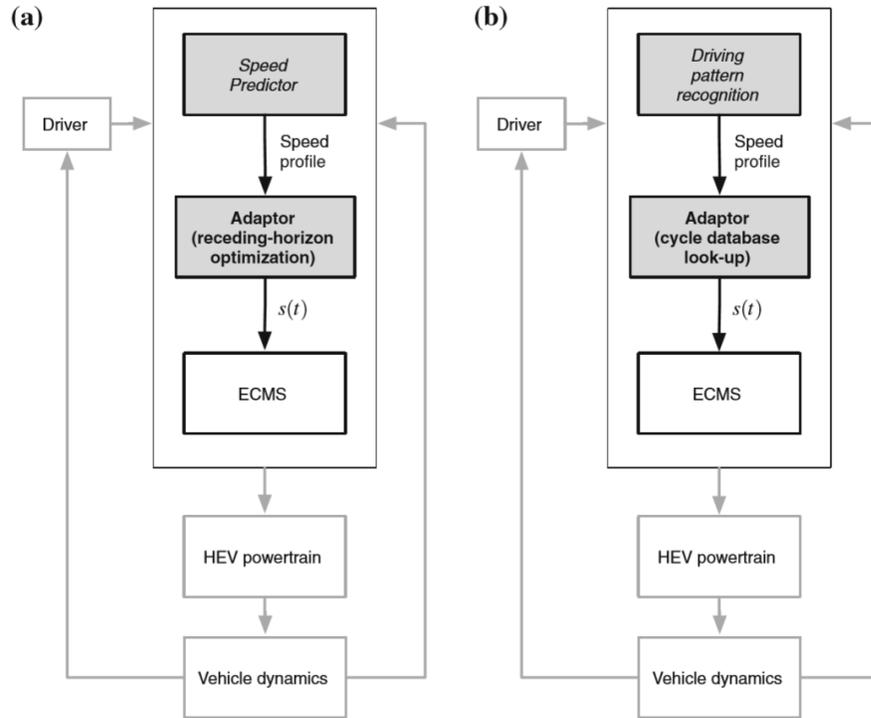


Figure 28: Schematic of adaptive energy management strategies [25]

The ECMS module is thus augmented with a device able to relate the control parameter,  $s(t)$ , to the current velocity profile. Figure 28 shows the A-ECMS control diagram: the identification of the driving mission given by the *Speed Predictor* is used as input to the *Adaptor* where the best value of the equivalence factor is found based on receding-horizon optimization [12].

To improve the execution time, a simplification was proposed for actual real-time implementation [26], consisting in the use of one equivalence factor, for both charge and discharge, thus introducing some approximation. The performance of A-ECMS is slightly inferior to the standard ECMS tuned on a perfectly known driving cycle, but in general, the results are quite good, and, most importantly, achievable in real-world application (if

enough computational power is available). In [30], the A-ECMS method strategy is based on speed prediction. The equivalence factor is estimated online based on a look-ahead horizon defined in terms of energy at the wheels, to determine at each instant the most likely behavior (charging or discharging) in the near future.

In [31] instead, an adaptation scheme similar to [29] is presented which uses a predictive reference signal generator (pRSG) in combination with a SOC tracking-based controller. The pRSG computes the desired battery SOC trajectory as a function of vehicle position such that the recuperated energy is maximized despite the constraints on the battery SOC. To compute the SOC reference trajectory, only the topographic profile of the future road segments and the corresponding average traveling speeds must be known. A constant reference SOC is considered, and A-ECMS implemented as in [29] is compared with a MPC-type controller based on the prediction of future torque demand, showing very similar performance of the two controllers.

### **Adaptation Based on Driving Pattern Recognition**

In parallel to the development of adaptive supervisory control schemes based on driving cycle prediction, an alternative adaptation scheme has been proposed exploiting the fact that equivalence factors are similar for cycles with similar statistical properties. Part of the studies below are cited from [12].

R. Bartholomaeus et al. [32] proposed an approach towards the real-time prediction of the speed profile of a vehicle operating in fixed-route service by leveraging the historic route

information for the predictive control of the hybrid drive train. The proposed algorithm utilized previously taken measurements, e.g., by a GPS device, of the vehicle motion during fixed-route travel. That history information together with measurements along the currently driven part of the route is processed in order to predict the future speed profile of the vehicle.

In a 2002 study by S. Jeon et al [33], a multi-mode driving control algorithm using driving pattern recognition is developed and applied to a parallel HEV. The multi-mode driving control was defined as the control strategy that switches a current driving control algorithm to the algorithm optimized in a recognized driving pattern. In the study, six representative driving patterns are selected, composed of three urban driving patterns, one expressway driving pattern, and two suburban driving patterns. A total of 24 parameters such as average cycle velocity, positive acceleration, kinetic energy, stop time/total time, average acceleration, and average grade are chosen to characterize the driving patterns.

An approach for A-ECMS based on driving pattern recognition is presented to obtain better estimation of the equivalence factor in different driving conditions is presented by B. Gu et al. in [34] and [35]. A pattern recognition algorithm is used to first identify which kind of driving conditions the vehicle is traversing, and then to select the most appropriate equivalence factors from a predefined set of values. The optimal values of  $s$  for several cycle typologies (city, highway, etc.) are precalculated and stored in memory; during vehicle operation, the adaptation algorithm uses the past and present driving conditions to determine the current cycle type, from which it selects the appropriate equivalency factor.

While the vehicle is running, a time window of past driving conditions is analyzed periodically and recognized as one of the representative driving patterns. This operation is performed in the *Driving pattern recognition* block of Figure 28. The *Adaptor* module then selects the more suitable values of  $s(t)$  from the equivalence factor database given the recognized driving patterns, and the ECMS is executed with the estimated value of  $s(t)$ .

### **Adaptation Based on Feedback from State of Charge**

After the equivalence between ECMS and PMP was formalized and a new interpretation of the ECMS was given as the optimal solution computed with PMP, it was understood that only one parameter needed to be adapted for online optimization, e.g., the co-state  $\lambda$ . Adaptive supervisory control approaches that rely on the instantaneous minimization of the Hamiltonian and have the PMP co-state  $\lambda$  as the single control parameter to adapt go under the name of A-PMP methods. The mechanism used to perform the adaptation in this case is feedback of the battery SOC.

Approaches developed to design adaptive optimal supervisory control methods using SOC feedback in [36], [37] and [38] are based on the idea of dynamically adjusting the value of the co-state at the present time (without using past driving information or attempting to predict future driving behavior), in order to contrast the SOC variation and thus maintain its value around the target value (reference SOC profile). In all these methods, SOC reference is considered constant. Performing the adaptation using a single parameter rather than two has a significant advantage in that it reduces the design and calibration complexity.

Conceptually, these approaches differ in that, while [36], [37] by J. Kessels and A. Chasse update the equivalence factor at each time instant, S. Onori et al. [38] rely on the concept of a charge-sustaining horizon, imposing charge-sustainability over a finite time horizon. If, on one hand these methods are easy to implement, robust (as they all rely on SOC feedback) and computationally cheap, on the other hand their performance relies on tuning of the parameters used in adaptation.

#### **1.4.4 Model Predictive Control (MPC)**

The MPC was introduced to tackle the issue of the DP algorithm, i.e., the global optimal control can be achieved only when all future information including the road shape, state of the vehicle, and the road loads are known in advance. Such conditions are impractical to obtain in advance for real-time applications, as discussed earlier in this chapter.

Model Predictive Control (MPC) is defined in a family of control techniques that make an explicit use of a model of the process to obtain the control signal by minimizing an objective function. Model predictive control represents the solution of a standard optimal control problem over a finite horizon, performed online using a model to predict the effect of the control on the system output. Therefore, MPC operates based on a receding-horizon control strategy with a predictive scheme using three main steps as described by E. F. Camacho et al in [39]:

- (i) calculating the optimal inputs over a prediction horizon to minimize the objective function subject to the constraints,

- (ii) implementing the first element of the derived optimal inputs to the physical plant, and
- (iii) moving the entire prediction horizon forward and repeating from step (i).

The optimal control problem in the finite domain is solved at each sampling instant, and control actions are obtained based on an online rolling optimization. However, the performance of the MPC is sensitive to the model quality. The mismatch of the models is represented in the models of the wheels, weather, road conditions, and sensor accuracy. To minimize this mismatch and disturbances, the horizon length has to be tuned, or GPS information is used with the MPC to improve the prediction results. According to G. Rizzoni et al [12], MPC is a control technique that requires high computational effort and an accurate model of the system in order to give good results; on the other hand, it can be applied to many cases and can be very effective if the reference trajectories are known. This does not happen in the case of vehicular applications, and some prediction techniques must be used in order to implement MPC as an energy management strategy for HEVs.

From the viewpoint of prediction algorithm, MPC can be split into two sub-classes: deterministic and stochastic. Like DP, deterministic-MPC has been used as a benchmark to evaluate other MPC-based energy management strategies, since it uses extensive future knowledge. On the other hand, stochastic MPC does not demand a priori information, and can easily adapt itself to changes in the stochastic parameters and high-order models. In literature, Markov chains are generally used to predict unknown future information or arbitrary processes such as driver demand, vehicle speed, power demand, road grade, left/right turns etc. as studied by [40], [41] and [42]. In [43], the authors use a Model

Predictive Control (MPC)-based strategy and utilize the information attainable from Intelligent Transportation Systems (ITS) to establish a prediction-based real-time controller structure.

The decisive advantage of MPC over ECMS is that MPC is neither short-sighted nor too sensitive. MPC is also used in conjunction with algorithms such as quadratic programming, nonlinear programming, Pontryagin's Minimum Principle and Stochastic Dynamic Programming based on the mission characteristics.

## Chapter 2. Design Space Exploration

In this chapter, the entire process of the optimal architecture selection of the series plug-in hybrid delivery truck is laid out with high level of detail. The architecture selection was made with the aid of a multitude of tools and input data, which culminated in the optimal layout to cater to the specific mission. This process is termed as Design Space Exploration for hybrid electric vehicles.

**Note:** This chapter is an extended version of the paper titled “Design Space Exploration for a Series Plug-in Hybrid Pickup and Delivery Truck Using Gaussian Process Optimization” accepted for publication by the SAE in April 2020.

In previous research work on this topic, several electrified powertrain architectures have been proposed, such as series- and parallel hybrids, with or without plug-in (PHEV) features, each having their own benefits. Multiple performance objectives are considered when optimizing the design of an electrified powertrain – fuel economy, emissions, component cost and weight, packaging, battery size, and drivability. Often times, these objectives are contradictory, i.e., one candidate may never be optimal with respect to all relevant objectives, but instead there are different trade-offs between different powertrain candidates.

Moreover, when comparing different segments of the transportation industry, for example, transit buses, or different classes of trucks for parcel delivery, it is clear that one powertrain may not be optimal in all situations. Hence, for each specific application, selecting electrified powertrain architecture and component sizes is an important task to find the optimal trade-off between fuel consumption and other performance objectives. This leads to the notion of *design for operation*, i.e. a design methodology that aims at finding optimal powertrain topology and component sizes for specific operation scenarios.

The problem of finding the optimal powertrain configuration is referred to as Design Space Exploration (DSE), which is often formulated as a Multi-Objective Optimization Problem (MOOP) as pointed out by D. Jung et al [44], [45]. In general, DSE is an optimization problem where the design space grows exponentially with the number of components that are optimized, thus requiring a considerable computational effort. Design Space Exploration of hybrid electric vehicles can be traced back to the 1990s, when the Good-Design Seeker, an architecture for exploring large design space of HEV using exhaustive search and dominance filtering, was developed by J. Josephson, G. Rizzoni et al [46]. This exercise has since then has adopted three main approaches to solve the design optimization problem:

- Brute force method, that is, an exhaustive search;
- Other search algorithms including genetic algorithm, the Nelder-Mead Simplex algorithm, PSO and DIRECT [47], [48];

- Gradient-based algorithms including convex optimization and sequential quadratic programming studied in [\[49\]](#), [\[50\]](#), [\[51\]](#).

## **2.1 Design Space Exploration for Delivery Trucks**

While the design optimization and the control design problems can be solved separately, we believe that it is necessary to solve a co-optimization problem in which the powertrain optimal control solution is a part of the design space as well. One typical example is to consider the component/subsystem sizing and control variables as a whole, and to optimize this bi-level system design using a combination of an iterative search algorithm and Dynamic Programming (DP).

A large chunk of previous research in the area of HEV design space exploration hinges on fixed HEV architectures. This approach is not holistic, and limits the scope of design optimization to sizes of only certain components and the final drive ratio as discussed in [\[52\]](#). Among those who take into consideration the generation and selection of different topologies, few explore a variety of series hybrid topologies. A general and inclusive framework for HEV design space exploration (DSE) is proposed in this paper, by specifying in detail the elements of input, constraints, cost function and output for each plausible architecture. Simplified pictorial representations of this process are shown in Figures 29 and 30.



Figure 29: Components of Design Space Exploration

Evaluating the performance of each powertrain can be time-consuming, because fuel consumption is evaluated by optimizing a specific powertrain for a given set of driving scenarios representing realistic driving missions. Therefore, in these situations, selecting a suitable search algorithm is important. In order to reduce the total computation time, a design space exploration algorithm is proposed which uses Gaussian Processes (GP) to select the powertrain candidate, in each iteration, that is most likely to be Pareto-optimal.

In this work, the powertrain design of a medium-duty Class 6 delivery truck is considered. The truck has a series hybrid powertrain with an internal combustion engine as range extender (REEV), where different powertrain topologies with plug-in features are explored. A simulation model of the powertrain is developed and Dynamic Programming

is used to compute the optimal control strategies and fuel consumption for each architecture.

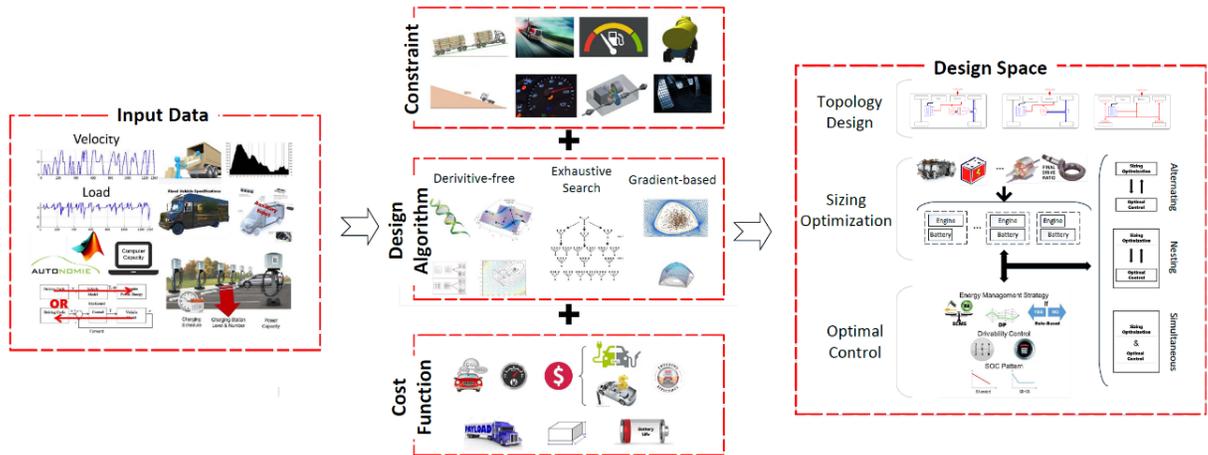


Figure 30: A holistic schematic of DSE

The first part of this chapter describes the modeling of the various components of the powertrain, and provides the details about the size of the design space and the GP algorithm used to optimize the exploration. DP is used to solve the optimal energy management problem to determine the sequence of control inputs that result in the lowest cost solution, i.e. the highest fuel efficiency, for each of the optimal candidates. In the second part, the solution of the problem is presented, and the results of the simulations are used to perform a cross-architecture performance and cost comparison.

## 2.2 Framework, Methodology and Problem Formulation

As already mentioned, the object of this study is to achieve an optimal architecture for a range-extender hybrid electric vehicle (REEV) with plug-in features. A general schematic representation of a REEV is presented in Figure 31. As shown, in this type of architecture,

the internal combustion engine (ICE) is used along with a generator in order to operate as a genset that is connected to the rest of the powertrain electrically. Figure 1 also shows which components are considered for the DSE: the electric motor (EM) and the transmission (Trans) and driveline. In the energy flow block diagram at the bottom of the figure, these elements are enclosed inside dashed lines. Depending on the transmission and EM choice, five REEV architectures are analyzed:

1. Two-speed e-Axle;
2. Three-speed AMT + Electric Motor;
3. Four-speed AMT + Electric Motor;
4. Direct Drive;
5. Dual Motor.

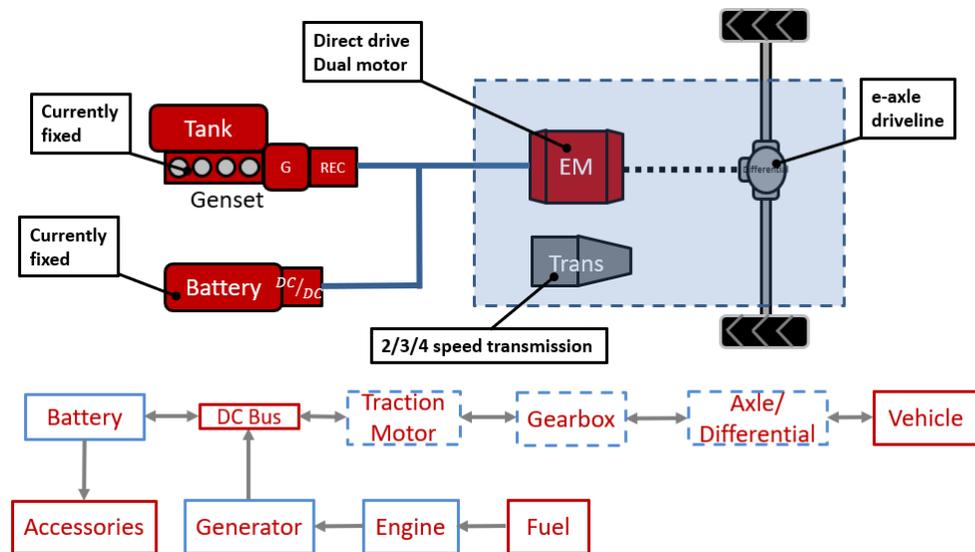


Figure 31: REEV architecture with representation of energy flow

In the following subsections, the models of the vehicle and powertrain components are described first. After that, the size of the design space is presented, and the need to reduce its volume is discussed, which leads to the description of the search algorithm used in the design space reduction. In addition, the formulation of the optimal energy management problem is described.

### 2.2.1 Vehicle Road Load Model

The vehicle road load is a result of Newton's Second Law of Motion, and is used to calculate the longitudinal vehicle speed:

$$v(t) = v(0) + \int_0^t \frac{F_{tract} - F_{load}}{1.1M} dt \quad (1)$$

For a REEV, the tractive force  $F_{tract}$  is passed down to the wheel level from the electric motor through the transmission (if present). On the other hand, the load force,  $F_{load}$ , is a combination of aerodynamic drag,  $F_{drag}$ , force from grade,  $F_{grade}$ , and rolling resistance.

$$F_{load} = F_{drag} + F_{grade} + F_{roll} + F_{inertia} \quad (2)$$

$$F_{drag} = \frac{1}{2} C_d \rho_{air} A_f v^2 \quad (3)$$

$$F_{grade} = Mg \sin \theta \quad (4)$$

$$F_{roll} = Mg \cos \theta \quad (5)$$

$$F_{inertia} = 1.1M \frac{dv}{dt} \quad (6)$$

The prominent parameters of the vehicle used in simulation are listed in Table 13. These correspond to a Class 6 pickup and delivery truck.

Table 13: Default vehicle parameters

Mass (kg)	8890
Rolling resistance coefficient	0.0072
Wheel radius (m)	0.4191
Frontal area (m <sup>2</sup> )	5.41
Aerodynamic drag coefficient	0.622
Auxiliaries power (W)	4000

The vehicle model across different simulators stay ideally the same. Although some differences exist due to the ways different simulators are constructed, the vehicle dynamics model, the powertrain model should be synced up as identical as possible. To address this syncing issue, an effort of converting vehicle data into class-defined objects are made.

The default vehicle parameters (Table 13) are stored in the ‘vehicle\_model’ class and will be automatically used during class object creation.

### 2.2.2 Powertrain Dynamics Model

All powertrain models are quasi-static energy conversion efficiency models, except the battery model, which contains an integral equation (state of charge). Thus, the combined vehicle and powertrain model has two dynamic states: vehicle velocity and battery state of charge.

### (i) Genset Model

The engine model is created from measured data of a mass-market 4.5L internal combustion engine, which is scaled up with the Willans Line model to match a 150 kW of maximum electrical power generation, coupled with a constant generator efficiency of 90%. The scaled genset efficiency map is shown in Figure 32. However, being mechanically disconnected from the driveshaft, the genset is manipulated to operate at its maximum efficiency on the optimal operation line (OOL), which is illustrated in Figure 33. The genset is modelled as a MATLAB class object with the properties described in Table 14.

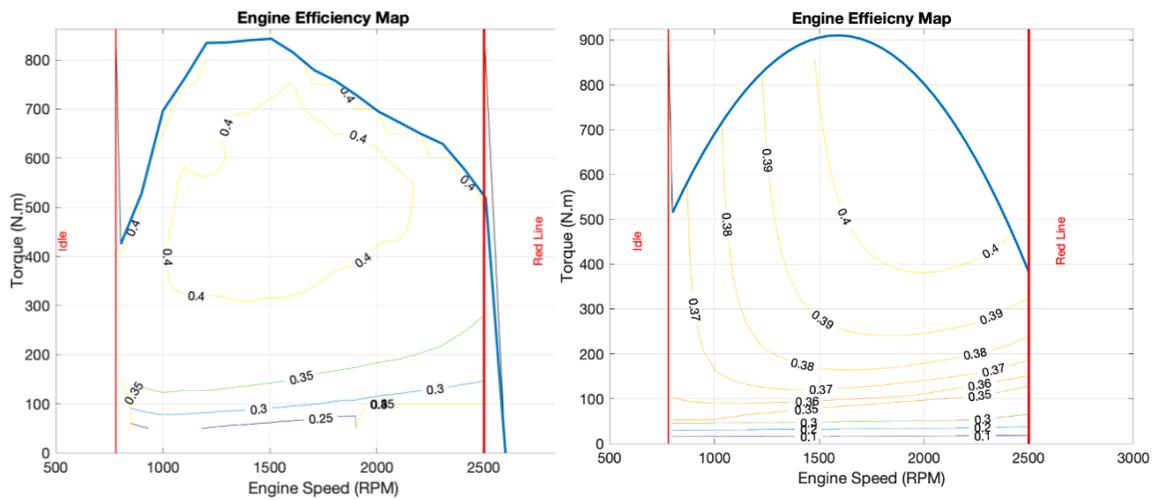


Figure 32: Original and scaled engine efficiency maps

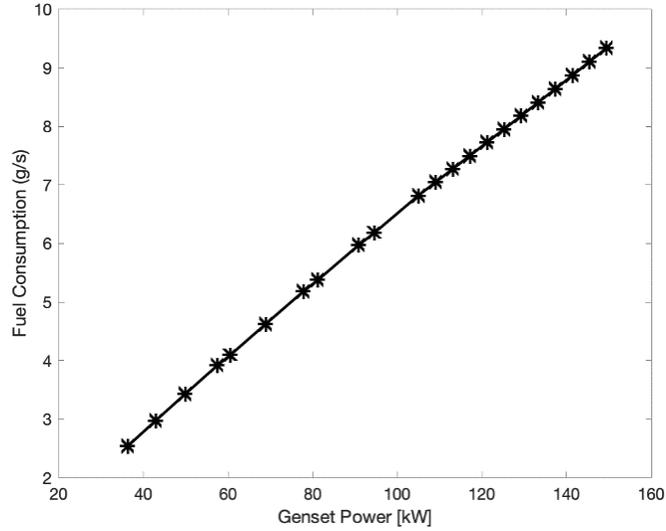


Figure 33: Scaled engine optimal operating line (OOL)

Table 14: Genset class definition

Name	Meaning
P_genset_list	Genset electrical output power (W)
mf_list	Fuel consumption rate (grams/sec)

**Note:** There exists inherent possibility of sudden switching from one engine (or electric motor) operation point to another because of the quasi-static model being used here. In realistic terms, these sudden switching of operating points wouldn't be acceptable. It is up to future revisions (if necessary) to implement these higher fidelity models of engine (and electric machine) that models the constraint on torque change and speed change to limit abrupt transitions.

## (ii) Electric Motor Model

The electric motor used in this study is a scaled up permanent magnet synchronous machine with a peak power output of 245 kW and a continuous power output of 165 kW. The electric motor map is generated using a linear scaling process: the speed range of the motor is stretched, such that it could meet top speed requirements of 80 mph with a final drive of 5.13, and wheel radius of 0.4191 m. The original motor map and the scaled up electric machine map are shown one after the other in Figure 34.

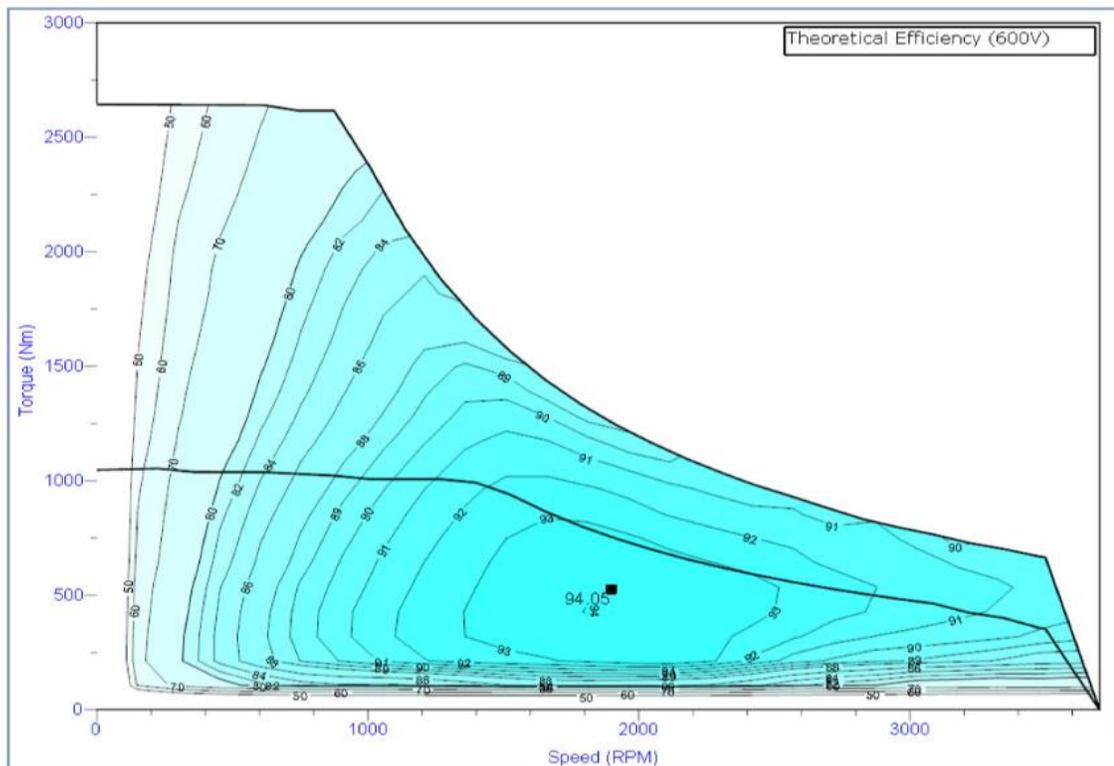
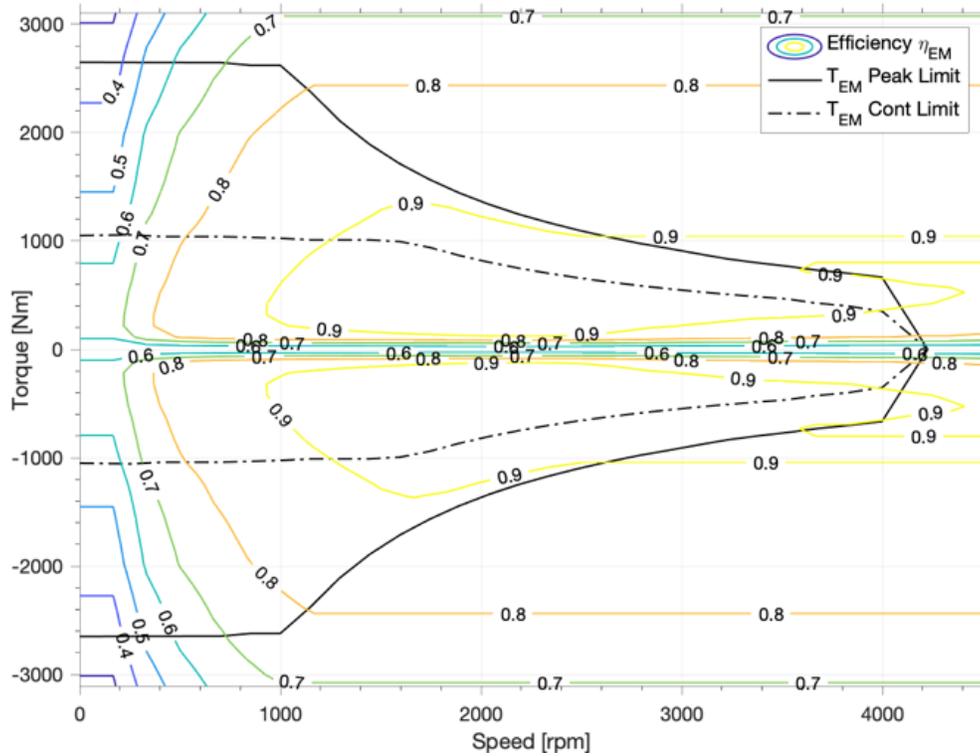


Figure 34: The original motor map (above) and the scaled-up map (below)

Figure 34 continued



The scaling of the motor took a simple but effective linear scaling approach. First, the speed range of the motor is stretched, such that it could meet top speed requirements of 80 mph with a final drive of 5.13 and wheel radius of 0.4191(m) or 16.5(in). Second, the torque range of the motor is stretched and obtained an optimal scaling ratio of 1.069 as a result of the DSE study.

The abovementioned scaling process resulted in a motor with continuous peak power of 165kW. The electric motor is modelled as a MATLAB class object with the properties listed in Table 15.

Table 15: Electric machine class definition

Name	Meaning
w_motor_list_eta	EM Speed List for $\eta_{EM}$ map
T_motor_list_eta	EM Torque List for $\eta_{EM}$ map
eta_motor	EM Efficiency ( $\eta_{EM}$ ) map
T_motor_list	EM Speed List for Maximum Torque Curve
T_peak_max_list	EM Peak Maximum Torque List
T_cont_max_list	EM Continuous Maximum Torque List

**(iii) Battery Model**

The battery model is a zeroth order equivalent circuit model, which assumes parallel and series connections of identical cells. The battery parameters are listed in Table 16.

The battery model follows simple zero order equivalent circuit model. Data of the battery chemistry and internal characteristics are taken from an OSU project directory. The battery pack is modelled as a MATLAB class object with the properties described in Table 16. Scaling of the battery was performed in a simplified way: By changing the number of parallel branches in order to match up with the assumed 74 kWh battery capacity suggested by the industry partner. The resulting number of parallel branches is a non-integer.

Table 16: Battery pack parameters

Property	Value	Property	Value
Rated Pack Voltage (V)	700	Rated Cell Voltage (V)	2.71
Number of cells in series	258	Number of cells in parallel	51.13*
Pack capacity (kWh)	74	Cell capacity (Wh)	5.61
Pack capacity (Ah)	27,306	Cell capacity (Ah)	2.07
Discharging C-rate	2.5	Charging C-rate	1

**\*Note:** The number of cells in parallel is not an integer because the scaling of the battery was performed using a continuous scaling factor.

The physics involved in modeling the battery pack are described below:

- $I_{cell} = \eta_{batt} \cdot \frac{1}{2R} (V_{oc} - \sqrt{V_{oc}^2 - 4RP_{cell}})$
- $SoC(t) = -\frac{1}{C_{cell}} \int_0^t I_{cell} dt + SoC(0)$
- $V_{OC}$  modeled by:  $V_{oc} = V_0 + \alpha_0(1 - \exp(-\beta SoC)) + \gamma SoC + \zeta(1 - \frac{\epsilon}{1-SoC})$
- R (internal resistance) is modeled by maps in Figure 35.

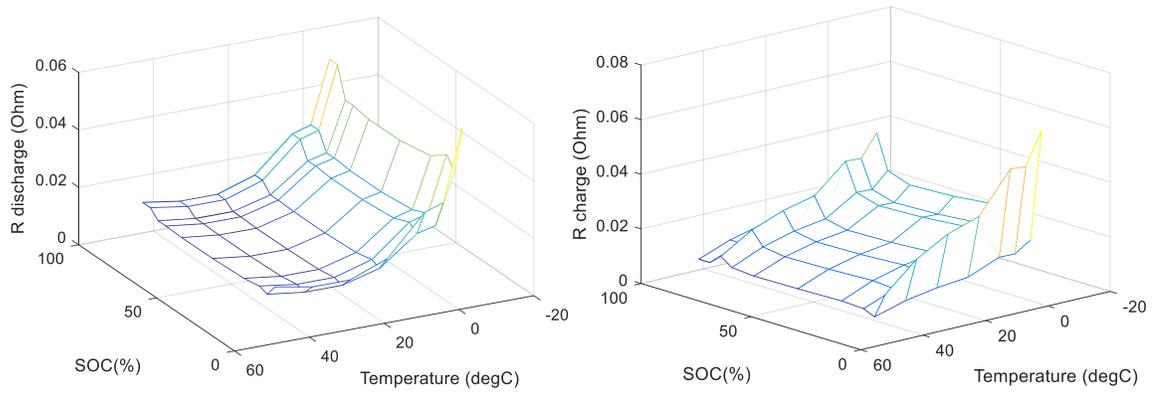


Figure 35: Battery internal resistance maps

#### (iv) Transmission Model

The transmission is simply modelled as a set of gear ratios with fixed efficiencies. The optimized gear ratios are listed in the tables for the corresponding architectures later in this chapter.

### (v) Backward Model

A backward simulator was built to make use of Dynamic Programming algorithm to provide a benchmark global optimal solution for the energy management problem. The name backward comes from the fact that the algorithm requires inverted vehicle dynamics calculation from the outcome speed of vehicle to the force and torque exerted on the vehicle powertrain. The ‘dpm’ toolbox requires that the problem is formulated in a particular format such that the toolbox could understand and solve it. The relationship diagram for this particular problem is shown in Figure 36.

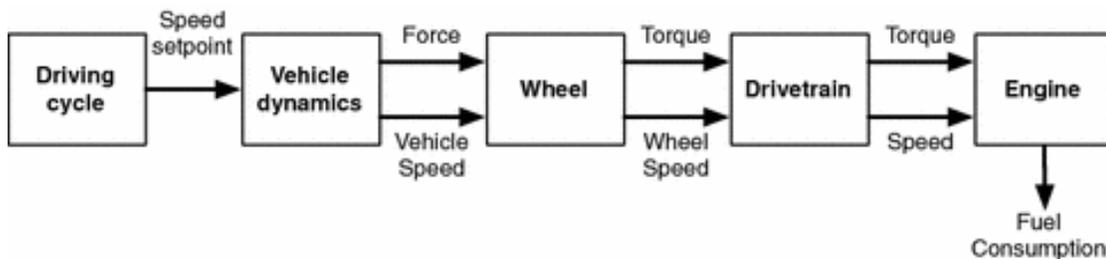


Figure 36: Backward powertrain model for a conventional vehicle

**Model Setup:** The vehicle and environment models are automatically setup by creating the vehicle class object described in Section 2.2.1. The only thing that needs effort is to translate those models to ‘dpm’ acceptable variables.

**Simulation options:** Apart from a scalable vehicle powertrain, the things that could be changed for the backward DP simulation are: upper and lower limits of SOC, initial SOC value and the drive cycle. The variables required by ‘dpm’ is listed in Table 17.

Table 17: Variables required to setup the dpm function

DPM variable	Meaning	Property
grd.X{1}	SOC	$\in [0, 1]$ , continuous
grd.X{2}	Engine State	$\in [0, 1]$ , discrete
grd.X{3}	Gear State	$\in [0, 1, 2]$ , discrete
grd.U{1}	Engine ON/OFF Request	$\in [0, 1]$ , discrete
grd.U{2}	Genset Power Level	$\in [1, 2 \dots, N]$ , discrete
prb.W{3}	Gear Request	$\in [1, 2]$ , discrete
prb.W{1}	Vehicle Speed	predetermined, stage by stage
prb.W{2}	Vehicle Road Load	predetermined, stage by stage
prb.W{3}	Mechanical Brake Torque	(optional*) predetermined, stage by stage

\*The mechanical brake torque is an optional feature required only when recording mechanical brake decisions from the Simulink forward simulator is to be injected here for comparison reasons (refer to Section 3.1 for more information on this topic)

### 2.2.3 Library of Components

The powertrain components simulated in this project include the following, and their ranges of specification are described in detail in Table 18.

1. Internal combustion engine
2. Gear ratios and mechanical gearbox
3. Electric machine(s)
4. Battery pack
5. Engine accessories and auxiliary loads
6. Brakes and tires

Table 18: Specification for the vehicle parameters and range for components

<b>Component</b>	<b>Specifications</b>
<b>Vehicle</b>	Class 6 Pick-Up & Delivery Truck GVW: 19,600 lbs. Frontal Area: 5.4 m <sup>2</sup> Coefficient of drag: 0.622 Transmission efficiency: 0.95, driveline efficiency: 0.93, e-axle efficiency: 0.93 Rolling resistance coefficient: 0.0072; Rotation mass factor: 1.1.
<b>Genset</b>	Using the optimal operating line (OOL) of combined Genset data
<b>Battery</b>	74 kWh permissible SOC range: 100% - 20% Specified starting SOC for drive cycle: 99% Specified terminal SOC boundary for drive cycle: 78% - 82%
<b>Traction Motor</b>	Scalable motor based on TM4 LSM200C-HV2600 176 kW continuous power PMSM electric machine
<b>Gearbox</b>	4-speed [4.5– 3.5], [3.5– 2.5], [2.5– 1.2]; 3-speed [4.0– 2.5], [2.5– 1.2]; 2-speed [3.0 – 1.5]; direct drive; dual-motor.
<b>Rear Axle Ratio</b>	5.13
<b>Accessories</b>	4 kW
<b>Tire Radius</b>	0.419 m

#### 2.2.4 Design Space Exploration

There are multiple architecture options and component sizes that can be used within the framework of delivery trucks. The powertrain is optimized with respect to freight ton efficiency (discussed in section 2.3), vehicle weight, packaging and operating cost. The complexity of on-line control algorithms for different architectures is not considered in this case study. Fuel economy is evaluated using Dynamic Programming (DP) and a mathematical model of each powertrain architecture. Since different candidates will have various trade-offs of the objective functions, the goal is to explore the Pareto-front to

understand the optimal trade-offs between the objective functions and to see which powertrain architectures and component sizes are relevant in different situations.

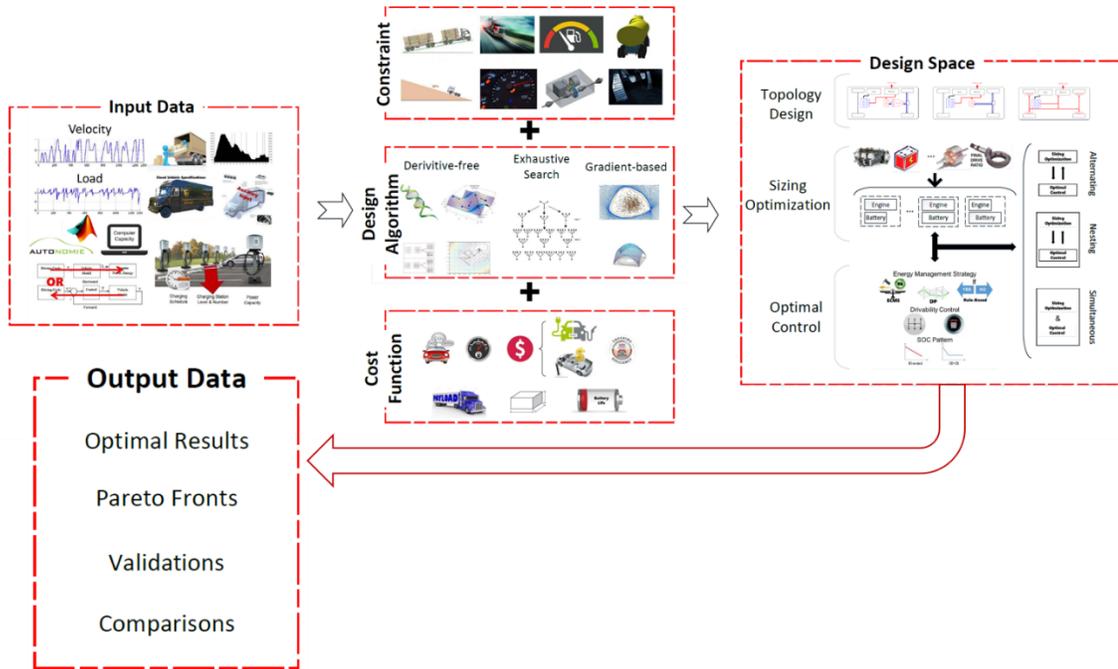


Figure 37: The progressive list of stages and outcomes of Design Space Exploration

In general, the driving missions are represented by driving cycles which describe the general characteristics for each driving mission, for example, average and maximum speed, and frequency of starts and stops. One powertrain can be optimized for multiple driving cycles simultaneously; however, in this study, we select one specific driving cycle that is representative of a typical package delivery application. The flow of actions in this process is represented in Figure 37.

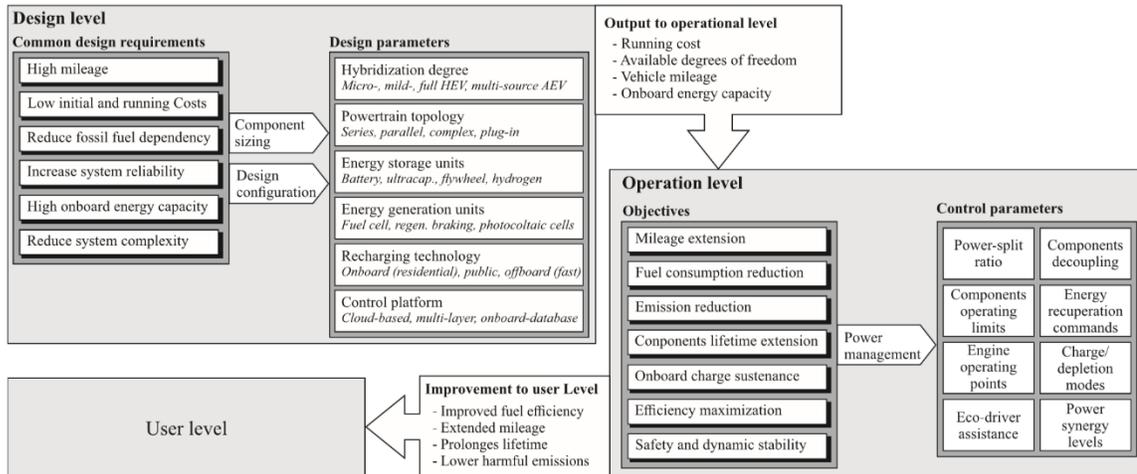


Figure 38: Design and operational objectives of Hybrid Electric Vehicles [15]

The overarching objectives under consideration when developing an HEV are categorized and discussed (Figure 38) in detail by Ahmed M. Ali et al [15]. To solve the design space exploration problem, a search algorithm is necessary that can evaluate candidates from different architectures. Design space exploration is, in general, a non-linear problem meaning that an exhaustive search approach is necessary to guarantee that the global optimum is found. However, since the search space grows exponentially with the number of components, an exhaustive search is only feasible if the search space is not too large. Here, an approach that reduces the design space while retaining near optimality is used.

### (i) Need for Design Space Reduction

The powertrain design variables considered in the DSE exercise include the transmission gear ratios and the electric motor power. Based on the vehicle parameters, a preliminary matching is performed between the transmission gear ratios and the electric machine power to meet the following requirements:

- desired acceleration and maximum speed;
- desired gradeability.

From this preliminary matching process, the range of values for both the gear ratios and the electric machine power output are obtained. For instance, for the four-speed automated manual transmission + electric machine (AMT+EM) architecture, the considered range of values is reported in Table 19.

Table 19: Design space variables, range and discretization

<b>Design variable</b>	<b>Value range</b>	<b>Discretization</b>
Gear 1	3.5 – 4.5	5
Gear 2	2.5 – 3.5	11
Gear 3	1.2 – 2.5	14
Final drive	5.13	-
Electric Machine	150 -200 kW	30

For this architecture, the size of the design space, based on the data reported in Table 19, would be 23,100. The process of evaluating the fuel economy with Dynamic Programming is computationally intensive, and becomes impractical when the design space becomes too large. Therefore, a brute-force exhaustive search is not feasible, and a reduction of the design space is needed. This is accomplished by means of a search algorithm that is able to filter out the candidates that are less likely to be optimal, as described in the next subsection. It will be shown in the following sections that with an approach called Gaussian Process Optimization, the number of evaluated candidates for this architecture is reduced

from 23,100 to 530. The performance and fuel economy calculations are then performed only on the remaining candidates.

### **2.2.5 Search Algorithms Used in DSE**

Evaluating the objective functions of each powertrain candidate is considered computationally costly, due to heavy simulations and optimizations of the powertrain fuel consumption. Therefore, the search algorithm should carefully select which candidates to evaluate to reduce the search time. The candidate selection procedure is allowed to take time if it helps to reduce the overall time of the design space exploration. A heuristic search algorithm is proposed which is able to explore relevant parts of the design space from multiple architectures while reducing the number of evaluated candidates.

#### **(i) Overview of Convex Optimization**

Willans lines scalable engine model has been used to model hybrid electric vehicles since the late 90's. PHEV life-cycle cost, fuel cost and emission cost has been studied for oversimplified vehicle model and a Peterson battery degradation model. DIRECT-method based series PHEV optimization with respect to battery size and motor size has been performed with synthesized drive cycle out of Markov chains based on collected driving data. Particle Swarm Optimization (PSO) method is applied to optimize a hybrid fuel-cell vehicle in terms of series and parallel numbers of fuel cells and supercapacitors. A comparison of four optimization methods for P2 Parallel HEV component size showed advantage of Simulated Annealing and DIRECT method over Genetic Algorithm and PSO. Optimizing the

parameters of a rule-based supervisory control strategy is studied based on a fixed HEV powertrain. Other methods used for HEV component size optimization include Parallel Chaos Optimization Algorithm (PCOA). Methods For parallel hybrid HEV with planetary gearsets, bond graphs have been abstracted to represent the gearing connection variants, and optimization of the bond graph structure leads to optimal powertrain selection.

The nature of the optimization-optimal control problem is two-fold: firstly, the powertrain parameters needs to be selected; secondly the selected powertrain needs to be properly controlled to do its best. A review of the approaches used to deal with such problem is summarized in. The approaches for attacking the co-design problem are categorized into sequential, nested, combined and alternating methods.

Combined methods combined the sizing optimization problem and vehicle optimal control in a drive cycle into a single optimal control problem to avoid running repeated vehicle simulations for all variants of powertrain architectures. Due to the mixed-integer programming nature of the problem caused by discrete switching dynamics of genset and gearshift, the problem is not solvable directly with convex optimization.

Researchers at Chalmers University has conducted a series of investigation of this combined method using convex optimization. Further simplification than the convexification model is explored, where linear programming (LP) is used assuming that the efficiencies of components are constant values, and a 4% difference in result is observed.

Convex optimization is built for sizing and continuous optimal control problem, and combined with heuristics to turn the discrete engine on/off dynamics into a switching threshold for turning on the engine.

An attempt was made by fusing a heuristic power threshold-based EMS strategy into a convexified HEV model, and optimizing the sizing of battery and genset with the control strategy at the same time cost. In this attempt however, the control strategy is heuristic and not globally optimal. In the same paper a second iterative Convex Optimization/Dynamic Programming (DP) approach is used, where DP is used to pass optimal engine on/off control trajectory for particular powertrain configuration to the problem as a fixed sequence, and the remaining sub-problem is handled by convex optimization to provide the next best powertrain configuration back to DP to generate an updated engine on/off trajectory.

## **(ii) Introduction to the Gaussian Process Optimization**

The design space exploration can be expressed as a Multi-Objective Optimization Problems (MOOP). MOOP are optimization problems where the objective is to simultaneously minimize a set of  $M$  objective functions

$$\min_{\bar{x}} (g_1(\bar{x}), g_2(\bar{x}), \dots, g_M(\bar{x}))$$

$$\text{subject to } \bar{x} \in X,$$

where  $g_i(\bar{x})$  is an objective function, and  $X$  denotes the design search space.

A candidate is called Pareto-optimal if there are no other candidates that dominates the solution, i.e. if another candidate is better with respect to all objective functions [53]. The

idea is to find the Pareto-optimal solutions in the design space that offers the best trade-offs in conflicting objectives. The set of Pareto-optimal solutions is called the Pareto-front.

Usually the knowledge of the Pareto-front requires an exhaustive search of all candidates, which in many cases are too computational demanding. A good approximation to the Pareto-front that doesn't require full exhaustive search is the "dominance" function that is only based on evaluated candidates. The function is defined as:

$$h(\bar{y}; \hat{Y}) = \frac{|\{\tilde{y} \in \hat{Y}: \tilde{y}_i \leq \bar{y}_i, \tilde{y}_i \neq \bar{y}_i, \forall i = 1, \dots, n\}|}{|\hat{Y}|} \quad (1)$$

where  $|\cdot|$  denotes the number of elements in the set. The set  $\hat{Y}$  contains the vectors  $\bar{y}$  for all evaluated candidates. The single-objective function  $h(\bar{y}; \hat{Y})$  measures the ratio of other candidates that dominates the given candidate. An advantage of the proposed measure (1) is that it is normalized with respect to the value ranges of the different objectives. Note that for Pareto-optimal candidates, the single-objective function  $h(\bar{y}; \hat{Y}) = 0$ . With the introduction of the "dominance" function, an algorithm that suggests next best candidates is also needed to keep looking for Pareto-likely candidates.

### **(iii) Candidate Selection Using Gaussian Process**

A search algorithm is necessary to identify which candidates in the design space to evaluate next in each iteration. A machine learning method called Gaussian Processes [54] is here proposed to identify which candidate to evaluate based on the previously evaluated candidates. Gaussian Processes have previously been proposed to solve global optimization

problems, see for example [55]. Here, the idea is to estimate  $h(\bar{y}; \hat{Y})$  for candidates that have not yet been evaluated. If any candidate is estimated to have  $h(\bar{y}; \hat{Y}) = 0$ , within some confidence interval, the candidate is considered relevant to be evaluated because it can potentially improve the Pareto-front. If it is estimated that  $h(\bar{y}; \hat{Y}) > 0$ , i.e. the candidate is not predicted to have better performance than the “so far” Pareto-optimal solutions, that candidate will not be considered for evaluation.

It should be noted that Gaussian Process is originally intended for problems with inherent probabilistic nature, but it is used here to help with a deterministic optimization problem. The legitimacy of this seemingly misuse is that one could treat a large deterministic optimization problem as a probabilistic one, in which the true value distribution of all candidates could never be exploited due to computational demand.

#### (iv) Gaussian Processes

A Gaussian Process (GP) is a non-parametric function that can be used to model a spatially correlated function  $f(u)$ . It uses observations of the function input  $u$  and output  $y$  to compute an estimate  $y = \hat{f}(u)$ . A GP is defined by its mean function  $\mu(u)$  and covariance function  $k(u, u')$ , which is also called a kernel function.

$$f(u) \sim GP(\mu(u), k(u, u'))$$

The conditional distribution  $p(\hat{f}|f)$  is computed as

$$A = K(u, u')K(u, u)^{-1}$$

$$P = K(u', u') - K(u', u)K(u, u)^{-1}K(u, u')$$

The covariance matrix  $K(u, u')$  is given by

$$K(u, u') = \begin{pmatrix} k(u_1, u'_1) & \cdots & k(u_1, u'_M) \\ \vdots & \ddots & \vdots \\ k(u_N, u'_1) & \cdots & k(u_N, u'_M) \end{pmatrix}$$

There are many different kernel functions to model spatial correlation. Here, the exponential kernel is used,

$$k(u, u') = \sigma^2 \exp\left(-\frac{r}{\rho}\right),$$

where  $r = \sqrt{(u - u')^T(u - u')}$  is the Euclidean distance and  $\sigma$  and  $\rho$  are tuning parameters.

The matrix  $P$  gives a confidence measure of the estimate, meaning that the estimate  $\hat{f}$  has greater uncertainties at points where the diagonal elements of  $P$  is large. By assuming that the true function  $f$  exceeds a lower bound of the confidence interval, it is possible to determine which points are most likely to be a global minimum. Since the objective is to find Pareto-optimal candidates by finding candidates such that  $h = 0$ , new candidates are selected such that the lower bound of the confidence interval is lower than zero. As more candidates are evaluated, the confidence interval close to the evaluated candidates will shrink and new candidates will be selected in other parts of the design space which has a lower confidence interval. This will help balance the search between global exploring of different areas of the design space which have not yet been explored and local exploring where many Pareto-optimal candidates have been found.

## (v) Evaluation Procedure

Gaussian Processes are used to estimate the function which maps a specific candidate  $\bar{x}$ , i.e., a set of component sizes for a given architecture, to an estimate of the distance to the Pareto front. A GP model is estimated for each powertrain architecture where the evaluated candidates, i.e. their component configurations, are input and the corresponding values  $h(\bar{y}; \hat{Y})$  are outputs in the training data.

In each iteration of the search algorithm, the GP models are updated for the architecture. Then, the GP model for each architecture is used to estimate for all remaining candidates which are likely to be Pareto-optimal, i.e. the candidates which have a confidence interval that includes zero. A new candidate  $\bar{x}$  is then selected from the architecture where the lower bound of the confidence interval has the lowest value.

If a candidate  $\bar{x}$  cannot be evaluated or is infeasible, i.e. it cannot fulfill the specified performance requirements, the single-objective function (1) has no value. These candidates are ignored when evaluating the single-objective function  $h(\bar{y}; \hat{Y})$ . However, note that it is likely that Pareto-optimal solutions are close to the set of infeasible candidates which means that the search algorithm is likely to select infeasible candidates. To avoid that the search algorithm focuses the search among the infeasible set, a penalty is added based on the distance from each candidate to the infeasible candidates. This is done by generating a second GP model for each architecture which is used to add a penalty to the estimated lower bound based on how close it is to evaluated candidates which were feasible and infeasible. The penalty is scaled to make sure that the search avoids candidates that are likely to be

infeasible but still evaluate those that are feasible. A GP model  $p = \hat{f}_p(\bar{x})$  is trained for each architecture using evaluated candidates  $\bar{x}$  as input where infeasible candidates have output one and feasible candidates have output minus one. Then, the penalty is computed as the maximum of the estimated output for a new candidate  $\hat{p} = \hat{f}_p(\bar{x}_{new})$  and zero to have a value that is non-negative and should not penalize candidates far from the infeasible candidates.

### **(vi) Initialization**

In order to make predictions which candidate to evaluate in each iteration requires there is a set of already evaluated candidates from all architectures. Therefore, the search algorithm is initialized by evaluating a randomly selected set of candidates from each architecture. The results from these candidates are then used to predict which next candidate to evaluate by the search algorithm.

### **(vii) Search Algorithm Summary**

The design space exploration search algorithm can be summarized in the following steps:

1. Define a search space including multiple architectures  $X = X_1 \cup X_2 \cup \dots \cup X_q$  where  $q$  is the number of architectures.
2. Select randomly for each architecture a number of candidates and evaluate the objective functions for these candidates.
3. Train a GP model for each architecture using the evaluated candidate component sizes as input and the value of the single-objective function (1) as output.

4. Train another GP for all architectures where feasible candidates are mapped to the value zero and infeasible candidates to a penalty value  $\gamma$ . This is used to penalize candidates that are likely to be infeasible.
5. Use the GP models to predict a lower bound of the confidence interval for all other candidates and the infeasibility penalty.
6. Select the candidate that has the lowest lower bound (after adding the infeasibility penalty). If the lower bound is larger than 0, then stop the search.
7. Otherwise, evaluate the selected candidate and go to step 3.

The iterative process involved in this exercise is illustrated in Figure 39.

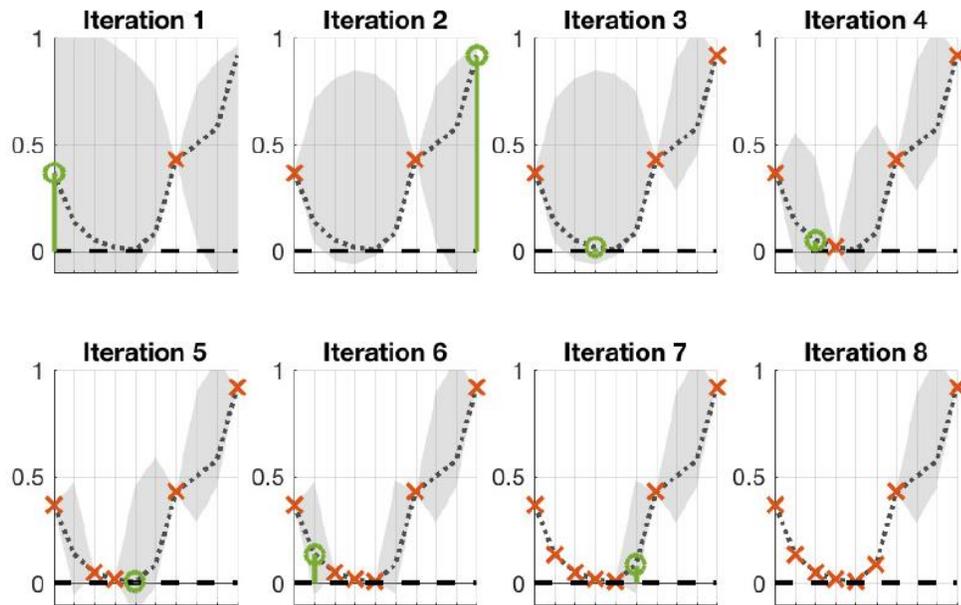


Figure 39: Arriving at the optimal candidate using the search algorithm in [44]

- Shaded area: 95% confidence interval
- Dotted line: True value

- **Red:** Evaluated candidates
- **Green:** Gaussian Process predicted values
- **Dotted line at bottom:** threshold for optimal results

The Gaussian Process Optimization could receive a speed boost if the initial candidates are close enough to the actual optimal ones in the design space. This could be achieved by utilizing the speed benefits of convex optimization to provide a set of approximately good candidates as the initial candidates for Gaussian Process Optimization.

### **2.2.6 Optimal Energy Management Strategy**

Once the design space has been reduced to a manageable size, Dynamic Programming [25] can be used to evaluate the efficiency of the remaining candidates. Since two energy sources are available on hybrid powertrains, an energy management strategy is required to control the energy flows in the HEV. In this study, DP is used to solve the optimal energy management problem to determine the sequence of system inputs that result in the lowest cost solution, i.e. the highest fuel efficiency, for each of the optimal candidates.

In any optimization problem, the cost function is the key to the problem formulation and solution. In this study, the primary goal is maximizing fuel economy while ensuring the minimum number of engine start-stops and gear shifts to keep emissions in check without a comprehensive emissions model. With this goal, the cost function  $J$  to be minimized has been formulated:

$$\min_u J = \min_u \int_{T_0}^{T_f} \dot{m}_f(x, u) dt + \Phi |\Delta u_{gen}| + \Omega |\Delta u_{gear}| dt \quad (8)$$

where  $\dot{m}_f$  is the fuel consumption in g/s,  $\Delta u_{gen}$  is the engine on-off switching event,  $\Delta u_{gear}$  is the gear change event,  $\Phi$  and  $\Omega$  are the cost penalties associated with engine on-off and gear shift, respectively.

A backward simulator is built from the sub-models described previously in section 2.1 in order to generate the optimal control strategy for the given scenario with the use of Dynamic Programming. The signal flow in the backward model is depicted in Figure 40.

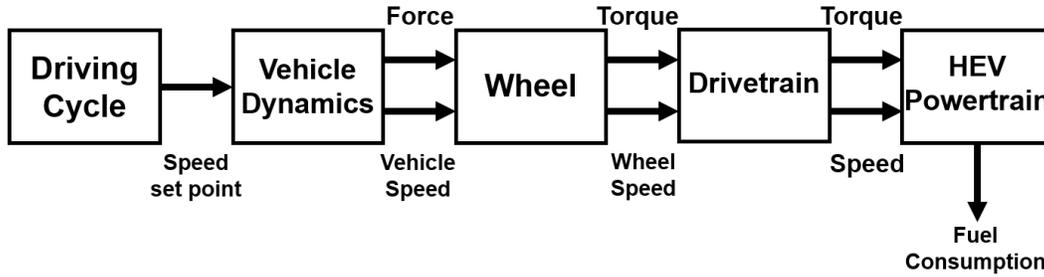


Figure 40: Backward simulator representation for a hybrid vehicle

The sequence of controls provided by DP represents the power split between the internal combustion engine and the rechargeable energy storage system at successive time steps. The cost here corresponds to the design objectives: fuel consumption and genset start/stop. The set of choices at each instant is determined by considering the state of each powertrain component and the total power request. The number of solution candidates that can be considered and evaluated is a compromise between the computational capabilities and the accuracy of the result.

The optimal power split and resultant fuel economy for each selected candidate is generated using the open source ‘dpm’ toolbox [11].

## **2.3 Simulation Results and Problem Solution**

In this section, the candidates of all the five REEV architectures are presented in detail. The performance of every architecture is evaluated after having reduced the size of the design space to a manageable level. Appropriate cost functions will be extended to all the architectures in order to perform Dynamic Programming and obtain the fuel economy of all likely candidates. The candidates with the best fuel economy will subsequently be compared against each other over parameters including freight ton efficiency, cost, weight and packaging.

### **2.3.1 Duty Cycles for the Delivery Truck**

The Design Space Exploration exercise dealt with in this paper is limited to a specified driving profile corresponding to a Class 6 pickup and delivery truck, with a REEV plugin hybrid architecture. The duty cycle in question is approximately 8 hours long, with zero geographical grade, and is representative of a start-stop driving scenario. The drive cycle that was initially synthesized for this purpose was too artificial to be considered a realistic set of scenarios, and was hence smoothed out to meet the necessary requirements for the application. The profile is depicted in Figure 41. The statistical features of the drive cycle are also reported in the Table 20.

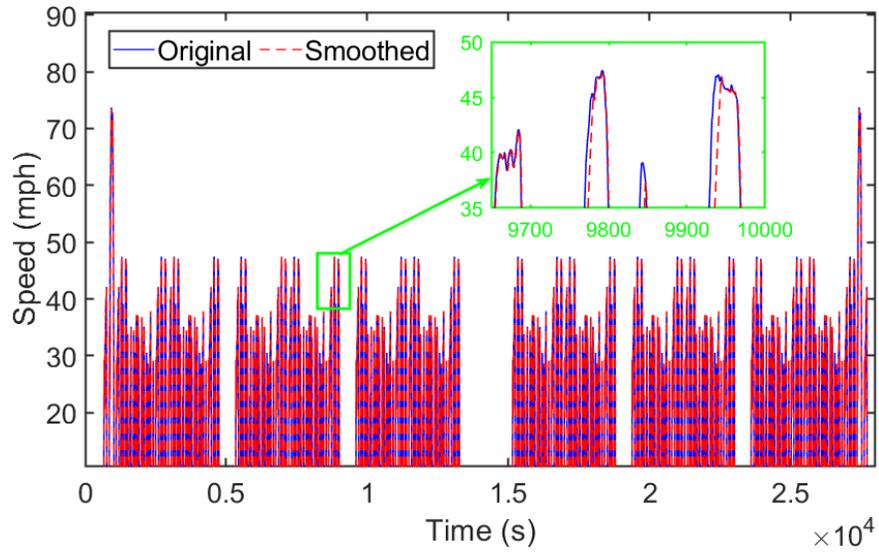


Figure 41: The selected pick-up and delivery drive cycle

Table 20: Driving cycle characteristics

Property	Original	Smoothed partial	Deviation
<b>Duration</b>	7.69 hours (27,703 sec)	1.39 hours (5,000 sec)	22703 seconds
<b>Distance covered</b>	96.73 miles	17.29 miles	79.44 miles
<b>Max Speed</b>	73.71 miles/h	73.32 miles/h	0.39 miles/h
<b>Average Speed</b>	27.79 miles/h	26.42 miles/h	1.37 miles/h
<b>Max Acceleration</b>	1.6413 m/s <sup>2</sup>	1.6413 m/s <sup>2</sup>	0

As per the benchmark set for the Class 6 pick-up and delivery truck application, the required operating points have been plotted on the electric machine map depicted in Figure 42, utilizing a direct drive architecture with a final drive ratio of 5.13. The size of the working points on the plot represents the power demand from the motor. This gives an early indication of the continuous and peak power requirement for the electric machine.

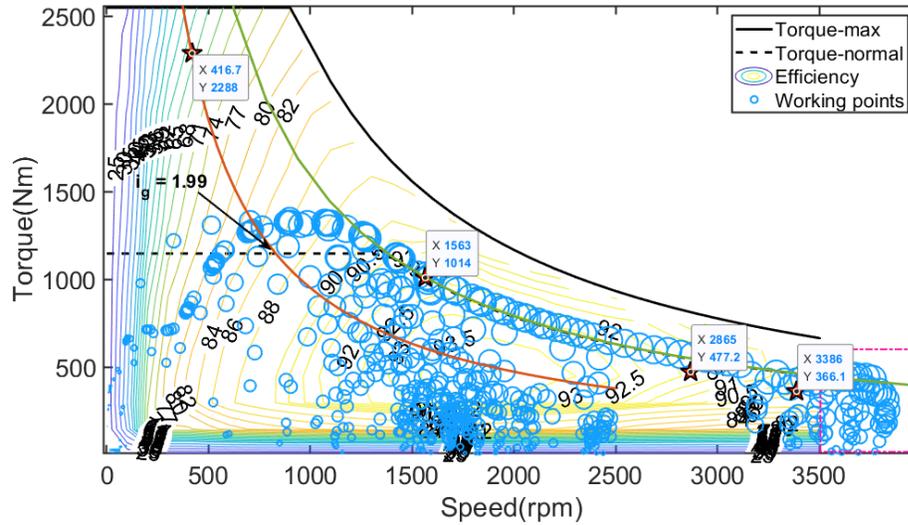


Figure 42: Electric motor map with operating points for the selected drive cycle

### 2.3.2 Constraints in the Optimization Process

The assumptions for the weight and cost of the powertrain, and the methodology for arriving at the required values are elaborated in Tables 21 and 22. The specifications quoted in these tables are obtained through market research for the corresponding product as well as by using advice from the industry partners involved in the CERC Truck Consortium.

Table 21: Weight and cost of the REEV powertrain

Components	Weight
ICE Engine	300 kg
Final Drive	100 kg
Genset	260 kg
AMT Transmission	50 + 25*gears kg
e-axle Transmission	25*gears kg
Battery	7 kg/kWh
Electric Motor (Interpolated)	[130kW, 195kg; 212kW, 340kg]
Other*	3000 kg

Components	Cost
ICE Engine	10,000 \$
Final Drive	1,000 \$
Genset	15,000 \$
AMT Transmission	2000 + 500*gears \$
e-axle Transmission	2000 + 500*gears \$
Battery	300 \$/kWh
Electric Motor	50 \$/kW
Fuel (gas)	2.80 \$/gal
Electricity	0.15 \$/kWh

Table 22: Calculation of the cost and weight of the REEV powertrain:

Components	Definition
Powertrain Weight	$w_{pt} = w_{genset} + w_{batt} + w_{motor} + w_{trans} + w_{fd}$
Curb Weight	$w_{GCWR} = w_{pt} + w_{other}$
Gross Weight	$w_{GVWR} = w_{GCWR} + w_{payload}$
Powertrain Cost	$c_{pt} = c_{genset} + c_{batt} + c_{motor} + c_{trans} + c_{fd}$
Fuel Cost	$c_{fuel} = u_{fuel} \times \frac{m_{fuel} L_{to\_Gal}}{1000 \rho_{gas}} \times N_{trip} \times 365 \times N_{year}$
Electricity Cost	$c_{elec} = u_{elec} \times E_{elec} \times N_{trip} \times 365 \times N_{year}$
Energy Cost	$c_{elec} = u_{elec} \times E_{elec} \times N_{trip} \times 365 \times N_{year}$
Operation Cost	$c_{operation} = c_{pt} + c_{energy}$

1. The maximum available payload is denoted as  $w_{payload}$
2. The electric energy used per run is denoted as  $E_{elec}$
3. The number of years of operation is assumed to be  $N_{year} = 10$
4. The number of daily trips run by the truck is denoted by  $N_{trip}$

Based on the chosen vehicle parameters, the minimum powertrain requirements are calculated in order to meet the performance benchmarks set by the industry partner. The broad list of powertrain requirements can be summarized as: Minimum power for acceleration and max speed; RPM requirement for presumed reduction gear ratio; Minimum

power and torque requirement for grade performance. The specific list of requirements and the corresponding powertrain requirements are tabulated in Table 23.

Table 23: Powertrain performance benchmark

<b>Presumed Axle Reduction Gear Ratio Values</b>	4.17	5.13	6.23
	Target	Time	Power Required
<b>Acceleration Time</b>	0-20 mph	9 sec	54.91~102.11 kW*
	0-30 mph	14 sec	82.83~151.11 kW
	0-40 mph	22 sec	102.24~179.49 kW
	0-50 mph	36 sec	114.05~187.75 kW
	25-35 mph	9 sec	Powertrain Specific
<b>Maximum Speed</b>	70 mph	711.45 rpm ( $i_0 = 1$ )	89.91 kW
		2966.7 rpm ( $i_0 = 4.17$ )	
		3815. rpm ( $i_0 = 5.13$ )	
		4432.3 rpm ( $i_0 = 6.23$ )	

The required power for acceleration events is calculated as the sum of two terms: a constant force term needed for acceleration, and a constant power term needed to sustain the attained speed. The expressions needed to determine the power demand are obtained from [56]. The left-hand side integral is broken into two parts: the 0 to  $v_b$  constant force operation and the  $v_b$  to  $v_f$  constant power operation:

$$m \int_0^{v_b} \frac{dv}{\frac{P_m}{v_b}} + m \int_{v_b}^{v_f} \frac{dv}{\frac{P_m}{v}} = t_f$$

Solving for  $P_m$ , we can get:

$$P_m = \frac{\delta m}{2t_f} (v_b^2 + v_f^2)$$

$$\left\{ \begin{array}{l} P_{m\_low} = \lim_{v_b \rightarrow 0} P_m + P_f = \frac{\delta m v_f^2}{2 t_f} + P_f \\ P_{m\_high} = \lim_{v_b \rightarrow v_f} P_m + P_f = \frac{\delta m v_f^2}{t_f} + P_f \\ P_f = v_f \left( mg C_{f0} + mg v_f C_{f1} + \left( \frac{1}{2} \rho C_D A_w + C_{f2} \right) v_f^2 \right) \end{array} \right.$$

Using the same approach taken for meeting the acceleration benchmarks, the powertrain requirements are estimated for meeting the gradeability benchmarks as in Table 24.

Table 24: Gradeability benchmarks for the powertrain

Speed (mph)	Grade (%)	Required Wheel Torque (Nm @ rpm)	Axle Reduction Gear Ratio	Motor Torque (Nm) and Speed (rpm) with gear ratio	Required Power (kW)
8	30	11727.70@81.31	$i_0 = 4.17$	2812.4@339.06	99.86
			$i_0 = 5.13$	2286.1@417.11	
			$i_0 = 6.23$	1882.5@506.56	
30	12	5197.53@304.91	$i_0 = 4.17$	1246.4@1271.5	165.96
			$i_0 = 5.13$	1013.2@1564.2	
			$i_0 = 6.23$	834.27@1899.6	
55	4	2445.80@559.00	$i_0 = 4.17$	586.52@2331	143.18
			$i_0 = 5.13$	476.76@2867.7	
			$i_0 = 6.23$	392.58@3482.6	
65	2	1876.08@660.63	$i_0 = 4.17$	449.90@2754.8	129.79
			$i_0 = 5.13$	365.71@3389.0	
			$i_0 = 6.23$	301.14@4115.7	

A graphical representation of the preliminary matching for the axle reduction ratios based on the speed and gradeability requirements is depicted in Figure 43. This figure indicates that for the range of maximum torque of the EM in the [2400, 3200] Nm bracket, and the maximum EM speed in the bracket of [3000, 4500] RPM, the ideal axle reduction gear ratio lies in the range of [3, 6.5].

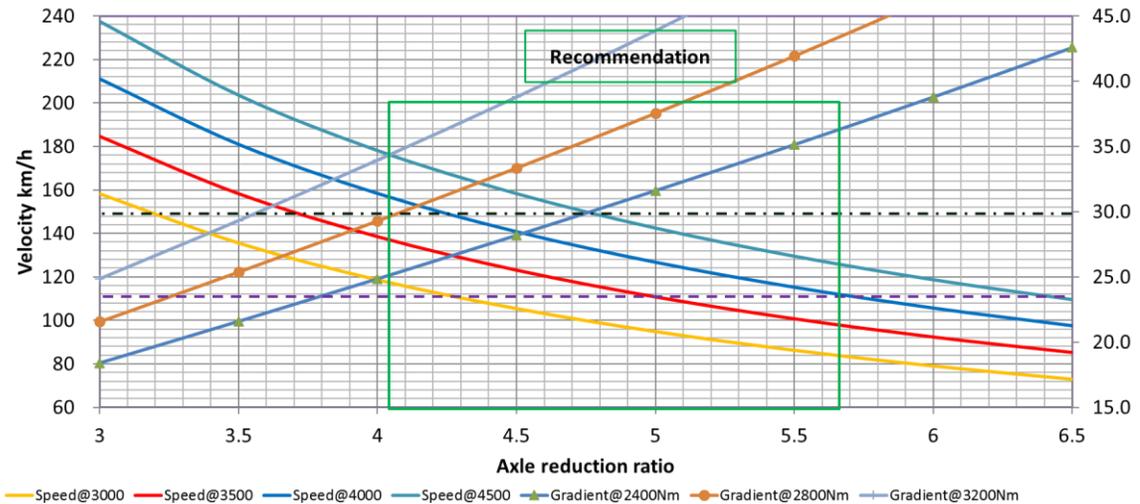


Figure 43: Preliminary matching for the axle reduction ratios with speed and grade

### 2.3.3 Conventional Powertrain

The conventional powertrain of the Class 6 delivery truck has a gasoline engine, with a 5-speed automatic transmission. The powertrain is represented in Figure 44, and the component characteristics are reported in Table 25. For this architecture the state and control variables are listed in Table 26, with fuel cost penalties added for engine start/stop and gear change in the objective function to minimize fuel consumption.

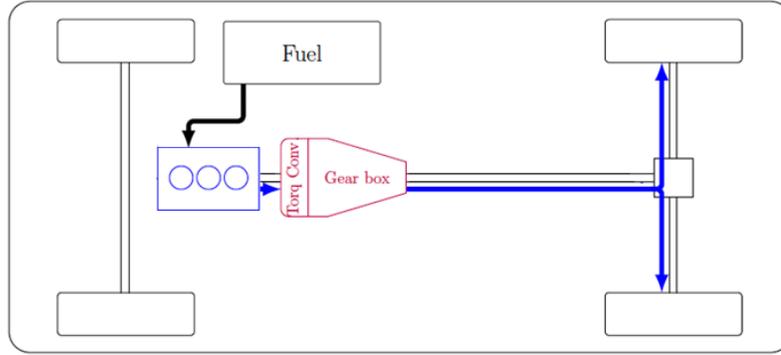


Figure 44: Conventional powertrain architecture

Table 25: Conventional powertrain specification range

Design Variable	Value
Engine Power (kW)	278
Engine Top Speed	5000 RPM
1 <sup>st</sup> gear ratio	3.1
2 <sup>nd</sup> gear ratio	1.81
3 <sup>rd</sup> gear ratio	1.41
4 <sup>th</sup> gear ratio	1
5 <sup>th</sup> gear ratio	0.71
Final Drive	5.13

Table 26: Conventional powertrain control parameters and cost function

<b>State Variables</b>	Engine State, $s_{gen}$ , ON or OFF	Current Gear, $s_{gear}$ (5-speed + Final drive)
<b>Control Variables</b>	Engine ON/OFF, $u_{gen}$	Selected Gear, $u_{gear}$
<b>Objective Function</b>	$\min_u J = \min_u \int_{T_0}^{T_f} \dot{m}_f(x, u) dt + \Phi  \Delta u_{eng}  + \Omega  \Delta u_{gear} $	

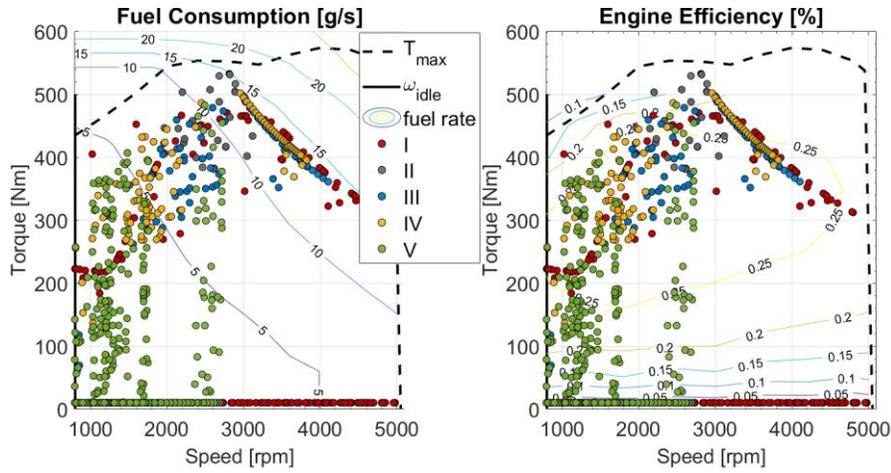


Figure 45: Engine efficiency and fuel consumption for the conventional powertrain

Fuel consumption for the conventional powertrain over the CERC P&D drive cycle is depicted in Figures 45 and 46. The conventional gasoline engine vehicle consumes 50.8 kg of fuel, which serves as a benchmark for the HEV architectures to be studied going forward.

Table 27 lists the significant specifications and performance of the IC engine truck.

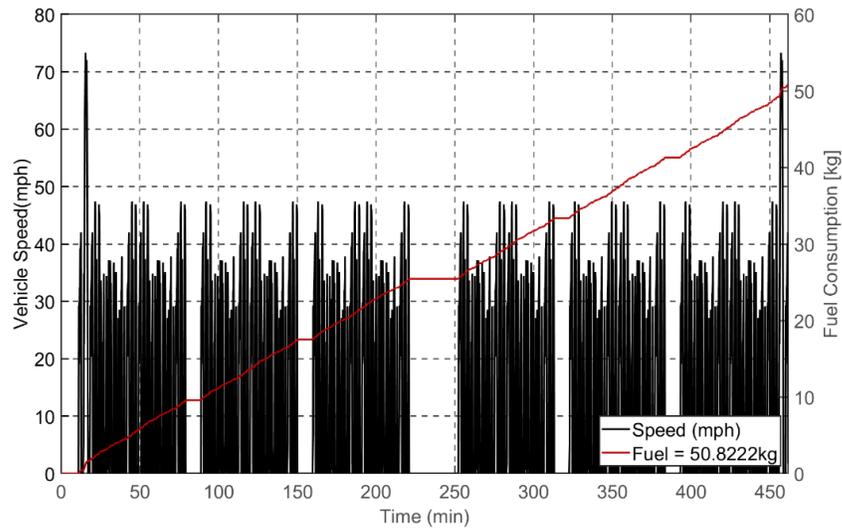


Figure 46: Fuel consumption for the conventional powertrain

Table 27: Conventional IC engine powertrain specifications

Final Drive Ratio	Trans Ratio	Fuel Economy (mpg)	Powertrain Weight (kg)	Powertrain Cost (\$)	10 Yrs. Energy Cost	Freight ton eff.
5.13	3.1, 1.8, 1.4, 1.0, 0.7	7.1	700.0	15,500.0	150,327.0	36.6

### 2.3.4 Hybrid Powertrains

The five proposed hybrid powertrain architectures are presented in more detail in the following sub-sections. The common features are the genset and the plug-in capability, while they differ in the size of the electric motor, transmission and driveline details.

#### (i) Two-speed e-Axle

The two-speed e-Axle powertrain is driven by an electric machine through a 2-speed transmission. The powertrain is shown in Figure 47, while the range extender genset and transmission characteristics are reported in Table 28. The state and control variables for this architecture are listed in Table 29, with fuel cost penalties added for engine start/stop and gear change in the objective function to minimize fuel consumption.

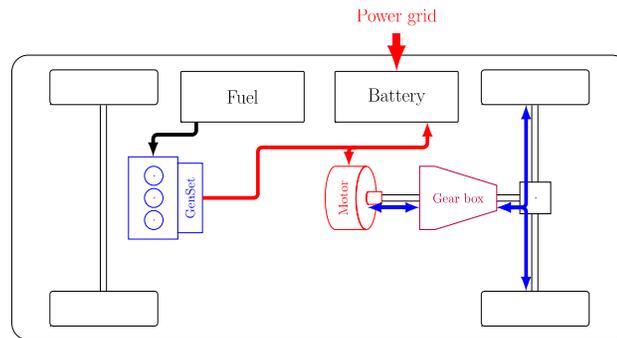


Figure 47: Two-speed e-Axle powertrain architecture

Table 28: Two-speed e-Axle powertrain parameters

Design Variable	Value Range
Motor Continuous Power (kW)	150-200
1 <sup>st</sup> gear ratio	1.5-3
2 <sup>nd</sup> gear ratio	1
Final Drive Range	5.13
Engine Top Speed	4000 RPM

Table 29: Two-speed e-Axle powertrain control parameters and cost function

State Variables	Genset State, $s_{gen}$ , ON or OFF	Current Gear, $s_{gear}$ (2-speed+Final drive)	State of Charge of the Battery, SOC
Control Variables	Engine ON/OFF, $u_{gen}$	Selected Gear, $u_{gear}$	Genset Power Level, $p_{gen}$
Objective Function	$\min_u J = \min_u \int_{T_0}^{T_f} \dot{m}_f(x, u) dt + \Phi  \Delta u_{eng}  + \Omega  \Delta u_{gear} $		

Multi-objective DSE is performed on two-speed e-axle candidates, shown in Figure 48.

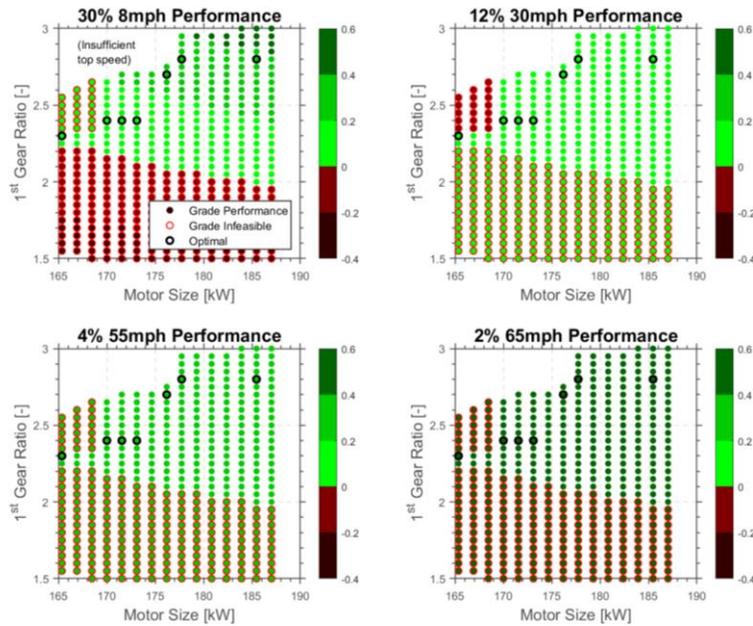


Figure 48: Pareto front generation

From the design space exploration for the 2-speed e-axle architecture, a number of observations are made. Different grade requirements have different electric machine size preferences. A grade of 30% at 8mph requires larger overall gear ratio (best: 3×5.13), a grade of 12% at 20mph requires smaller overall gear ratio (best: 2×5.13), a 4% grade at 55mph and 2% grade at 65mph requirements are met easily, and are only sensitive to the electric motor size, not the gear ratios. They can both be satisfied through a direct drive from the electric machine to the wheels. The ease of satisfying the gradeability requirement is indicated by the torque margin factor.

$$\text{Torque margin factor} = \frac{\text{Available Torque}}{\text{Required Torque}} - 1$$

The study of the grade capability requirements has led to a number of conclusions. With the current gear ratio selection of [1.5-3] and current motor range [150-200] kW, the 3<sup>rd</sup> and 4<sup>th</sup> grade requirements are easy to meet. The 30% grade at 8mph requirement asks for a larger gear ratio, and with an electric motor of 165kW continuous power, this ratio is at least 2.25. With the increase in motor size, the required gear ratio will drop to 2.0. The 12% grade at 30 mph requirement asks for higher EM power, and delivers the best performance when the motor has the highest power delivery at 30 mph. This requirement is generally met unless a highly undesirable combination is chosen, such as: the electric motor is under 170kW and the 1<sup>st</sup> gear ratio is above 2.3.

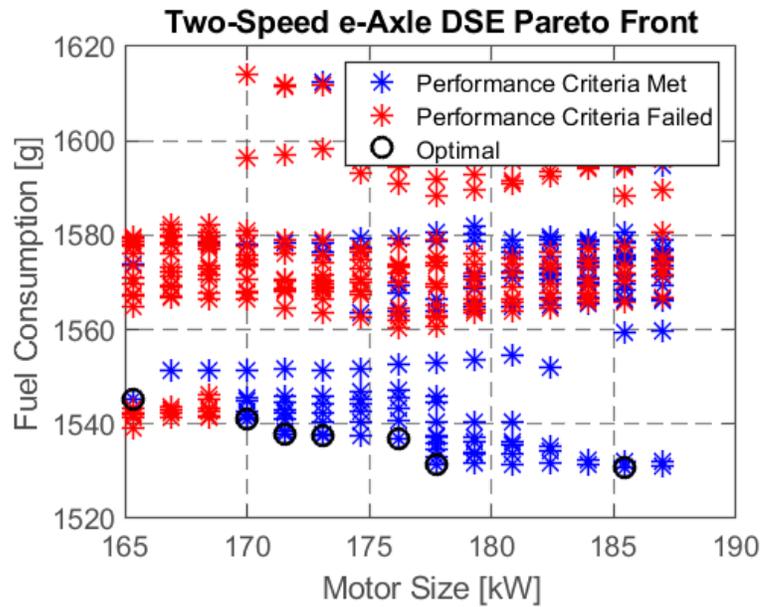


Figure 49: Pareto front generation for two-speed e-axle candidates

An illustration of the pareto front generation for the 2-speed e-axle architecture is shown in Figure 49 as an example. In this scenario, a trade-off exists between motor size and fuel consumption. Among the most optimal candidates, there is a fuel consumption difference of 15g and a motor size difference of 20 kW. The worst candidate has 85g more fuel consumption than the best candidates.

A minimum motor power of 165 kW is required to meet drive cycle and grade requirement benchmarks. Moreover, a higher 1st gear ratio offers better fuel economy. In order to meet grade requirements, gear ratios around 2.3 are best suited for all four designated grade requirements even with a relatively small motor. This observation is evident from the gear ratio vs motor size plot in Figure 50.

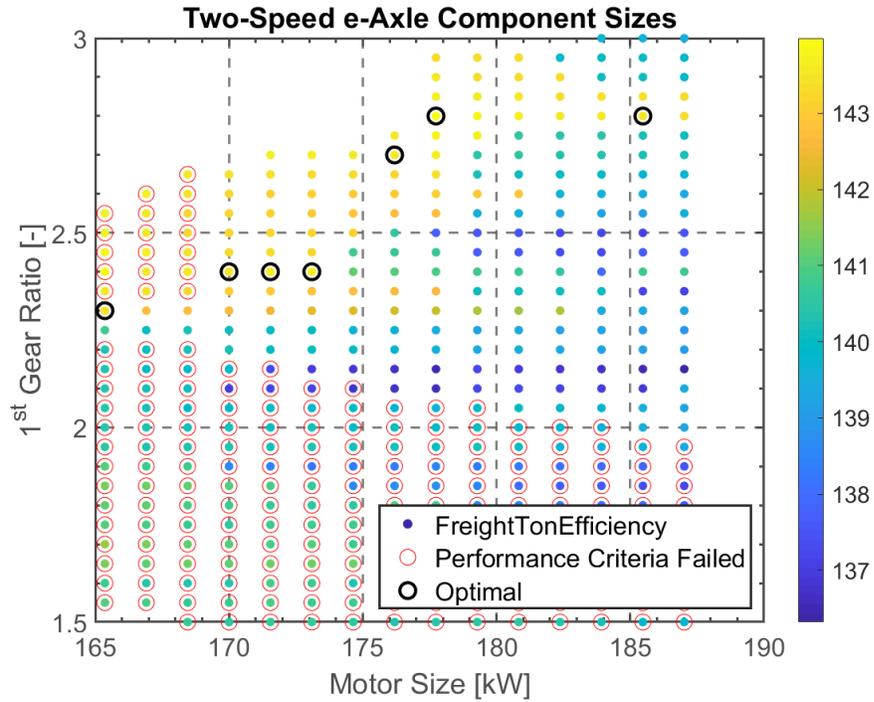


Figure 50: Optimal candidate selection for two-speed e-axle candidates

Table 30: Two-speed e-axle – best candidate; genset penalty analysis

Fuel Consumption Comparison for Different Genset Start Penalty								
Iteration	Discharge Limit	Charging Limit	Penalty for Genset Start [g fuel]	Number of Genset Start	Fuel Consumption [kg]	Final SOC [%]	Fuel Consumption for Genset Start Penalty [g]	Additional Idle Fuel Loss Due to High Penalty on Engine Start [kg]
1	2C	1C	10	190	14.08	19.38	12.18	0
2	2C	1C	20	165	15.87	19.44	12.57	0.39
3	2C	1C	30	143	17.41	19.43	15.51	3.33

As the fuel penalty increases, the number of genset start-stop events will drop, but on the other hand, there will be more unnecessary idling time when the fuel is not converted into useful work. When the fuel penalty for start-stop is 30g instead of 10g, a considerable 3.33 kg of fuel is wasted, at the cost of 47 less genset starts, from 190 starts to 143 starts.

Motor efficiency falls mostly in either very high (90-95%) regions, or very low (<80%) regions. The room for low efficiency region performance improvement is limited by flexibility of gear ratios, and characteristic of drive cycle. 39% time in 1<sup>st</sup> gear, 61% time in 2<sup>nd</sup> gear, as shown in Figure 51.

Table 31: Electric machine operating time against efficiency

Motor efficiency (%)	Operation time (%)
95-99	0
90-95	27.78
85-90	5.71
80-85	3.12
<80	63.38

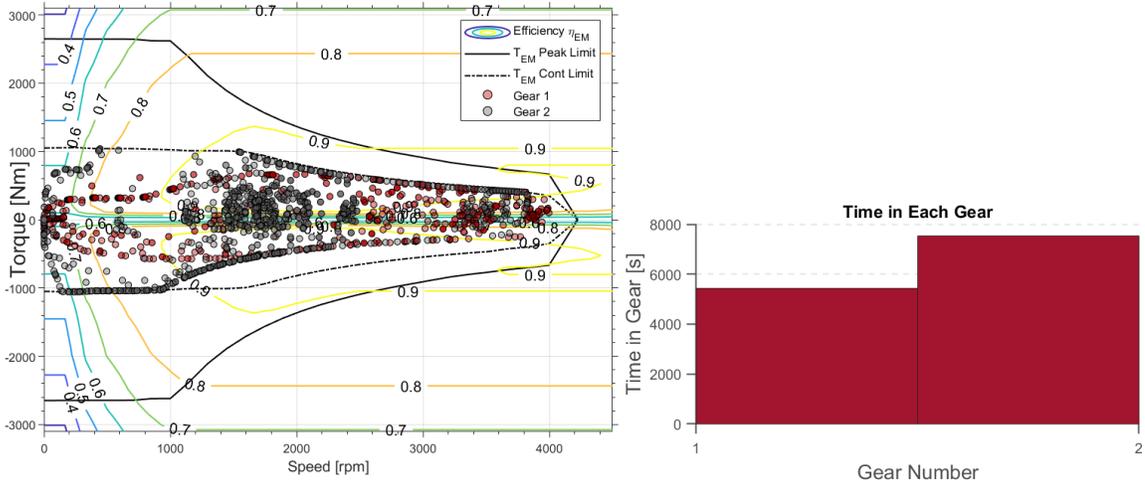


Figure 51: EM optimal operating points and gear shifts

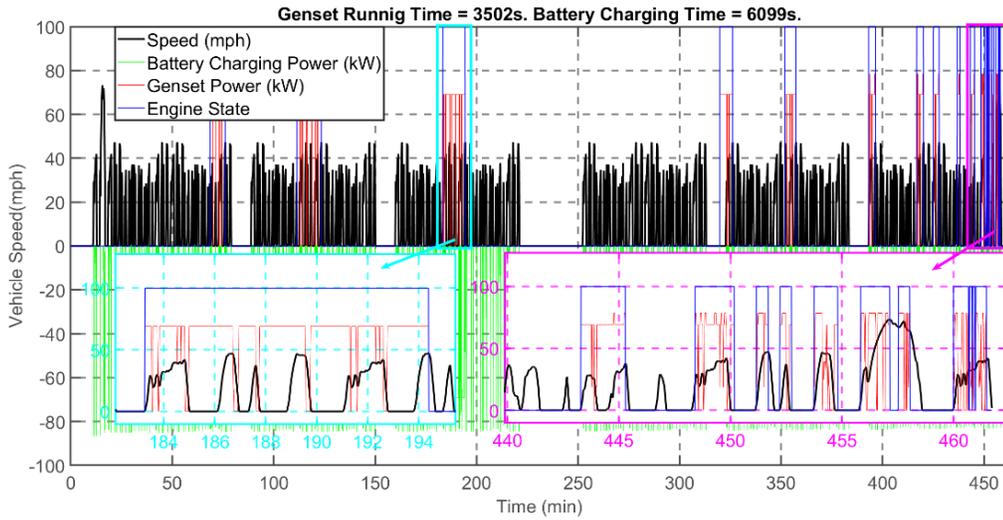


Figure 52: Genset power, engine state, battery power for the speed profile

A number of optimal powertrain operation properties are summarized below.

- Discharging C-rate: 10C
- Charging C-rate: 1C
- Engine on: 50.87 min

- Battery charging: 101.65 min
- Charging from Engine: 44.3 min
- Engine turns on 20 times
- Idling events are seen during hard regen braking

**(ii) Three-speed AMT + Electric Motor**

The three-speed AMT+EM powertrain, depicted in Figure 53, is driven by an electric machine through a 3-speed automated manual transmission, with a genset range extender. The characteristics of the components are listed in Table 32. The state and control variables, and the cost function for this architecture are listed in Table 33.

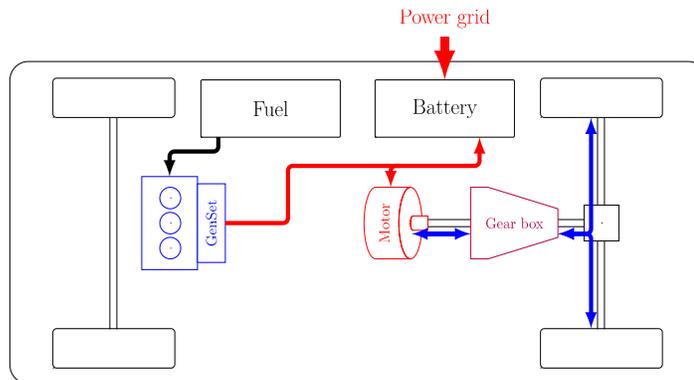


Figure 53: Three-speed AMT+EM powertrain architecture

Table 32: Three-speed AMT+EM powertrain parameters

Design Variable	Value Range
Motor Continuous Power (kW)	150-200
1 <sup>st</sup> gear ratio	2.5-4
2 <sup>nd</sup> gear ratio	1.2-2.5
3 <sup>rd</sup> gear ratio	1
Final Drive Range	5.13
Engine Top Speed	4000 RPM

Table 33: Three-speed AMT+EM powertrain control parameters and cost function

<b>State Variables</b>	Genset State, $s_{gen}$ , ON or OFF	Current Gear, $s_{gear}$ (3- speed+Final drive)	State of Charge of the Battery, SOC
<b>Control Variables</b>	Engine ON/OFF, $u_{gen}$	Selected Gear, $u_{gear}$	Genset Power Level, $p_{gen}$
<b>Objective Function</b>	$\min_u J = \min_u \int_{T_0}^{T_f} \dot{m}_f(x, u) dt + \Phi  \Delta u_{eng}  + \Omega  \Delta u_{gear} $		

Multi-objective DSE is performed on three-speed AMT+EM candidates, shown in Fig 54.

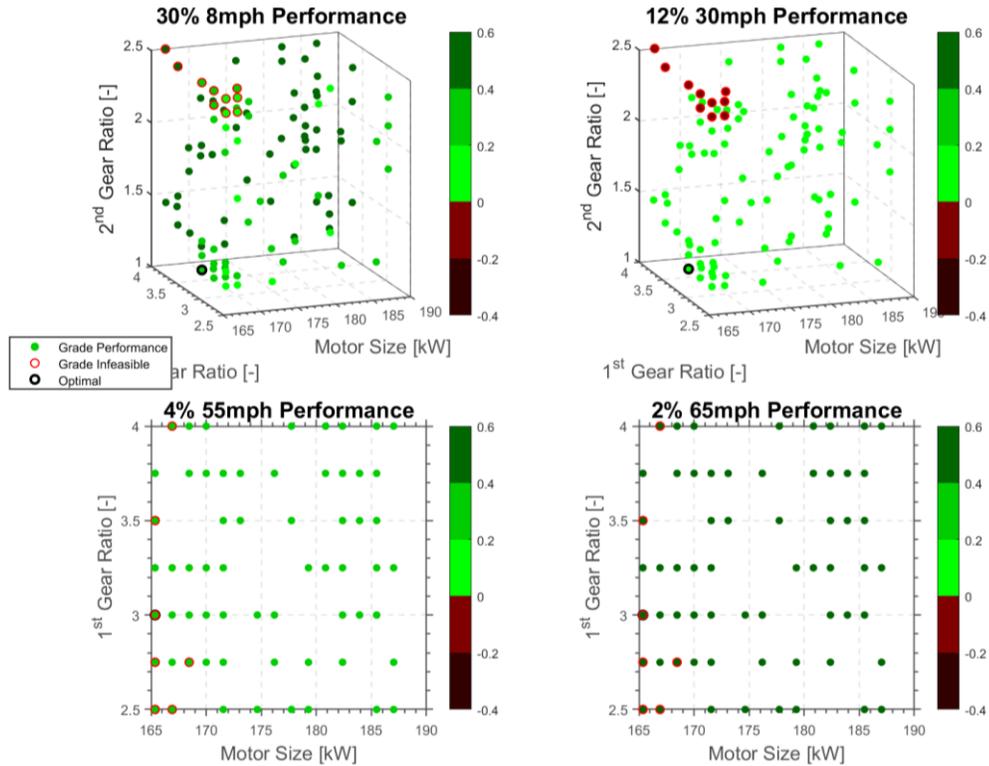


Figure 54: Pareto front generation

From the design space exploration for the 3-speed e-axle architecture, a number of observations are made. Different grade requirements have different electric machine size preferences. The 30% grade at 8mph requires a large 1<sup>st</sup> gear ratio, and has no dependence on 2<sup>nd</sup> gear ratio. The 12% grade at 20mph requirement is focused on the 2<sup>nd</sup> gear ratio, slightly preferring a smaller 2<sup>nd</sup> gear ratio. The 4% grade at 55mph and 2% grade at 65mph requirements are met easily, and is only sensitive to the electric motor size, not the gear ratios. They can both be satisfied by a direct drive (3<sup>rd</sup> gear) to the wheels.

The study of the grade capability requirements has led to a number of conclusions. With the current gear ratio setup at [2.5-4] and [1.2-2.5], and current electric motor range [150-200] kW, the 3<sup>rd</sup> and 4<sup>th</sup> grade requirements are easy to meet. The 1<sup>st</sup> grade requirement is met by all candidates, because the 1<sup>st</sup> gear ratio starts at 2.5, which is more than enough for the 1<sup>st</sup> grade requirement for any motor size. The 2<sup>nd</sup> grade requirement is focused on the gear ratio of 2<sup>nd</sup> gear, because, in order to meet that requirement, the vehicle has to be in 2<sup>nd</sup> gear, not 1<sup>st</sup> or 3<sup>rd</sup>. Furthermore, the requirement on 2<sup>nd</sup> gear ratio is similar to the two-speed e-axle case, where a slightly smaller gear ratio is preferred, and when motor size is less than 170kW, the 2<sup>nd</sup> gear ratio should be less than 2.4. In general, the grade performance requirement is not a concern as long as the 1<sup>st</sup> gear ratio is taller than 2.4, and the 2<sup>nd</sup> gear ratio is shorter than 2.4.

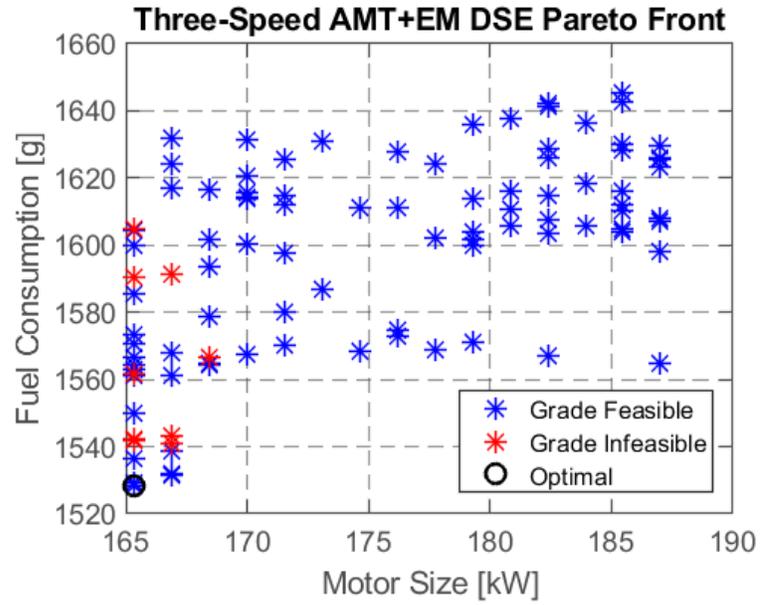


Figure 55: Candidate feasibility study for 3-speed AMT+EM

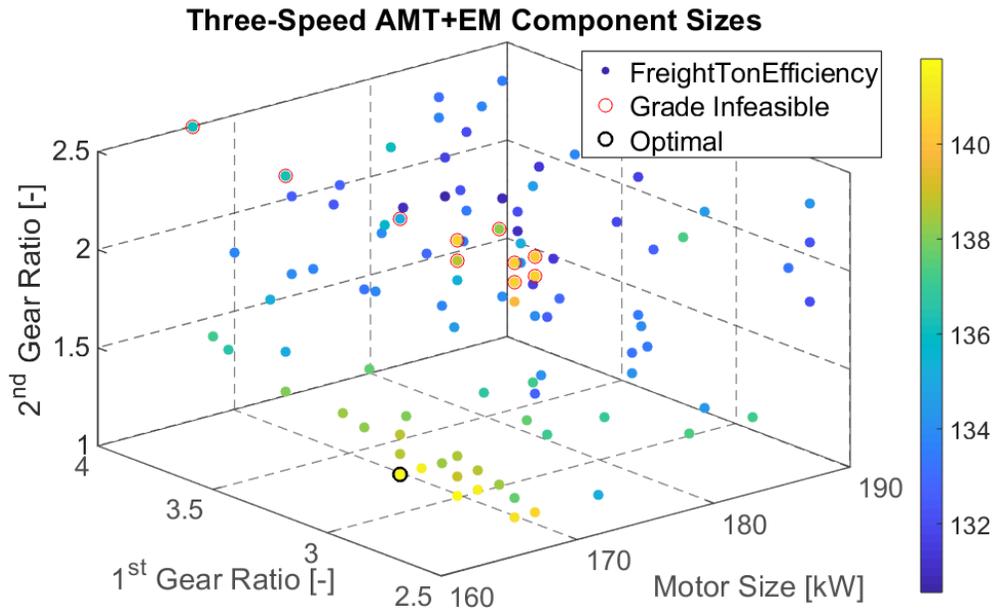


Figure 56: Optimal component sizes

A single best solution stands out as a result of the converging tendency of smaller motor and lesser fuel consumption, and gear ratio optimization for fuel consumption. Transmission ratio provided enough range. Only a few combinations fail the grade test. The worst candidate has 117g more fuel consumed than the best candidate. A minimum 165kW motor is required for drive cycle and grade requirements. Most gear ratio combinations could meet grade performance requirements. Optimal result pointed to one single candidate.

The choice of gear ratio is solely targeted at optimizing fuel consumption with the smallest possible electric motor, as grade requirements are out of the picture. For the 1<sup>st</sup> gear ratio, the choice gear ratio value 3 could be explained as the best option to place the low speed low torque operations into an optimal region on the motor efficiency map. For 2<sup>nd</sup> gear ratio, the choice of ratio equal to 1.2 could be explained as the tendency to require more diversified small gear ratios slightly above 1 in order to improve motor efficiency for low speed low torque operations.

Motor efficiency falls mostly in either very high (90-95%) regions, or very low (<80%) regions. The room for low efficiency region performance improvement is limited by flexibility of gear ratios, and characteristic of drive cycle. Timewise, 3% is spent in 1<sup>st</sup> gear, 33% in 2<sup>nd</sup> gear, 64% in 3<sup>rd</sup> gear as indicated in Figure 57.

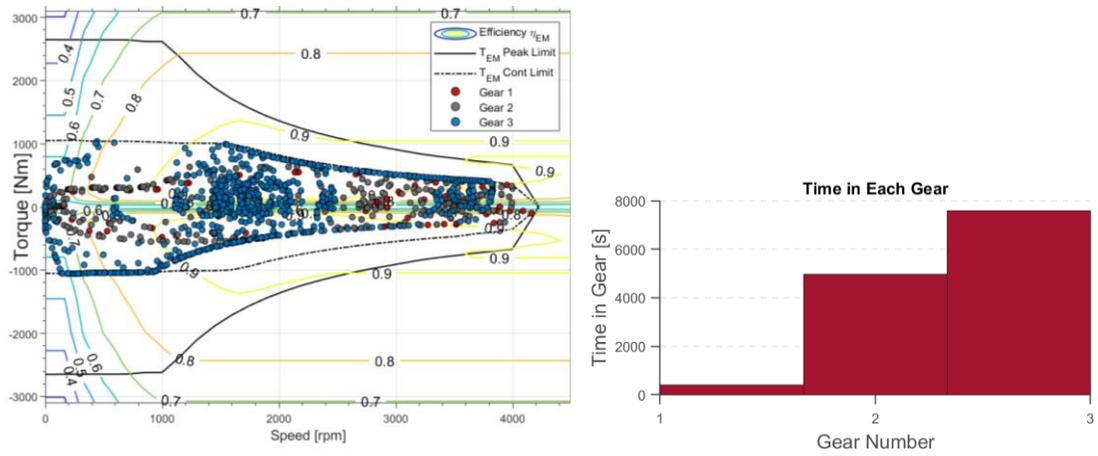


Figure 57: EM optimal operating points and gear shifts

Table 34: Electric machine operating time against efficiency

Motor efficiency (%)	Operation time (%)
95-99	0
90-95	27.85
85-90	5.81
80-85	3.07
<80	63.27

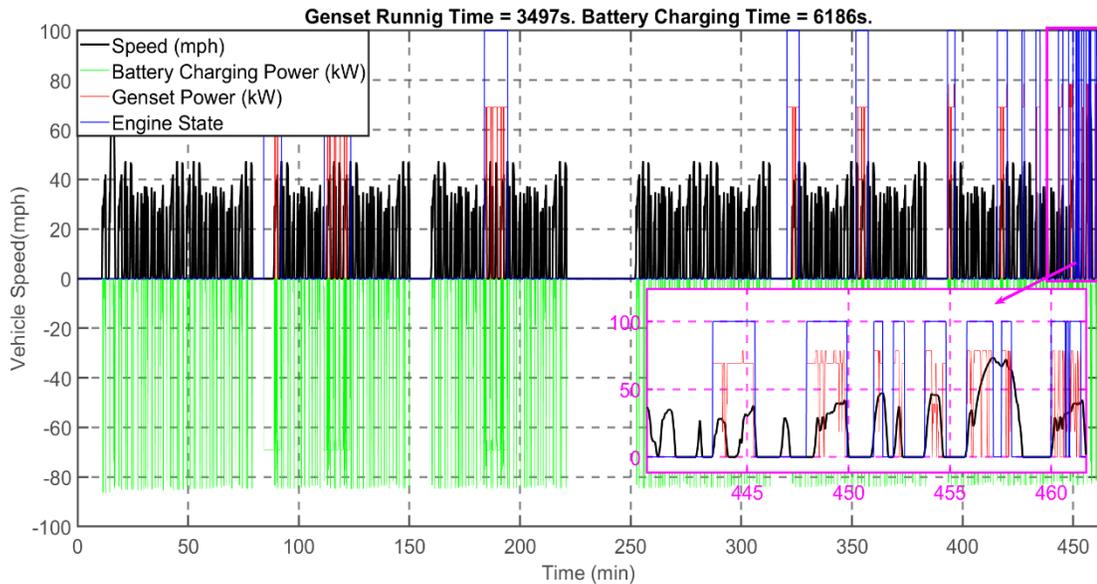


Figure 58: Genset power, engine state, battery power for the speed profile

A number of optimal powertrain operation properties are summarized below.

- Discharging C-rate: 10C
- Charging C-rate: 1C
- Engine on: 58.28 min
- Battery charging: 103.1 min
- Charging from Engine: 45.0 min
- Engine turns on 21 times
- Idling events are seen during hard regen braking

### (iii) Four-speed AMT + Electric Motor

The four-speed AMT+EM powertrain, see Figure 59, is driven by an electric machine through a 4-speed automated manual transmission, with a genset range extender. The



Multi-objective DSE is performed on four-speed AMT+EM candidates, shown in Fig 60.

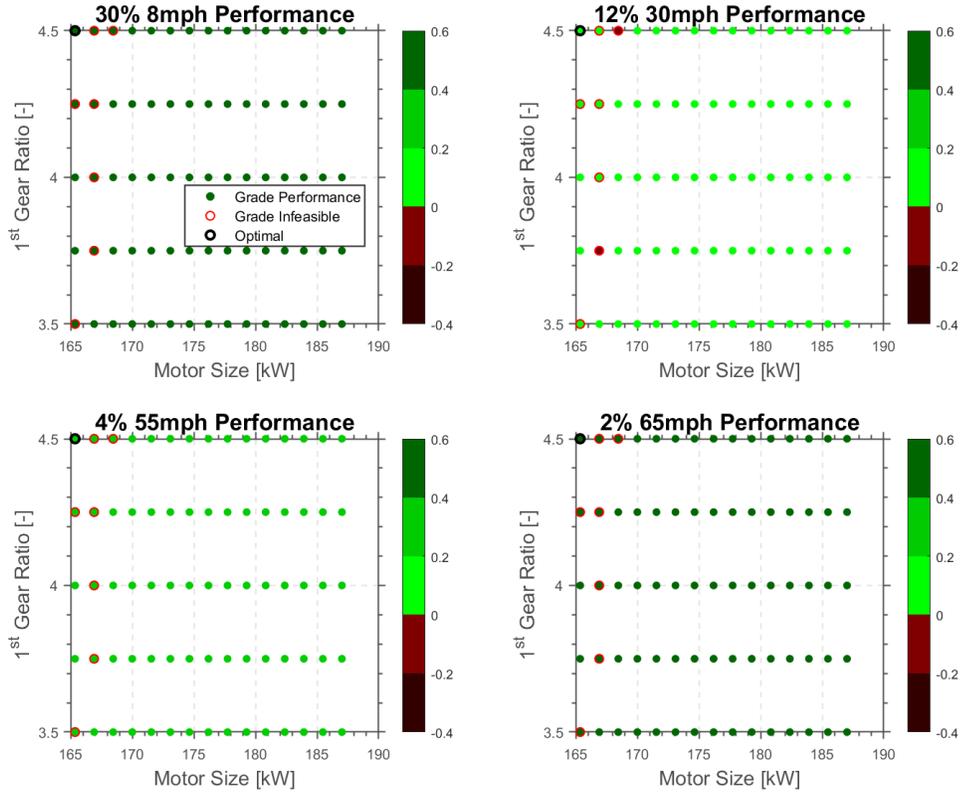


Figure 60: Generation of Pareto-optimal candidates

From the design space exploration for the 4-speed e-axle architecture, a number of observations are made. Similar to the case for 3-speed architecture, grade requirements are generally met. Only a few cases exist where the 12% grade at 30mph requirement is not met, due to a lack of gear ratios below 2.4. The 2<sup>nd</sup> grade performance benchmark (12% grade at 30mph) is the most challenging requirement due to its power demand, the best TAF for it is 16%.

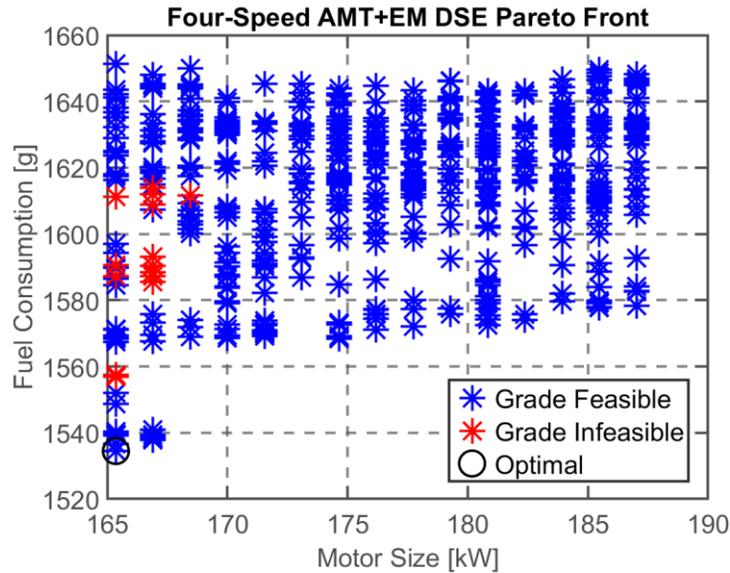


Figure 61: Four-speed AMT+EM Pareto front

The study of the grade capability requirements has led to a number of conclusions. A single best solution stands out as a result of the converging tendency of smaller electric motor size and lesser fuel consumption, and gear ratio optimization for fuel consumption. The selected range of gear ratios proved to be sufficient. Only a few combinations fail the gradeability test. The worst candidate has 117g more fuel consumed than the best candidate. A minimum continuous power of 165kW is required for the electric machine to meet the drive cycle and grade requirements. Most gear ratio combinations could meet the gradeability performance requirements. Compared to three-speed AMT+EM architecture, the four-speed version doesn't provide noticeable energy cost drop. For the 1<sup>st</sup> gear ratio, the choice of value at 4.5 could be explained as the best option in order to place the extreme low speed low torque operations into an optimal region on the motor efficiency map.

For the 2<sup>nd</sup> gear ratio, the choice of value at 3.0 could be considered as the best option to place the low speed low torque operations into an optimal region on the motor efficiency map. For the 3<sup>rd</sup> gear ratio, the choice of 1.2 could be explained as the tendency to require more diversified small gear ratios slightly above 1 in order to improve motor efficiency for high speed operations by slightly reducing motor speed and increasing torque.

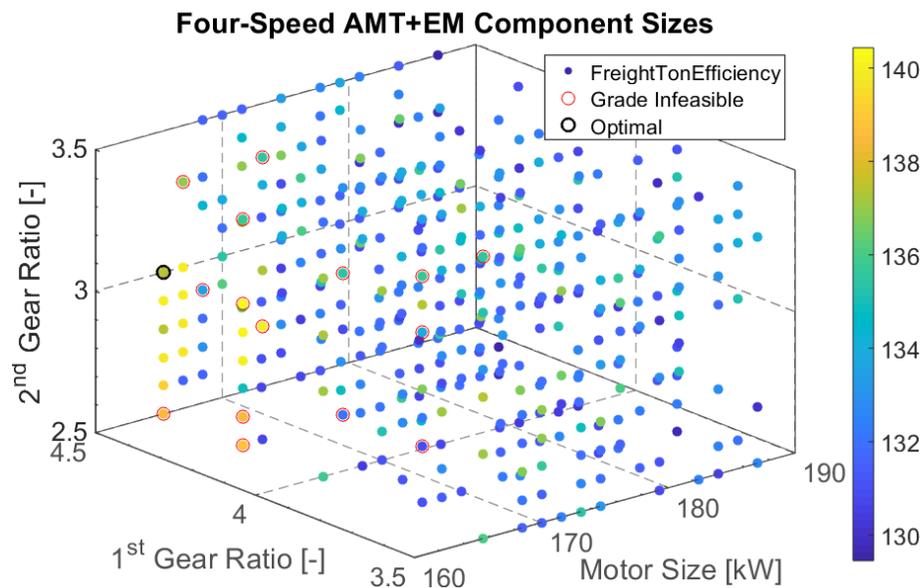


Figure 62: Optimal component sizes

Motor efficiency falls mostly in either very high (90-95%) regions, or very low (<80%) regions. Timewise, 1% is spent in 1<sup>st</sup> gear, 23% in 2<sup>nd</sup> gear, 46% in 3<sup>rd</sup> gear, 30% in 4<sup>th</sup> gear as shown in Figure 63.

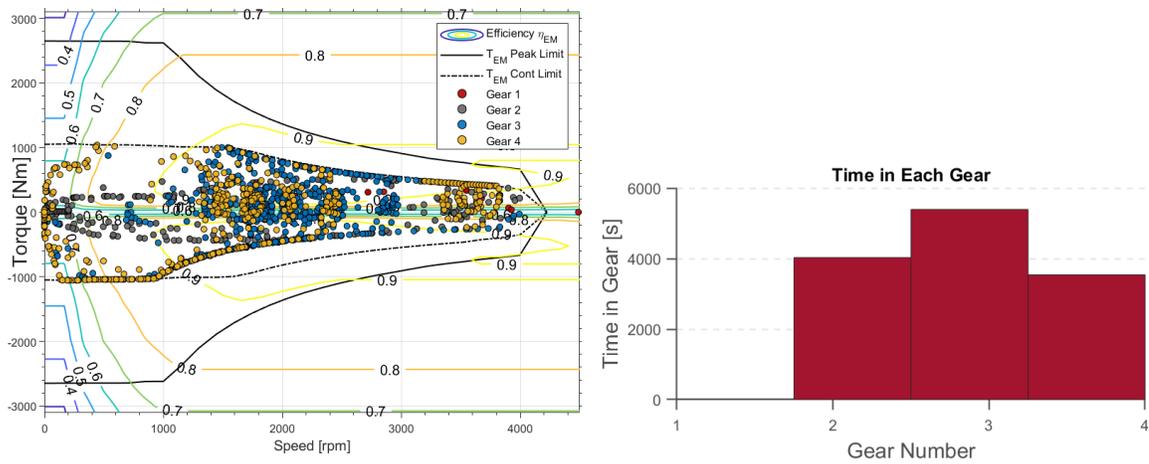


Figure 63: EM optimal operating points and gear shifts

Table 37: Electric machine operating time against efficiency

Motor efficiency (%)	Operation time (%)
95-99	0
90-95	29.74
85-90	4.49
80-85	2.46
<80	63.30



Figure 64: Genset power, engine state, battery power for the speed profile

A number of optimal powertrain operation properties are summarized below

- Discharging C-rate: 10C
- Charging C-rate: 1C
- Engine on: 61.70 min
- Battery charging: 103.48 min
- Charging from Engine: 45.18 min
- Engine turns on 21 times
- Idling events are seen during hard regen braking

#### (iv) Direct Drive

The direct drive powertrain, see Figure 65, is driven by an electric machine through a final drive ratio of 5.13, with a genset range extender. The component characteristics are listed

in Table 38. The state and control variables, and the cost function for this architecture are listed in Table 39.

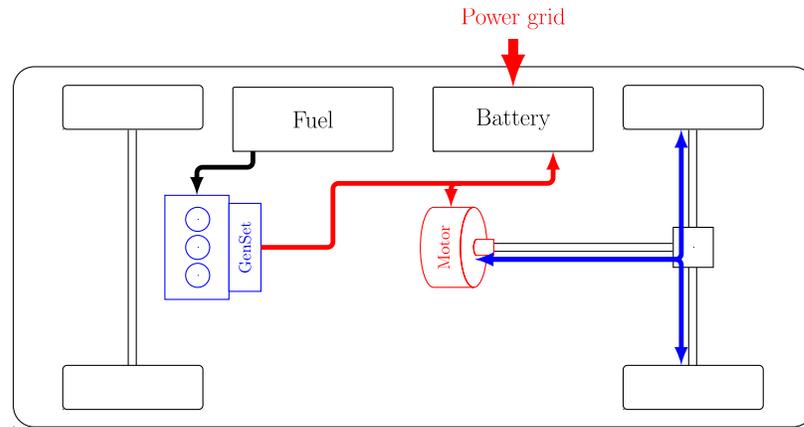


Figure 65: Direct drive powertrain architecture

Table 38: Direct drive powertrain parameters

Design Variable	Value Range
Motor Continuous Power (kW)	150-450
Final Drive Range	5.13
Engine Top Speed	4000 RPM

Table 39: Direct drive powertrain control parameters and cost function

<b>State Variables</b>	GenSet State, $s_{gen}$ , ON or OFF	State of Charge of the Battery, SOC
<b>Control Variables</b>	GenSet ON/OFF, $u_{gen}$	GenSet Power Level, $p_{gen}$
<b>Objective Function</b>	$\min_u J = \min_u \int_{T_0}^{T_f} \dot{m}_f(x, u) dt + \Phi  \Delta u_{gen} $	

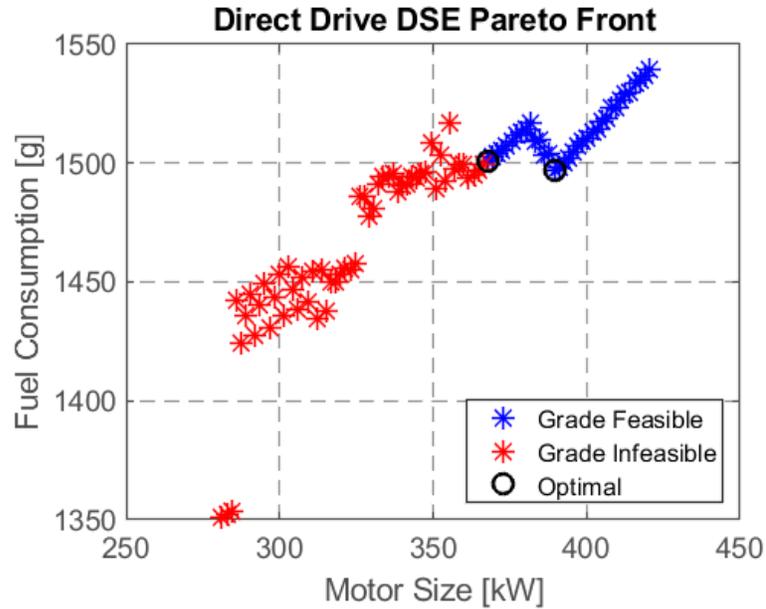


Figure 66: Direct drive pareto front

From the design space exploration for the direct drive architecture, a number of observations are made. A trade off exists between electric motor size and fuel consumption. As there's no transmission in this architecture, the motor size needs to be much larger to pass the grade performance tests. The worst candidate has 42g more fuel consumed than the best candidate. Compared to the e-axle AMT+EM architectures, the direct drive layout provides better fuel economy due to the fact that there are no transmission losses. But this saving is partially offset by the lack of a transmission to further optimize motor operation across the operating range.

The drawback of the direct drive layout is obvious: an oversized electric motor increases powertrain weight and cost. Motor efficiency falls mostly in either very high (90-95%) regions, or very low (<80%) regions. The room for low efficiency region performance improvement is limited by the lack of a gearbox for torque multiplication.

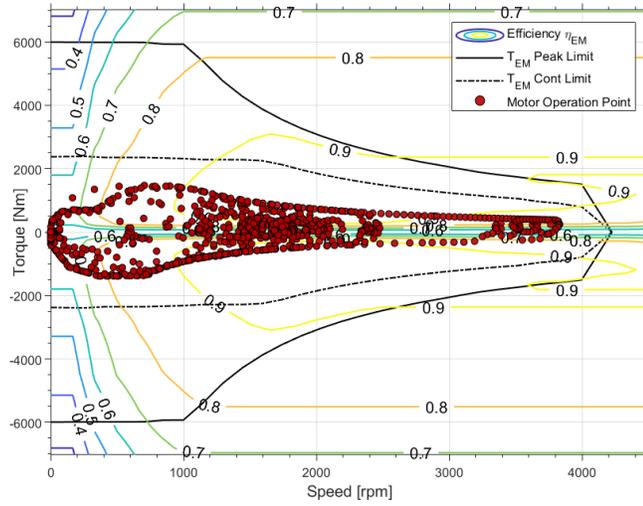


Figure 67: EM optimal operating points

Table 40: Electric machine operating time against efficiency

Motor efficiency (%)	Operation time (%)
95-99	0
90-95	15.90
85-90	7.53
80-85	3.24
<80	73.32

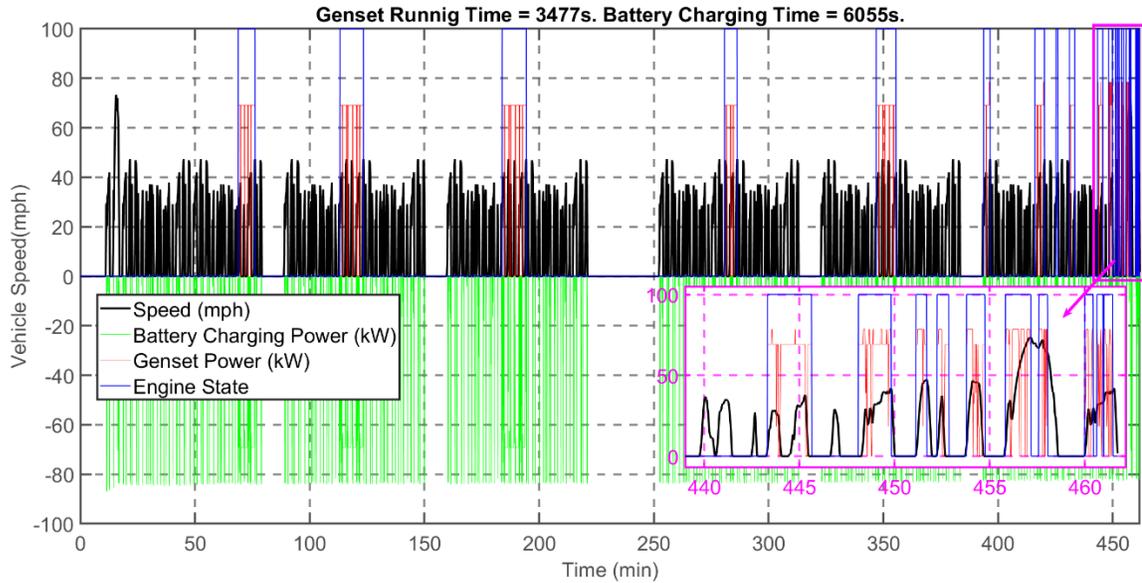


Figure 68: Genset power, engine state, battery power for the speed profile

A number of optimal powertrain operation properties are summarized below.

- Discharging C-rate: 10C
- Charging C-rate: 1C
- Engine on: 58.0 min
- Battery charging: 100.92 min
- Charging from Engine: 43.93 min
- Engine Idling: 14.07 min
- Engine turns on 19 times
- Idling events are seen during hard regen braking

**(v) Dual Motor**

The dual motor powertrain is shown in Figure 69. The component characteristics are presented in Table 41. The state and control variables, and the cost function are listed in Table 42.

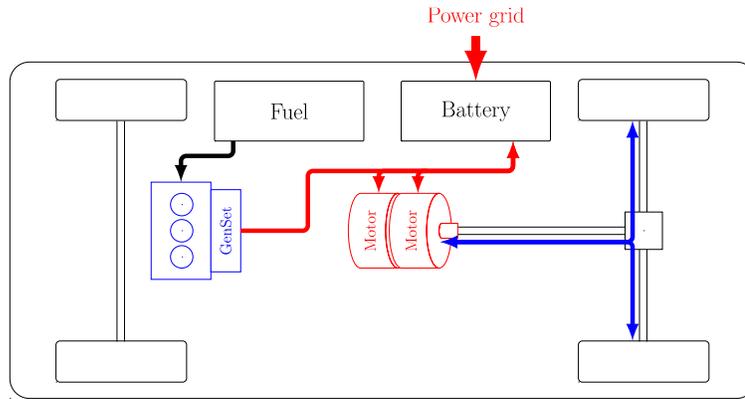


Figure 69: Dual motor powertrain architecture

Table 41: Dual motor powertrain parameters

Design Variable	Value Range
Motor 1 Continuous Power (kW)	150-300
Motor 2 Continuous Power (kW)	50-150
Final Drive Range	5.13
Engine Top Speed	4000 RPM

Table 42: Dual motor powertrain control parameters and cost function

<b>State Variables</b>	GenSet State, $s_{gen}$ , ON or OFF	Current Gear, $s_{gear}$ (4-speed+Final drive)	State of Charge of the Battery, SOC
<b>Control Variables</b>	GenSet ON/OFF, $u_{gen}$	GenSet Power Level, $p_{gen}$	Torque Split Factor, $f_{split}$
<b>Objective Function</b>	$\min_u J = \min_u \int_{T_0}^{T_f} \dot{m}_f(x, u) dt + \Phi  \Delta u_{gen} $		

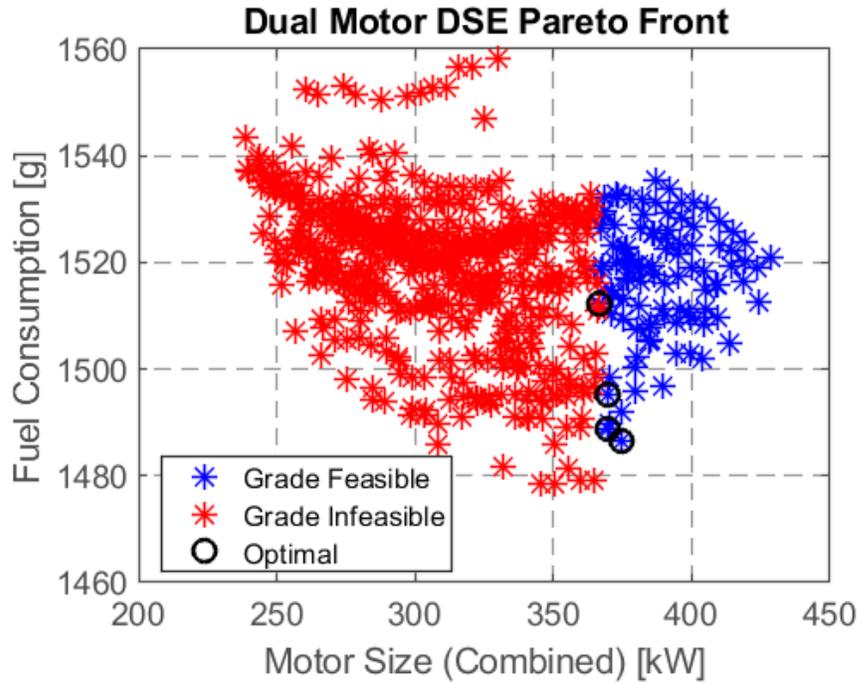


Figure 70: Dual motor pareto front

From the design space exploration for the dual motor architecture, a number of observations are made. Several optimal candidates exist, but the combined motor size is consistent. Minimum combined electric motor power stands at 370kW. Compared to the single-motor direct drive architecture, the total motor power required remains the same. But the candidate with the best fuel consumption is slightly better than that of the direct drive layout by a margin of just 1%.

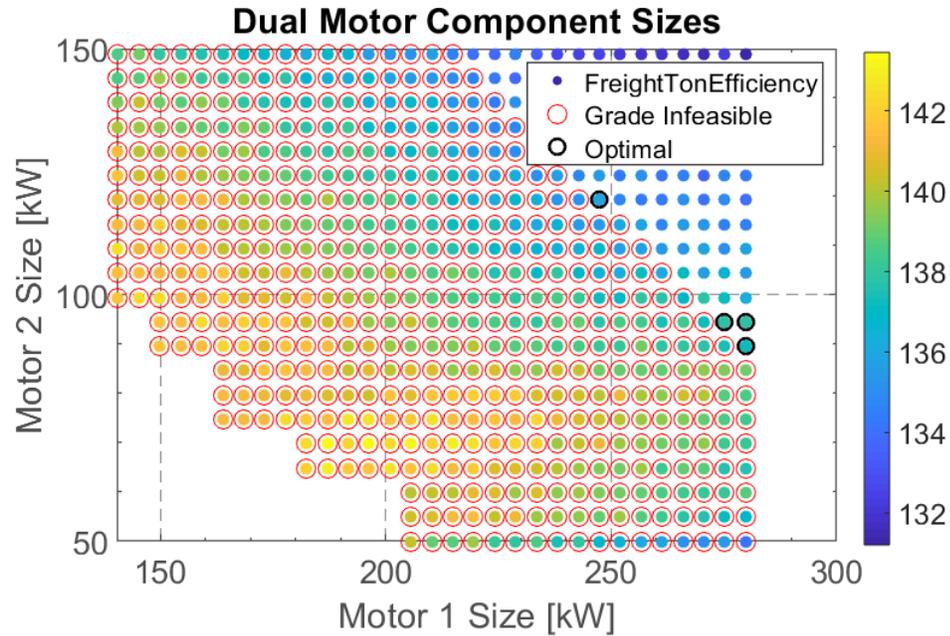


Figure 71: Dual motor architecture component sizes

A compromise between Motor 1 size and Motor 2 size exists in this architecture. In combination, they need to reach the torque required for every individual gradeability performance requirement. Compared to e-axle architectures, the dual motor layout provides minor advantage in fuel savings due to the fact that, although there is no transmission loss, the relatively low efficiency of motor operation compromises the savings on transmission losses. Motor efficiency falls mostly in either very high (90-95%) regions, or very low (<80%) regions. Motor 1, which is bigger in size, would be better off working in low power situations or be turned off, so that the combined system could minimize energy losses during certain operating points on the drive cycle.

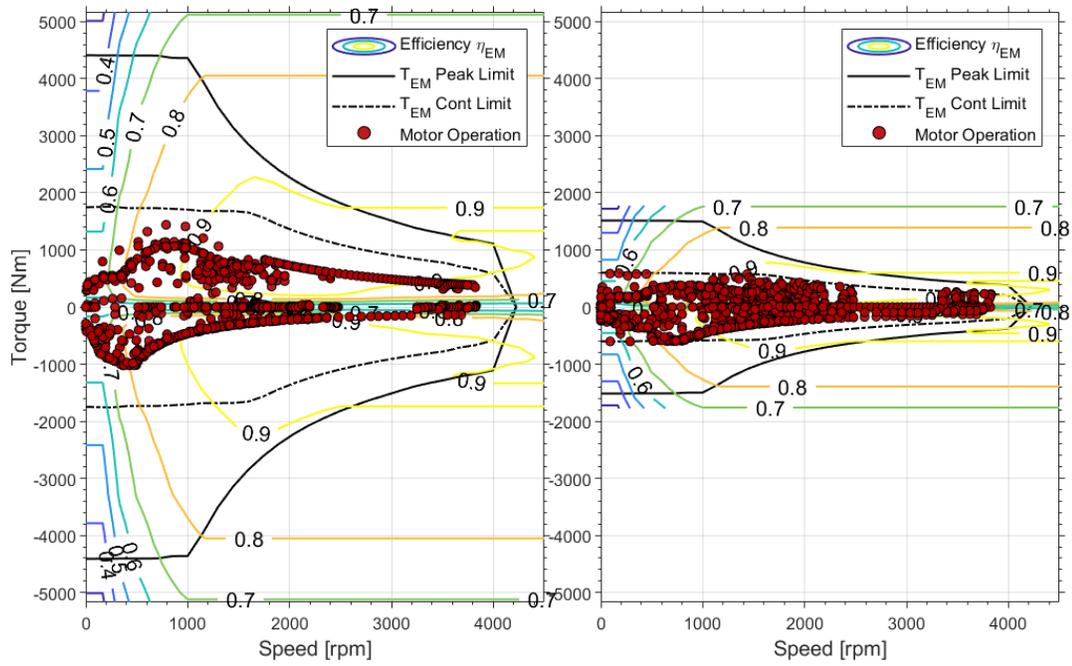


Figure 72: Dual motor operating maps

Table 43: Electric machine operating time against efficiency

Motor efficiency (%)	Motor 1 (%)	Motor 2 (%)
95-99	0	0
90-95	12.30	22.52
85-90	5.42	5.19
80-85	1.83	2.94
<80	80.44	69.34

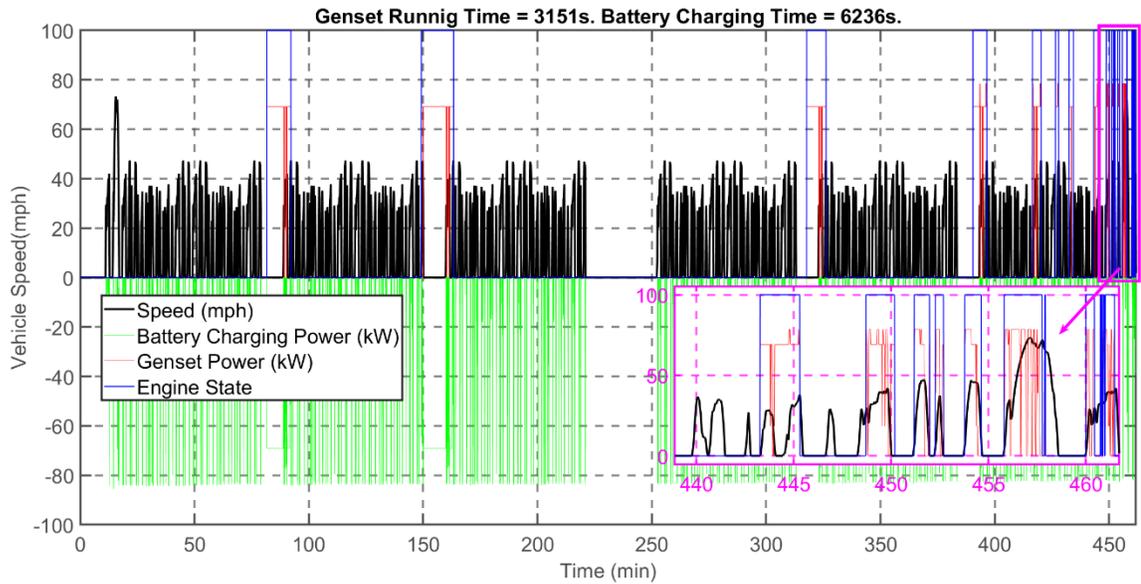


Figure 73: Genset power, engine state, battery power for the speed profile

A number of optimal powertrain operation properties are summarized below.

- Engine on: 52.5 min
- Battery charging: 103.93 min
- Charging from Engine: 44.07 min
- Engine turns on 19 times

### 2.3.5 Cross-Architecture Performance and Cost Evaluation

At this stage, multi-objective design space exploration is performed on each architecture. Using the near-optimal component size range for each REEV layout, fuel economy values are evaluated using Dynamic Programming. For delivery trucks, freight ton efficiency is defined as the number of tons of freight moved for one mile for each gallon of fuel:

$$\eta_{freight-ton} = \frac{W_{payload} \times D_{trip}}{V_{fuel}} = W_{payload} * MPG \quad (9)$$

where  $W_{payload}$  is the maximum available payload of the truck in tons,  $D_{trip}$  is the trip distance in miles and  $V_{fuel}$  is the volume of fuel used per trip in gallon.

As discussed earlier, the primary metrics for evaluating the selected architectures are powertrain cost, powertrain weight and freight ton efficiency. For all the five architectures, these three parameters are obtained and compared against each other. A number of assumptions have been made for the cost and weight evaluation of the various powertrain components, based on industry trends and available data. Calculation of the cost and weight of the REEV powertrains are performed based on the methodology summarized in Section 2.3.2.

The results of the cross-architecture performance comparison are presented in Figures 74 to 77. Figures 74 and 75 show the relation between freight ton efficiency, powertrain weight and cost for the REEV architectures, while Figure 76 incorporates the conventional powertrain into the picture. It is evident from Figure 76 that the two-speed e-axle architecture has the highest freight ton efficiency, since adding more speeds into the transmission adds to the losses in the driveline while increasing the cost of the powertrain at the same time.



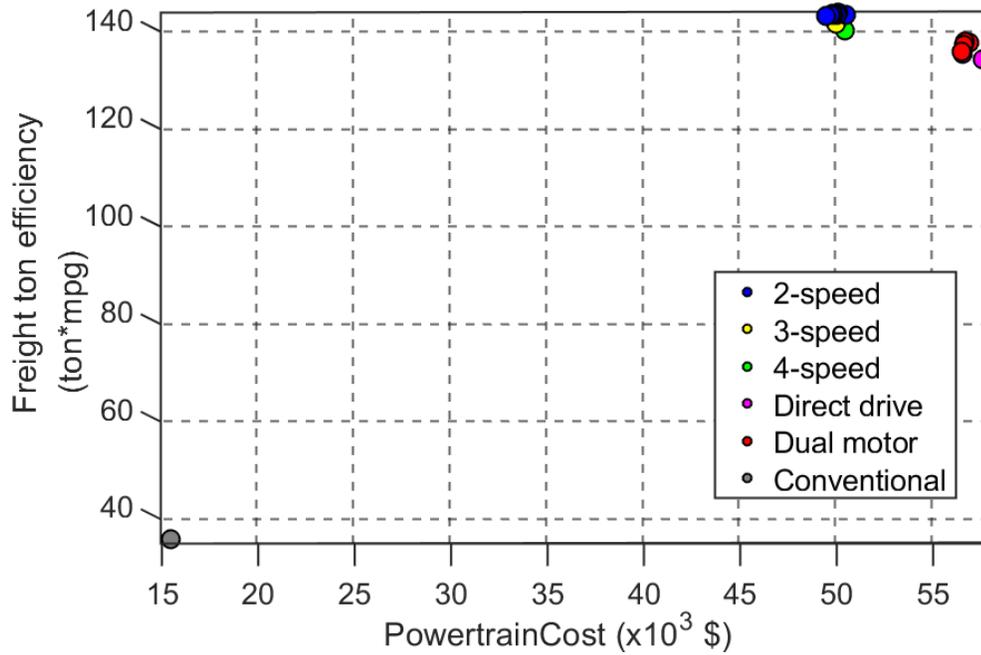


Figure 76: Comparison against the conventional powertrain

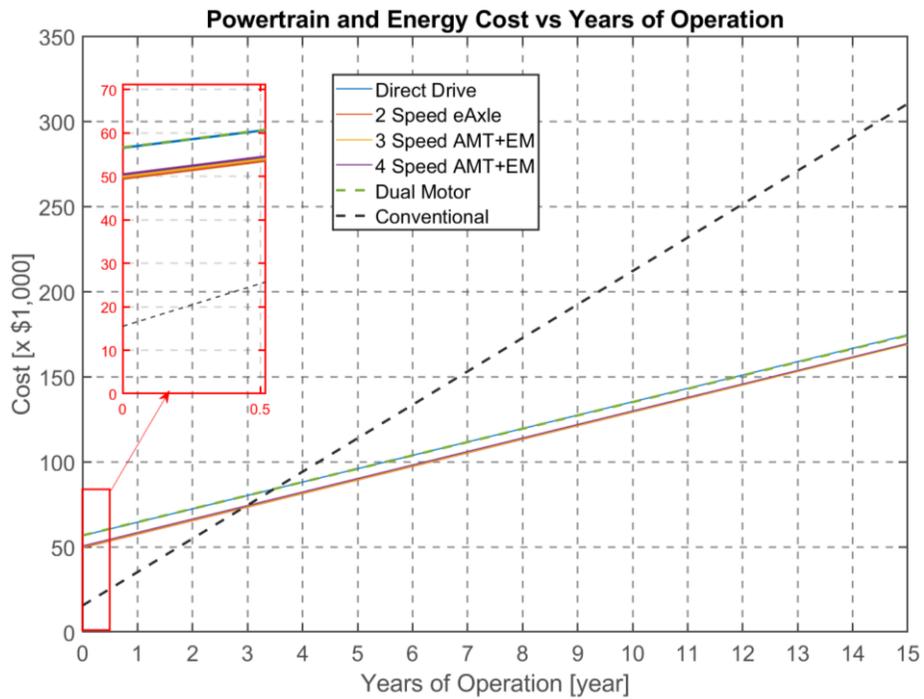


Figure 77: Powertrain cost comparison over the lifetime of operation

In comparison to the conventional powertrain, all the REEV architectures have far superior freight ton efficiency, although at a much higher powertrain cost. The hybrid powertrains have more components and hence incur a much higher initial cost. This is expected to be offset by the decreased cost in fuel consumption over time. The REEV powertrains cost in the range of 50,000 USD, against the 15,000 USD initial investment for the conventional powertrain, as shown in Figure 77. Furthermore, the direct drive and dual motor architectures demand greater initial investment over the two-speed e-Axle, three-speed and four-speed AMT+EM architectures largely due to the higher cost of the electric motor/s.

Figure 77 shows that the conventional architecture may have a lower initial investment cost, but the high fuel cost adds up and starts offsetting the initial cost advantage. Therefore, the range extender candidates start to surpass the conventional powertrain after 3 years of operation. The dual motor and direct drive candidates have a small price margin over AMT+EM solutions because of the larger motor size requirement. Note that maintenance cost for the vehicle is not included in this evaluation.

Additional comments can be made comparing the results shown in Figures 78 to 81, where the influence of each architecture on the powertrain efficiency, motor size and weight is shown. These figures also highlight the statistical spread of the effect of the component size range on these parameters using box plots. They convey the variability of the performance parameters for each architecture with change in component size.

Figures 78 and 80 show that in terms of freight ton efficiency and powertrain weight, the two-speed e-Axle stands out among the other architectures.

For the direct drive architecture, a trade-off exists between motor size and freight ton efficiency. As there is no transmission, the motor size needs to be much larger to pass the grade performance tests. Compared to AMT+EM and two-speed e-Axle architectures, the direct drive layout is expected to offer better freight ton efficiency due to the fact that there are no transmission losses. But this saving is partially offset by the lack of a transmission to further optimize motor operation. The drawback of direct drive is evident: oversized motor increases powertrain weight and cost.

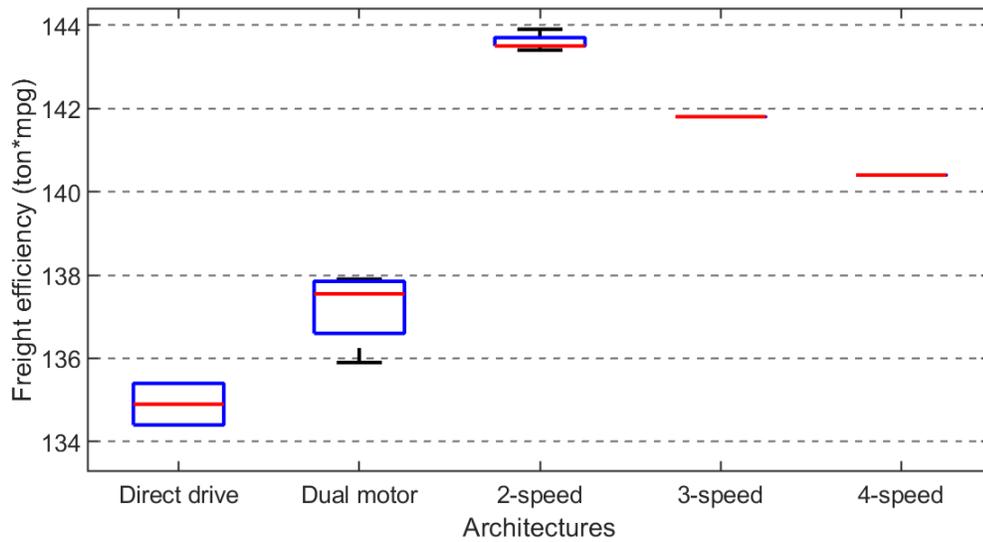


Figure 78: Freight ton efficiency comparison

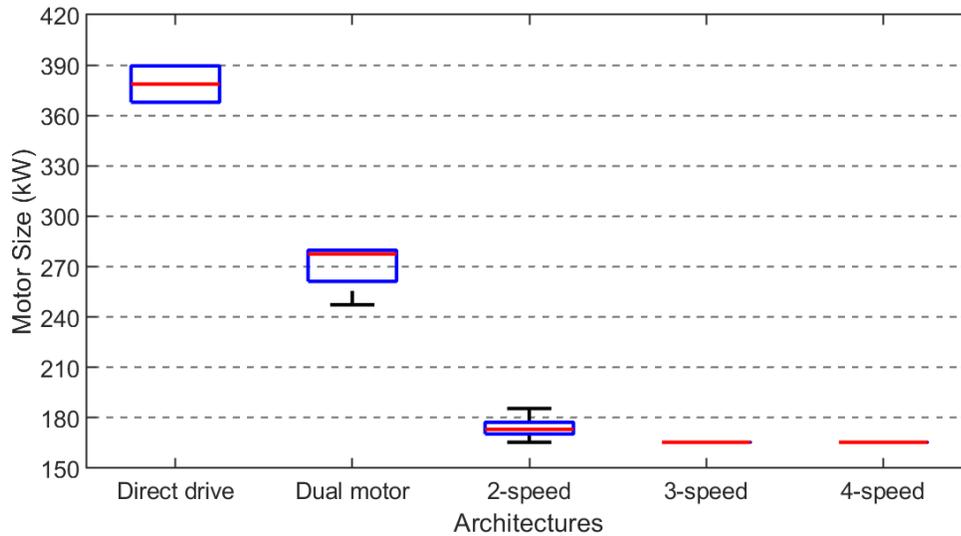


Figure 79: Motor size comparison

The dual motor architecture has a compromise between motor 1 and motor 2 size. When combined, they need to reach the torque required for each individual grade performance requirement. Compared to e-Axle architecture, dual motor provides a minor advantage in fuel savings due to the fact that there are no transmission losses.

When it comes to powertrain weight and cost, the two-speed e-axle architecture once again stands out among the rest – refer to Figures 80 and 81. The higher cost and weight of the direct drive and dual motor architectures take them out of consideration for the optimal candidate. Although the motor size is slightly better for three-speed and four-speed architectures, the two-speed architecture outshines the other two in terms of freight ton efficiency, powertrain weight and cost. Hence, the overall trend for this particular application is clearly in favor of the two-speed e-axle topology.

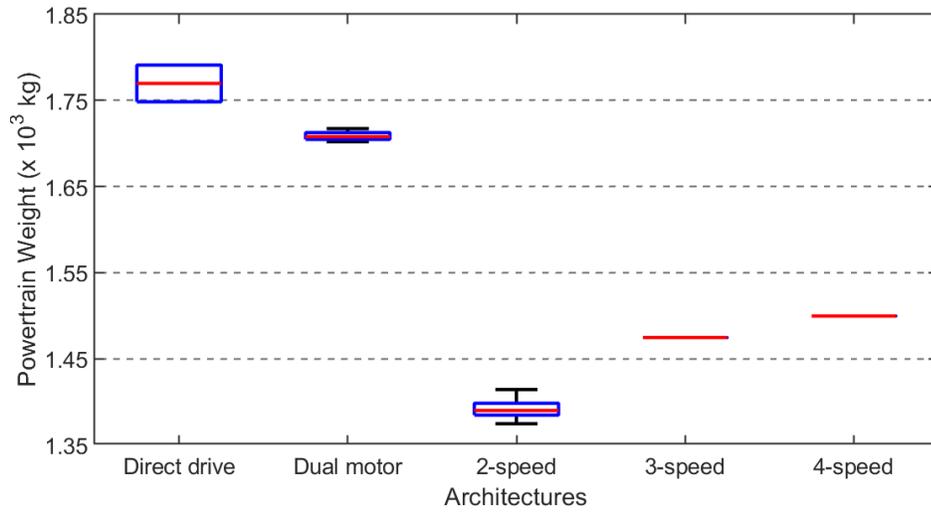


Figure 80: Powertrain weight comparison

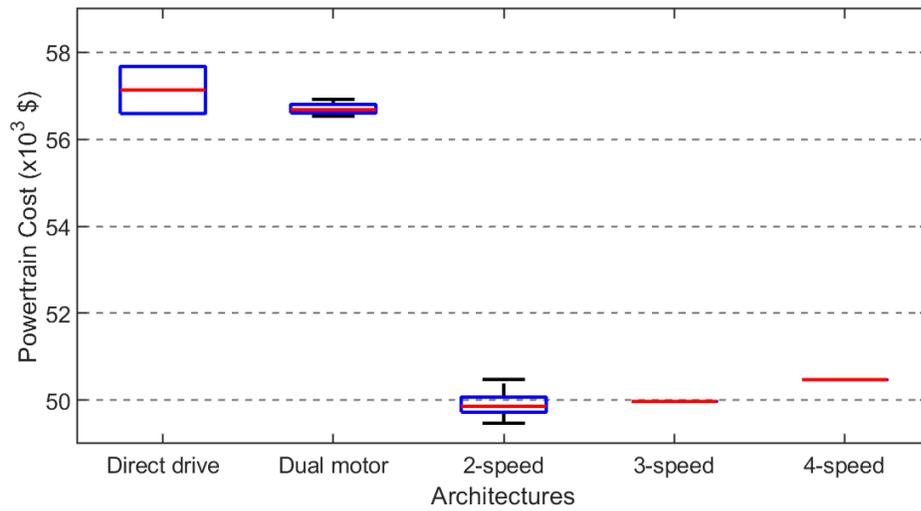


Figure 81: Powertrain cost comparison

### 2.3.6 Optimal Architecture and Conclusions

In this work, the *design for operation* notion is applied to find the optimal powertrain topology and component sizes for a specific operation scenario. In particular, the vehicle considered in this study is a Class 6 pickup and delivery truck, which is typically used in

urban driving with frequent start/stop events. In the first part of this paper, the modeling of the powertrain and vehicle components has been addressed, and an overview of the Design Space Exploration (DSE) has been presented. The problem of reducing the design space size has also been introduced. Five different range-extended PHEV powertrain architectures have been proposed for the specific operation scenario: two-speed e-axle, three-speed AMT+EM, four-speed AMT+EM, direct drive and dual motor. A multi-objective DSE, using Gaussian Process for design space reduction, has been performed. The objectives of the analysis are minimum fuel consumption and genset start/stops. A backward simulator for every powertrain architecture has been developed to determine the optimal energy management strategy using Dynamic Programming. The resulting best candidates from each REEV architectures have been compared against each other and against the conventional powertrain in terms of freight ton efficiency, cost, size and weight. Overall, the two-speed e-Axle stood out in terms of freight ton efficiency, powertrain weight and cost, while maintaining the requisite competitive performance among all architectures. A two-speed transmission is sufficient for the selected pickup and delivery duty cycle in addition to meeting the grade capability requirements.

## **2.4 Summary of Results**

Overall, the two-speed e-Axle stood out in terms of freight ton efficiency, powertrain weight and cost, while maintaining the requisite competitive performance among all architectures. A two-speed transmission is sufficient for the selected pickup and delivery duty cycle in addition to meeting the grade capability requirements. Further scope of this

study in the upcoming chapters involve designing an online optimal control strategy for the selected architecture, and evaluating the robustness of the implemented controller over realistic driving scenarios.

## **Chapter 3. Performance Metrics to Enhance Power Management in HEVs**

The thrust areas in this chapter include the choice of specific performance metrics that are crucial to the optimal operation of a Class 6 range-extended hybrid delivery truck. The study revolves around the performance metrics which are most relevant to the CERC Truck project and discusses technologies that can be leveraged to improve these metrics over a baseline IC engine truck. Although the focus is on metrics that enhance range-extended delivery trucks, the results of this chapter can be very well extended to other hybrid vehicle architectures and market segments.

After arriving at the relevant metrics, the discussion moves on to literature survey and essential quantification of the benefits of look-ahead knowledge on the performance of an REEV controller. A forward simulator with an adaptive ECMS controller is the primary tool utilized in the evaluation of the potential performance benefits.

### **3.1 Performance Metrics for Delivery Trucks**

Based on the mission, a hybrid electric vehicle may have a multitude of objectives that need to be fulfilled. A few of them pertaining to a Class 6 pickup and delivery truck are highlighted in this study. The first and foremost target for the market segment is to

minimize operational cost, which directly translates to reducing initial investment, fuel consumption and other operational costs relating to maintenance.

All commercial operations must also adhere to the prescribed emissions targets laid out by the governing authorities that they fall under. This is of special importance today with most densely populated city centers around the world clamping down on localized air pollution caused by vehicular operation by mandating different levels of zero emission operation and chalking out designated areas in the city, namely geofences, that enforce EV operation for transportation. The fuel economy and emissions from an REEV can be directly impacted by the engine start-stop strategy employed by its controller in an attempt to minimize fuel usage. Frequent start-stops can push the aftertreatment system out of its optimal operating temperature range and risk emissions control. Furthermore, this can also compromise the durability of the genset components.

In addition to the above factors governing controller objectives, the longevity of the product is of importance to make commercial sense. The electrical energy storage system (ESS) may need dedicated controller attention to ensure that it operates within safe limits, while not getting cycled at unreasonable charge/discharge rates that accelerate ageing. This may apply to other components of the electrified powertrain as well. From a driver's perspective, the powertrain must offer acceptable levels of NVH (noise, vibration, harshness) and drivability to facilitate safe and stress-free operation. What follows is a

detailed exploration of a number of these prime factors that are seminal to the optimal operation of the REEV under consideration.

### **3.1.1 Fuel Economy**

According to studies cited by Schittler et al [61], as much as about 30% of the life cycle cost of a medium duty truck comes from the cost of fuel. Since the average mileage for a US Class 6 truck is of the order of 100,000 miles per year, minimizing the fuel consumption by a few percentages will translate into significant cost and emissions reductions.

It has been estimated that that approximately one quarter of the global CO<sub>2</sub> emissions come from the transportation sector, bulk of which corresponds to roadway transportation [61]. Hybridization of medium and heavy duty (MD/HD) vehicles has a strong potential to improve real world fuel economy, decrease principal pollutants as well as greenhouse emissions. The dent that can be made by fuel saving in this market segment can be far more significant compared to the passenger car segment due to their volume of fuel consumption. However, the market penetration of hybridized products in this segment is still very miniscule.

The hybridization challenges of medium-duty trucks are not limited to the choice of the right configuration alone. The driving mission can be highly variable within the same Class 6 weight category. For instance, a delivery truck operates over a very different driving profile compared to a goods truck, even though both vehicles may have a similar GVW classification. A typical delivery truck spends its first part of the driving mission getting to

the start of its daily route and then switches to a stop-and-go route with peak speeds between 20 mph and 30 mph limited to an urban or housing neighborhood scenario. It can also be noted that there are significant idle events during the drive cycle observed with package drop-off. These findings are supported by research which found that both application and driver behavior have significant effects on the fuel economy of a particular vehicle application [2], [6] and [7].

The general consensus is that medium-duty vehicles offer greater flexibility and opportunity for various regeneration strategies, as well as the option of having one axle being driven by one source, say, a conventional powertrain, and the other by a purely electric powertrain.

The concept of pairing of hybridization with the higher efficiency and lower CO<sub>2</sub> output of a diesel combustion engine (as opposed to a gasoline engine) is an attractive prospect. However, the main deterrent is the incremental hybrid system cost over the baseline equivalent, which drives up initial investment. Another point of note is that modern-day diesel engine, even with their high thermal efficiency, demand complicated and expensive aftertreatment systems to comply with emission certification. As a result, depending on the market nature and investment constraints, a tradeoff needs to be made between increasing fuel economy and reducing emissions by tailoring a suitable hybrid system and strategy.

The final product configuration will vary based on the specific vehicle application and market.

In conclusion, the driving force behind hybridization is the combination of fuel and running costs. As we cross peak oil production and the cost of fuel increases, customers may opt for medium-duty hybrids to reduce their operating costs over the lifetime of the vehicle. Some mechanisms to accelerate deployment include enforced government regulations as well as the reducing costs of batteries and electrical component development and production. At the end of the day, the balance of these competing factors will ultimately dictate the deployment and adoption of MD HEVs. [9]

### **3.1.2 Engine Start/Stop Events**

A hybrid electric vehicle may have, on several occasions, conflicting ideas within its list of objectives. A prime example is that of the trade-off between fuel economy and engine start-stop events. While turning off the genset when the load demand from it equates to zero is an action that directly slashes idling fuel consumption, the emissions and powertrain durability penalties that come with it are conflicting with the initial objective in the first place. The energetic costs of IC engine start-stop events in plug-in hybrid vehicles have been exceedingly well-documented by Engbroks et al. [62] with experimental and simulation methods. Following is a brief summary of the major conclusions arrived at by

this study focusing on experimental evaluation of the energetic expense for engine starts in PHEVs under various conditions.

The study distributes over 80,000 engine starts in a combined IC engine/electric machine characteristic map to quantify the losses involved in these events. The energetic expenses are broken down to the individual powertrain components in order to quantify the principal causes. The influence of cold engine oil temperatures is also investigated in this context. With a share of about 80%, the internal combustion engine has the greatest influence on the energetic costs for the IC engine start in every single operating point. There are numerous reasons for this: Even in the stationary best efficiency point, the gasoline engine operates with efficiencies not better than 35%.

It was concluded that the influence of this effect is largely dependent on the engine-off time as well as on the exhaust gas temperature dynamics. The impact of the electrical losses in this event was observed to be relatively marginal. Energy losses in the electric machine(s), battery pack and power electronics are primarily related to the power demand by the controller which spiked during start-stops events.

At 0°C, the effort for starting the IC engine is tripled at low loads. This was justified by the increasing viscosity of the engine oil with cold temperatures. In further research, it was shown that the effort for engine start was increased by a factor of 1.5 to 3 in terms of increased fuel consumption for a temperature of 0°C, largely attributed to low oil temperatures. This can be reasoned by an increasing viscosity of the engine oil with cold temperatures. The impact of the resulting friction torque was much larger in the part-load

conditions than towards higher engine loads. The experiment utilized a V6 gasoline engine which had electric costs varying between 0.5 Wh and 4.0 Wh in the range of 1000-2000 RPM. A critical observation was that the added fuel cost per engine start lied between 0.4 g and 2.3 g. The results also showed a strong dependence on the IC engine operating point. While the required electric energy largely varied with torque demand, the fuel usage showed a strong dependence on power demand.

The existing mix of medium-duty Class 6 trucks have turbocharged diesel powered powertrains equipped with advanced aftertreatment systems that curtail emissions. These diesel engines are inherently bulky with very heavy rotating masses and associated auxiliary loads, which are highly cumbersome in the event of frequent start-stop events. To add to that, they have turbocharger systems which demand a continuous feed of lubricating oil powered by the engine oil pump to prevent coking and bearing deterioration as the high-speed assemblies “coast down” without pressure-fed oiling during short stop and go events. The aforementioned advanced after treatment equipment will also be affected if hybrid power-split operations are not coordinated with diesel particulate filter (DPF) regeneration and catalyst temperature dynamics. If not carefully managed, the additive effects of

frequent engine start-stop events will adversely impact the durability of engine and emission systems.

The influence of engine Stop/Start systems on the NVH quality of vehicles and the potential countermeasures for improving Stop/Start NVH for vehicles equipped with automatic engine start-stop systems was studied in detail by Wellmann et al in [63].

In lighter weight classes, it has been shown that adopting “blended” hybrid mode offers an alternative to an engine stop start strategy, which provides the opportunity to have both the drivability of a conventional vehicle without the adverse durability impacts of a start-stop engine strategy, while achieving increased fuel economy and reduced emissions. It is to be mentioned that such an architecture will enforce a smaller hybrid powertrain with little to no pure electric range although its offers cost and performance advantages on a per kW basis.

### **3.1.3 Emissions**

There is a growing awareness of the effect of green-house gases such as CO<sub>2</sub> on global warming, and realization that reducing CO<sub>2</sub> emissions needs to be addressed on either voluntary or mandatory basis. In the case of vehicles running on fossil fuels, reduction of CO<sub>2</sub> emission is linked directly to fuel consumption, and thus the additional stimulus for a strong research focus on energy efficiency of vehicle propulsion.

The overall vehicle fuel economy fundamentally depends on the efficiency of the fuel converter (e.g. IC engine), vehicle mass and losses, e.g. air drag, rolling resistance,

driveline losses etc. While in the case of passenger cars there are obvious opportunities for improvements in all of aforementioned categories, in the case of trucks the choices become severely limited. Most of the heavier trucks already use highly efficient diesel engines and fuel conversion efficiency can be improved only incrementally. Truck weight is dictated by the payload: development of light weight structures is likely to increase the payload carrying capacity, rather than reduce the gross vehicle weight. Dimensions of the payload carrying section limit the chances for reducing the aerodynamic drag. Consequently, hybridization of the propulsion system offers the best potential for significant reduction of truck fuel consumption. Hybridization provides the flexibility in controlling engine operation, regeneration of braking energy, possible engine downsizing and prolonged engine shut-down intervals [64].

However, per the extensive research by Minarcin et al [9], the emission warranty obligation engine manufacturers face has been an integral part of deployment. Engine manufacturers have extensive experience with the engine-based components that affect emission compliance. It is to be noted that hybridization is a new emission control that many engine manufacturers have little or no experience with or control over as it relates to component and by extension emission warranty. In the volumes currently deployed, the risk is minimized but the durability limits cannot be adequately characterized absent significant increases in volume. Minarcin et al [9] also point out that this places the full hybrid compliance obligation on the engine manufacturer as the certificate holder. Engine manufacturers must then weigh the benefits of emission warranty obligation against the

benefits of a hybrid system. This has often prevented the certification of medium-duty HEVs in California and will continue to have major cost implications [9].

Even though parallel hybrids are easier to implement with greater level of expertise in the industry, they do not offer the flexibility of operation needed in the medium-duty delivery vehicle segment. Parallel hybrids are effective in regeneration as shown by Buchvald et al. [65], Wu et al. [66] and Filipi et al. [67], but the emissions advantages typically diminish for duty cycles including predominantly start-stop driving. In contrast, a series system provides full flexibility in controlling engine operation under any conditions [64], and the goal of this study is to explore the opportunities for maximizing fuel economy with the series hybrid concept.

The subject of emissions in series HEV applications is discussed in more detail in Sec 4.5.

### **3.1.4 Introduction to the Forward Simulator**

The selected medium to evaluate the potential for improvement of the fuel economy, engine start-stop and emissions performance of the truck is a forward simulator. The Forward Simulator is a Simulink-based simulator that incorporates the selected optimal range extender HEV architecture arrived at in Chapter 2 with the Design Space Exploration exercise. The data for the powertrain components are sourced from industry partners as well as utilizing market research. The full disclosure on the component modeling in the forward simulator is discussed here.

## **Vehicle Road Load Model**

The vehicle road load is a result of Newton's Second Law of Motion, and is used to calculate the longitudinal vehicle speed, in the exact same procedure presented in Sec 2.2.1.

## **Powertrain Dynamics Model**

All powertrain models are quasi-static energy conversion efficiency models, except the battery model, which contains an integral equation (state of charge). Thus, the combined vehicle and powertrain model has two dynamic states: vehicle velocity and battery state of charge.

### **(i) Genset Model**

The engine model is created from measured data of a mass-market 4.5L internal combustion engine, which is scaled up with the Willans Line model to match a 150 kW of maximum electrical power generation, in the exact same procedure presented in Sec 2.2.1. The genset is modelled as a MATLAB class object with the same properties as discussed in Chapter 2.

### **(ii) Electric Motor Model**

The electric motor used in this study is a scaled up permanent magnet synchronous machine with a peak power output of 245 kW and a continuous power output of 165 kW. The electric motor map is generated using a linear scaling process described in Chapter 2.

### **(iii) Battery Model**

The battery model is a zeroth order equivalent circuit model, which assumes parallel and series connections of identical cells, as described in Chapter 2.

### **(iv) Transmission Model**

The transmission is simply modelled as a set of gear ratios with fixed efficiencies. The optimized gear ratios are those corresponding to the optimal 2-speed architecture arrived at in the Design Space Exploration process discussed in Chapter 2.

As a Simulink model, this simulator provides unique benefits in that it has visually intuitive interface, has a good library of modules for controller prototyping, and easy to be shared with other team members as the Simulink platform is prevalent in engineering education.

### **3.1.5 Model Setup with MATLAB**

The overall model structure is a driver-powertrain-vehicle dynamics interaction. The driver model is currently implemented as a simple PID controller that tracks the reference speed. The current driver settings are:  $K_P = 5$ ;  $K_I = 10^{-5}$ ;  $K_D = 0$ . The hybrid powertrain structure is illustrated in Figure. The powertrain takes in commands from the powertrain controller, and provides tractive force to the rear axle. Currently, a front axle model has not been built, and all traction and braking torque are modelled on the rear axle. The Simulink schematics of the forward simulator are depicted in Figures 82 and 83.

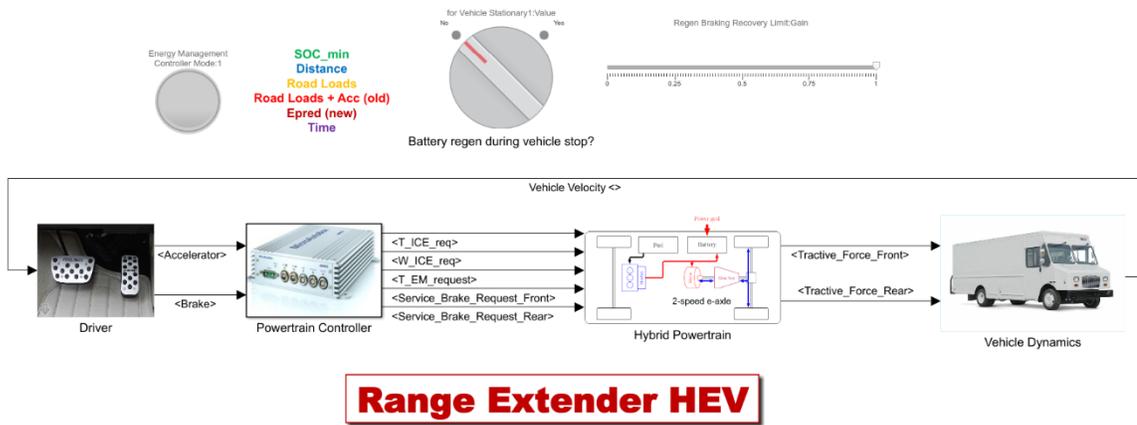


Figure 82: Forward simulator for the selected 2-speed e-axis architecture

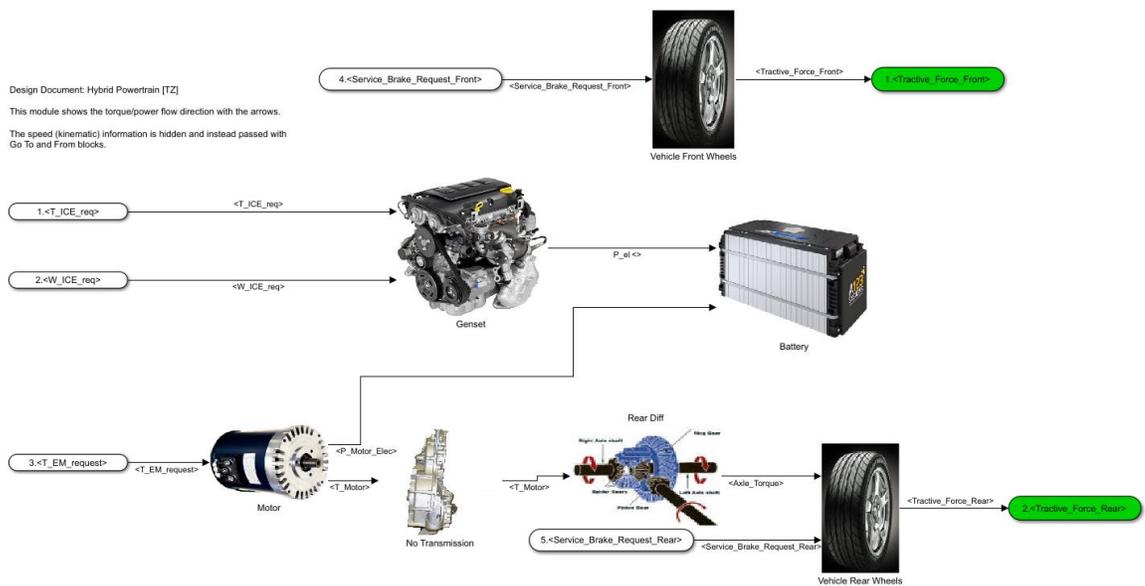


Figure 83: Powertrain module inside the forward simulator

All sub-modules of the powertrain (genset, battery, electric motor etc.) inherit the class definitions mentioned in section Chapter 2. Prior to the Object-Oriented Design refactoring, these sub-modules were manually modified to match up with the models in the backward DP simulator. The work is complete with the ultimate and long-term reliable approach to have an Object-Oriented Design pipeline throughout the different simulators.



Figure 84: Object oriented design of the simulator environment

### 3.1.6 Powertrain Controller: A-ECMS

The powertrain controller accepts the inputs from the driver model, namely the accelerator pedal position (APP) and brake pedal position (BPP) commands, and sends forward the following requests to the powertrain: Torque from IC Engine, IC engine speed, Motor Torque request and hydraulic brake request. The controller is implemented in three main blocks which will be discussed in the following subsections.

**Gear Shifting Strategy:** The gear shifting policy is simplistic as it is done only on the basis of the engine speed. Even though the model has the capability of accepting APP position as a parameter, in the current implementation it is invariant with APP request.

$$\omega_{upshift} = f(gear, APP) = 1047.09 \text{ (constant)}$$

$$\omega_{downshift} = f(gear, APP) = 26.49 \text{ (constant)}$$

Thus, the implementation is straightforward: the system decides upshifts and downshifts using a *Stateflow*® block that takes into account  $\omega_{upshift}$  and  $\omega_{downshift}$ . In order to avoid rapid shifting, the state flow takes into account a threshold time for which a particular gear should hold irrespective of the engine speed crossing the speed thresholds.

**Motor Regeneration and Service Brake requests:** This part of the controlled can be divided into several individual tasks:

- Evaluating the maximum traction braking that can be applied.

- Comparing the maximum traction possible at the wheel and converting it into a motor torque value. This is then compared with the maximum possible motor torque at that particular speed (a dynamic quantity).
- Deciding between APP and BPP requirements in case both are transmitted to the controller.
- Deciding the power discharge from the battery based on motor power demand. The final motor torque demand is thus limited by the battery power *Stateflow*®.

The calculation of maximum traction torques and comparison is shown below:

$$T_{brake,max} = \frac{0.7mg}{r_{gear} * r_{diff} * r_{wheel}}$$

$$T_{brake} = \min(T_{brake,max}, T_{max,negative,motor}) * BPP$$

$$where, T_{max,negative,motor} = f(\omega_{motor})$$

$$Saturation\ block\ to\ ensure\ 0 \leq T_{brake} \leq T_{brake,max}$$

The resolution between BPP and APP commands is shown below:

$$If\ APP > 0.01$$

$$T_{EM,req} = APP * T_{EM,max,positive}$$

$$where, T_{EM,max,positive} = f(\omega_{motor})$$

$$else\ if\ regeneration = 1$$

$$T_{EM,req} = T_{brake}$$

**Power-Split Strategy:** The power split strategy depends on several subdivisions which are summarized in the below subsections.

**Accessory Load Definition:** The simulator assumes a constant accessory load of 4000W on the Battery when the vehicle is in motion. In case the vehicle is stationary, there is no accessory load on any powertrain component.

**SOC Deviation Calculation:** The deviation of SOC is based on 2 quantities: a desired SOC value and the actual SOC value from the battery. In order to calculate the desired SOC, the current controller setting relies on the concept of work done by the vehicle. The controller has a 1D lookup table that stores the remaining work demand as a function of vehicle distance travelled. The desired SOC is then calculated using the following methodology.

$$W_{dmd,r} = \textit{Work demand remaining}$$

$$W_{dmd,r} = f(V_{veh}): \textit{Implemented as a lookup table}$$

$$\Delta SOC = \textit{Deviation from the desired SOC}$$

$$\Delta SOC = SOC_{Batt} - \max \left( SOC_{final}, SOC_{final} + \frac{SOC_{initial} - SOC_{final}}{W_{dmd,total}} W_{dmd,remaining} \right)$$

The ECMS section of the controller decides the appropriate power for the Genset and the Battery. In order to do this, first the power requirement, now referred to as  $P_{consumption}$  is evaluated as follows:

$$P_{consumption} = P_{motor} + P_{accessories}$$

$$P = \min(\max(P_{Genset,array}), \max(0, P_{consumption}))$$

The power P is now discretized in steps of 1 percent such that:  $P_i = 0.01 * P * i$

The fuel consumption of the genset is calculated for all discretized values through a lookup table and the battery power request is calculated for the remaining part of the power consumption as summarized below:

$$m_{f,i,ICE} = f(P_i)$$

$$P_{batt,request,i} = P_{consumption} - P_i$$

$$m_{f,total,i} = m_{f,i,ICE} + \frac{P_{batt,i} * \gamma}{Q_{LHV}}$$

The calculation of  $P_{batt,i}$  is done using a separate battery model and is described in subsequent subsections.

From the total array of total fuel consumption, the lowest value of the consumption is selected and the same indexed values from genset and battery power vectors are selected as final requests.

$$P_{gen,req,ECMS} = P_{i*}$$

$$P_{batt,req,ECMS} = P_{batt,request,i*}$$

It must be noted that the value of  $\gamma$  is not modified throughout the cycle. The SOC control is based on enforcement of lower and upper SOC limits.

**Battery model and calculations:** The battery model in the controller calculates the current and voltage based on the 0th order model. The calculation of open circuit voltage and resistance are implemented as lookup tables.

$$P_{batt} = (V_{OC} - IR_0)I$$

$$I = \frac{V}{2R_0} - \sqrt{\left(\frac{V}{2R_0}\right)^2 - \frac{P_{batt,requested}}{R_0}}$$

$$V = \frac{P_{batt,requested}}{I}$$

$$0 \leq V \leq V_0$$

$$V_0 = f(SOC)$$

$$R_0 = f(SOC, T, \text{sgn}(I))$$

$$P_{batt} = VI + Q$$

$$= I^2 R_0 N_{series} N_{parallel}$$

It can be seen from the equations that the net power from the battery includes the requested power as well as the losses from heating. Because of the saturation on the voltage, the controller ensures that the power requests to the battery are reasonable and in no violation of battery limits. In the current simulation, the temperature is assumed to be constant

throughout and is equal to 25 C. The genset and battery power requests as a result of the ECMS block are coupled in a bus and will now be collectively referred to as  $P_{req,ECMS}$ .

**Charging Mode:** A similar set of genset and battery power requests is defined in the charging mode:

$$P_{gen,req,charging} = P_{genset,elec,maxEta}$$

$P_{genset,elec,maxEta}$  is the power at maximum generator set efficiency

$$P_{batt,req,charging} = P_{consumption} - P_{gen,req,charging}$$

Thus, in charging mode,

$$P_{gen,req,charging} = P_{genset,elec,maxEta}$$

$$P_{batt,req,charging} = P_{consumption} - P_{gen,req,charging}$$

**Request Routing:** It is clear that we have 2 sets of requests, in ECMS mode and charging mode and the controller needs to resolve which set of requests is to be routed further. The decision between these requests is resolved through two limiting values of SOC, which are dynamic in nature.

The two values,  $dSOC_{lower}$  and  $dSOC_{upper}$  are defined as:

$$dSOC_{lower} = -0.5 * \frac{t_{cyc,total} - t}{t_{cyc,total}}$$

$$dSOC_{upper} = 0.5 * \frac{t_{cyc,total} - t}{t_{cyc,total}}$$

The resolution on the routed requests is based on the following logic:

*If  $dSOC < dSOC_{lower}$*

$$P_{req,routed,genset} = P_{gen,req,charging}$$

$$P_{req,routed,battery} = P_{batt,req,charging}$$

*If  $dSOC > dSOC_{upper}$*

$$P_{req,routed,genset} = P_{gen,req,ECMS}$$

$$P_{req,routed,battery} = P_{batt,req,ECMS}$$

The routed power requests are then fed to an "Engine restart minimizer" block which eliminates continuous ON OFF requests to the engine. The end result of the block is an engine speed and torque request. From the "Motor Regeneration and Service brake requests" block, we get a set of torque requests for the motor and hydraulic brake request.

A result of simulating the CERC P&D cycle with initial battery SOC of 99% is depicted in Figure 85 with the outcome of the SOC profile and the fuel consumption superimposed.

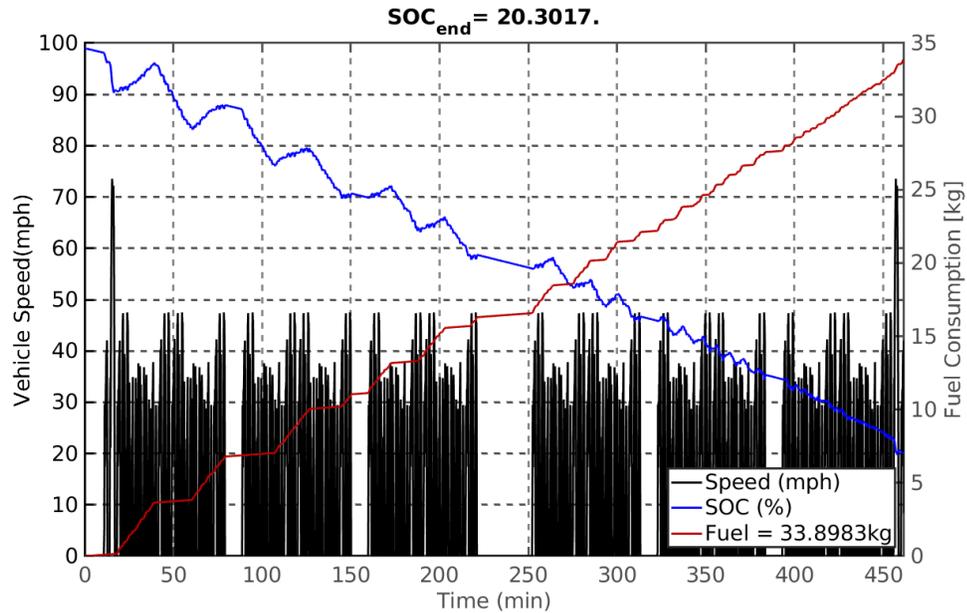


Figure 85: Sample results from the forward simulator with an ECMS controller

### 3.1.7 Cross-validation between Forward and Backward Models

Dynamic programming conclusively offers the best global solution to the optimal control problem discussed in this thesis. Before delving into the improvement of the online implementable controller in the selected candidate, it is important to analyze the potential for improvement in fuel economy on top the baseline controller. This section is a brief attempt to quantify the exact potential for improvement.

The backward DP simulator and Simulink forward simulator were developed in different time frames. Thus, it is necessary to have them cross-validated. Since model differences exist between backward and forward simulation, the comparison requires bridging the differences first. Several efforts have been applied towards achieving this goal.

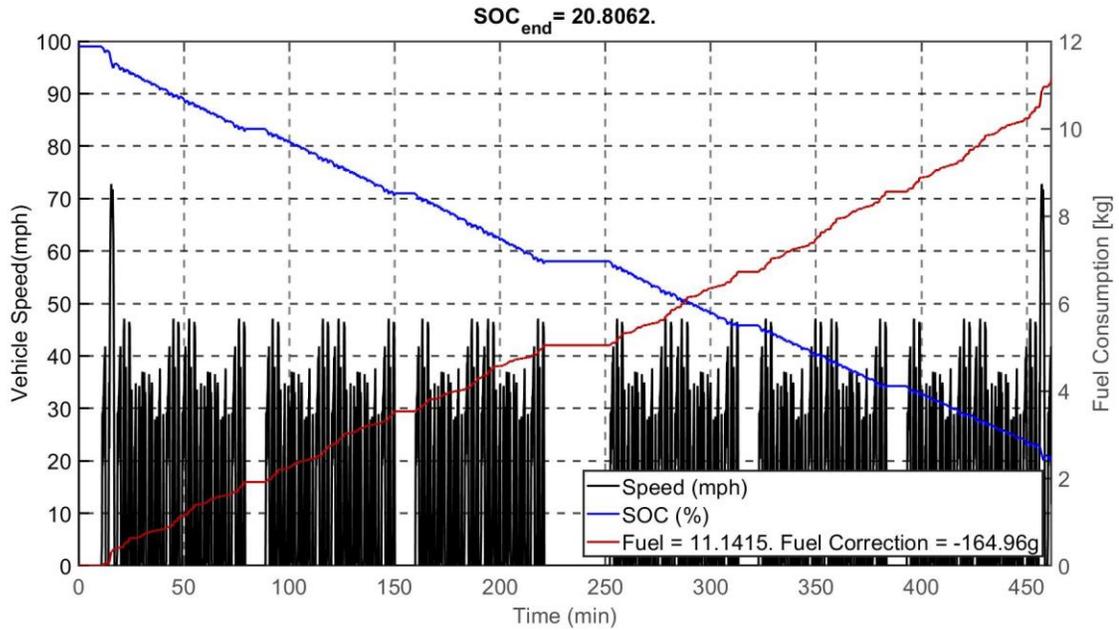


Figure 86: Sample results from the backward simulator using Dynamic Programming

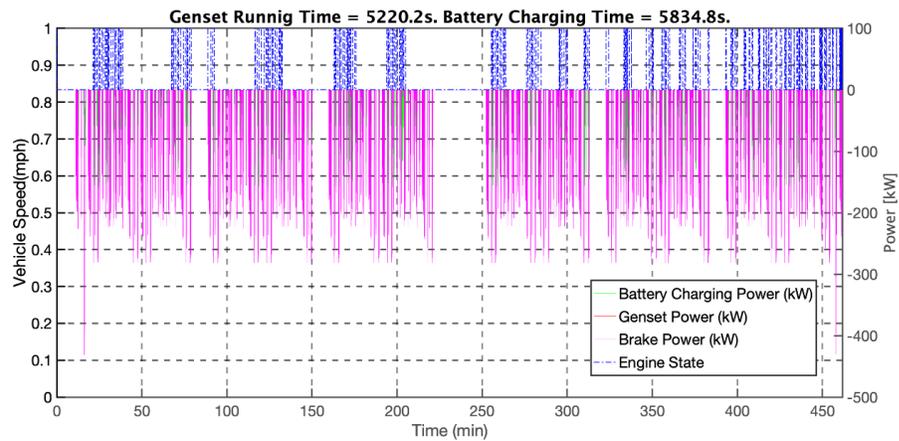


Figure 87: Powertrain states observed during the P&D drive cycle

In order to perform cross-validation between the backward and forward simulation, the benchmark fuel economy for the selected architecture is evaluated using Dynamic Programming and the results are summarized.

1. The results of the backward simulator following CERC Pickup and Delivery cycle are represented in Figures 86 and 87.
2. First, the velocity profile produced by the forward simulator driver is exported to be running on the backward simulator, the resulting energy numbers are represented in the following figure.
3. Second, the forward simulator's mechanical brake decisions are exported to be running on the backward simulator, the resulting energy numbers are represented in Figure.
4. The forward simulator with a driver following CERC Pickup and Delivery cycle is represented in Figure 85.
5. Fuel economy results are compared between DP and ECMS in Table 44, with percentage differences for three different levels of velocity profile matching that highlight the potential for fuel efficiency improvement in the forward simulator. This exercise is also aided by the Sankey diagram in Figure 88.

Table 44: Comparison of fuel economy results between DP and ECMS

Gross Wt.	DP	DP (fwd. cycle)	DP (fwd. + brake)	A-ECMS	% Delta
19501 lbs.	15.2 MPG	14.2 MPG	11.8 MPG	11.2 MPG	35%; 26%; 5%

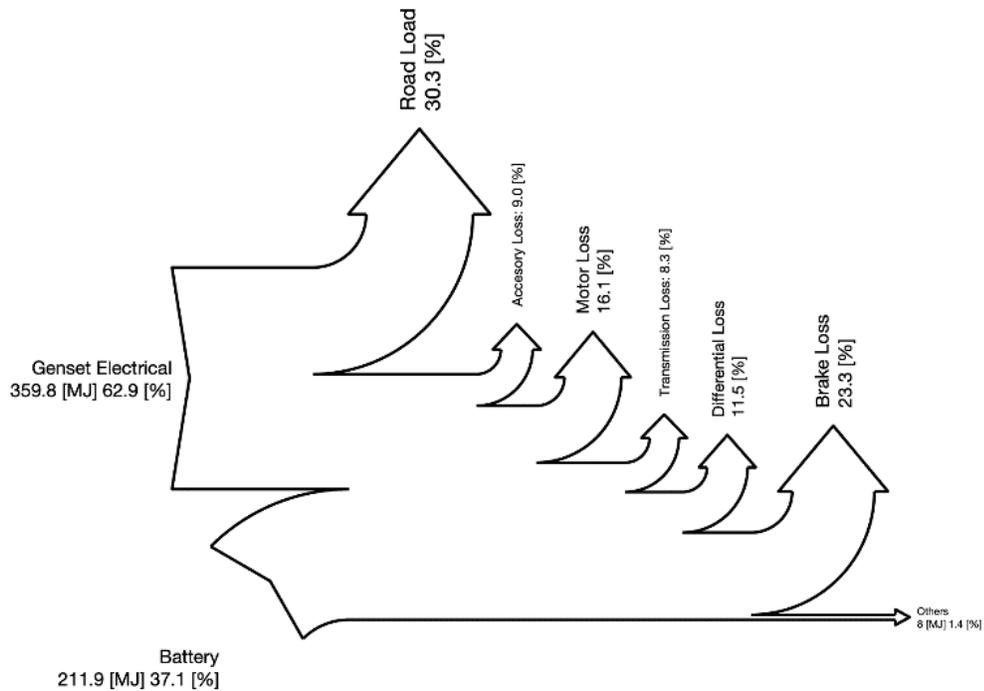


Figure 88: Sankey diagram showing the energy flow in the forward simulator

### Energy Comparison Between Backward and Forward Simulators

Figure 89 depicts the component-wise energy split starting from the two sources of energy – generator set and the battery – all the way down to the wheels. The Sankey diagram in Figure 90 offers further insights into the areas of the forward simulator that may be deficient in modeling accuracy and points towards avenues of potential fuel economy improvement going forward.

- 1 Backward DP with original CERC\_PnD cycle
- 2 Backward DP with driver-produced cycle
- 3 Backward DP with driver-produced cycle and brake-actuation
- 4 Forward simulation

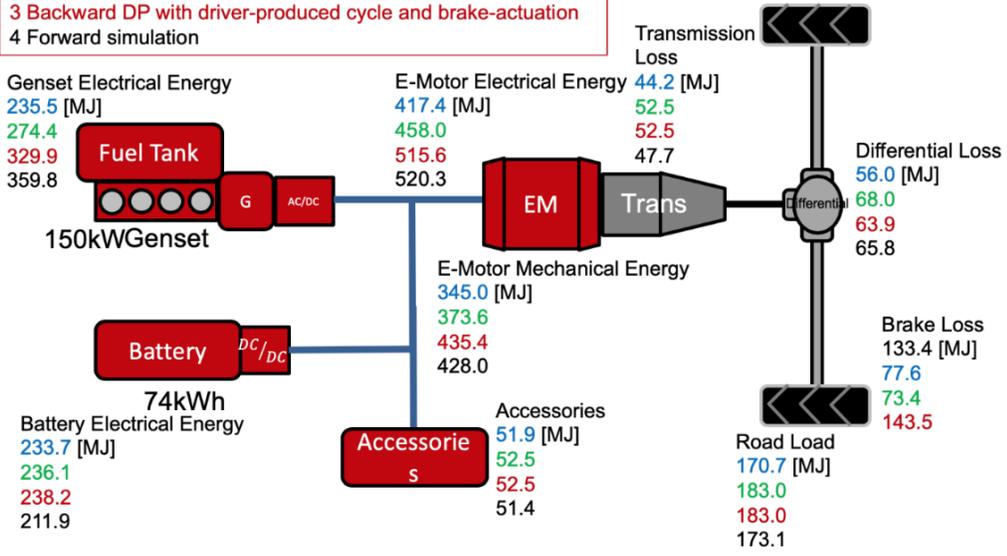


Figure 89: Energy split at component level for forward and DP simulations – the comparison is performed at four levels of driver behavior matching

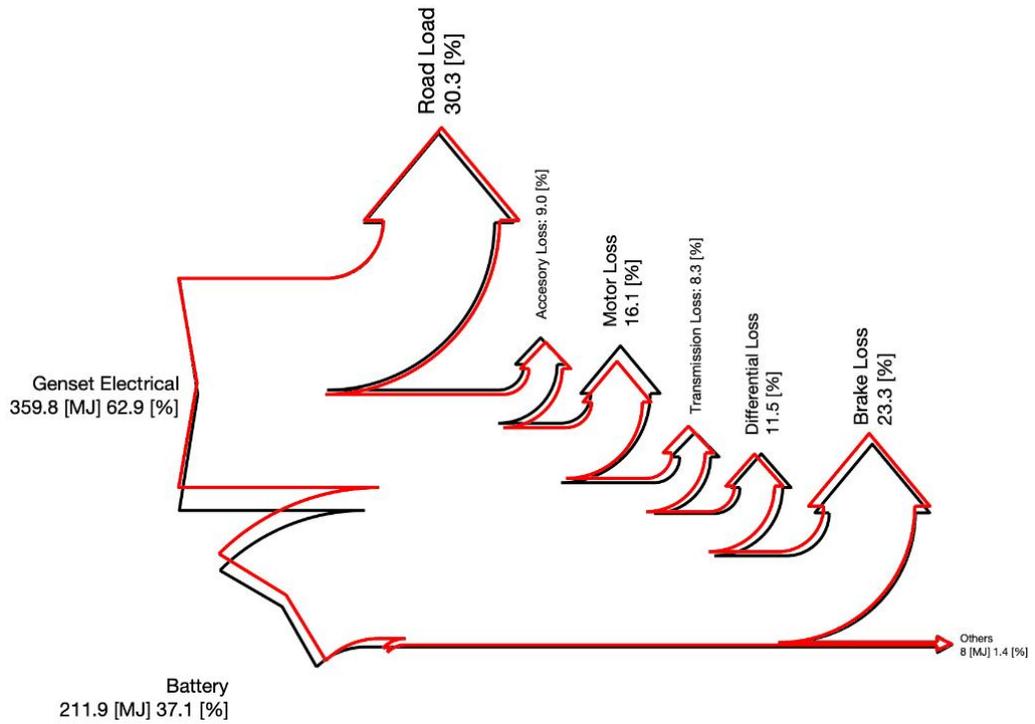


Figure 90: Sankey diagram with normalized comparison of energy flow between the forward simulator (black) and DP (red); Note: the driver inputs are matched between the two simulations in order to equate the power demand from the duty cycle

### **3.2 Literature Review: Look-Ahead Information in Delivery Trucks**

Improvement of efficiency of powertrains – electrified or otherwise – have been traditionally focused on enhanced performance of each individual component in the propulsion system and how they interact with each other. IC engine and transmission efficiencies have been incrementally optimized over the course of a century since the advent of the automobile. A similar approach has been ongoing in the area of electric drivetrains and energy storage system with a push to increase component level efficiency and minimize losses at the interfaces of power transmission. With diminishing returns offered by these approaches, the next leap in powertrain efficiency gains is expected to come from the smart utilization of big data, connectivity, sensor fusion and high computational power. These technologies offer real-time streams of information with a prescient view of the scenarios that may be encountered by vehicles, which can be leveraged to prepare the powertrain to strive towards an optimal response. The look-ahead data include measured quantities at the current time and generally have information about the spatial future.

Control strategies are adapting to take advantage of the aforementioned information available to vehicles – especially those that have predictable driving missions such as delivery vehicles and refuse trucks. In existing literature, the methodologies used range from statistical approaches that store and retrieve historical driving data to predict the future driving demand with high certainty, to neural networks that assign weighting factors

to parts of the drive cycle to dictate power source, and receding horizon controllers (e.g. MPC) that utilize look-ahead data in their predictors.

Feedforward optimal control algorithms such as Dynamic programming, PMP and convex optimization are acausal and demand complete a priori information. In the real-world, it is unrealistic to assume full knowledge of the exact future trajectories. Even in the unlikely scenario of complete driving mission information, the computational complexity of these algorithms rules out their real-time implementation.

The underlying utility of look-ahead information is that it allows one to predict the temporal or time-based future trajectories of the vehicle state (velocity) based on the knowledge of its spatial future in the driving route. Spatial future consists of the vehicle's surrounding and how they may manifest in space in the future, such as traffic density, traffic light status, stop signs etc. If the application is a delivery vehicle, the spatial elements in the future of the driving route may be more predictable than that of an average passenger vehicle. This prediction is used to compute an optimal policy for the powertrain operation and such controllers are called look-ahead controllers due to the type of the data utilized in prediction. The following literature survey inherits results from Chapter 5 on look-ahead energy management written by Hegde. B, 2018 [68].

According to existing literature [68], predictive energy management strategies for HEVs can be classified based on its two major components: 1) prediction methodology; and 2) optimization algorithm. The prediction methodology culminates in the prediction of the velocity trajectory that the vehicle may follow based on the information from the look-

ahead enablers. The optimization algorithm is then tasked with devising an optimal way to prepare the powertrain and utilize the energy sources available on board to fulfil the control objectives.

Another approach to classifying these control strategies in literature is based on whether or not the control algorithm has access to the future velocity trajectory. Prediction methodologies without look-ahead information include frozen-time, exponentially varying, stochastic, and prescient MPC algorithms. Frozen-time – as the name suggests – assumes that the vehicle’s velocity will remain the same as current time for the future, and exponentially varying method assumes the vehicle’s velocity decreases exponentially in the future [69] [70] [71]. Stochastic methods utilize historical data to construct a Markov transition model for vehicle’s states. The Markov model is then used to predict the vehicle’s velocity into the future as a statistic. Such methods are highlighted in [70], [72], [73], [74] and [75]. Prescient algorithm offers a velocity prediction that is identical to that in the future, since it assumes that the full knowledge of the future trajectory is available in advance [70].

Look-ahead controllers are generally feedback controllers and their feedback not only come from the vehicle but external environment as well. Sensors such as GPS, RADAR and map databases can be used for to aid the look-ahead controller in predicting the future. They are generally implementable by virtue of their causality, however, the significant computation burden that comes with processing large amounts of look-ahead data may

make them expensive to implement on a production vehicle [76], [77], [25]. Further exploration of this topic is dealt with in Section 3.3.1.

A study by Ambuhl and L. Guzzella [78] predicts a velocity trajectory by using the current vehicle state and average vehicle speed over the entire trip. The authors in [79], [80] and [81] estimate the velocity profile of the vehicle by using traffic light phase information which in turn helps in the minimization of fuel consumption over the entire mission. [82] and [83] have explored the use of road grade, turns and speed limit information in the prediction of future vehicle speed profiles. The future velocity was assumed to be known based on the velocity trajectory of a leader vehicle in [84]. Future vehicle velocity trajectories are predicted by using traffic data and geographical relief preview in the works of [85] and [86] respectively.

The classification of controllers based on their optimization algorithm includes the likes of linear programming, quadratic programming, non-linear optimization, model predictive control etc. MPC with a linearized model for optimization is highlighted in [88]. ECMS-based methods [26] are used for predictive control in [87], [89] and [90]. Stochastic MPC is used in [72] and [70]. MPC and dynamic programming have been used in conjunction in the works of [71], [91], and [92] [93]. The effect of the optimality of the MPC control based on changing prediction horizon lengths has been discussed by Rezaei et al. in [91]. Gissing et al. [94] explored the predictive energy management problem with cabin heating considerations. Both the studies assumed that the exact future velocity trajectory of the

vehicle is known a-priori. Huang et al. [95] have published an excellent review on the usage of MPC for HEV energy management.

### **3.3 Look-Ahead Energy Management**

Look-ahead energy management is a predictive control policy that leverages information about some of the future events facing the vehicle in its impending route. The anticipated velocity profile of the vehicle is then estimated by bringing together this host of information, and the hybrid power-split is then accomplished by means of an optimal control algorithm. The conditions change during a drive mission due to perturbations such as delays due to traffic, or dynamic parameters such as the vehicle gross weight. The control robustness in a predictive algorithm is increased by feedback and the approach discussed here involves repeatedly calculating the fuel-optimal power-split in real-time. The technique is simplified and illustrated in Figure 91. At point A, the optimal solution is sought for the control problem that is defined over the horizon which is obtained by truncating the entire driving mission horizon. Only one time-step worth of control is applied to the system, and then this process is repeated at point B. A schematic for the

determination of the time horizon for predictive control based on various information inputs through connectivity and sensors is illustrated in Figure 92.

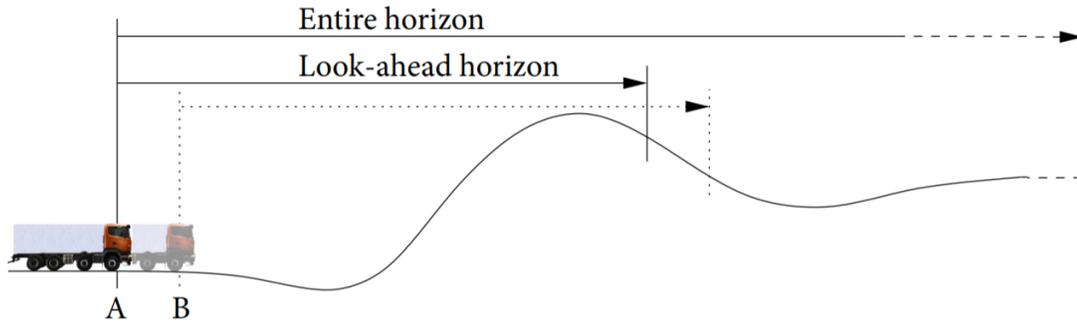


Figure 91: Illustration of a look-ahead control methodology [61]

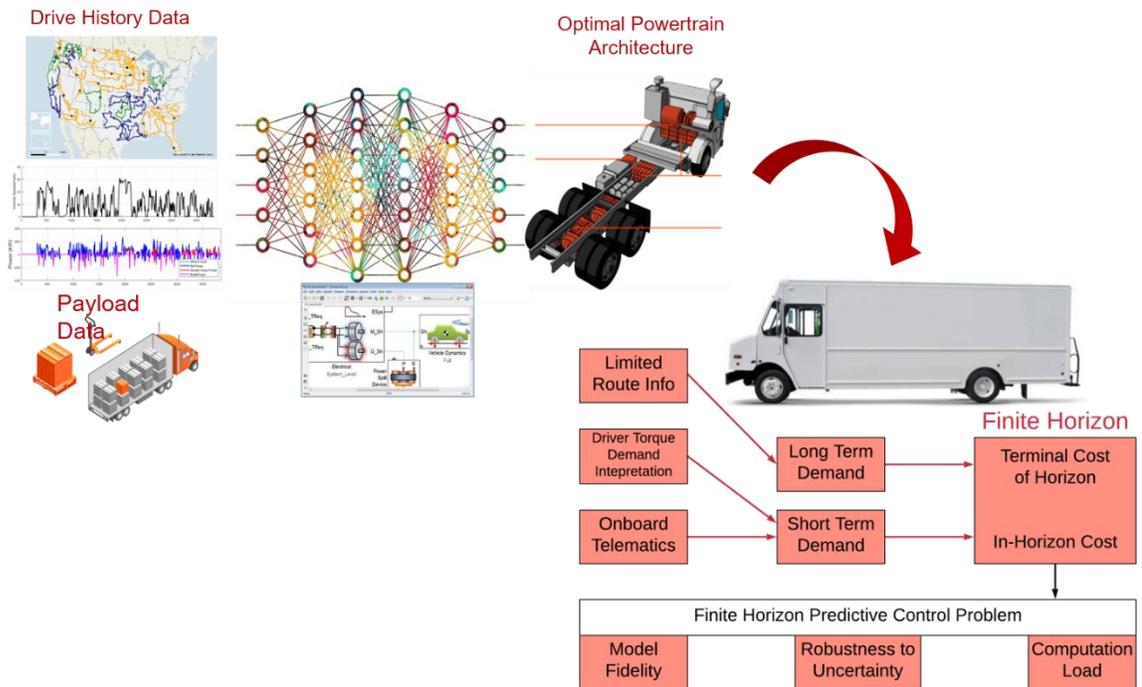


Figure 92: Determination of the horizon for predictive control based on various inputs

### 3.3.1 Feasible Look-Ahead Technologies

Various techniques and devices can be used to derive velocity profiles and power predictions from the available host of look-ahead data. It is understood that the reliability of look-ahead forecasts depends on the accuracy of sensors on-board the vehicle and their integration with the processing system. Four types of look-ahead predictors are highlighted in this section, introduced in the increasing order of their information fullness and intricacy. The model-based nature of these predictors sure that physical limits of the powertrain and driver behavioral uncertainties are incorporated in the predictions.

A cumulative enumeration of the different levels of velocity predictors is presented here (as inherited from Chapter 2 of the study on this topic by A. V. Rejendran [96]. At each incremental level, additional information is added towards refining the velocity prediction, and hence requires additional technological support to go up each level in the list. The general consensus is that the higher the predictor level, the better the accuracy of velocity prediction.

**1. Type 1 or SL-Predictors:** These predictors rely on road speed limit (SL) and grade information from maps. They use speed limit data of different roads along the route to forecast future speed. The inbuilt powertrain model needs to recognize the infeasibility of step increase in speed and predict a gradual acceleration that is in accordance with powertrain physical limitations and a predefined driver behavior model. Grade information is used to make road load predictions. However, it does not account for vehicle stops at stop signs and intersections.

**2. Type-2 or SLSS-Predictors:** Type 2 adds to Type 1 predictors. These predictors include Stop Sign (SS) information, and account for presence of stop signs and traffic light controlled intersections along the route. They also use a priori knowledge of the vehicle specifications to predict acceleration and deceleration performance requirements.

**3. Type 3 or SLSSLT:** Type 3 improves speed forecasts from Type 2 predictors by using known information about the route and expected turns during the route. This predictor is designed based on observations that human drivers reduce speed prior to sharp turns following which they accelerate and try to match the speed limit of the new road. Type 3 predictors recognize this driver behavior and improves upon the prediction from Type 2 predictor.

**4. Type 4 or V2X (Vehicle to X):** This predictor directly uses the velocity trajectory information from other vehicles that traversed a given route at some point in the past. Predictions are made either by extracting the speed of a prior vehicle as a function of location, or data from multiple vehicles can be fused together to yield a prediction. The fusion process may be defined such that the speed data is weighted differently based on vehicle type, time of travel etc.

The underlying assumption of this predictor is that human drivers are best indicators of another human's on road behavior. In this case, actual speed information from a similar vehicle that traversed a given road in the past is used to detect other factors that cause drivers to reduce speed apart from stop signs, traffic lights and turns. This includes the effect of crossing intersections with larger roads (where drivers reduce speed for safety

reasons) or the effect of curvy roads along the route. A practical implementation of V2X predictors is currently implemented in General Motors' Cadillac products under the brand name of Super Cruise [97]. Using LiDar mapping technology, Cadillac has mapped over 200,000 miles of compatible highways in the US, and established vehicle-to-vehicle connectivity between Cadillacs to enable hands-free driving.

Figure 93 is a graphical representation of how each level of velocity predictors assist in the forecasting of the future vehicle trajectory. Starting with the road speed limit alone, the profile follows a step function-like behavior which is highly unrealistic. Smoothing it out with blended acceleration and deceleration inputs make it more practical. Adding stop signs and turns further enhance the accuracy of the prediction. Finally, the speed data from V2X connectivity completes the velocity trajectory prediction with exceeding resemblance to that generated by a human driver.

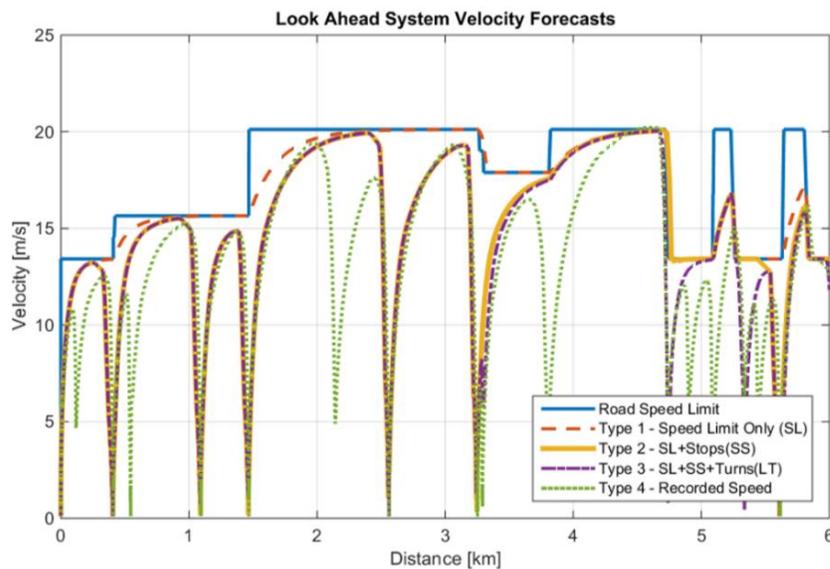


Figure 93: Driving profile estimation from different levels of look-ahead data [96]

## **List of Available Look-Ahead Technologies**

The exercise of driving a look-ahead energy management strategy towards an optimal direction is dependent on the effective utilization of future information from the enablers. Out of the host of data collected by these enablers, the right information needs to be extracted that can reliably predict the impending velocity profile to facilitate an optimal control action. In many scenarios, this will require the system to explore information that are beyond the confines of the vehicle operation and immediate vicinity, such as those provided by localized radar and imaging sensors. Tapping on the status of the transportation system and its surroundings can add value in improving the prediction accuracy of the future trajectories. The collection of sensors, methods and sources that can collate information which can contribute towards the estimation of future driving conditions are called ‘Technology Enablers’.

This section summarizes the results of a brief study that was conducted to classify the types of information that can be accrued from existing technologies, and how they come together to create a beneficial environment for predictive energy management of hybrids. The look-ahead parameters of interest that are derived from the road and traffic information can be classified into the following broad categories:

1. Speed limit
2. Effective dynamic speed limit
3. Stop signs and traffic lights
4. Traffic light status

5. Grade
6. Leader speed and distance
7. Leader type
8. Traffic density

A summary of enabler technologies, the information furnished by them, their utility and practical implementation cases are highlighted below:

**GPS:** The Global Positioning System feeds in the parameters such as position, velocity and direction of motion of the vehicle. This aids in the triangulation of the vehicle within its route, and giving it with feedback on its bearings. GPS finds application in maps and V2V communication.

**Inertial Measurement Units:** IMUs provide information on the instantaneous longitudinal, lateral and vertical acceleration values of the vehicle. This suite of information can be utilized in localization feedback, speed determination, detecting turns and road condition measurement. This finds direct application in driver assistance systems such as Electronic Stability Program (ESP), traction control and slip detection.

**Camera:** Camera imagery can be processed to recognize landmarks, obstructions ahead, lateral traffic light in the visible range and lane localization. This host of information can be used in vehicle localization, serves as an indicator of traffic density, tweaking the regeneration time and in determining driver intent recognition. The safety systems that are

derived from this information include collision avoidance, lane departure warning, lane keep assist, pedestrian detection, speed limit and sign detection.

**Radar:** Radar relays information on the relative speed of the vehicle ahead, headway and lateral and longitudinal obstructions. Radar information can contribute towards efficient adaptive cruise control, serve as indicator of traffic and identify or predict lane change/passing. Safety related features such as adaptive cruise control, collision avoidance, pedestrian detection, lane change assist, beam control, blind spot monitoring and turning assistant depend on the radar information.

**Lidar:** Lidar also transmits information on the relative speed of the vehicle ahead, headway, lateral and longitudinal obstructions and lane localization in specialized cases. This information can cater to efficient adaptive cruise control, serve as an indicator of traffic and identify or predict lane change/ passing. The emergence of lidar in automotive applications has sharpened the driver assistance features such as adaptive cruise control, collision avoidance, pedestrian detection, lane change assist, beam control, blind spot monitoring and turning assistant.

**Maps – offline:** Offline maps provide a multitude of information such as the grade of the road ahead, grade profile for multiple routes, routes to destination and speed limits. use: road load estimation, optimal routing; safety: hill descent systems, adaptive cruise control (speed limit), wrong-way driving warning

**Maps – online:** Adding on top of the features offered by offline maps, the online counterpart can offer routes, weather, wind, traffic information and construction zone information in real-time. This array of information can be exploited for online routing update, sensing effective speed limit, and determining optimal routing. The added safety benefits are adaptive cruise control, (based on road condition, speed limit), crosswind information and weather warnings.

**V2I/S:** Vehicle-to-infrastructure connections furnish real-time data on traffic light broadcasts, traffic density, traffic speed broadcast, weather and road conditions. This can serve towards velocity optimization or increased regeneration time, traffic density information and ramp-up shaping. The driver assistance systems of relevance to this technology are intersection assistant (with GPS), turning assistant, traffic light control, crosswind information and weather warnings.

**V2V:** Vehicle-to-vehicle communication facilitates sharing of braking pedal signal, accelerator pedal signal and self-classification data between vehicles on the road. This information can be direct indicators of traffic and ramp-up shaping and further contribute towards the improvement of collision avoidance, adaptive cruise control, lane change assist, beam control, blind spot monitoring and turning assistant.

### **3.3.2 Quantifying the Benefits of Look-Ahead Information**

The vehicle's environment determines how the vehicle is driven on a certain route. Fortunately, some of the vehicle's environment is static with respect to distance traveled

by the vehicle. For instance, stop signs remain at the same locations and road slope doesn't change based on time of the day. Certain other aspects of the vehicle's environment such as traffic light phases or traffic density changes with time. Some of the environmental features can only be described statistically for the lack of simple models to describe them; they include wind direction, wind speed, behavior of other drivers on the road etc. While both travel distance and time constitute the future, features that are static in distance domain remain the same as time progresses. Hence, it is trivial to predict the status of such features into the future. For features that evolve with time, if we can measure their current state and understand how they evolve in time, then those features can also be predicted into the future with some uncertainty.

Sensor technologies and communication systems elaborated in Section 3.3.1 can provide data about certain features of the environment. The easiest elements to get the data for are stationary in space. The features evolving in time are harder to get data for but aren't entirely impossible to track. Look-ahead data can be used to accurately predict the future states of the vehicle and hence enable more informed, globally optimal energy management strategies [68]

To begin with, determining the power demand of the vehicle over the course of a driving mission is the starting point to achieving a globally optimal energy management strategy. Just like in the case of Dynamic Programming, a priori knowledge of all the prospective regeneration events, and the occasions where electrical energy can be used to offset fuel consumption allows the control algorithm to make decisions that are globally optimal.

Using the road method, the power demand of a vehicle can be expressed by the equation:

$$P_{veh} = V_k \left[ \frac{\rho_a C_d A_f V_k^2}{2} + MgC_{roll} \cos(\theta_k) + Mgsin(\theta_k) + M\dot{V}_k \right]$$

where  $V$  is the velocity of the vehicle,  $\dot{V}$  is the acceleration,  $M$  is the mass of the vehicle,  $g$  is acceleration due to gravity,  $\rho_a$  is density of air,  $C_d$  is co-efficient of aerodynamic drag,  $A_f$  is effective aerodynamic frontal area,  $C_{roll}$  is co-efficient of rolling friction,  $\theta$  is road grade in radians, and  $k$  suggests that it varies with distance.

Since it can be reasonably assumed that the road grade encountered by a vehicle is static across the route, and can be accounted for quite easily, to predict the power demand of the vehicle, it is imperative that the vehicle's velocity must be forecasted. The features of the environment of the vehicle into the future can be measured with look-ahead information. The challenge is to structure a methodology that directly links information about the future to the vehicle's velocity prediction.

Hence, it is highly advantageous to have the ability to accurately predict the velocity trajectory of the vehicle into the future. A reliable prediction of the velocity profile can lead to a well-informed energy management strategy that prepared the powertrain states in advance to best tackle the power demand ahead, rather dealing with it in a reactive manner that may likely be sub-optimal.

## **Impact of Look-Ahead Data Analyzed from the LEMS Project at OSU (2018)**

Studies by B. Hegde et al [68] showed that ‘information vs. benefit’ pareto front was subject to variation with traffic conditions. Traffic predictions and Ranging sensors did not exhibit improvement at low traffic conditions. But traffic predictions and ranging sensors were found to be valuable under heavy and moderate traffic.

The delivery truck characteristics corresponding to this study include lower speeds limits, frequent stops & turns, shorter traffic congestions. The result showed that speed limits and stop locations played key role in fuel economy, additional benefit due to ranging is not significant, traffic look-ahead adds significant value only if congestions are significant, turns and stops locations may add benefits in urban driving and that traffic look-ahead will be very useful in congested traffic scenarios.

The energy management system proposed in this work uses the look-ahead data to optimize fuel consumption over a trip with exact a-priori knowledge of the future. The implemented controller extracts velocity trajectories over varying horizon lengths and processes that data for energy management with an adaptive ECMS controller. This exercise is performed in order to quantify the best-case scenario of efficiency improvement, given an ECMS-based controller is implemented in the simulator.

Simulation results are first presented for a baseline *non-adaptive ECMS controller* with a fixed equivalence factor ‘s’, which brings the final SOC to a desired lower limit of ~20%. The drive cycle used as reference for the look-ahead information is the CERC Pickup and

Delivery Cycle, and the forward simulator used to implement the controller evaluation coincides with the model that was introduced in Section 3.1. Figure 94 depicts the SOC trajectory resulting from the baseline ECMS controller evaluation with a tuned equivalence factor to bring the final SOC to 20%.

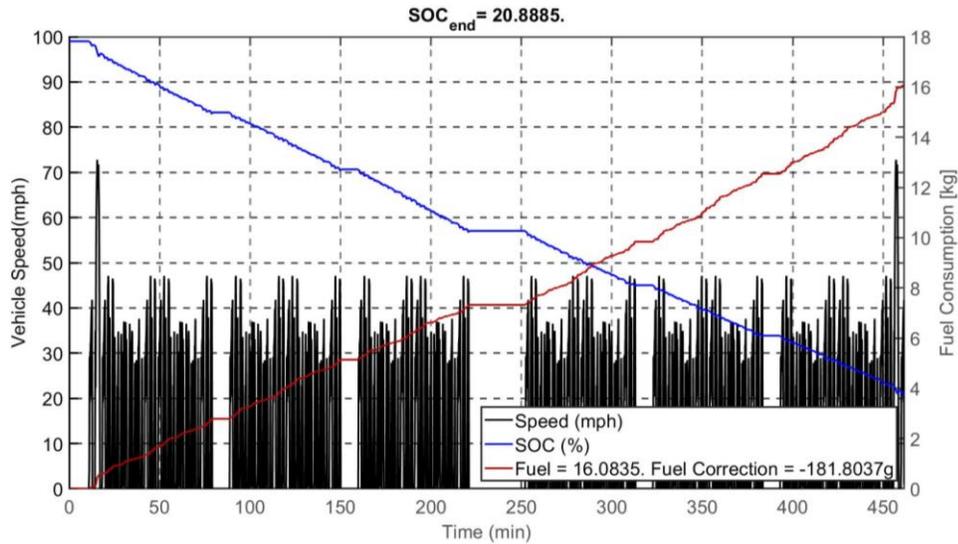


Figure 94: SOC profile and fuel usage resulting from the baseline ECMS controller

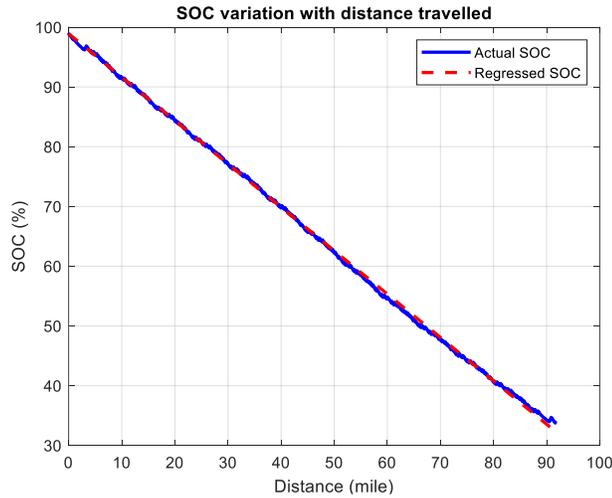


Figure 95: Distance-based reference SOC profile for the ECMS controller

Figure 95 represents the distance-based reference SOC profile that is enforced upon the ECMS controller to follow over the course of the driving mission. The linear trajectory can be expressed by the equation:  $SOC = SOC_0 - \alpha d$ , where  $\alpha = 0.726\%$  is the slope of the line (or the percentage SOC drop per mile) and  $d$  is the total trip distance. The value of the tuned equivalence factor in this case is 1.91, and the total fuel consumed comes out to be 16.08 kg, including the correction applied for the remaining energy in the battery. The number of engine start-stop events tally up to 503 in the absence of any limiting mechanism on it. This set of results obtained without any look-ahead information forms the benchmark for the adaptive ECMS technique.

### **3.3.3 Incorporating Look-Ahead Information in the Control Problem**

Following on the heels of the previous section, this part focuses on the implementation of an *adaptive ECMS* strategy that hinges on look-ahead information from a prescribed driving cycle to dynamically adapt the equivalence factor and hence enable fuel economy enhancement with no prior tuning of the said factor.

A Simulink implementation of the equivalence factor adaption block is depicted in Figure 96, along with indications of the velocity prediction, and a distance-based reference SOC tracking. The default horizon length (and hence, the frequency) of the equivalence factor adaption is 60 seconds. The reference trajectory of the SOC adjusts itself with every passing horizon based on the distance information available for the next 60 seconds, which, in turn, triggers the updating of the s-factor.

The s-factor adaption block does not refresh the equivalence factor if the vehicle is stationary for 60 seconds or longer. Additionally, this methodology is robust even in the absence of a suitable initiation of the equivalence factor.

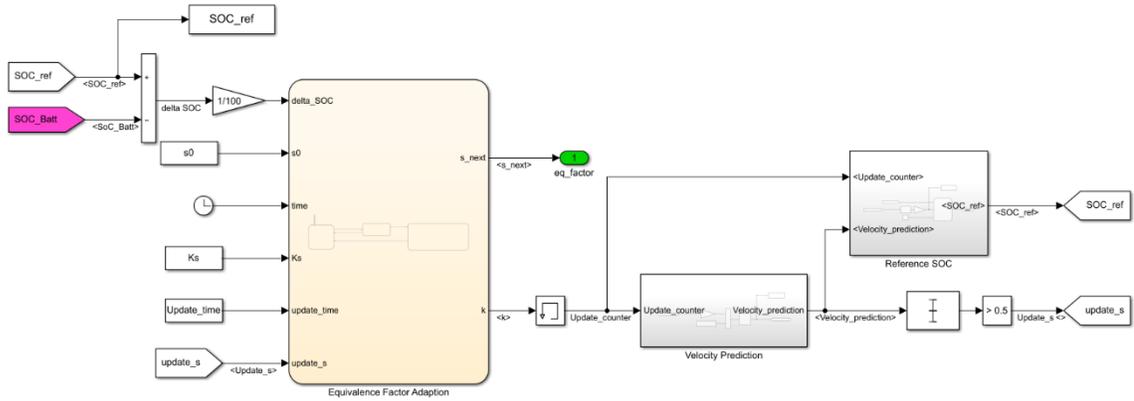


Figure 96: Simulink block with the Adaption Method for the Equivalence Factor

The ECMS equivalence factor is now actively adjusted by tracking the deviation of the actual SOC with respect to the reference SOC. The adaption rate can be varied based on the value of the update time variable. This results in a reliable self-correcting mechanism for the reference SOC tracking.

Based on the work by Rizzoni et al [26], the new equivalence factor for each upcoming horizon step ( $k+1$ ) can be adapted as follows.

$$s_{k+1} = 0.5(s_k + s_{k-1}) + K_s(SOC_{ref} - SOC(t_k))$$

Where,  $s_k$  and  $s_{k-1}$  are the equivalence factors corresponding to the previous two horizons,  $K_s$  is a calibration parameter (equal to 1.8 in this case),  $SOC_{ref}$  is the reference SOC at the current time and  $SOC(t_k)$  is the actual SOC at the current time  $t_k$ .

The velocity prediction uses the CERC Pickup and Delivery drive cycle as the accurate estimate of the future horizon (full information). The extraction procedure for the predicted velocity is depicted in the Simulink representation in Figure 97.

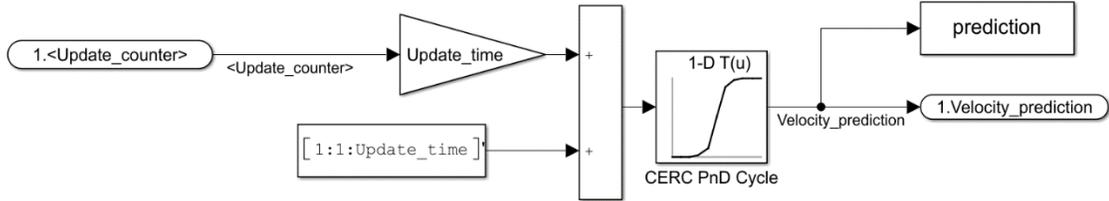


Figure 97: Velocity prediction block using the CERC P&D drive cycle

The distance covered in each impending horizon is determined through the integration of velocity for that horizon, and the reference SOC for each horizon step is calculated as:

$$d_{prediction} = \int_{t_k}^{t_{k+1}} v_{prediction}(t) dt$$

$$\Delta SOC_{prediction} = \alpha d_{prediction}$$

$$SOC_{ref}(t_{k+1}) = SOC_{ref}(t_k) - \Delta SOC_{prediction}$$

The methodology for SOC adaption is implemented in Simulink as depicted in Figure 98.

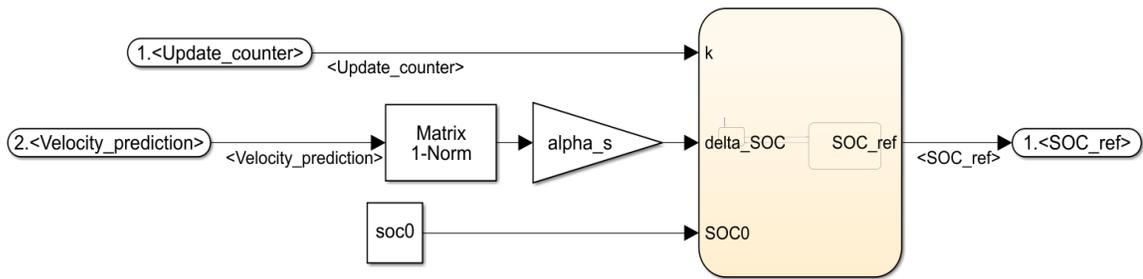


Figure 98: Reference SOC profile generation

Simulation results are presented for the *adaptive ECMS controller* with the equivalence factor ‘s’ being updated with every upcoming horizon, and finally bringing the final SOC to a desired lower limit of ~20%. Figure 99 depicts the SOC trajectory resulting from the A-ECMS controller evaluation with an arbitrary initial equivalence factor of 2. The resulting fuel economy and engine start-stop events are highlighted in contrast with the baseline controller in Table 45. It is evident that even without any calibration effort to find an optimal s-factor, the fuel consumption retains the same value as that of the benchmark controller. This is validated by the fact that the average equivalence factor of the A-ECMS controller over the entire mission coincides with that of the tuned value of its baseline counterpart.

The significant reduction in the number of start-stop events with the implementation of the A-ECMS controller directly translates to an improved emissions performance for the vehicle at no absolute cost in fuel consumption. This can be attributed to the timely

adaption of the s-factor which simultaneously helps follow the reference profile, while not demanding excessive state changes from the engine to do so.

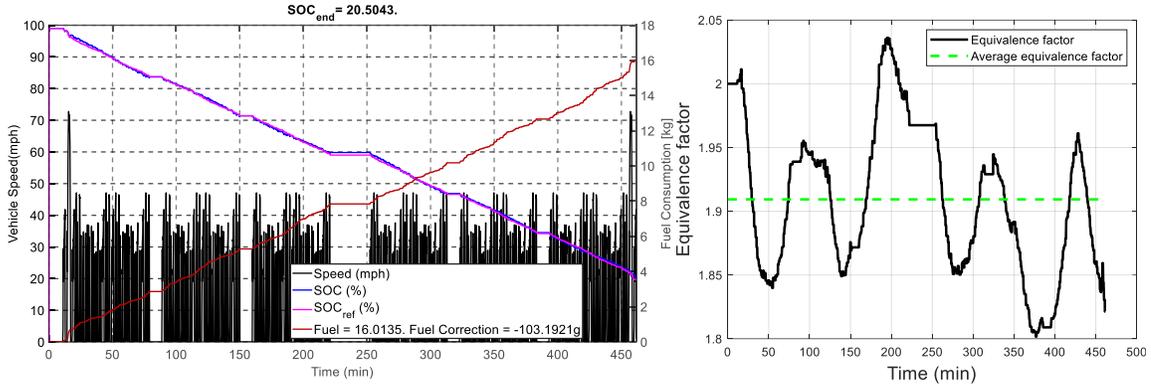


Figure 99: SOC trajectory, fuel consumption and s-factor adaption with A-ECMS

Table 45: Comparison of the benchmark ECMS controller against adaptive ECMS

Parameter	ECMS	A-ECMS with distance based SOC prediction
Fuel consumption	16.08 kg	16.01 kg
Engine Start-stops	503	453 (emissions benefit)
S-factor	1.91	1.909 (average)
Horizon length change	NA	20/60/300 seconds
Information needed	NA	Adaption, calibration parameters and look-ahead information

In order to evaluate the impact of horizon lengths on the results, two more iterations are performed with horizon lengths of 20 seconds and 300 seconds respectively. The results for the SOC trajectory and the equivalence factor evolution are summarized in Figures 100 and 101.

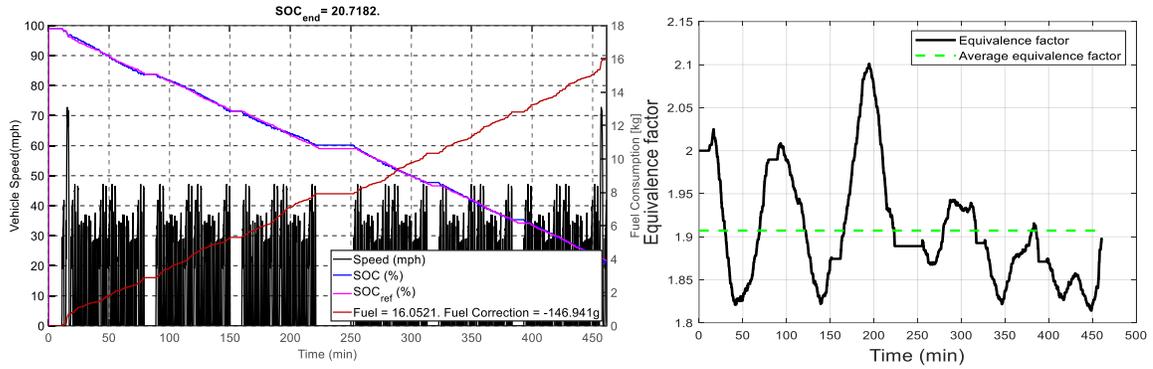


Figure 100: SOC trajectory, fuel consumption and equivalence factor adaption with the A-ECMS controller (horizon length: 20 seconds)

Changing the horizon lengths did not yield any notable variation in fuel consumption which hovered around the baseline value of 16 kg. On a similar note, the number of engine start stop events remained fairly steady in comparison to that in the 60-second horizon case. From Figures 100 and 101, it is also evident that the s-factor adaption becomes unstable with higher length of horizon, and that the SOC deviates more from the reference as the horizon is widened.

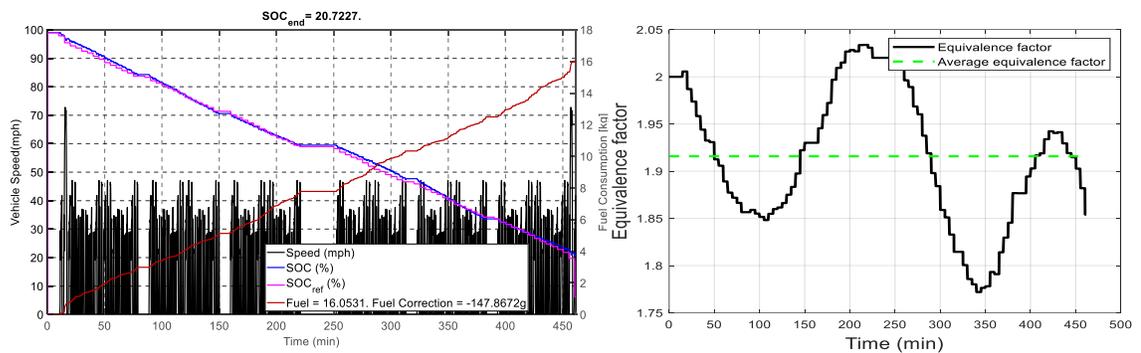


Figure 101: SOC trajectory, fuel consumption and equivalence factor adaption with the A-ECMS controller (horizon length: 300 seconds)

In order to evaluate the impact of the initial value of the equivalence factor on the results, two more iterations are performed with s-factors 3 and 1 respectively. The results for the SOC trajectory and the equivalence factor evolution are summarized in Figures 102 and 103.

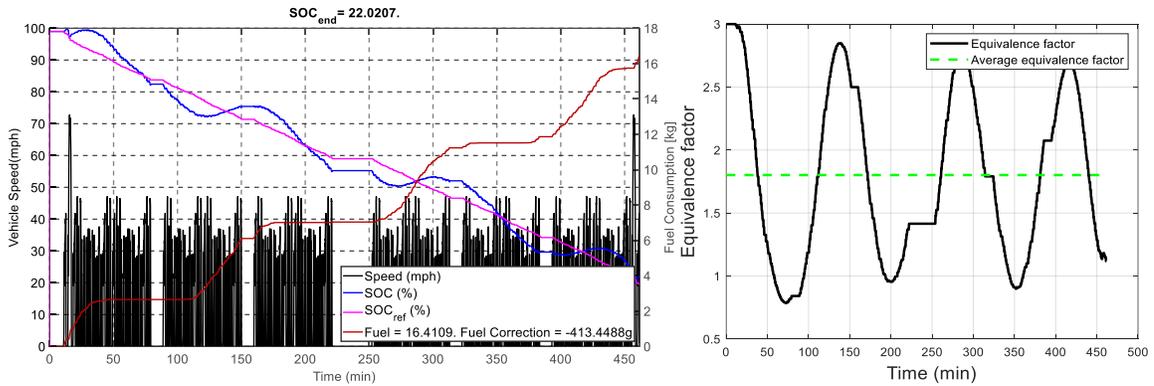


Figure 102: A-ECMS controller (initial equivalence factor = 3)

Changing the initialization of the equivalence factor yielded slightly higher values of fuel consumption over the baseline, which hovered around 16.5 kg. On the engine start-stop front, the former resulted in much fewer start-stop events (411), while the latter induced a higher count at 475, although both numbers fare better than the baseline ECMS count. From Figures 102 and 103, it is also evident that the s-factor adaption remains smooth with varying initialization values, however, the SOC deviates more from the reference in both cases depending on the calibration parameter  $K_s$ .

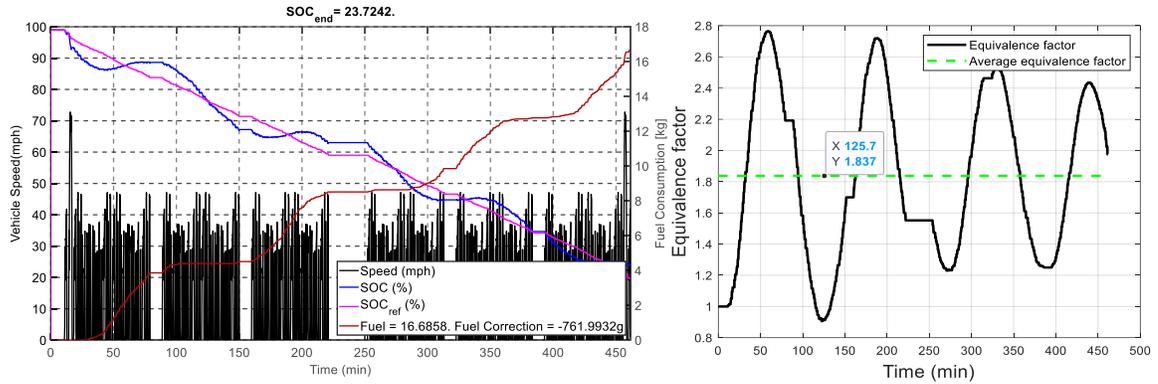


Figure 103: SOC trajectory, fuel consumption and equivalence factor adaption with the A-ECMS controller (initial equivalence factor = 1)

### 3.4 Summary of Results

The outcome of the study in this chapter suggests the limitation of the A-ECMS controller even with full knowledge of the drive cycle available in a charge depleting series HEV, in comparison to the level of efficiency that can be achieved the global optimal dynamic programming. Recall from Section 3.1.7 that DP can provide up to 35% better fuel efficiency in this particular architecture without the driver inputs matching (actual velocity and braking strategy), and up to 5% better efficiency with driver inputs equated.

Since receding horizon controller applications such as model predictive control (MPC) are beyond the scope of this thesis, the primary advantage from integrating an online adaptive ECMS algorithm with the selected candidate lies in the fact that its fuel economy and engine start-stop outcomes are relatively robust even with the lack of calibration of the equivalence factor specific to the drive cycle. The ideal horizon length is in the range of one minute, and the results are comparable to that of a tuned ECMS controller, all the while

providing enhanced engine start-stop control authority, which translates to a direct emissions advantage.

The shortcomings of the A-ECMS controller even with full look-ahead information of the velocity profile may stem from the fact that ECMS as a standalone is not adequately equipped to tackle optimization of a hybrid nature – ones that include both continuous and discrete states such as electric machine/genset power and gear selection/genset state respectively. Hence, a rule-based controller that is capable of handling different types of states may be necessary to further improve fuel economy and minimize start-stops on top of what was achieved by A-ECMS in this application.

## **Chapter 4. Traffic-in-the-Loop Simulation for Controller Evaluation**

The central idea behind this chapter is the realization that fuel economy extends beyond the realm of powertrains and is influenced by many factors that are beyond the physical confines of the vehicle systems. These features constitute the broader environment outside the vehicle, and are termed environmental factors. These aspects have seemingly random behavior and are non-causal in nature, which make isolating their effects a daunting task. A traffic-in-the-loop (TIL) simulator exposes the vehicle systems to these factors to varying degrees and duration in order to verify the robustness of the controller in response to them, independent of the driving mission.

### **4.1 Introduction to Traffic-in-the-Loop Simulation**

As detailed in Section 3.1.1, a fixed drive cycle-based approach to evaluate controllers fails to expose the powertrain to scenarios related to vehicle interaction with varying traffic density and aggressiveness, infrastructure and the causal nature of events that are part of real-world driving. The overall interdependency of the transportation system on various elements makes these factors unpredictable. For example, the velocity trajectory of a vehicle is affected by its interactions with traffic lights, speed of other vehicles, road grade, driver behavior etc. Standard drive cycles subject vehicles to temporally fixed set of inputs on every iteration, and fail to account for random events.

To evaluate the effectiveness of a controller, there needs to be a simulator that goes beyond the realm of driving cycles. Such a system will need to have the capability to facilitate a two-way interaction between the powertrain and the environment in every step of the action. The perspective shifts from following designated velocity profiles to viewing it from the standpoint of the entire transportation system. Such a scheme facilitates a setting that allows more reliable calibration of controllers, and enables accelerated development of intelligent powertrains and safety systems.

The subsequent sections of this chapter describe various components of TIL simulation, including its architecture and concludes with a demonstration of results obtained from it.

### **Impact of Driver Behavior on Fuel Economy**

Experimental validation by fuel economy testing authorities over the years, coupled with academic research by Sharer et al [98] and Ahmed et al. [99] point out that variation in fuel economy caused by driving styles and on-road conditions can add up to 35%. While driver behavior has been cited as the highest contributing element towards variation of real-driving fuel economy from certified estimates, the second and third most influential factors are interaction of vehicles with traffic, and infrastructure such as stop signs, speed limits, traffic lights etc. respectively. These are natural factors which affect driver behavior that go unaccounted for in synthetic driving cycles traditionally used for fuel economy evaluation. As a result, it is observed that the aforementioned three elements are interrelated to a certain degree.

Such social and infrastructure related interactions on driver behavior can account for variation in trip parameters like trip-time and fuel economy caused by random events. Moving to a traffic-in-the-loop based simulation helps recreate a good degree of randomness encountered by human drivers, given that the calibration of the traffic generation is realistic and that the powertrain has sufficient fidelity to respond to stimuli. This methodology can be exceedingly better in replicating fuel economy and related performance results (e.g. emissions) that are obtained through driving a vehicle in real world scenarios as opposed to a dynamometer testing.

### **Consequence of Real-world Driver Behavior**

Although it may appear in the onset that realistic driving behavior is completely governed by causality of events in a real-time manner, the truth is that any present driver action may have more complicated consequences. For instance, an aggressive acceleration event before approaching a traffic signal that is soon to turn red will enforce the application of brake and a mandatory stop event. On the other hand, a calculated speed trajectory by the driver can result in cruising uninterrupted through the traffic signal without the need for a braking event by allowing sufficient time for the lights to go green.

A more advanced example of non-causal effect is the artificial limitation or de-rating of the powertrain induced by driver behavior, powertrain limitations, traffic and interaction with the infrastructure. Advanced driver assistance systems (ADAS) adaptive cruise control (ACC) or speed control enforced by stability systems can also induce dynamic effects on fuel economy that have no room for consideration in traditional drive cycle-based

powertrain simulation. Once again, an integrated powertrain-traffic simulation environment has the scope to quantify these impacts in a bid to make controller evaluations all the more representative.

### **Types of Traffic Simulators**

Traffic simulation is the mathematical modeling of transportation systems. These simulated environments accommodate for the presence of stop signs, traffic lights, lane changes and traffic with a spread of different vehicles; driven by drivers of various types. It also imposes varying properties for roads including, road-based speed limits, lane widths and prohibited lanes. The behavior of member vehicles to the above parameters are described by discrete rules or mathematical formulations that are often differential equations [[100](#)].

The velocity profiles followed by vehicles on a traffic simulator are not predefined. Each vehicle's behavior is determined by its acceleration limits, driver behavior model, and road speed limits. Given a traffic free virtual road, vehicles in this environment accelerate to a steady speed and maintain that speed in the absence of any external disturbances like traffic or stop signs. This steady state speed for each vehicle, depends on the vehicle properties and type of driver assigned to it. Certain driver models may cause a minor oscillation of this steady state speed, about a given mean with a pseudo-random amplitude and frequency. Further, member vehicles decelerate to avoid collisions in the presence of other vehicles or when approaching traffic lights and stop signs. This steady state speed for each vehicle, depends on the type of driver assigned to it. Further, the vehicles members decelerate to avoid collisions in the presence of other vehicles or ahead of traffic lights and stop signs.

Traffic simulation techniques can be classified into 3 basic types: Macroscopic, Mesoscopic and Microscopic. The study in this thesis is limited to microscopic traffic simulators, i.e. every vehicle element in the traffic environment serves as an independent entity with the ability to adhere to traffic laws and are governed by 'behavioral sub-models' [101].

**Microscopic Traffic Simulations** track individual vehicle movements on a second by second or sub-second basis. In such simulations, there will be a 'warm-up' period before the system reaches a desired traffic density on its network. SUMO, the traffic simulation package used in this work falls under this category. The computation load for such simulation techniques increase with the number of vehicles being simulated.

**Mesoscopic Traffic Simulation:** When dealing with larger simulation areas or higher traffic count, microscopic simulation techniques can become slow as they are process intensive and compute the behavior of every vehicle individually. To simplify this problem, transportation elements can be analyzed in small groups. Within each group the elements are considered homogeneous. Vehicles can be grouped according to various factors but mostly based on direction of traffic flow.

**Macroscopic traffic simulations** deal with aggregated characteristics of transportation elements, such as combined traffic flow dynamics & zonal-level travel demand analysis. E.g.: Google Maps.

## 4.2 Literature Review: Traffic Simulators for Controller Evaluation

Various attempts have been made towards utilizing the concept of traffic simulation in conjunction with numerical powertrain simulators for fuel economy prediction. A summary of these attempts is presented in this section. The traffic-in-the-loop framework introduced here differs from these prior attempts since it forms a closed loop co-simulation where the powertrain model and the traffic simulation environment interact dynamically through an in-built two-way communication link. The co-simulation is setup such that the powertrain dynamics and the driver properties of the host vehicle affects its behavior in the traffic simulator at every time step and vice versa.

A highly comprehensive study on the most important models and theories that characterize the flow of highway and city traffic in its many facets was conducted by the Committee on Traffic Flow Theory and Characteristics in 2001 [102]. They present the various models that have been developed to characterize the relationship among the traffic stream variables: speed, flow and concentration. This publication also explores in high detail the human element in the context of the driver-vehicle system, and the discrete components of its performance. Macroscopic car flow models, traffic impact models, flow in signalized traffic intersections, safety schemes, fuel consumption and emission models are also explored in detail. The study concludes with a traffic simulation model that explores a wide variety of dynamical problems.

Elsewhere, Bajpai et al [103] proposed an efficient interface to couple adaptive control strategy and traffic simulator. This interface mediates between traffic control system and

traffic simulator and provides online interaction to simulation from the control strategy. Additionally, a module to estimate the vehicular delay due to the control strategy is developed and tested in SUMO. Studies by Doniec et al. [104] proposed a behavioral multi-agent model for road traffic simulation. Their contribution involves simulation of traffic in a real intersection and comparison of the simulated traffic flow with the real flow to highlight the relevance of the approach.

J. Maroto et al. [105] described a microscopic model that is able to simulate traffic situations in an urban environment in real time for use in driving simulators. Two types of vehicles were considered in the simulation, namely the user-driven vehicle at the center of the simulation model and the other vehicles that interact with it and its surroundings, which configure the developed traffic model. Simulation was performed in a reduced zone, called the control zone, surrounding the user-driven vehicle. Such a model is immediately applicable to large-scale driving simulators for driver training, traffic control studies, and safety studies.

### **4.3 Cross-Platform Simulation with Urban Drive Cycles**

By combining traffic simulation with a physics-based powertrain model, the simulation framework proposed in this chapter overcomes drawbacks of drive cycle-based powertrain simulation as detailed above. It allows users to run a virtual road test of a vehicle, in a simulated environment with various sensors and control algorithms in loop. Given the nature of vehicle behavior in traffic simulators, velocity profiles generated from such techniques are more realistic, provided the traffic and infrastructure elements are tuned

appropriately with accurate data. Once calibrated, infinite number of drive scenarios can be obtained by varying 'seed numbers' corresponding to each simulation run- in the traffic modeling software.

Simulation of Urban Mobility (SUMO), is an open source, highly portable, microscopic, multi-modal traffic simulation. It allows to simulate how a given traffic demand which consists of single vehicles moves through a given road network. The simulation allows addressing a large set of traffic management topics. It is purely microscopic: each vehicle is modelled explicitly, has an own route, and moves individually through the network. Simulations are deterministic by default but there are various options for introducing randomness.

SUMO provides the capability to turn street map into trajectories for 2-dimensional vehicle models to run on. The additional features include the addition of traffic lights, speed limits, environment vehicles to interact with the ego-vehicle. The project started in October of 2019, with an objective to connect the Simulink-based simulator with traffic environment.

The project objectives include prototyping of the powertrain architecture and investigation of proper control strategy. The objectives span a wide spectrum from vehicle modeling to control design. To address these objectives with a relatively small and efficient team, a flexible suite of simulation tools is required. Combining the past wisdoms and lessons learned at the Center for Automotive Research, a solution of providing not just one, but cluster of interdependent tools is proposed and being implemented.

To represent and analyze the aspect of powertrain optimization, hybrid energy management, driver, external environment, available information for control, single tool that does all doesn't exist. Instead, a pool of different tools is brought together. In this case, three distinctive simulators are used as shown in Figure 104.

- A MATLAB script-based simulator that follows a predefined drive-cycle, doesn't have a driver and runs Dynamic Programming (DP) as the control algorithm;
- A MATLAB Simulink-based simulator that implements a PID-based driver who tries to track a predefined drive-cycle, and runs A-ECMS as the control algorithm;
- A SUMO-based traffic simulator that simulates road map and traffic for a host vehicle.

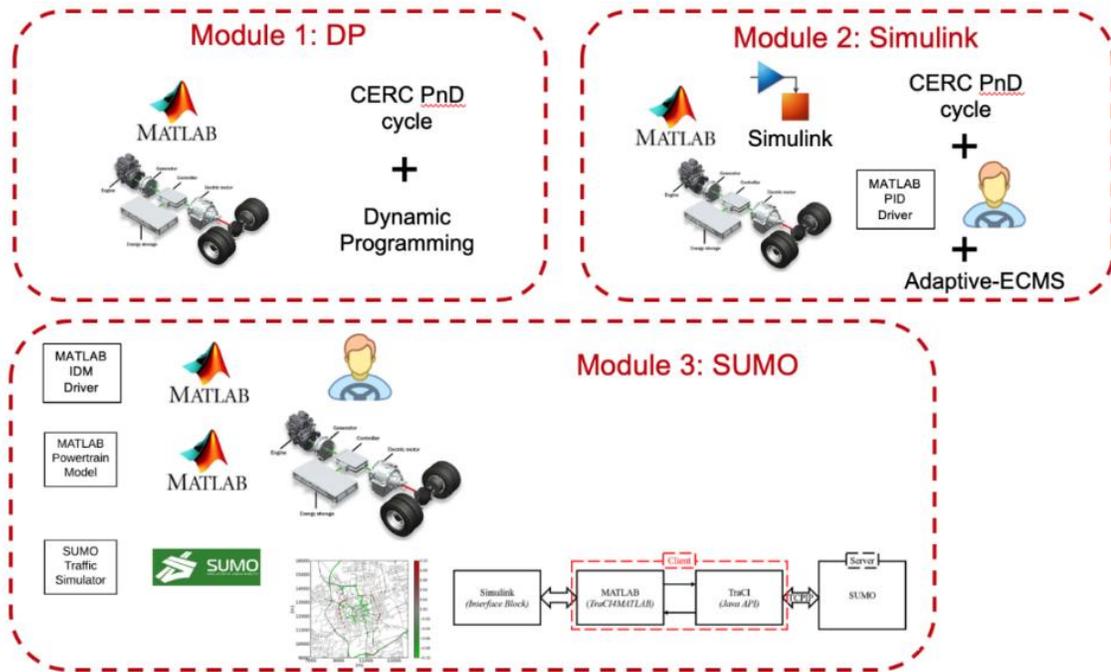


Figure 104: Cross-platform simulation

#### **4.4 SUMO – Simulation of Urban Mobility**

To overcome the deficiencies in current practices pertaining to powertrain simulation as discussed earlier, a framework that aims to integrate the real-world factors such as non-causality, vehicle-infrastructure interaction and traffic behavior into a conventional powertrain simulator is presented here. This enables the user to capture dynamics that are associated with real-world systems and study the effect of factors external to the powertrain on the overall system efficiency and performance. In the proposed framework, the online coupling of SUMO and Simulink allows the adaptation of vehicles behavior during simulation runtime.

Further, TIL helps to capture the mean and variance of on-road fuel economy without the need for physical road testing. In theory, the precision of such estimations is constrained only by the fidelity of models used. The primary goal of this co-simulation is to describe a ‘host vehicle’ in the SUMO environment, whose dynamics and controls are governed by a Simulink based powertrain model. As this host vehicle moves in the virtual world; a Simulink based dynamic powertrain plant model is utilized at the back-end, to compute the parameters such as, but not limited to: Host speed for next time step, instantaneous fuel consumption, engine operating points, gear shifts etc., while simultaneously exercise various control actions (e.g. acceleration limiting, torque split etc.). The speed computed by the powertrain model is then fed into SUMO as an ‘command input’ before every simulation step. This would make sure that the behavior of vehicle model within SUMO is realistic, and within dynamical bounds as defined in the vehicle model. This closed loop

simulation allows all associated causality to be strictly maintained. The following Section 3.5.1 describes the powertrain simulator's integration with a traffic simulation package that helped achieve the goals of this work. The salient features of SUMO are as follows:

- SUMO includes all applications needed to prepare and perform a traffic simulation (network and routes import, DUA, simulation)
- Simulation features:
  - Space-continuous and time-discrete vehicle movement
  - Different vehicle types
  - Multi-lane streets with lane changing
  - Different right-of-way rules, traffic lights
  - A fast openGL graphical user interface
  - Manages networks with several 10.000 edges (streets)
  - Fast execution speed (up to 100.000 vehicle updates/s on a 1GHz machine)
  - Interoperability with other application at run-time
  - Network-wide, edge-based, vehicle-based, and detector-based outputs
  - Supports person-based inter-modal trips
  - Network Import
  - Imports VISUM, Vis-sim, Shapefiles, OSM, RoboCup, MATsim, OpenDRIVE and XML-Descriptions
  - Missing values are determined via heuristics
  - Routing
  - Microscopic routes - each vehicle has an own one

- Different Dynamic User Assignment algorithms
- High portability
- Only standard C++ and portable libraries are used
- Packages for Windows main Linux distributions exist
- High interoperability through usage of XML-data only
- Open source (GPL)

Figure 105 elucidates the simulation paradigm followed by SUMO in bringing together its capabilities in order to prepare an environment suitable to introduce a host vehicle for performing traffic-in-the-loop simulation. The procedure starts with selecting the route map through Open Street Maps resources and concludes in trace extraction that corresponds to the trajectory of the traffic elements that are generated in the simulation space. At the end of this process, the stage is set for the host vehicle to enter the simulation and exchange information to start the co-simulation with MATLAB.

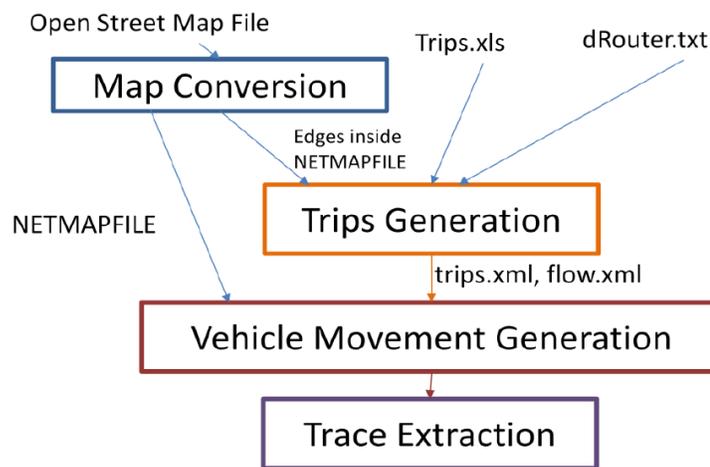


Figure 105: Vehicle mobility simulation process in SUMO

#### **4.4.1 Forward Simulator**

The Forward Simulator is a Simulink-based simulator that incorporates the selected optimal range extender HEV architecture arrived at in Chapter 2 with the Design Space Exploration exercise. The data for the powertrain components are sourced from industry partners as well as utilizing market research. The full disclosure on the component modeling in the forward simulator is discussed in Chapter 3.1.

#### **4.4.2 Open Street Maps for Route Generation**

SUMO has evolved into a full featured suite of traffic modeling utilities including a road network capable to read different source formats, demand generation and routing utilities from various input sources (origin destination matrices, traffic counts, etc.), a high performance simulation usable for single junctions as well as whole cities including a “remote control” interface (TraCI) to adapt the simulation online.

##### **Road Network Generation**

To run a traffic simulation, a network file is required. A network file is simply an XML file that contains information about how junctions (nodes) are connected with roads (edges consisting of one or more lanes), how lanes are connected at junctions, what are the top speeds and shapes of lanes and also the information about traffic light logic. There are several ways to obtain such a network file, but most of them mean using the NETCONVERT tool that comes with SUMO. One option is to provide SUMO-specific XML files as input for NETCONVERT. Those inputs include an XML file to list all nodes,

another one to list all edges, one for describing edge types (i.e. what is the maximum speed, number of lanes, allowed/disallowed vehicle types etc. for a given edge type), and finally a file to describe how edges are connected in nodes (e.g. to specify if left turns are allowed in a given junction or not). The edge type and connection files are optional and SUMO uses default values if edges don't have any type information specified and guesses the connections at junctions if unspecified. It is also possible to generate a random road network using the NETGENERATE tool. It builds three different kinds of abstract road networks: “Manhattan”-like grid networks, circular “spider-net” networks, and random networks. Each of the generation algorithms has a set of options, which allow adjusting the network’s properties. Figure 106 shows examples of the generated networks.

However, using a network that exists in the real world is likely to be more desirable. The road network importer NETCONVERT converts networks from other traffic simulators such as VISUM, Vissim, or MATSim. It also reads other common digital road network formats, such as shapefiles or OpenStreetMap (OSM). Besides these formats, Netconvert is also capable to read fewer known formats, such as OpenDRIVE or the RoboCup network format.

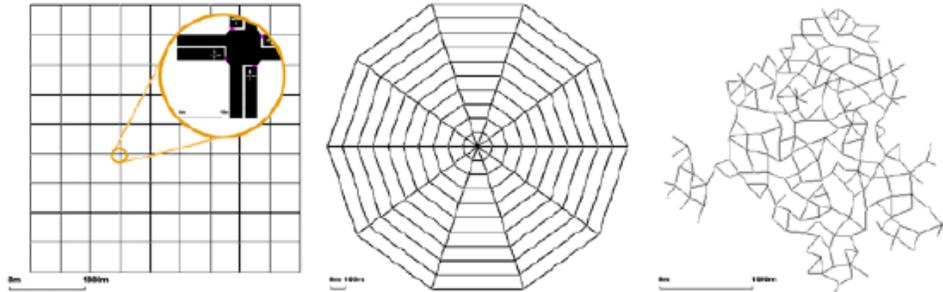


Figure 106: Examples of abstract road networks as built using “netgenerate”; from left to right: grid (“Manhattan”, spider, and random network)

### Road Network Using OSM Map

To use an OSM map for creating a road network for SUMO, map data needs to be downloaded. That can be done preferably from <http://www.openstreetmap.org/> by either selecting a rectangular area from the map or specifying a bounding box with geographic coordinates. But it is also possible to obtain a map programmatically by sending an HTTP request to the OSM API. SUMO even provides a Python script for making the API call. The downloaded OSM map is simply an XML file that contains a list of data primitives - nodes (a point on the earth's surface), ways (an ordered list of nodes) and relations (a data structure that documents relations between other elements). This OSM map file can then be given as input to NETCONVERT to transform it into a network file i.e. Netmap file that SUMO can use.

For this work, a map of the area around The Ohio State University was downloaded in OSM format from Open Street Maps and imported to SUMO. A sample route for CERC application is charted out in Figure 107.



Also, depending on the used input formats and set processing options, one can also find

- Districts;
- Roundabout descriptions.

#### 4.4.3 Inserting the Host Vehicle into the Co-simulation

The fundamental components in this simulator are summarized in this section. A high-level overview of the proposed co-simulation is illustrated in Figure 108.

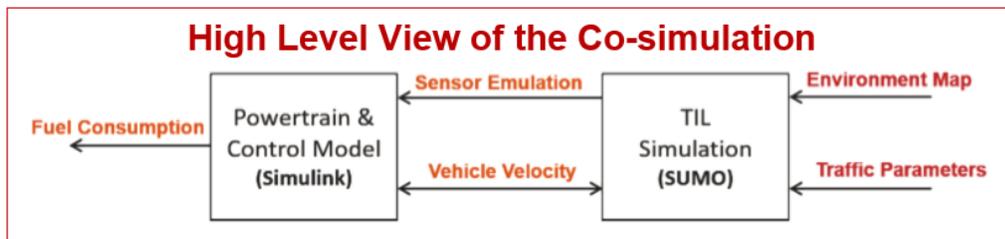


Figure 108: SUMO framework – a high level overview

**SUMO and Associated Support Structures:** These provisions are in place to facilitate the communication between SUMO and Simulink through MATLAB.

**Sensor Modules:** Outputs from SUMO such as (but not limited to) distance to and speed of vehicle ahead, current phase of upcoming traffic light en route etc. can be considered as sensor outputs. In this case, these values can be considered as output from a RADAR sensor and a V2I receiver respectively. Sensor noise can then be modeled separately by understanding individual sensors behavior and creating a custom noise pattern that can be added to the sensor outputs from the SUMO before being fed into the controller.

**Powertrain Controller:** The powertrain controller can range from a simple start stop controller for mild hybrids that use the sensor data to perceive the environment to make on-off decisions to more complex controllers where sensor data and robustness to uncertainty is crucial for optimal operation. These controllers process inputs from SUMO and yield control inputs to the powertrain.

**Powertrain and Vehicle Dynamics Model:** These models help calculate energy consumption and fuel economy values. Based on the fidelity of the system model, one can choose to observe dynamics involved in various components and study their effects on the environment or vice versa.

#### **4.4.4 Intelligent Driver Model (IDM) for Velocity Profile Generation**

The IDM predictor is deterministic and uses information such as headway distance, speed of the vehicle ahead, current velocity of the host vehicle and its acceleration limits to determine the host vehicle's predicted speed in the next time-step (set using the parameter window in Table 46). The IDM based velocity predictor in Simulink (Figure 109) is calibrated using parameters from the host vehicle's (SUMO) parameters at the beginning of each simulation.

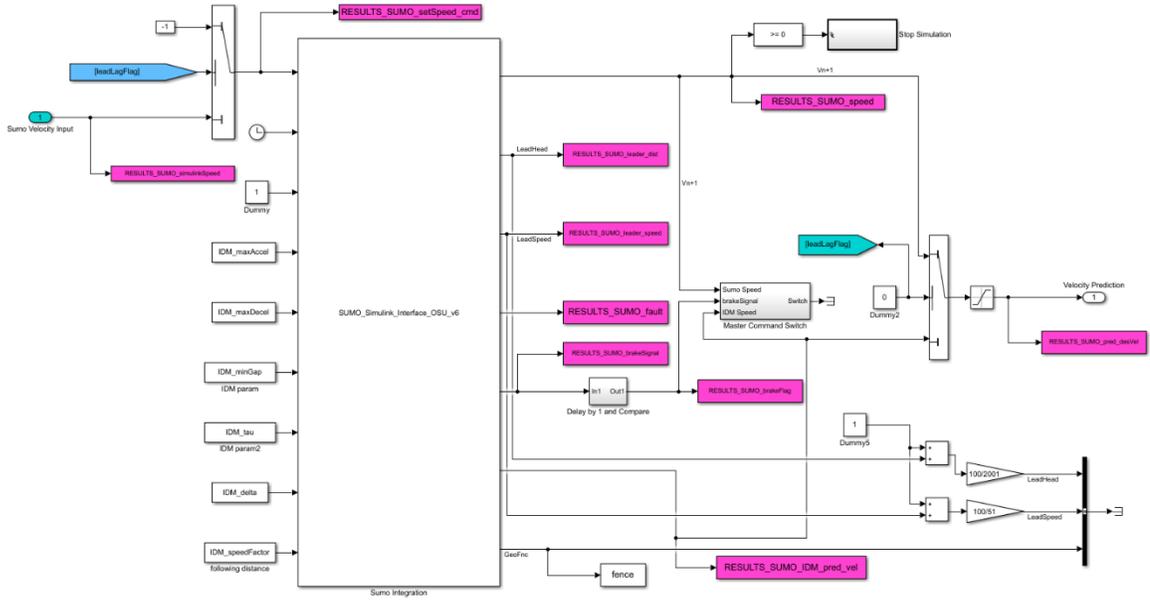


Figure 109: Simulink implementation of the Intelligent Driver Model (IMD) which is the interface for information exchange between MATLAB and SUMO

The IDM acceleration is a continuous function incorporating different driving modes for all velocities in freeway traffic as well as city traffic [106]. Besides the bumper-to-bumper distance ‘ $s$ ’ to the leading vehicle, the desired speed  $v_0$  and the actual speed  $v$ , the IDM also takes into account the velocity difference (approaching rate)  $\Delta v = v - v_1$  to the leading vehicle. The IDM acceleration function is given by

$$a_{IDM}(s, v, \nabla v) = \frac{dv}{dt} = a \left[ 1 - \left( \frac{v}{v_0} \right)^\delta - \left( \frac{s * (v, \Delta v)}{s} \right)^2 \right], \quad (1)$$

$$s * (v, \Delta v) = s_0 + vT + \frac{v\Delta v}{2\sqrt{ab}}, \quad (2)$$

Initialization of parameters of the Intelligent Driver Model (IDM) used in the simulations are provided in Table 46. These are varied in the later stages of the co-simulation to setup desired traffic scenarios, as shown in Figure 110. A simplified flowchart depicting the SUMO-MATLAB integration via the Intelligent Driver Model is given in Figure 111.

Table 46: Intelligent Driver Model (IDM) parameters

Parameter	Car	Truck
Desired speed, $v_0$	120 km/h	85 km/h
Free acceleration exponent, $\delta$	4	4
Desired time gap, T	1.5 s	2.0 s
Jam distance	2.0 m	4.0 m
Maximum acceleration, a	1.4 m/s <sup>2</sup>	0.7 m/s <sup>2</sup>
Desired deceleration, b	2.0 m/s <sup>2</sup>	2.0 m/s <sup>2</sup>

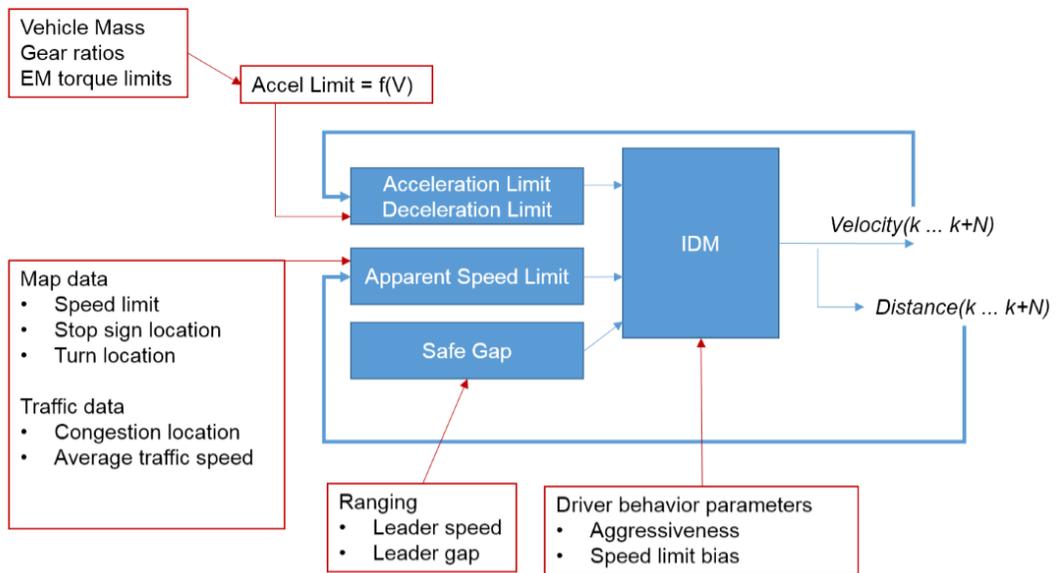


Figure 110: Paradigm of the Intelligent Driver Model (IDM) within SUMO

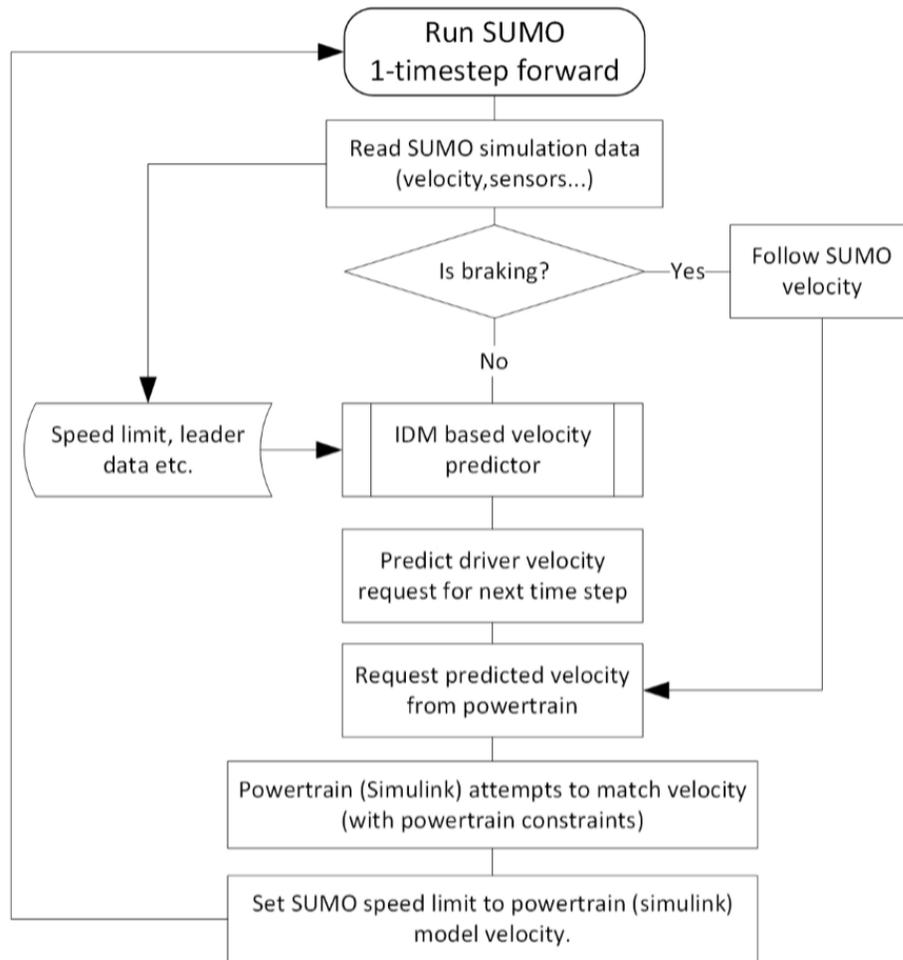


Figure 111: Flowchart depicting the SUMO-MATLAB integration via the IDM [96]

#### 4.4.5 Co-simulation Framework and Sensor Emulation

To cope with the deficiencies in current practices pertaining to powertrain simulation, a framework that aims to integrate the real world factors such as non-causality, vehicle-infrastructure interaction and traffic behavior into a conventional powertrain simulator in an effort to capture the dynamics that are associated with real-world systems and study the effect of environmental factors external to the powertrain on the overall system efficiency

and performance is elaborated here for common understanding. In this perspective, the functionality of the TraCI4MATLAB utility is essential to understand.

### **TraCI4MATLAB**

TraCI4MATLAB is an API (Application Programming Interface) developed in MATLAB that allows the communication between any application developed in this language and the urban traffic simulation software, SUMO. The functions that comprise TraCI4MATLAB implement the TraCI (Traffic Control Interface) application level protocol, which is built on top of the TCP/IP stack, so the application developed in MATLAB, which is the client, can access and modify the simulation environment provided by the server (SUMO). TraCI4MATLAB allows controlling SUMO objects such as vehicles, traffic lights, junctions etc., enabling applications like traffic light predictive control and dynamic route assignment, among others.

### **Co-Simulation Framework**

The integration of the Simulink-based powertrain simulator with SUMO traffic simulation has been accomplished in a two-step manner. First, the TraCI4MATLAB interface is used to link MATLAB and SUMO in a 'server-client' manner, where SUMO acts as the server with MATLAB acting as the client. TraCI4MATLAB uses a Java API to establish a TCP/IP communication link between SUMO and MATLAB. To ensure connectivity, an explicitly specified port (port 30000) to establish this communication passage is utilized. Further, this architecture ensures that multiple vehicle models running on various different

physical/virtual machines can simultaneously access the simulation parameters in a centrally operating SUMO server by connecting to the same port via TraCI. This is represented graphically in Figure 112. In this figure, the dotted lines enclosing the ‘TraCI’ and ‘MATLAB’ blocks signify that they operate as one integral unit by virtue of the fact that TraCI4MATLAB is implemented as MATLAB functions and reside within the MATLAB environment.

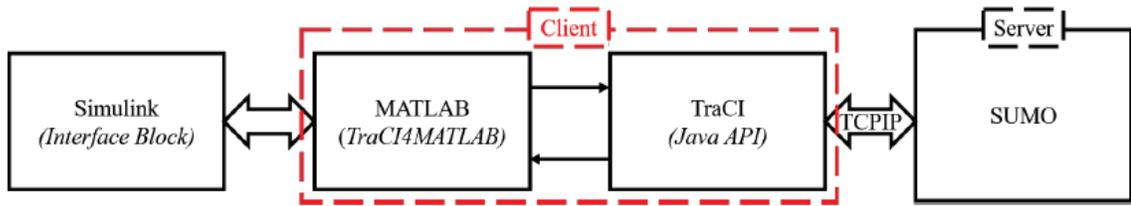


Figure 112: Information flow between Simulink and SUMO

Second, a communication link is established between the Simulink model and TraCI4MATLAB via an interface that resides within the powertrain model. This interface supports two-way flow of information, and thus works to close the loop between SUMO and the powertrain model. Hence, it is embedding a more detailed and customizable powertrain and control model into the traffic simulation. This architecture is shown graphically in Figure 113, which describes the closed loop information exchange that takes place in the designed framework.

The overall goal behind organizing the simulation framework is such that, given a-priori knowledge of the ‘host’ vehicle’s speed in the next time step, the vehicle model can treat it as a ‘virtual’ driver speed request. This speed request ( $\vec{v}_{k+1}$ ) is then subject to control and

vehicle dynamic limitations, hence resulting in modified speed request  $v_{k+1}$ . Further, speed  $\vartheta_{k+1}$  from the vehicle model is then sent to SUMO as a speed input. This method of Simulink leading the simulation and SUMO following the speed input, ensures that the SUMO and Simulink models are always tightly coupled. However, vehicle velocity in a future time step is not available as an output from SUMO. Thus, the authors set the Intelligent Driver Model (IDM) model as the driver model for the host vehicle within SUMO environment, and implemented the same equations in Simulink.  $\varphi_k$  carries all the required parameters from SUMO to be used by the IDM model shown above to predict the vehicle's velocity in the next time step  $\vec{\vartheta}_{k+1}$ .

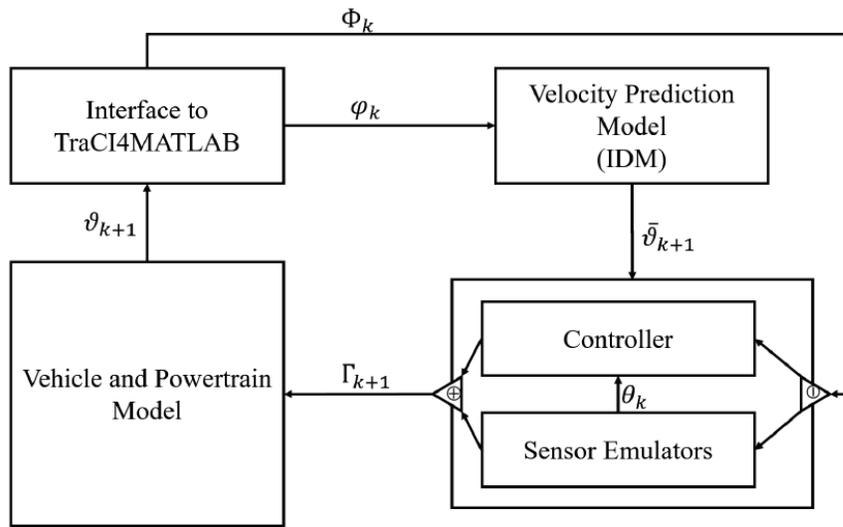


Figure 113: Information flow in closed-loop simulation

Since the model equations for the IDM are the same in both the SUMO and Simulink environments, the velocity prediction for step 'k+1' is accurate. This predicted velocity  $\vec{\vartheta}_{k+1}$  is fed into the vehicle controller as desired velocity. Subsequently, the controller

applies a set of inputs, given by  $\vartheta_{k+1}$  to the vehicle model. Based on control and component level dynamics, the velocity  $\bar{\vartheta}_{k+1}$  attained by the vehicle is computed by the vehicle model, given the latest control inputs and powertrain limits. This now forms the velocity input which is then set as an input to the SUMO vehicle via the interface. Upon receiving this input, SUMO carries out one simulation step, and the whole process repeats for the next time step. Hence, a closed loop co-simulation with traffic-in-the-loop is achieved. From a systems perspective, one can consider this integration process to be akin to replacing the driver in the vehicle model with the traffic simulator.

The co-simulator framework developed in this work is depicted in figure 114. This high-level figure illustrates all associated components and information flows. There are four basic components in this simulator: host powertrain, lead vehicle, traffic and driver.

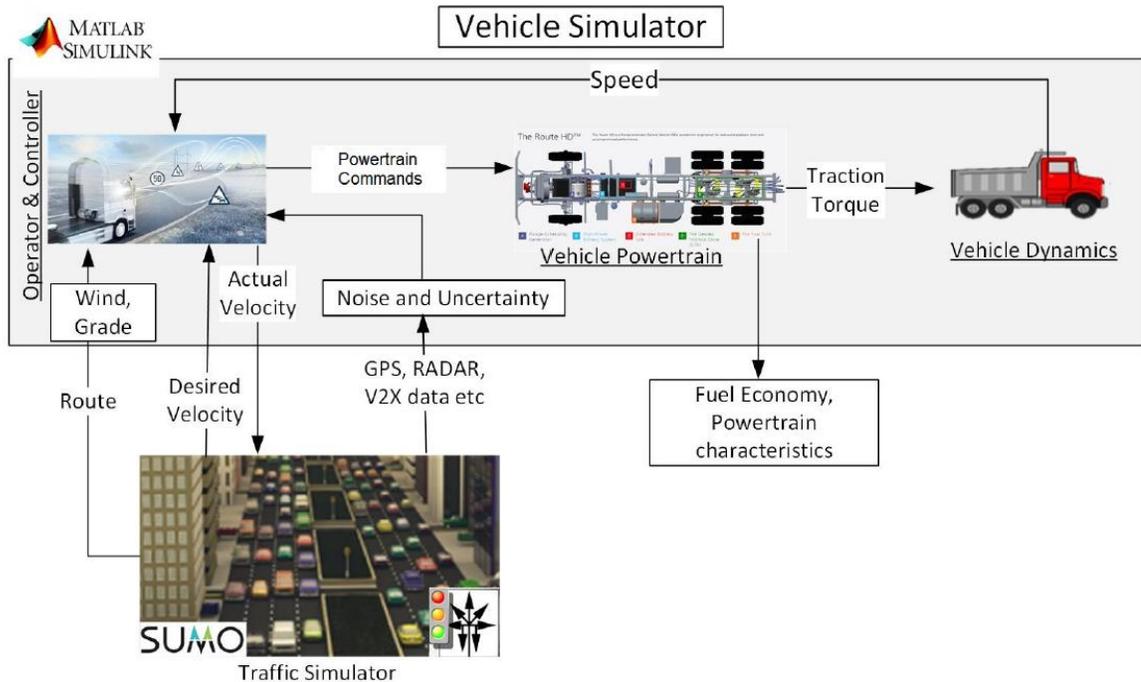


Figure 114: Bird's-eye view of the simulation framework

### Traffic-in-the-loop Powertrain Simulation

The primary objective of the co-simulation is to declare a 'host vehicle' in the SUMO environment, whose dynamics and control are governed by a Simulink-based powertrain model. As this host vehicle moves in the virtual world, the dynamic powertrain model is utilized in the backend in a closed loop manner to compute the parameters such as (but not limited to) host speed for next time step, instantaneous fuel consumption etc., and simultaneously exercise various control actions (e.g. acceleration limiting). The speed computed by the powertrain model is then fed into SUMO as a 'command input'. This would make sure that the behavior of the vehicle model within SUMO is realistic and within the physical and dynamic bounds as defined by the Simulink model.

There are several ‘TraCI’ commands, as shown in Table 47, which are used to interface the communication between MATLAB and SUMO. To appreciate how MATLAB communicates with SUMO to get specific information such as VehicleID, position, speed and LaneID etc. about a particular vehicle, for instance 'testVehicle', through ‘TraCI’ commands, please refer to the commands described in Table 47.

Table 47: TraCI commands to extract sensor information

<b>TraCI Commands/functions</b>	<b>Information</b>
traci.vehicle.getPosition('testVehicle')	Self-localization
traci.vehicle.getPosition('VehicleID')	Location of other vehicle(s)
traci.vehicle.getSpeed('testVehicle')	Self-speed
traci.vehicle.getLeader_osu('TestVehicle')	Leader ranging
traci.vehicle.getAccel(LeaderID)	Leader acceleration
traci.vehicle.getVehicleClass(LeaderID)	Leader classification
traci.vehicle.getLaneID(vehID)	Lane identification
traci.lane.getWidth (laneID)	Road width
traci.lane.getLength(laneID)	Road localization
traci.lane.getMaxSpeed(LaneID)	Speed limit
traci.edge.getLastStepVehicleIDs(edgeID)	Road traffic number
traci.edge.getLastStepOccupancy(edgeID)	Road utilization %
getLastStepVehicleIDs(edgeID) traci.vehicle.getVehicleClass(VehicleIDs) traci.edge.getLastStepLength(EdgeID)	Road demographics
traci.vehicle.getLastStepMeanSpeed(edgeID)	Edge average speed
traci.vehicle.getLastStepHaltingNumber(edgeID)	Number of vehicles in jam
traci.vehicle.getVehicleLights()	Traffic light position
traci.trafficlights.getPhase()	Traffic light phase
traci.vehicle.getVehicleLights()	Time to traffic light green
traci.vehicle.getVehicleLights() – post processed	Distance to green light
traci.simulation.getDistanceRoad(Current Edge, Current Pos, Destination Edge, Destination Pos, isDriving)	Distance to the destination

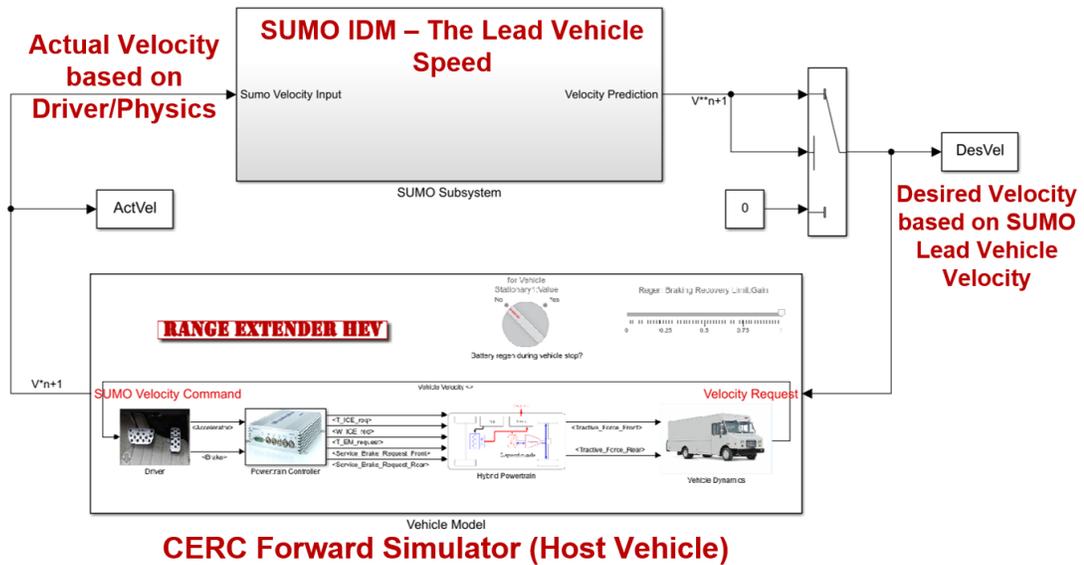


Figure 115: The complete layout of the powertrain-traffic co-simulation environment

## 4.5 Simulation Results and Discussion

A successful co-simulation has been obtained by integrating the forward simulator of the selected two-speed e-axle HEV candidate with the SUMO traffic-in-the-loop simulation environment for evaluation of the controller performance under various driving scenarios.

### 4.5.1 Drive Cycle Characterization

The specific driving route under consideration is a randomized traffic generated in the Columbus OSM street map elaborated in Section 4.4.2, with the host vehicle injected into it. This route is utilized here for the controller evaluation by subjecting the host vehicle to a multitude of traffic parameters such as driver aggressiveness, following distance, acceleration and deceleration limits during a driving scenario lasting approximately 20

minutes. The response of the host vehicle to its environment is limited by the ability of its default driver (driven by a PID controller) to follow the demanded trajectory.

#### **4.5.2 Analysis of the ECMS Controller Performance**

With the co-simulation environment in place, the ECMS controller can now be evaluated against a variety of traffic conditions that a vehicle may be subjected to in real-world driving. This section explores the impact of driver and traffic-induced factors such as maximum and minimum acceleration of the vehicle(s) driving ahead of the host vehicle, minimum time gap between these vehicles and the following distance.

The set speed by the SUMO simulator is essentially the lead vehicle speed in the SUMO traffic simulator. The host vehicle does not track the velocity set by the SUMO simulator accurately because the forward simulator driver is not adequately tuned for every scenario encountered in the traffic simulation. Hence, this is effectively a way of testing the robustness of the forward simulator physics as much as it is of the controller.

#### **4.5.3 Effect on Trip Fuel Economy and Start-Stops**

In general, the trends in fuel economy and start-stop events is heavily influenced by the aggressiveness of the lead vehicle driver (IDM), which compels the host vehicle to follow in response. A very low aggressiveness in the lead vehicle driver results in higher fuel economy, and allows the host vehicle to track the speed more accurately.

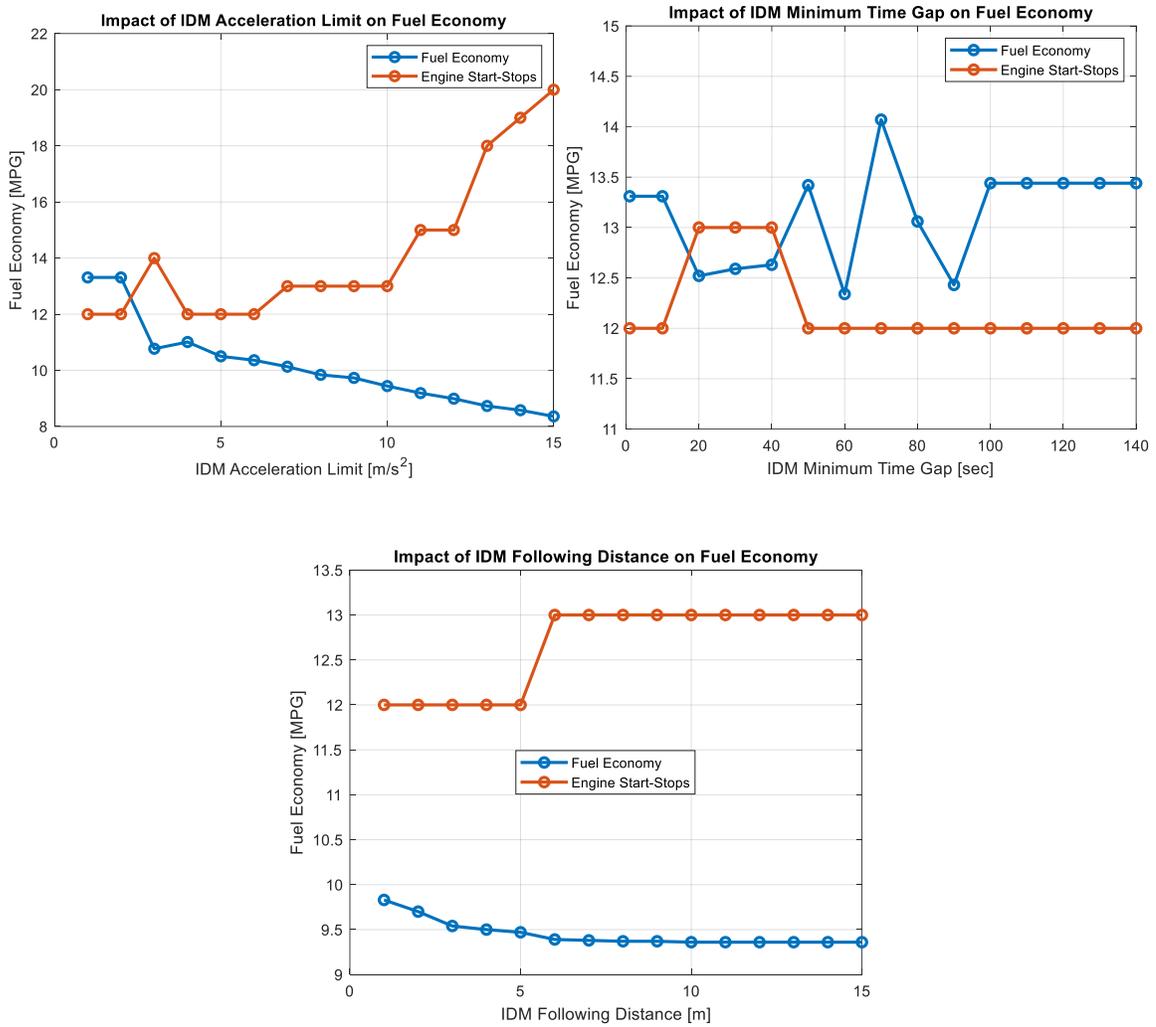


Figure 116: Impact of driver behavior on fuel economy and engine start-stops

Table 48: Statistical data: dependence of powertrain performance on IDM behavior

Parameter	Fuel Economy		Engine Start-Stops	
	Mean	Standard Deviation	Mean	Standard Deviation
IDM Acceleration Limit	10.15	1.51	14.2	2.7
IDM Minimum Time Gap	13.13	0.5	12.2	0.41
IDM Following Distance	9.42	0.14	12.72	0.46

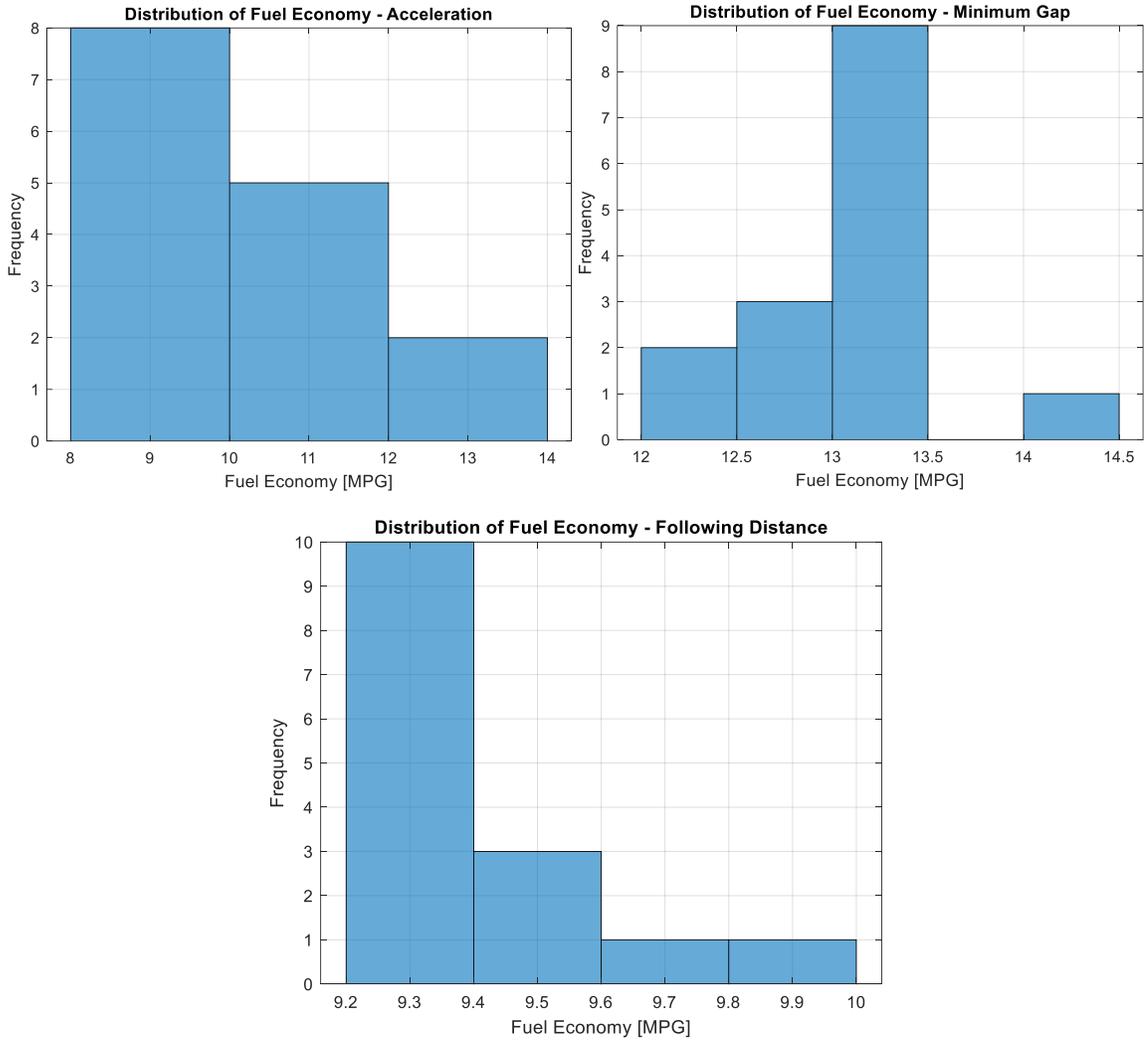


Figure 117: Statistical data: dependence of powertrain efficiency on IDM behavior

As expected, the statistical data in Table 48, and Figures 116 and 117 show that the highest deviation in the fuel economy is inflicted by change in acceleration limits of the driver, while the least impact is due to the lead vehicle following distance. The time gap maintained by the lead vehicle had an intermediate level of influence on fuel economy. In terms of the number of genset start-stop events, a clear trend was observed only in the case

of changing acceleration limits. Following distance and time gap did not yield any conclusive impacts on engine start-stop trends.

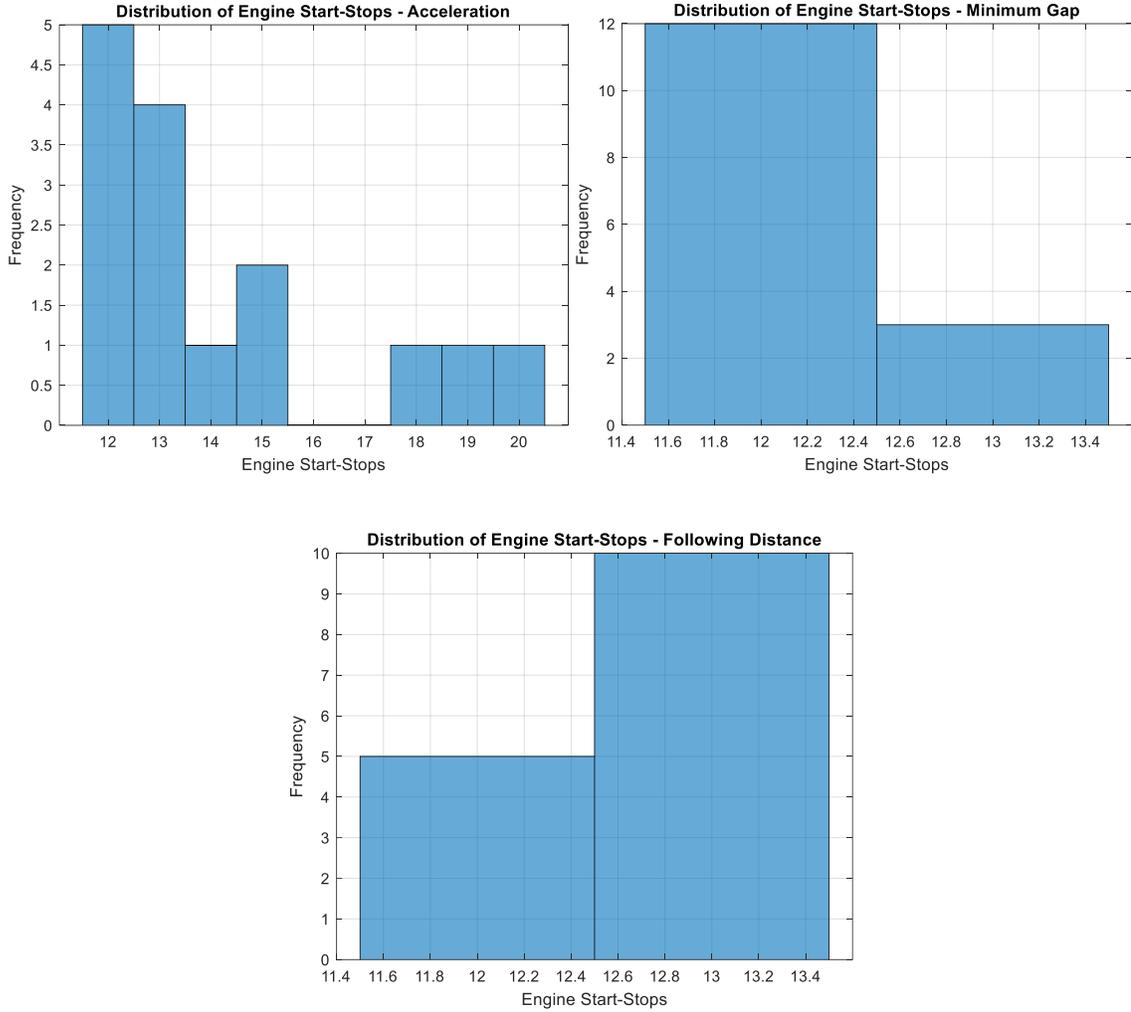


Figure 118: Statistical data on the dependence of genset start-stops on IDM behavior

The frequency distribution of the number of engine start stops highlighted in Figure 118 attest that the above driver behavior elements do not have a verifiable effect on the genset ON-OFF events, partly owing to the relatively small number of test cases (60 driving scenarios).

#### **4.5.4 Effect on Trip Emissions**

As elaborated in Section 3.1.3, trip emissions are largely down to the quantity of fuel consumed. Additionally, they can be heavily influenced by the number of genset start-stop events that impacts the temperature dynamics of the aftertreatment system in the case of the selected range-extended hybrid candidate.

A detailed exploration of the aftertreatment model dynamics is beyond the scope of this thesis. However, a simplified lump thermal model with an energy equivalence between the exhaust gas thermal energy and the catalyst temperature dynamics is implemented in the forward simulator. The following discussion throws light on the details of this exhaust thermal model, as well as the results generated with the A-ECMS controller.

#### **An Aftertreatment Model for the Forward Simulator**

The aftertreatment system dynamics is modeled here as a simple lumped thermal system with heat input from the generator set exhaust flow, which raises its temperature to the optimum degree needed for suitable conversion of the exhaust species (Figures 119 and 120). The catalyst also loses heat to the surroundings by means of convective losses. This convective heat transfer is accounted for in the following expressions.

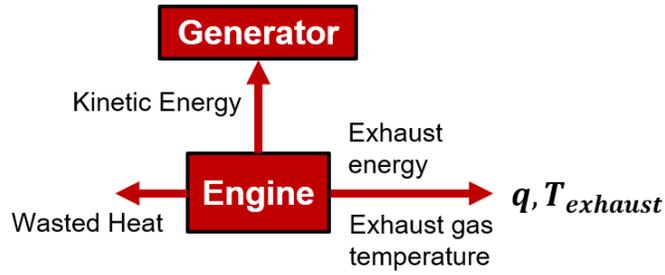


Figure 119: Energy dynamics in the generator set system

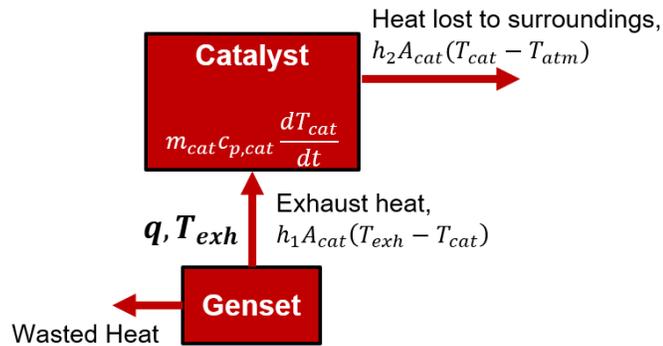


Figure 120: Thermal dynamics of the two-way catalyst

Convection heat transfer rate,  $q = hA(T_1 - T_2)$ ,

Where 'h' is the coefficient of convective heat transfer

The thermal resistance associated with convection is given by the expression:

$$R_t = \frac{\Delta T}{q} = \frac{1}{hA}$$

Thermal capacitance:  $T - T_0 = \frac{1}{C_t} \int_0^t q dt$ ,  $C_t = Mc$

Heat energy added to the catalyst:  $q = mc_p \Delta T$

Experimental data needed to execute this model in simulation include the map of the aftertreatment inlet temperature as a function of the engine torque and speed:  $T_1(Torque, \omega)$ , the map of the aftertreatment inlet mass flow rate as a function of torque and speed:  $\dot{m}(Torque, \omega)$ .

The temperature dynamics of the catalyst is given by:

- Rate of heat energy added to the catalyst (W):  $\dot{q} = m_{cat} c_{p,cat} \left( \frac{dT_{cat}}{dt} \right)$
- Heat transferred to the catalyst (W) =  $h_{exh\ to\ catalyst} A_{cat} (T_{exh} - T_{cat}) = h_1 A_{cat} (T_{exh} - T_{cat})$
- Heat lost to the surroundings (W) =  $h_{catalyst\ to\ air} A_{cat} (T_{ca} - T_{atm}) = h_2 A_{cat} (T_{cat} - T_{atm})$
- Energy balance for the aftertreatment system gives:

$$m_{cat} c_{p,cat} \frac{dT_{cat}}{dt} = h_1 A_{cat} (T_{exh} - T_{cat}) - h_2 A_{cat} (T_{cat} - T_{atm}) \quad (1)$$

The parameter values that are required execute this model in simulation include the mass, area and the coefficient of convective associated with the exhaust flow, the catalyst and the environment such as  $m_{cat}$ ,  $c_{p,cat}$ ,  $h_1 A_{cat}$ ,  $h_2 A_{cat}$ ,  $T_{exh}$ ,  $T_{cat}$  as functions of engine operating points. Note that the exothermic heat generated by catalysis is not modeled separately here, and is assumed to be lumped with the first term on the right-hand side of equation (1).

A map of the engine's exhaust temperature is built based on the data obtained from experimental results. Then a catalyst thermal model is integrated with the forward simulator. Note that the catalyst cooling during the engine-off periods is strongly affected by the temperature that is assumed for the exhaust flow. If it is assumed that when the engine is off, no exhaust flow occurs, the exhaust temperature drops to some fixed value (e.g. 150 C), and the catalyst will cool down faster, because it will reject heat both to the atmosphere and the exhaust. Also note that the exhaust temperatures are quite high in general, and since the engine is operated on the optimal operating line for the most part, the catalyst usually stays in that higher range of its temperature spectrum. Therefore, the results show that only engine stops of around half an hour can significantly cool down the catalyst below the optimum operating range.

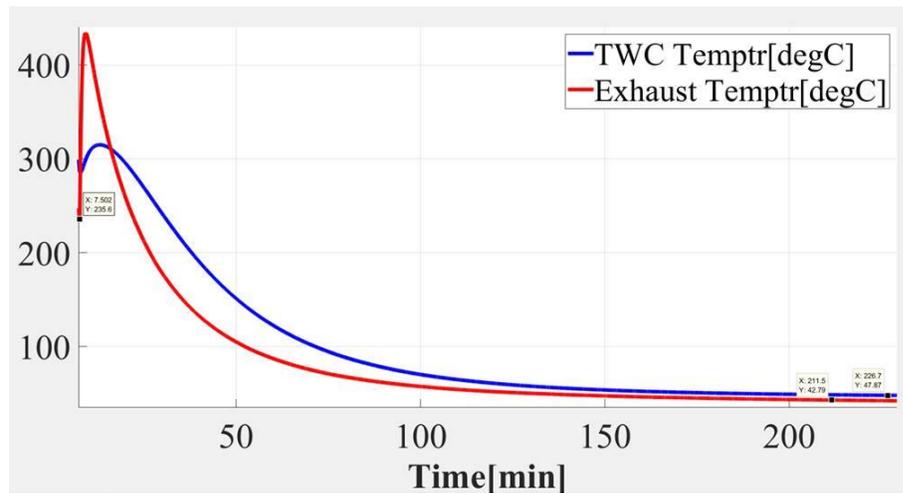


Figure 121: Temperature dynamics of the catalyst based on the exhaust gas flow

The exhaust temperature during cooling down is not the same as the previous value noted when the engine is on. It also cools down as depicted in the experimental result in Figure 121. The model parameters are such that there is a non-zero heat transfer coefficient from

the exhaust flow to the catalyst when there is no exhaust flow, i.e. no convection heat transfer. The model can then capture the heat soaking section (the very first increase of temperature in the plot where the engine is already off). Otherwise, the modeled two-way catalyst (TWC) temperature will fall straightaway instead of going up first.

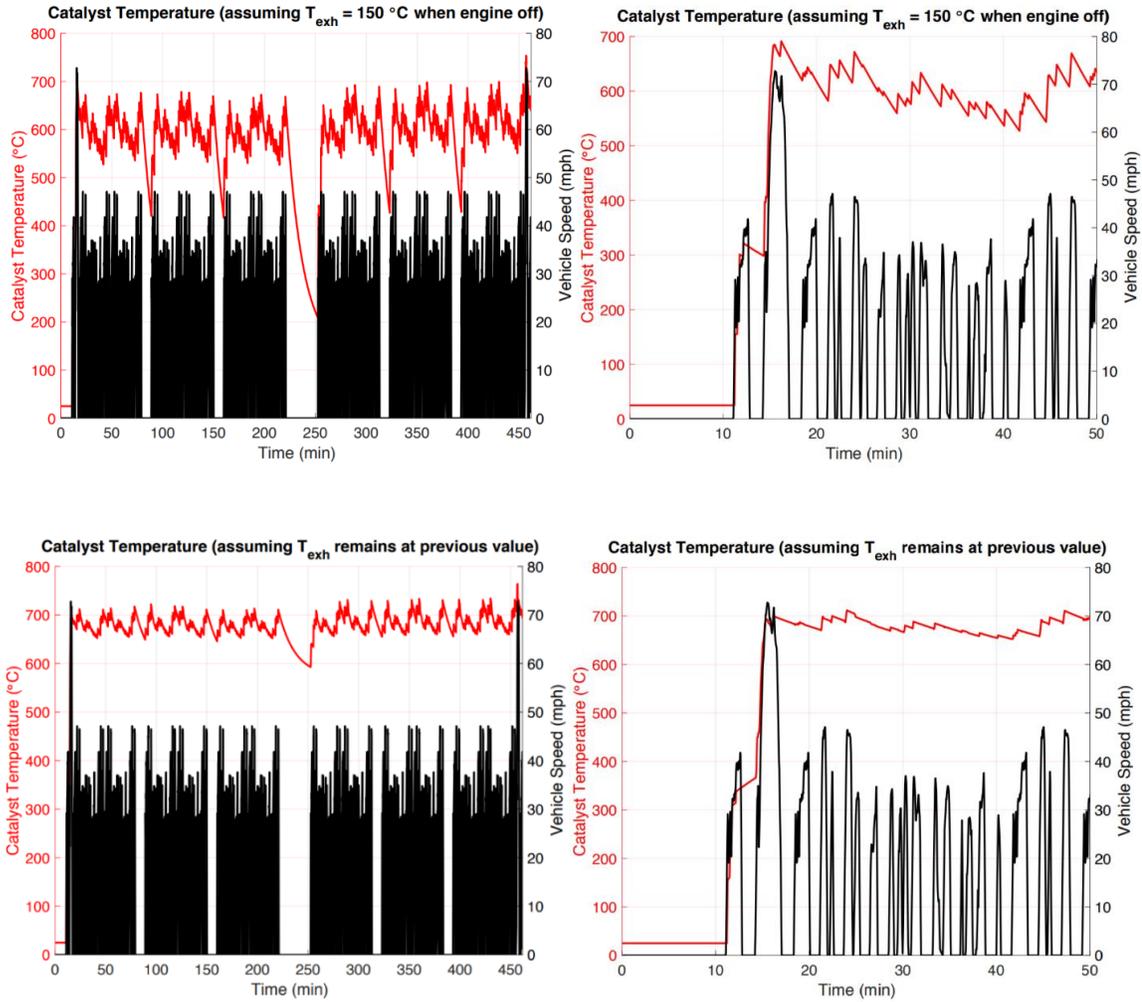


Figure 122: Study of the catalyst temperature dynamics over the drive cycle

However, the benefit of that is unused given the fact that the heat soak phenomenon is not modeled. That means catalyst temperature only goes downward, and the exhaust

temperature is a passive parameter in the controller operation, with the lower threshold temperature never realized in the simulation window, except for the duration of the mid-day break in the drive cycle.

From the simulation results of the aftertreatment dynamics indicated in Figure 122, it is clear that the two-way catalyst temperature does not fall below the light-off temperature assumed at the cut off value of 300 C. As a result, the emission penalty that comes with uncontrolled start-stop events is trivial. It is however a concern that the number of start-stops can adversely affect the durability of the genset hardware, as elaborated in Section 3.1.2. This is a consequence of the limitation of the A-ECMS controller in being inherently unable to penalize start-stop events with a penalty function. It has the ability to access neither the past trajectory of the drive cycle nor the future events while performing the instantaneous power split to follow the reference SOC profile. Hence, a rule-based controller would be necessary to have a meaningful authority over start-stop events.

#### **4.5.5 Advantage of Traffic Co-simulation over Deterministic Drive Cycles**

Fixed drive cycle-based simulation techniques are inadequate to fully represent the effects related to vehicle interaction with infrastructure and other vehicles and its non-causal treatment of driving profile information. Thus, a pressing need was felt to develop a simulation framework where the vehicle is subjected to a real-world environment and is required to interaction with surrounding traffic by virtue of imposed driver behavior variation. This work has developed a framework that aims to integrate the real world factors such as causality, vehicle infrastructure interaction and traffic behavior into a hybrid

powertrain simulator in an effort to capture the dynamics that are associated with real-world systems, and study the effect of factors external to the powertrain on the overall fuel economy, engine start-stop events and emissions.

In the proposed framework, the online co-simulation allows the adaptation of vehicle behavior during the simulation runtime. Hence, in a statistical sense, the methodology of the simulation that is described in this thesis helps to reliably capture the mean and standard deviation in on-road fuel economy, constrained only by the fidelity of the models used without the need for on-road tests.

#### **4.6 Summary of Results**

A co-simulation has been achieved with the integration of the selected powertrain model with a dynamic traffic environment by marrying MATLAB and SUMO in a real-time manner. The impacts of the driver and traffic behavior on the fuel economy, start-stop performance and emissions of the vehicle have been evaluated to test the controller robustness in an environment that is far more realistic than synthetic drive cycles.

## Chapter 5. Conclusion and Future Scope

The research presented in this thesis dealt with the formulation of a suitable design space exploration search scheme capable of leveraging realistic drive cycle data and a collection of powertrain component information to culminate in an optimal architecture for a Class 6 range-extended plugin hybrid electric delivery truck. An online implementable energy management strategy was designed and implemented to optimize the selected performance metrics of the vehicle over a specific driving mission. Finally, the robustness of the controller was tested with an integrated powertrain-cum-traffic-in-the-loop simulation. The entire exercise has led to a set of conclusions that aided in making an informed decision about the final implementation of the specified real-world product.

### 5.1 Summary of the Study

A *design for operation* notion that relies intensively on operational data collection and large-scale simulations is the cornerstone of this project. In the past two decades, a number of studies have addressed this design notion that uses optimization tools and simulation models to find optimal powertrain topology and component sizes for specific operation scenarios. In this work, this design for operation notion was revisited with a specific combination of optimization and control techniques that promises accurate results with relatively fast computational time. A Gaussian Process (GP) based statistical learning approach was used to refine the search space for the most accurate, optimal designs. Five

hybrid powertrain architectures, namely two-speed e-axle, three-speed automatic manual transmission with electric motor (AMT+EM), four-speed AMT+EM, direct drive and dual motor were explored, and a set of Pareto-optimal designs were found for a specific driving mission that represents the variations in a hypothetical operational scenario. The entire range of modeling and optimization processes were performed on the MATLAB-Simulink platform. A cross-architecture performance and cost comparison was performed, which showed that two-speed e-axle is the optimal architecture for the specific application.

Given the inability of fixed drive cycle-based simulation techniques to fully represent effects related to vehicle interaction with infrastructure and other vehicles and its non-causal approximation of driving profiles, a strong need was felt to develop a simulation framework where the vehicle drives in a virtual real-world condition and is subjected to interaction with surrounding vehicles and infrastructure. This work has developed a framework that aims to integrate the real-world factors such as causality, vehicle infrastructure interaction and traffic behavior into a hybrid powertrain simulator in an effort to capture the dynamics that are associated with real-world systems and study the effect of environmental factors external to the powertrain on the overall system efficiency and performance.

In the proposed framework, the online traffic-powertrain co-simulation allows the adaptation of vehicle behavior during simulation runtime. Hence in a statistical sense, the frame work of the simulation that is described in this thesis helps to reliably capture the

mean and variance in on-road fuel economy, constrained only by the fidelity of the models used without the need for on-road tests.

## **5.2 Future Scope**

Firstly, it is established that the Design Space Exploration exercise resulted in satisfactory results for the powertrain architecture by minimizing the computation cost to a great extent. Such a study can also be extended to parallel or power-split vehicle architectures in the same commercial truck weight class to generate schemes that are more universally applicable.

While the adaptive ECMS controller provided robust fuel economy performance for the vehicle by utilizing accurate a priori information, better results can be obtained by incorporating a receding horizon controller such as model predictive control (MPC) or MPC in conjunction with dynamic programming over short horizons. Such techniques have been successfully demonstrated in passenger vehicle by undertakings such as the ARPA-E NEXTCAR led by the Center for Automotive Research, OSU [13]. With such a versatile controller, multiple levels of velocity predictors can be employed to obtain varying degrees of look-ahead information, and predict the future velocity. Such an exercise would also facilitate the cost-benefit analysis of the addition of cumulative sensor suites in production vehicles. Furthermore, there is potential for more realistic intelligent driver models to be integration with the forward simulator for look-ahead control as well as traffic-in-the-loop simulation with SUMO. Additionally, controller development and validation for

applications similar to geo-fencing – constraining the powertrain operation mode based on location data – can also be performed using the SUMO framework.

Developing a higher fidelity aftertreatment model to be integrated with the simulator is another area of potential improvement. This model can be integrated with dynamic programming with added penalties for failing to adhere to temperature limits of the two-way catalyst. Results would offer the absolute best scenario of fuel economy and engine start-stop counts to serve as a universal benchmark for this architecture going forward.

Further improvement of fuel economy and start-stop performance can be achieved with advanced control of the regenerative strategy to curtail energy conversion losses between the genset and wheels by maximizing the battery throughput. This may necessitate the exploration of a rule-based controller to further improve fuel economy and start-stops on top of A-ECMS. Additionally, the SUMO co-simulation can be extended to study impacts of different levels of look-ahead information – speeds limits, stop signs, traffic density et al on the selected vehicle performance metrics.

Finally, the traffic-in-the-loop simulation can be scaled up to perform a higher number of driving scenarios to better quantify the trends in the impact of driver behavior on fuel economy and start-stops.

## Bibliography

1. U.S.-China Clean Energy Research Center - Truck Research Utilizing Collaborative Knowledge. Retrieved April 10, 2020, from <https://cerc-truck.anl.gov/>
2. IEA. (2019). Global EV Outlook 2019 (pp. 1–232).
3. Keegan, M. (2018, December 12). Shenzhen's silent revolution: world's first fully electric bus fleet quietens Chinese megacity. Retrieved April 10, 2020, from <https://www.theguardian.com/>
4. Poon, L. (2018, June 22). Inside China's Electric Bus Capital. Retrieved April 10, 2020, from <https://www.citylab.com/>
5. Vehicle technology cost curves. Retrieved April 10, 2020, from <https://theicct.org/vehicle-technology-cost-curves>
6. IEA. (2017). The Future of Trucks
7. IEA, "CO2 emissions from heavy-duty vehicles in the Sustainable Development Scenario, 2000-2030", IEA, Paris
8. Kalia, A. V., & Fabien, B. C., "On Implementing Optimal Energy Management for EREV Using Distance-Constrained Adaptive Real-Time DP. Electronics, 2020. doi: 10.3390/electronics9020228.

9. Minarcin, M., Rask, E., and Smith, M., "Challenges and Opportunities in Adoption of Hybrid Technologies in Medium and Heavy-Duty Applications," SAE Technical Paper 2011-01-2251, 2011, <https://doi.org/10.4271/2011-01-2251>
10. D. Tran, M. Vafaiepour, et al, "Thorough state-of-the-art analysis of electric and hybrid vehicle powertrains: Topologies and integrated energy management strategies", Renewable and Sustainable Energy Reviews, Volume 119, 2020, ISSN 1364-0321, <https://doi.org/10.1016/j.rser.2019.109596>.
11. DPM Toolbox. Retrieved April 12, 2020, from <https://idsc.ethz.ch/research-guzzella-onder/downloads.html>
12. Onori S., Serrao L., Rizzoni G. "Dynamic Programming. In: Hybrid Electric Vehicles". SpringerBriefs in Electrical and Computer Engineering. Springer, London
13. NEXTCAR. (2016, November 2). Retrieved April 12, 2020, from <https://arpa-e.energy.gov/?q=arpa-e-programs/nextcar>
14. Alzorgan, M., Carroll, J., Al-Masalmeh, E., and Mayyas, A., "Look-Ahead Information Based Optimization Strategy for Hybrid Electric Vehicles," SAE Technical Paper 2016-01-2226, 2016, <https://doi.org/10.4271/2016-01-2226>.
15. M. Ali, Ahmed and Söffker, D. (2018). "Towards Optimal Power Management of Hybrid Electric Vehicles in Real-Time: A Review on Methods, Challenges, and State-Of-The-Art Solutions". Energies. 11. 10.3390/en11030476.
16. Paganelli, G., Guerra, T. M., Delprat, S., Santin, J.-J., Delhom, M., and Combes, E. (2000). "Simulation and assessment of power control strategies for a parallel hybrid

- car”. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering. <https://doi.org/10.1243/0954407001527583>
17. Panday, A., & Bansal, H. O. (2014). “A Review of Optimal Energy Management Strategies for Hybrid Electric Vehicles”. International Journal of Vehicular Technology, 2014, 1–19. doi: 10.1155/2014/160510
  18. Serrao, L., Onori, S., and Rizzoni, G. (March 25, 2011). "A Comparative Analysis of Energy Management Strategies for Hybrid Electric Vehicles." ASME. J. Dyn. Sys., Meas., Control. May 2011; 133(3): 031012. <https://doi.org/10.1115/1.4003267>
  19. S. Delprat, J. Lauber, T. M. Guerra and J. Rimaux, "Control of a parallel hybrid powertrain: optimal control," in IEEE Transactions on Vehicular Technology, vol. 53, no. 3, pp. 872-881, May 2004.
  20. T. H. Pham, J. T. B. A. Kessels, P. P. J. van den Bosch, R. G. M. Huisman and R. M. P. A. Nevels, "On-line energy and Battery Thermal Management for hybrid electric heavy-duty truck," 2013 American Control Conference, Washington, DC, 2013, pp. 710-715.
  21. J. T. B. A. Kessels, M. W. T. Koot, P. P. J. van den Bosch and D. B. Kok, "Online Energy Management for Hybrid Electric Vehicles," in IEEE Transactions on Vehicular Technology, vol. 57, no. 6, pp. 3428-3440, Nov. 2008.
  22. H. Yu, M. Kuang and R. McGee, "Trip-Oriented Energy Management Control Strategy for Plug-In Hybrid Electric Vehicles," in IEEE Transactions on Control Systems Technology, vol. 22, no. 4, pp. 1323-1336, July 2014.

23. D. Ambühl, O. Sundström, et al, “Explicit optimal control policy and its practical application for hybrid electric powertrains”, *Control Engineering Practice*, Volume 18, Issue 12, 2010, Pages 1429-1439, ISSN 0967-0661.
24. Kim, N.W., Lee, D.H., Zheng, C. et al., “Realization of PMP-based control for hybrid electric vehicles in a backward-looking simulation”. *Int. Jour. Automotive. Tech.* 15, 625–635 (2014). <https://doi.org/10.1007/s12239-014-0065-z>
25. Onori, S., Serrao, L., & Rizzoni, G. (2016). “Hybrid electric vehicles: energy management strategies”. London: Springer.
26. C. Musardo, G. Rizzoni, et al, “A-ECMS: An Adaptive Algorithm for Hybrid Electric Vehicle Energy Management”, *European Journal of Control*, Volume 11, Issues 4–5, 2005, Pages 509-524, ISSN 0947-3580, <https://doi.org/10.3166/ejc.11.509-524>.
27. Pei, D., and Leamy, M. J. "Dynamic Programming-Informed Equivalent Cost Minimization Control Strategies for Hybrid-Electric Vehicles." *ASME. J. Dyn. Sys., Meas., Control.* September 2013; 135(5): 051013. <https://doi.org/10.1115/1.4024788>.
28. C. Musardo, B. Staccia, “Energy management strategies for hybrid electric vehicles”, Master’s thesis (Politecnico di Milano, 2003).
29. C. Musardo, B. Staccia, S. Bittanti, Y. Guezennec, L. Guzzella, G. Rizzoni, “An adaptive algorithm for hybrid electric vehicles energy management”, *Proceedings of the FISITA World Automotive Congress* (2004).
30. A. Sciarretta, M. Back, L. Guzzella, “Optimal Control of Parallel Hybrid Electric Vehicles”. *IEEE Trans. Control Syst. Technol.* 12(3), 352–363 (2004).

31. D. Ambühl, L. Guzzella, “Predictive reference signal generator for hybrid electric vehicles”. *IEEE Trans. Veh. Technol.* 58, 4730–4740 (2009).
32. R. Bartholomaeus, M. Klingner, M. Lehnert, “Prediction of power demand for hybrid vehicles operating in fixed-route service”, *IFAC Proceedings Volumes, Volume 41, Issue 2, 2008, Pages 5640-5645*, <https://doi.org/10.3182/20080706-5-KR-1001.00951>.
33. S. Jeon, S. Jo, Y. Park, J. Lee, “Multi-mode driving control of a parallel hybrid electric vehicle using driving pattern recognition”. *ASME J. Dyn. Syst. Meas. Control* 124, 141–149 (2002).
34. B. Gu, G. Rizzoni, “An adaptive algorithm for hybrid electric vehicle energy management based on driving pattern recognition”, *Proceedings of the 2006 ASME International Mechanical Engineering Congress and Exposition (2006)*
35. B. Gu, “Supervisory control strategy development for a hybrid electric vehicle”, *Master’s thesis (The Ohio State University, 2006)*.
36. J. Kessels, M. Koot, P. van den Bosch, D. Kok, “Online energy management for hybrid electric vehicles”. *IEEE Trans. Veh. Technol.* 57(6), 3428–3440 (2008).
37. A. Chasse, A. Sciarretta, J. Chauvin, “Online optimal control of a parallel hybrid with costate adaptation”, *Proceedings of the 6th IFAC Symposium Advances in Automotive Control (2010)*.
38. S. Onori, L. Serrao, G. Rizzoni, “Adaptive equivalent consumption minimization strategy for hybrid electric vehicles”, *Proceedings of the 2010 ASME Dynamic Systems and Control Conference (2010)*.
39. Camacho, E. F., & Bordons, C. (2007). “*Model Predictive Control*”. London: Springer.

40. Liang Li, Sixiong You, Chao Yang, Bingjie Yan, Jian Song, Zheng Chen, "Driving-behavior-aware stochastic model predictive control for plug-in hybrid electric buses", *Applied Energy*, Volume 162, 2016, <https://doi.org/10.1016/j.apenergy.2015.10.152>.
41. G. Ripaccioli, D. Bernardini, S. Di Cairano et al., "A stochastic model predictive control approach for series hybrid electric vehicle power management," *Proceedings of the 2010 American Control Conference*, Baltimore, MD, 2010, pp. 5844-5849.
42. J. Wang, Y. Huang, H. Xie , G. Tian, "Driving Pattern Prediction Model for Hybrid Electric Buses Based on Real-World Driving Data", KINTEX, Korea, May 3-6, 2015.
43. L. Fu, Ü. Özgüner, P. Tulpule, V. Marano, "Real-time energy management and sensitivity study for hybrid electric vehicles", *Proceedings of 2011 American Control Conference* (2011).
44. Jung, D., Ahmed, Q., Zhang, X., and Rizzoni, G., "Mission-based Design Space Exploration for Powertrain Electrification of Series Plugin Hybrid Electric Delivery Truck," *SAE Technical Paper 2018-01-1027*, 2018.
45. T. Zhao, Q. Ahmed and G. Rizzoni, "Influence of Battery Charging Current Limit on the Design of Range Extender Hybrid Electric Trucks," *2018 IEEE Vehicle Power and Propulsion Conference (VPPC)*, Chicago, IL, 2018, pp. 1-6.
46. Josephson, John & Chandrasekaran, B. & Carroll, Mark & Iyer, Naresh & Wasacz, Bryon & Rizzoni, Giorgio & Li, Qingyuan. (2000). "An Architecture for Exploring Large Design Spaces".

47. O. Hegazy and J. Van Mierlo, "Particle swarm optimization for optimal powertrain component sizing and design of fuel cell hybrid electric vehicles", 2010 12<sup>th</sup> International Conference on Optimization of Electrical and Electronic Equipment, pp. 601–609, May 2010.
48. B. Desai and S. S. Williamson, "Optimal design of a parallel hybrid electric vehicle using multi-objective genetic algorithms," in 2009 IEEE Vehicle Power and Propulsion Conference, pp. 871–876, Sep. 2009.
49. N. Murgovski, L. Johannesson, J. Sjberg, and B. Egardt, "Component sizing of a plug-in hybrid electric powertrain via convex optimization," *Mechatronics*, vol. 22, no. 1, pp. 106 – 120, 2012.
50. M. Pourabdollah, E. Silvas et al., "Optimal sizing of a series PHEV: Comparison between convex optimization and particle swarm optimization", 2015. 4th IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling 2015.
51. M. Pourabdollah, B. Egardt, N. Murgovski, and A. Grauers, "Convex optimization methods for powertrain sizing of electrified vehicles by using different levels of modeling details," *IEEE Transactions on Vehicular Technology*, March 2018.
52. E. Silvas, T. Hofman, N. Murgovski, L. F. P. Etman, and M. Steinbuch, "Review of optimization strategies for system-level design in hybrid electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 66, pp. 57–70, Jan 2017.

53. Deb, K. (2014). Multi-objective optimization. In Search methodologies (pp. 403-449). Springer US.
54. Rasmussen, C. and Williams, C., "Gaussian processes for machine learning," MIT press Cambridge, 2006, vol. 1.
55. Osborne, M. A., Garnett, R., & Roberts, S. J. (2009, January). "Gaussian Processes for Global Optimization". In 3rd international conference on learning and intelligent optimization (LION3) (pp. 1-15).
56. Iqbal Husain (2010). "Electric and Hybrid Vehicles: Design Fundamentals", 2nd Edition. CRC Press: Boca Raton, FL.
57. W. Gao and C. Mi, "Hybrid vehicle design using global optimization algorithms," vol. 1, pp. 57–70, 2007.
58. P. Research Fellow, N. Kaushal, C. Hendrickson et al., "Optimal plug-in hybrid electric vehicle design and allocation for minimum life cycle cost, petroleum consumption, and greenhouse gas emissions," Journal of Mechanical Design, vol. 132, 01 2010.
59. R. Patil, B. Adornato, and Z. Filipi, "Design optimization of a series plug-in hybrid electric vehicle for real-world driving conditions," apr 2010.
60. Bayrak, Y. Ren, and P. Papalambros, "Design of hybrid-electric vehicle architectures using auto-generation of feasible driving modes," in 15th International Conference on Advanced Vehicle Technologies; American Society of Mechanical Engineers, 2013.

61. Erik Hellström, Maria Ivarsson, Jan Åslund and Lars Nielsen, "Look-ahead control for heavy trucks to minimize trip time and fuel consumption", 2009, Control Engineering Practice, (17), 2, 245-254. <http://dx.doi.org/10.1016/j.conengprac.2008.07.005>
62. Engbroks, L., Knappe, et al., "Energetic Costs of ICE Starts in (P)HEV - Experimental Evaluation and Its Influence on Optimization Based Energy Management Strategies," SAE Technical Paper 2019-24-0203, 2019, doi:10.4271/2019-24-0203.
63. Wellmann, T., Govindswamy, K., Orzechowski, J., and Srinivasan, S., "Influence of Automatic Engine Stop/Start Systems on Vehicle NVH and Launch Performance," SAE Int. J. Engines 8(4):1938-1946, 2015, <https://doi.org/10.4271/2015-01-2183>.
64. Kim, Y. and Filipi, Z., "Simulation Study of a Series Hydraulic Hybrid Propulsion System for a Light Truck," SAE Technical Paper 2007-01-4151, 2007, <https://doi-org.proxy.lib.ohio-state.edu/10.4271/2007-01-4151>.
65. Buchwald, P., Christensen, H., Larsen, H. and Pedersen, P. S., "Improvement of City bus Fuel Economy Using a Hydraulic Hybrid Propulsion System-a Theoretical and Experimental Study", SAE Paper number 790305, Warrendale, PA, USA, 1979.
66. Wu, B., Lin, C.-C., Filipi, Z., Peng H., Assanis, D., "Optimal Power Management for a Hydraulic Hybrid Delivery Truck", Journal of Vehicle System Dynamics, Vol. 42, Nos. 1-2, 2004, pp. 23-40.
67. Filipi, Z., Louca, L., Daran, B. et al., "Combined optimization of design and power management of the hydraulic hybrid propulsion system for the 6 x 6 medium truck", International Journal of Heavy Vehicle Systems, Vol. 11, No 3-4, 2004, pp. 372-402.

68. B. Hegde, “Look-Ahead Energy Management Strategies for Hybrid Vehicles”, PhD Dissertation (The Ohio State University, 2018).
69. C. Sun, X. Hu, S. J. Moura, and F. Sun. “Velocity Predictors for Predictive Energy Management in Hybrid Electric Vehicles”, *IEEE Transactions on Control Systems Technology*, 23(3):1197–1204, May 2015.
70. Giulio Ripaccioli, Daniele Bernardini, Stefano Di Cairano et al., “A stochastic model predictive control approach for series 201 hybrid electric vehicle power management”, *American Control Conference (ACC)*, volume 2010, pages 5844–5849, 2010.
71. H. A. Borhan, C. Zhang, A. Vahidi, A. M. Phillips, M. L. Kuang, and S. Di Cairano, “Nonlinear Model Predictive Control for power-split Hybrid Electric Vehicles”, *49th IEEE Conference on Decision and Control (CDC)*, pages 4890– 4895, December 2010.
72. L. Johansson, M. Asbogard, and B. Egardt, “Assessing the potential of predictive control for hybrid vehicle powertrains using stochastic dynamic programming”, *Proceedings of 2005 IEEE Intelligent Transportation Systems*, 2005.
73. M. Bichi, G. Ripaccioli, S. Di Cairano, D. Bernardini, A. Bemporad, and I. V. Kolmanovsky, “Stochastic model predictive control with driver behavior learning for improved powertrain control”, *49th IEEE Conference on Decision and Control* 2010.
74. S. Kermani, S. Delprat, T. M. Guerra, R. Trigui, and B. Jeanneret, “Predictive energy management for hybrid vehicles”, *Control Engineering Practice*, 20(4):408– 420, 2012.
75. S. D. Cairano, D. Bernardini, A. Bemporad, and I. V. Kolmanovsky, “Stochastic MPC With Learning for Driver-Predictive Vehicle Control and its Application to HEV Energy Management”, *IEEE Transactions on Control Systems Technology*, 2014.

76. Guzzella, L., & Sciarretta, A. (2015). *Vehicle Propulsion Systems Introduction to Modeling and Optimization*. Berlin: Springer Berlin.
77. A. Sciarretta and L. Guzzella, "Control of hybrid electric vehicles", *IEEE Control Systems Magazine*, vol. 27, no. 2, pp. 60-70, April 2007.
78. Ambühl, Daniel & Guzzella, Lino, "Predictive Reference Signal Generator for Hybrid Electric Vehicles", *Vehicular Technology, IEEE Transactions*, 2009.
79. Asadi, Behrang & Vahidi, Ardan, "Predictive Use of Traffic Signal State for Fuel Saving", *IFAC Proceedings Volumes*. 42, 2009.
80. Behrang Asadi and Ardan Vahidi, "Predictive Cruise Control: Utilizing Upcoming Traffic Signal Information for Improving Fuel Economy and Reducing Trip Time", *IEEE Transactions on Control Systems Technology*, 2010.
81. C. Raubitschek, N. Schütze, E. Kozlov and B. Bäker, "Predictive driving strategies under urban conditions for reducing fuel consumption based on vehicle environment information," *2011 IEEE Forum on Integrated and Sustainable Transportation Systems*, Vienna, 2011, pp. 13-19.
82. Daliang Shen, Valerie Bensch, Steffen Miiller, "Model Predictive Energy Management for a Range Extender Hybrid Vehicle using Map Information", *IFAC-PapersOnLine*, 2015, ISSN 2405-8963, <https://doi.org/10.1016/j.ifacol.2015.10.038>.
83. van Keulen, T., Naus, G., de Jager, B., van de Molengraft, R., Steinbuch, M., and Aneke, E., "Predictive cruise control in hybrid electric vehicles", *World Electric Vehicle Journal*, 2009.

84. Tae Soo Kim, Chris Manzie and Rahul Sharma, “Model Predictive Control of Velocity and Torque Split in a Parallel Hybrid Vehicle”, Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics, October 2009.
85. 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, Vol. 59, No. 3, March 2010.
86. Kim, T. S., Manzie, C., & Sharma, R., “Optimal control of a parallel hybrid vehicle with a traffic preview”, Proceedings of the Institution of Mechanical Engineers, 228(7), pp. 719–733. <https://doi.org/10.1177/0954407013506567>.
87. T. S. Kim, C. Manzie, and R. Sharma, “Model predictive control of velocity and torque split in a parallel hybrid vehicle”, IEEE International Conference on Systems, Man and Cybernetics, 2009. SMC 2009, pages 2014–2019, October 2009.
88. R. Beck, F. Richert, A. Bollig, D. Abel, S. Saenger, K. Neil, T. Scholt, and K. E. Noreikat, “Model Predictive Control of a Parallel Hybrid Vehicle Drivetrain”, Proceedings of the 44th IEEE Conference on Decision and Control, pages 2670–2675, December 2005.
89. R. Beck, A. Bollig, and D. Abel, “Comparison of Two Real-Time Predictive Strategies for the Optimal Energy Management of a Hybrid Electric Vehicle”, Oil & Gas Science and Technology - Revue de l’IFP, 62(4):635–643, July 2007.
90. Lars Johannesson, Stefan Pettersson, and Bo Egardt, “Predictive energy management of a 4qt series-parallel hybrid electric bus”, Control Engineering Practice, December 2009.

91. Amir Rezaei and Jeffrey B. Burl, “Effects of Time Horizon on Model Predictive Control for Hybrid Electric Vehicles”, IFAC-PapersOnLine, 48(15):252–256, 2015.
92. M Debert, Golin Yhamailard, and Guzzella A. Ketfi-herifellicaud, “Predictive energy management for hybrid electric vehicles - Prediction horizon and battery capacity sensitivity”, IFAC Proceedings Volumes, 43(7):270–275, July 2010.
93. C. Zhang and A. Vahidi, “Route Preview in Energy Management of Plug-in Hybrid Vehicles”, IEEE Transactions on Control Systems Technology, 20(2):546– 553, 2012.
94. Jorg 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, 2015.
95. Yanjun Huang, Hong Wang, Amir Khajepour, Hongwen He, and Jie Ji, “Model predictive control power management strategies for HEVs: A review”, Journal of Power Sources, 341:91–106, February 2017.
96. A. V. Rajendran, “A Methodology for Development of Look Ahead Based Energy Management System Using Traffic In Loop Simulation”, Master’s Thesis (The Ohio State University, 2018).
97. GM Cadillac Super Cruise. Retrieved from <https://www.cadillac.com/world-of-cadillac/innovation/super-cruise> on April 13, 2020.
98. Sharer, P., Leydier, R. and Rousseau, Aymeric, “Impact of Drive Cycle Aggressiveness and Speed on HEVs Fuel Consumption Sensitivity”, 2007, 10.4271/2007-01-0281.

99. Xuchen Li, Qadeer Ahmed, Yousaf Rahman, “Improving the Electrified Powertrain Performance based on Retrospective Behaviour of Battery state of charge and engine fuel consumption”, IFAC-2018, <https://doi.org/10.1016/j.ifacol.2018.10.033>.
100. E. Lieberman and A. K. Rathi, “Traffic simulation,” in Traffic Flow Theory A State-of-the-Art Report, ch. 10, pp. 10–1–10–25, Committee on Traffic Flow Theory and Characteristics, 2001. Accessed on: 30-Aug-2017.
101. A. Bajpai and T. V. Mathew, “Development of an interface between signal controller and traffic simulator,” in Conference of Transportation Research Group of India (CTRG), (Bangalore, India), 2011. Accessed on: 30-Aug-2017.
102. Committee on Traffic Flow Theory and Characteristics. (2001). Traffic Flow Theory A State-of-the-Art Report (pp. 1–386).
103. Ashutosh Bajpai Tom V Mathew, “Development of an Interface between Signal Controller and Traffic Simulator”, CTRG 2011.
104. Arnaud Doniec, René Mandiau, Sylvain Piechowiak, Stéphane Espié, “A behavioral multi-agent model for road traffic simulation”, Engineering Applications of Artificial Intelligence, 2008, <https://doi.org/10.1016/j.engappai.2008.04.002>.
105. Joaquín Maroto, Eduardo Delso, Jesús Félez, and Jose Ma. Cabanellas, “Real-Time Traffic Simulation with a Microscopic Model”, IEEE Transactions on Intelligent Transportation Systems, Vol. 7, No. 4, December 2006.
106. Kesting, Arne & Treiber, Martin & Helbing, Dirk, “Enhanced Intelligent Driver Model to Access the Impact of Driving Strategies on Traffic Capacity”, Philosophical transactions. Mathematical, physical, and engineering sciences, 2010.

## Appendix A. Further Reading

1. Further reading on the CERC Truck consortium: <https://cerc-truck.anl.gov/>
2. More information on the SUMO traffic simulator:  
<https://sumo.dlr.de/docs/index.html>
3. Downloads from the ETH Zurich Dynamic Programming code package:  
<https://idsc.ethz.ch/research-guzzella-onder/downloads.html>
4. Open Street Maps for traffic environment setup: <https://www.openstreetmap.org/>
5. International Energy Agency Global EV Outlook Report 2019:  
<https://www.iea.org/reports/global-ev-outlook-2019>
6. International Energy Agency: The Future of Trucks, Technology Report 2017  
<https://www.iea.org/reports/the-future-of-trucks>

## **Appendix B. Contact Information**

1. Author email: [vijaysankar170@gmail.com](mailto:vijaysankar170@gmail.com)
2. Other work by the author: <https://blackswanmoments.wordpress.com/>
3. Center for Automotive Research, OSU: <https://car.osu.edu/>
4. CERC Truck consortium: <https://cerc-truck.anl.gov/about-cerc-truck/united-states-team/>