Powertrain and Vehicle Longitudinal Motion Control for Personalized Eco-driving of P0+P4 Mild Hybrid Electric Vehicles

by

Yu He

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mechanical Sciences and Engineering)
in the University of Michigan-Dearborn
2022

Doctoral Committee:

Assistant Professor Youngki Kim, Chair
Assistant Professor Zhen Hu
Professor Dewey Dohoy Jung
Assistant Professor Doohyun Kim
Assistant Research Scientist Kyoung Hyun Kwak

Yu He
heyuz@umich.edu
ORCID iD: 0000-0003-4293-0049

ACKNOWLEDGMENTS

First and foremost, I would like to express my most significant appreciation to my academic advisor, Professor Youngki Kim, for his unwavering support and guidance. His extensive knowledge and valuable advice not just inspired my academic study throughout my Ph.D. career but also solidified my confidence in pushing forward. Moreover, his attitude towards colleagues, students, and challenging problems proves that he is an excellent advisor, not just in the academic area. I very much appreciate Prof. Dewey DoHoy Jung and Prof. Oleg Zikanov. They introduce Prof. Youngki Kim to me at the beginning of my Ph.D. Journey.

I am thankful to the committee members, Prof. Zhen Hu, Prof. Dewey DoHoy Jung, Prof. DooHyun Kim and Dr. Kyoung Hyun Kwak, for taking their time and interest in evaluating my work and providing constructive feedback. I believe that their insightful comments have led this dissertation to be more thorough and complete. I would like to acknowledge the assistance from Dr. Kyoung Hyun Kwak. He constantly provides me with creative ideas throughout my Ph.D. journey and helps me improve the structure of my papers.

Many thanks to the financial support by the Hyundai-Kia America Technical Center, Inc., also known as HATCI. I also wish to thank Brian Link and Dr. Jason Hoon Lee who offered me a chance to work in HATCI during the Summer of 2022. This opportunity allows me to further improve Chapter 5's work within a human-in-the-loop environment. Thank should also go to all collaborators in the CVD team, including Heeseong Kim, Justin Holmer, Yueming (Max) Chen, and John Harbor. Much help from them has made the exciting experiment in Chapter 5 go smoothly. Special thanks to Shihong Fan, my friend and colleague during the internship, for setting up the HIL environment and repetitively test-driving my algorithms.

I am also grateful to all my roommates and friends who accompany my side during the COVID-19 pandemic. Their encouragement and mutual motivation make the journey much easier than it should be.

Last but not least, I want to thank my parents, Zhizhou He and Baohong Tai, for their support and faith in me. Without their unconditional and perennial support, I would not have completed this journey.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF TABLES

TABLE

LIST OF ACRONYMS

ACC Adaptive Cruise Control

ADAS Advanced Driver Assistance Systems

A-ECMS Adaptive Equivalent Consumption Minimization Strategy

APS Acceleration Pedal Position

AWD All-Wheel-Drive

BEV Battery Electric Vehicle

BSD Blind Spot Detection

BSFC Brake Specific Fuel Consumption

BSZ Blind Spot Zone

CA Constant Acceleration

CAFE Corporate Average Fuel Economy

CS Constraints Satisfaction

DCT Dual-Clutch Transmission

DDPG Deep Deterministic Policy Gradient

DEco-ACC Defensive Ecological Adaptive Cruise Control

DLO Daylight Openings

DP Dynamic Programming

DQN Deep Q-Learning

DRD-PB Desired Distance Personalized Braking

DRL Deep Reinforcement Learning

Eco-ACC Ecological Adaptive Cruise Control

ECU Electronic Control Unit

EMS Energy Management Strategy

FE Fuel Economy

FWD Front-Wheel-Drive

HER Hindsight Experience Replay

HEV Hybrid Electric Vehicle

HIL Human-in-the-Loop

ICE Internal Combustion Engine

IDM Intelligent Driver Model

LHS Latin Hypercube Sampling

LHV Lower Heating Value

MHEV Mild Hybrid Electric Vehicle

MPC Model Predictive Control

MPD Markov Decision Process

NGSIM Next Generation Simulation

NMPC Nonlinear Model Predictive Control

NV Neighboring Vehicle

OPD One-Pedal-Driving

PE Prioritized Exploration

PEER Prioritized Exploration and Experience Replay

PER Prioritized Experience Replay

PI Proportional Integration

PMP Pontryagin's maximum principle

POPD Personalized One-Pedal-Driving

PSO Particle Swarm Optimization

RWD Rear-Wheel-Drive

SAC Soft Actor-Critic

SOC State-of-Charge

SUV Sports Utility Vehicle

TD Temporal Difference

TD3 Twin-Delayed Deep Deterministic Policy Gradient

TPD Two-Pedal-Driving

UNECE United Nations Economic Commission for Europe

V2I Vehicle-to-Infrastructure

V2V Vehicle-to-Vehicle

WOT Wide-Open Throttle

ABSTRACT

Due to the increasing trend of greenhouse gas emissions, the United States Environmental Protection Agency (EPA) has started to publish strict regulations regarding emissions for different types of vehicles. Battery electric vehicles (BEVs) have drawn much attention in recent years because they potentially eliminate all tailpipe emissions. However, due to charging speed and capacity limitations on battery, currently, EV users are facing the problem of range anxiety and the lack of charging stations. Hybrid electric vehicles (HEVs), which possess the advantages of both conventional vehicles and BEVs, appear to be a viable solution to cope with such strict emission regulations while mitigating range anxiety. Among all types of hybrid electric powertrain systems, a P0+P4 system possesses distinct advantages: two electric motors located on the front and rear axles allow brake energy to be recovered from both axles. Moreover, the dual motor configuration enables the driver to switch among front-drive, rear-drive and all-wheel-drive modes. Particularly, a 48V P0+P4 HEV requires less expensive wiring and electric shock protection and hence it is considered to be the most cost-effective HEV for reducing GHG emissions.

This dissertation focus on improving the energy efficiency, ride comfort, and safety of a 48V P0+P4 MHEV. To achieve these goals, this dissertation proposes a hierarchical control design among domains of power-split and vehicle longitudinal motion of 48V P0+P4 MHEV. In the domain of power-split, two real-time implementable controllers are proposed: (1) the optimization-based controller and (2) the learning-based controller. In the optimization-based control design, the approximated adaptive equivalent consumption minimization strategy (AA-ECMS) with a suboptimal braking distribution derived from dynamic programming (DP) analysis is proposed to capture the global optimal operation trends of the P0 motor operation, front/rear tire force distribution. In the learning based control design, twin delayed deep deterministic policy gradient with prioritized exploration and experience replay (TD3+PEER), a novel prioritized exploration approach, is proposed to encourage the deep reinforcement learning (DRL) agent to explore states with complex dynamics. Both proposed power-split controllers achieve better fuel economy during the test trips compared to state-of-art rule-based and learning-based controllers.

In vehicle longitudinal motion control design, two controllers have been developed using model predictive control (MPC): (1) the defensive ecological adaptive cruise control (DEco-ACC) and (2) the personalized one-pedal-driving (POPD). The DEco-ACC is a novel car-following algorithm

that balances fuel economy, ride comfort, and avoidance of blind spots from neighboring vehicles. In DEco-ACC, a novel continuous and differentiable penalty function is proposed to describe the projection of several neighboring vehicles' blind spots to the ego vehicle's traffic lane. The proposed MPC-based controller considers this blind spot penalty function as a soft constraint within its prediction horizon; and is able to make its own decision to either yield, pass, or stay within the blind spots based on the MPC's cost function and the traffic scenario. The simulation results show that with two neighboring vehicles present simultaneously, the defensive ecological adaptive cruise control (DEco-ACC) reduces 29.5% of dwelling time in the blind spot and only scarifies 0.4% of fuel consumption. The POPD is a novel personalized one-pedal driving method that can learn the individual driver's preference during everyday driving. In POPD, two types of MPC constraints that represent distinct driver's behavior are identified by analyzing 450 real-world drivers' data. And then, the POPD algorithm is validated in both the simulation environment and the human-in-the-loop (HIL) traffic simulator. The experiment shows that the brake pedal usage is reduced from 31.3% to 5.25% for human driver A and from 16.7 to 3.56% for human driver B.

In summary, a hierarchical control design approach between power-split and longitudinal motion improves the energy efficiency, ride comfort, and safety of a P0+P4 MHEV. Two power-split algorithms (optimization-based and learning-based) improve energy efficiency. The DEco-ACC at automated driving scenario ensures safety while neighboring vehicles exist without sacrificing energy efficiency and comfort. The POPD in human driving scenarios enhances comfort by generating desired braking profiles for a target driver.

CHAPTER 1

Introduction

1.1 Background of a 48V P0+P4 System

The United States Energy Information Administration (EIA) estimates that if the current policy and technology trend continues, energy consumption and carbon dioxide (CO_2) emissions will increase from 2020 through 2050 as a result of economic growth and population growth [1]. According to

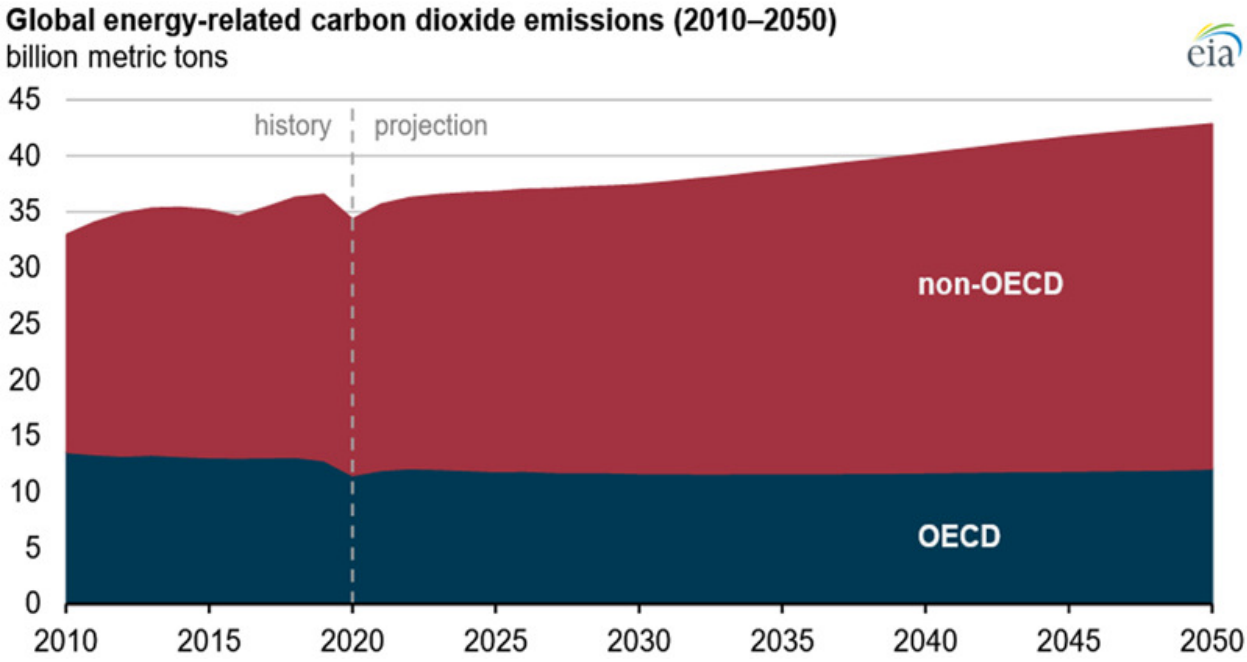

Figure 1.1: CO_2 emission from the year 2010 to 2050. (Projection) [1]

EIA's International Energy Outlook, the projection of CO_2 emission in 2050 will be 42.5 billion metric tons worldwide, which is 21% higher than it was in 2021. In addition, the Annual Energy Outlook 2022 (AEO2022) has reported that transportation-related CO_2 emission has dominated the total CO_2 in the U.S. Area since 2016 due to the increasing demand for transportation. Although

coal consumption has declined over the projection period, the increasing CO_2 from natural gas and petroleum still intensifies the greenhouse effect.

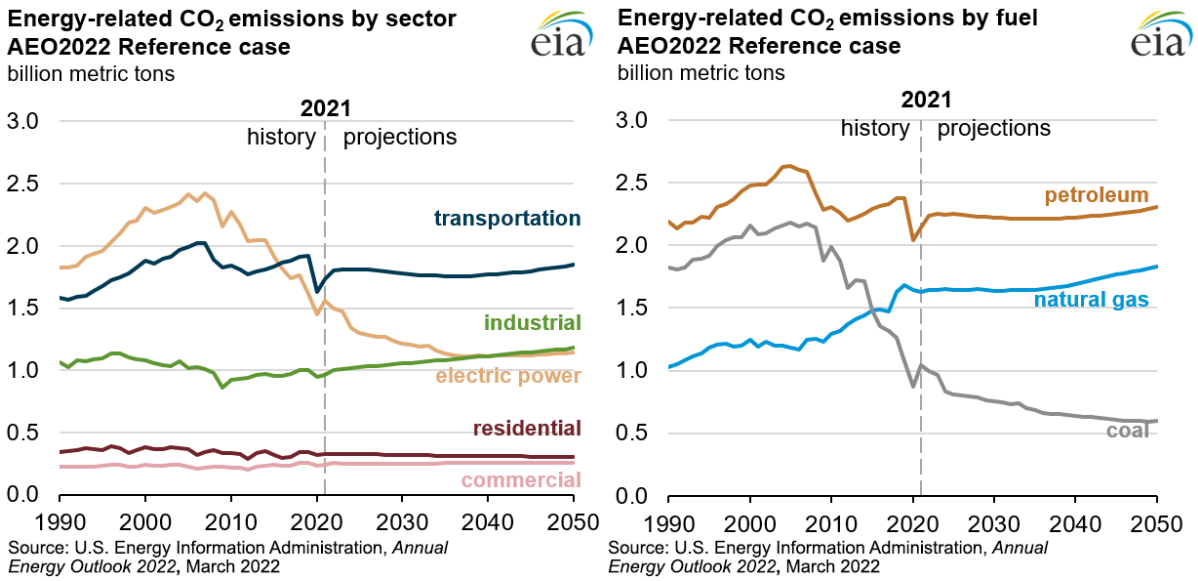

Figure 1.2: Projection of CO_2 emission in the U.S. area: by sector and by fuel. [1]

In 2017, the United States Environmental Protection Agency (EPA) published a strict regulation regarding greenhouse gas emissions and corporate average fuel economy (CAFE) standards for 2017 and later model year light-duty vehicles [4]. In December 2021, EPA revised the greenhouse gas emission standard again [2] for passenger cars and light trucks for the model year 2023 to 2026. The updated standard requires that the CO2 emitted per mile of each type of vehicle decreases at least 5% each year from 2023 to 2026, and the total changes of emission per mile of each type of vehicle are over 29% from 2023 to 2026.

Model year	Cars CO_2 (g/mile)	Light trucks CO_2 (g/mile)	Fleet CO_2 (g/mile)
2022 (SAFE reference)	181	261	224
2023	166	234	202
2024	158	222	192
2025	149	207	179
2026 and later	132	187	161
Total change 2022–2026	−49	−74	−63

Figure 1.3: Target CO_2 emission of each type of vehicle from 2022 to 2026. [2]

Due to the abovementioned regulations, vehicle electrification is becoming necessary as compliance with the strict CAFE regulations when using standalone internal combustion (IC) engines

becomes more difficult. Battery electric vehicles (BEVs) eliminate all tailpipe emissions but face major challenges on the battery front, such as charging time and range anxiety in long hauls [5]. Thus, many studies have explored hybrid electric vehicles (HEVs) from various perspectives, such as the system architecture, the energy management strategy, and the potential fuel economy (FE) gain. Compared to a conventional vehicle with a standalone internal combustion engine (ICE), it is known that a well-designed HEV with a decent control strategy can bring a significant fuel-saving benefit [6].

Among various HEVs, 48 V mild hybrid electric vehicles (MHEVs) have been drawing attention because of their potential to enhance the advantages of hybridization without compromising electric power and performance. The increased voltage system from the conventional 12 V still avoids the additional cost of the expensive wiring and electric shock protection mandated for a higher-voltage arrangement [7, 8]. Moreover, it offers a platform for more capable electric machines that may empower an extended stop/start function on top of boosted torque assist and regeneration [5].

To stretch such benefits of the 48 V system, hybrid configurations with more than one electric machine have also started to be investigated actively in recent years [9–11]. Most of the 48 V MHEVs on the market are under P0 architectures, which minimizes the necessity of powertrain design modification compared to the P1/P2/P3 hybrid variations. Although a P0 architecture may be the most attainable configuration in terms of cost [10], it inherits the limitations of a belt-driven system and often excludes the usage of pure electric driving mode. On the other hand, to cope with the recent market where the sales share of sport utility vehicles (SUVs) has substantially increased, the additional P4 hybrid architecture is being spotlighted. A bigger electric motor integrated on the rear axle of a vehicle enables electric all-wheel-drive (eAWD) capability as well as much more aggressive regenerative braking.

To promote the utilization of the hybrid system and eAWD capability even further, a P0+P4 48 V MHEV system with a dual-motor configuration, such as a less-expensive P0 motor combined with a P4 module, can be considered [11]. The P0+P4 system has the following advantages: (i) regenerative braking at both axles allows for maximized energy recuperation; and (ii) additional P4 motor allows for eAWD capability that caters to the mass-market demands. Unlike the other hybrid powertrain systems with a single motor, the P0+P4 MHEV has power sources on both the front and the rear axles: an engine and a P0 motor on the front axle and a P4 motor on the rear axle. This inherently allows the P0+P4 MHEV to switch among front-wheel-drive (FWD), rear-wheel-drive (RWD), and eAWD during vehicle operation.

This dissertation aims at improving the energy efficiency, safety and comfort of 48V P0+P4 MHEV. To address multiple goals at the same time, control optimization should be done in both vehicle's power-split and vehicle longitudinal motion perspectives. A prior art [12] has proved that

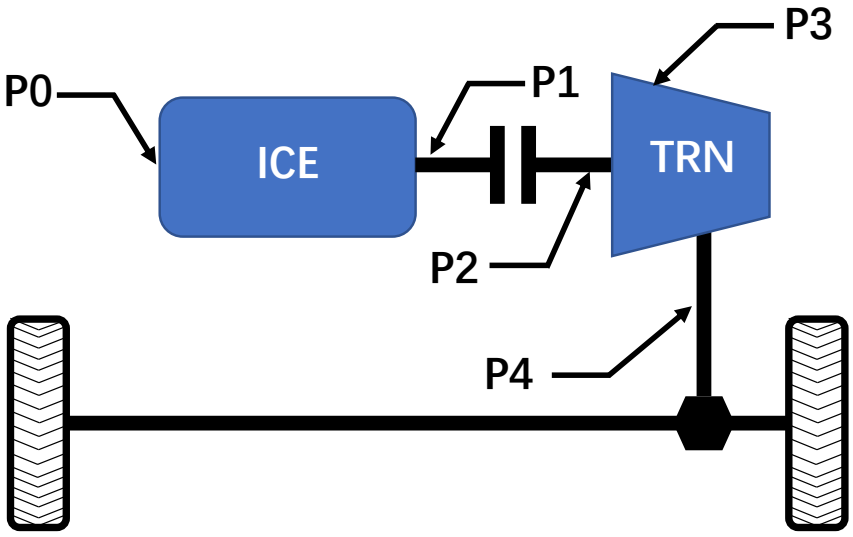

Figure 1.4: MHEV powertrain architecture: electric machine locations. [3]

co-optimization between vehicle velocity and hybrid powertrain components introduces additional control difficulties, which makes the co-optimization method hard to be real-time implementable. Furthermore, the results from [12] show that the co-optimization approach only achieves negligible improvements in energy efficiency when a certain level of passenger comfort is required. Therefore, this dissertation develops a hierarchical control approach to optimizing the vehicle's power-split and longitudinal vehicle motion in a sequential manner.

Optimization-based and learning-based strategies for the considered unique P0+P4 HEV system are proposed for the vehicle's power-split to seek energy-optimal operations. For velocity control, this dissertation further exploits the potential of Advanced Driver Assistance Systems (ADAS) and vehicle connectivity technologies, proposing defensive ecological adaptive cruise control (DEco-ACC) and personalized one-pedal-driving (POPD). The DEco-ACC further improves the existing Eco-ACC algorithms, allowing the ego vehicle to avoid staying in neighboring vehicles' blind spot zone during car-following scenarios. The POPD is able to perform personalized braking by learning the current driver's preferred driving style. The DEco-ACC is mainly designed for highway operation, and the POPD is mainly developed for urban/highway mixed trips. A schematic of the proposed hierarchical control approach is shown in Fig. 1.5.

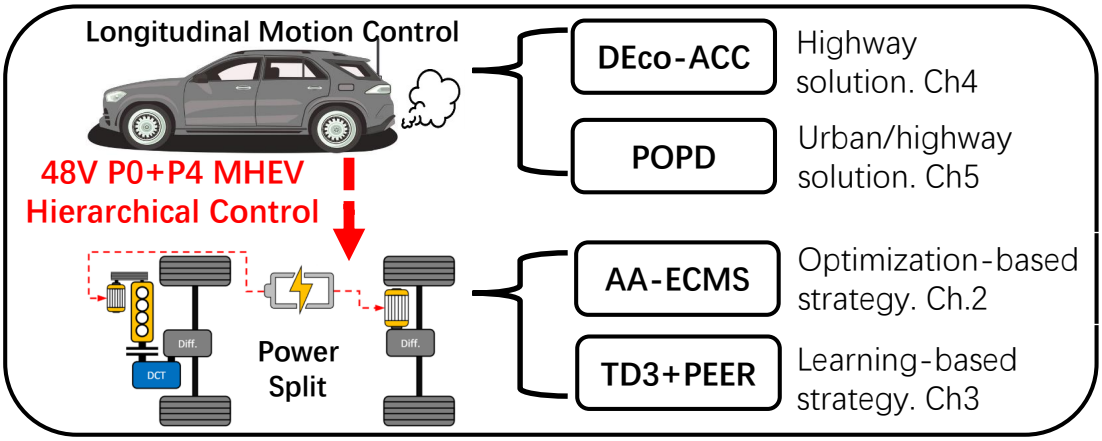

Figure 1.5: Dissertation outline.

1.2 Energy Management Strategies with Realistic Operational Constraints

Unlike conventional IC engine vehicles or single-motor EVs that only possess one power source, the instantaneously demanding power/torque of P0+P4 MHEVs can be satisfied with infinite combinations among different power sources. Influenced by architecture limitations and engine/motor characteristics, some combinations yield less overall fuel consumption than others. Currently, the study of a real-time optimal torque split strategy, which captures the most efficient combination among power sources, is an active area.

1.2.1 Optimization-based Approaches

Several studies in the literature have reported fuel economy improvement by the P0+P4 hybrid powertrain system [11, 13, 14]. The authors in [14] investigated the impact on the fuel economy of P0+P4 HEV architecture and design with connected and automated technology. However, a rule-based torque split control was considered, and there is room for further fuel economy improvement through optimal torque split. In [13], DP was used in control optimization for FE improvement of a P0+P4 HEV; however, the main focus was on the implementation of an efficient DP algorithm to assess the variations in optimal controls trajectories rather than on the development of a real-time-implementable control strategy. In [11], the authors presented the performance of fuel consumption reduction through P0+P4 hybridization with a realistic rule-based control strategy in comparison with an optimal control strategy. However, the control problem formulation and the approach to vehicle modeling were not provided in detail. Notably, most studies about P0+P4

hybrid powertrain systems in the literature have not considered important dynamics, including the longitudinal load transfer, the nonlinear tire effects, and realistic constraints on the regeneration of a P4 motor by braking force distribution. Moreover, detailed analysis and control design for effective real-time torque-split for P0+P4 MHEVs has not yet been rigorously studied.

Despite the P0+P4 MHEVs' many advantages, the overall energy performance heavily relies on the energy management strategy (EMS) that coordinates between the engine and motors. Previously, rule-based methods were widely applied to the torque split of HEVs for its simple structure and real-time implementation [15]. However, the preset rule can only achieve a limited level of optimality [16]. Moreover, the performance is also heavily influenced by human calibration, and driving conditions [17, 18]. The authors in [14] have reported that a rule-based control method calibrated by human and particle swarm optimization achieves drastically different powertrain operating behaviors.

An optimization-based method can be considered as a solution to eliminating human influence and further increasing optimality. In [19], a cost function of equivalence fuel that consists of instantaneous fuel rate and battery power is designed. By minimizing the equivalence fuel at every step of control with model predictive control (MPC), the real-time energy consumption performance can be further increased. However, the method of MPC requires a control-oriented model known ahead for prediction, and the problem complexity also influences the computation speed. When it comes to P0+P4 MHEVs, the additional control variable brings additional computation expenses, which makes the implementation of the MPC-based method a challenging problem in real-time. To the author's best knowledge, there is no optimization-based power-split strategy available for 48 V P0+P4 MHEV in the existing literature. Hence, this dissertation will fill the research gap by developing an optimization-based power-split strategy for the enhance performance of a 48 V P0+P4 MHEV.

1.2.2 Learning-based Approaches

Recent developments in the energy management of hybrid electric vehicles (HEVs) have shifted their focus to learning-based control algorithms [20–22]. Benefiting from advancements in neural network techniques, reinforcement learning (RL) has shown its astonishing adaptability to challenging control problems in many types of tasks [23, 24]. With RL, an agent can be developed to learn a control policy by interacting with a vehicle or with a vehicle model in a virtual environment.

The research works in [25] and [26] show that deep Q-network (DQN) learning-based torque split controllers can achieve a similar global optimality level to an optimization-based controller for hybrid electric tracked vehicles. In [27], Q-learning is combined with a recurrent neural network. The authors claim that from an energy-reduction perspective their proposed strategy outperforms

a rule-based strategy by at least 29% for a parallel HEV. It is worth mentioning that Q-learning was initially designed for problems with a finite number of discretized actions, such as turning left or turning right [23]. However, HEV power management problems are usually formulated with continuous actions, and hence the performance of Q-learning methods is inevitably limited by the level of discretization.

Many studies have found an "actor-critic" structure to be a potential solution to address continuous action space problems. In an "actor-critic" structure, the actor network is responsible for generating the control action as its output based on the defined state as input. The output of a neural network is inherently continuous, eliminating the error caused by discrete action. The critic network predicts the Q value, which estimates the long-term reward of each transition that the actor performed. The training process is still done through backpropagation with the temporal difference error (TD error) between the current critic prediction and a better TD target prediction for the same transition. The deep deterministic policy gradient (DDPG) is a state-of-the-art technique of actor-critic structure [28]. By converting Q-learning to DDPG, the studies in [29–31] observed different extents of fuel economy improvement. However, as a deterministic policy, the DDPG considers the optimal action at every step, which results in the overestimation of certain actions. This overestimation can easily distract the control decision with even random noise. Therefore, to prevent DDPG from exploiting certain regions in the action space, two variants of DDPG were proposed almost simultaneously: the soft actor-critic (SAC) [32] and the twin delayed DDPG (TD3) [33]. The SAC introduces an entropy term into the cost function of an actor network, which encourages the action selected to be as sparse as possible compared with the previous decision. Thanks to the sparse action, the training time in SAC is significantly reduced compared with DDPG and DQN. For example, in [34], the authors adopted SAC for power management of a hybrid electric bus with battery thermal and health constraints and have shown that their method achieves a 96.3% training time reduction compared with DQN. On the other hand, TD3 resolved overestimation issues by making the following three modifications to the DDPG [33]:

- Introduce additional critic network, only use the lowest Q prediction to generate TD-error.
- Delay the actor network update.
- Noise is also included in the target actor network.

The first bullet resolves the overestimation of certain transitions. The second and third bullet ensures the stability of the training process. In [35], TD3 was adopted for power management of a single-motor HEV, showing that 2% further fuel-consumption reduction could be achieved compared with DDPG. The authors in [36] showed that SAC and TD3 can outperform each other on different tasks, and both perform better than DDPG in most scenarios.

Aside from the algorithm architecture, the method called "experience replay" plays an essential role in improving the convergence speed and convergence performance of the RL training process. In [36], the authors showed that the same TD3 algorithm with hindsight experience replay (HER) or prioritized experience replay (PER) can improve the fuel economy by 3.2% and 2.1%, respectively, compared with random experience replay. Although both HER and PER can enhance the utilization of existing samples in the experience buffer, they do not encourage agents to explore complex dynamics regions in the state space. Without sufficient experience, the critic may not be able to predict the correct value in those regions. In addition, the agent may miss a potentially better solution from those regions. It is found that the current literature lacks a novel prioritized exploration technique that can encourage the agent to actively explore states in which Q values are sensitive to action selection.

1.3 Optimizing Longitudinal Motion in a Car-following Scenario

On the other hand, price dropping of cameras/sensors and stronger VCUs in electrified vehicles allow researchers to develop more advanced driver assistant systems (ADAS). ADAS technologies can enhance vehicles' driving safety and reduce driving efforts through features that either change vehicle's velocity or provide warnings during driving. Such features include emergency brake, front collision warning, blind-spot detection, and adaptive cruise control. In recent years, vehicle automation technologies, together with vehicle connectivity, have allowed the achievement of the full potential benefits of automated vehicles via safe, comfortable, and efficient driving, so-called eco-driving [37]. Under the concepts of eco-driving, two branches of study have drawn much attention and shown clear benefits to fuel economy, safety, and ride comfort to a 48 V P0+P4 MHEV: adaptive cruise control (ACC) and braking optimization in deceleration events.

1.3.1 Adaptive Cruise Control

Recently, much effort has been invested in the area of adaptive cruise control (ACC), one of the key features of ADAS for controlling longitudinal vehicle dynamics for highway car-following scenarios. In particular, ecological adaptive cruise control (Eco-ACC) considers the improvement of energy-efficiency as another critical objective in addition to maintaining a safe distance from a preceding vehicle or a time gap (or time to collision) [38, 39].

Since the energy efficiency of the ego vehicle is significantly influenced by its preceding vehicle, most Eco-ACC algorithms use information about the preceding vehicle. Furthermore, various forecasting algorithms provide the future behavior of the preceding vehicle with a certain

accuracy by using data from vehicle connectivity such as vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) [40] [41]. This availability of future information and the necessity of handling safety constraints and multi-objective cost functions make model predictive control (MPC) one of the notable trends toward eco-driving. For instance, the authors in [42] propose an adaptive cruise controller that functions by setting control decisions as multi-stage MPC constraints. This controller can handle both cruise control and adaptive cruise control scenarios. The work in [43] presents an Eco-ACC system that takes advantage of radar and traffic light-to-vehicle communications to predict the future trajectory of the preceding vehicle, leading to a 17% improvement compared to a traditional ACC. In [39], a nonlinear MPC-based ACC strategy is proposed for energy-optimal operation of electric vehicles. Safety and comfort requirements are implemented as state and input constraints and strictly enforced. In [44], an MPC-based ACC controller using a control barrier function is proposed for improving ride comfort and safety for an autonomous vehicle.

Although the aforementioned Eco-ACC algorithms have demonstrated their safety and energy efficiency performance, maintaining a safe/comfortable distance from a preceding vehicle is not the only factor that drivers consider during driving. Vehicles traveling in adjacent lanes also influence the driving strategy of the ego vehicle. For example, a neighboring car cannot be observed by the ego vehicle's driver if this vehicle is located in the blind spot zone of the ego vehicle. The blind spot zone (BSZ) is the region where an object is completely invisible to the driver without sufficiently tilting his or her head. Making a lane shift without checking for the existence of another car in the blind spot can be dangerous. Thus, to prevent collisions during lane changes, blind spot detection (BSD) and lane-change alert systems have been developed and equipped in modern vehicles [45]. These systems monitor the unnoticed vehicles in BSZs around the vicinity of the ego vehicle and warn the driver.

Even though most modern vehicles are equipped with a lane departure warning system or a BSD system, many do not have such advanced driver-assistant systems, particularly older vehicles. Moreover, the BSD system does not provide any information on whether the ego vehicle is located inside the BSZs of neighboring vehicles, which may not have BSD systems. Once the ego vehicle enters the blind spot of the neighboring car, there exists a high risk that the neighboring car does not have a BSD system and will make a lane shift into the ego vehicle. An experienced driver usually tries to avoid entering the blind spot or passes the blind spot of the neighboring vehicle at a faster speed to minimize the risk. For instance, the authors in [46] proposed a collision risk assessment algorithm based on probabilistic motion prediction of surrounding vehicles. However, the algorithm does not consider interactions between the ego vehicle and the surrounding vehicles. In summary, the current literature lacks an adaptive cruise control algorithm that can actively avoid the blind spots of neighboring vehicles for improved vehicle safety.

1.3.2 Braking Optimization in Deceleration Events

Under ADAS concepts, many researchers also focus on optimal regenerative braking for electrified vehicles, aiming to maximize energy regeneration during the braking process [47–49]. However, human factors such as ride comfort and sense of control have not been sufficiently considered. For example, a large braking force is needed to achieve the highest regenerative braking performance, which often results in discomfort to the human driver. In [47], the average deceleration values from simulations are ranged from -3.79m/s^2 to -7.09m/s^2 , which are much higher than the average deceleration value of -2.1m/s^2 that human drivers usually perform [50]. The increment in the magnitude of deceleration often introduces discomfort to the driver and reduces driving pleasure.

Realizing the importance of the human factor in regenerative braking control, researchers also have considered the individual driver's characteristics. In [51], the authors have proposed a Pontryagin's maximum principle (PMP)-based energy regeneration algorithm with the individual driver's speed preference considered. First, the long-term optimal control is optimized with the PMP approach. Thereafter, an MPC-based heuristic is proposed to track the PMP solution in real time. The authors claim that the proposed algorithm achieves 98% of the optimal energy recovery calculated by dynamic programming. The authors in [51] also point out that, depending on different drivers, there can be a difference of energy consumption as large as 5% between the human desired operation and the energy-optimal operation. Therefore, it is crucial to take the human factor into consideration while developing a regenerative braking algorithm. The research of [52] shows a nonlinear model predictive control (NMPC)-based car-following controller with different phases introduced. Based on relative distance and relative speed to the preceding vehicle, the ego vehicle operations are classified into "free driving," "approaching," or "unconscious following" phases and assigned with different reference velocity/distance values. The authors claim that after proper calibration, the trajectory of the ego vehicle becomes smooth and human-like. In [53], the authors have proposed a deceleration planning algorithm based on the intelligent driver model (IDM) [54], which consists of parameters reflecting the driver's personal characteristics. Those parameters are updated online using the driver's historical data to improve the prediction precision.

Although the design and implementation of regenerative braking methods considering the human factor have been systematically explored, most existing research is limited to two-pedal driving (TPD) vehicles. As one of the recent features of BEVs on the market, one-pedal driving (OPD) has only a few algorithms considering the human factor. In [55], the tractive torque of OPD is determined through acceleration pedal position (APS) and current vehicle velocity with rule-based methods. On the other hand, in [56], similar inputs are used, but the tractive torque is determined through a lookup table. In [57], a robust controller is designed to ensure OPD safety during a car-following scenario. However, none of this available research bridges the human factor with the OPD operation. Still, the on-market OPD vehicles such as Nissan Leaf, BMW i3, and Tesla Model

S have already attained their popularity [58] with a relatively simple logic:

- When the acceleration pedal is pressed, the vehicle accelerates, similar to a conventional vehicle.
- With the acceleration pedal slightly released, the vehicle starts coasting.
- With the acceleration pedal further released, the vehicle performs a regenerative braking action.

The most noticeable benefit of OPD is that a driver can use the brake pedal less frequently during city traffic, especially with many stop-and-go events. This feature also ensures that the EV recaptures as much energy as possible during braking. For example, the authors in [56] reported that the OPD algorithm could save up to 2–9% of energy compared to a parallel regeneration algorithm based on the same driving speed in city and rural driving.

However, to maintain the desired speed during driving, this OPD requires the driver to press the pedal constantly and carefully to a certain angle with forces applied. The driver also needs to pay extra attention to a single pedal-degree control. Although most test drivers can adapt to the new feature quickly [59], the transition from TPD into OPD still confuses drivers. As proposed in [58, 60, 61] and [57], the inter-vehicle spacing control methods based on OPD reduce a certain amount of the driver's effort within a platoon. However, these OPD algorithms do not sufficiently consider the individual driver's behavior. As a result, the driver might feel a sense of intrusion after these algorithms are activated. Hence, the current literature lacks a personalized one-pedal-driving algorithm that can learn a specific driver's driving behaviors. Once this algorithm learns the driver's behaviors, it will significantly save the driver's efforts on the brake pedal and increase ride comfort during braking.

1.4 Organization and Contributions

Due to the drawback of existing methods in HEV power management and vehicle longitudinal motion mentioned in Sections 1.2 and 1.3, simple integration of existing methods from the literature may not yield a decent solution in terms of fuel economy, safety, and ride comfort for a 48 V P0+P4 MHEV. Therefore, to resolve the drawbacks of the aforementioned control methods, this dissertation develops vehicle and powertrain control algorithms from the following two perspectives: optimizing power split among three power sources and optimizing vehicle longitudinal motion.

Chapters 2 and 3 focus on vehicle power splits optimization. The optimal torque split among the two electric machines and the ICE considering realistic operational constraints and longitu-

dinal load transfer is a considerably complex optimization problem; hence, it is not tractable in real-time. In Chapter 2, the control trajectories of the DP solutions satisfying global optimality are analyzed, and a simple and effective torque-split strategy using an adaptive equivalent consumption minimization strategy (A-ECMS) and a suboptimal force distribution strategy is proposed. Chapter 3 proposes a real-time energy management strategy for P0+P4 MHEV based on TD3. As prior art [29] has proven benefits of expert knowledge to DRL training speed and converged returns, our proposed TD3 framework will incorporate the expert experience from DP analysis in Chapter 2. Moreover, an innovative prioritized exploration technique is proposed to encourage the TD3 agent actively explore states in which their Q values are action-sensitive. The proposed framework, named TD3 with prioritized exploration and experience replay (TD3-PEER), allows the agent to learn a near-optimal control policy compared to existing DRL methods.

Chapter 4 proposes a novel DEco-ACC algorithm using MPC in consideration of neighboring vehicles' BSZs to further improve the potential safety of the ego vehicle without significant deterioration of fuel economy and drivability. The optimal cruise control problem can easily include constraints related to vehicle safety and riding comforts such as minimum and maximum distance from a preceding vehicle, minimum and maximum acceleration, and speed limits; thus, MPC is exploited to formulate the DEco-ACC problem. More specifically, the neighboring vehicles' BSZs are converted into state constraints, and a continuous and one-time differentiable penalty function is introduced to penalize the dwelling time in the BSZs of neighboring vehicles. As recent studies in the literature (e.g., [62, 63]) have shown that V2V technology is matured and capable of providing precise surrounding vehicle velocity information in actual operation, this study assumes that the velocity and position information from neighboring vehicles and a preceding vehicle and their BSZs are attainable. For optimizing and evaluating the performance of the proposed DEco-ACC, real-world traffic data from Next Generation Simulation (NGSIM) are used to analyze and generate car-following scenarios during highway driving. Especially, in consideration of the most probable case that one neighboring vehicle exists at one adjacent lane, a parametric study is conducted to investigate the impact of the weighting factors on the performance of the DEco-ACC.

Chapter 5 develops a more advanced one-pedal driving system using MPC, personalized one-pedal driving (POPD), inspired by the optimal regenerative braking of traditional two-pedal-driving (TPD). Similar to OPD, the POPD allows the driver to control the vehicle's acceleration with a single acceleration pedal. The upcoming braking event can automatically be handled by the POPD when the driver releases the acceleration pedal, thanks to the predictability of MPC design. To mimic a driver's braking behavior in the MPC control design, we consider headway and perceptual constraints; in particular, we have analyzed 450 drivers' real-world on-road data to investigate the constraints' dependence on a driver. In addition, we introduce a learning framework in the POPD where the weights of the MPC cost function is optimized with particle swarm

optimization. In addition, to investigate the impact of prediction accuracy on POPD performance, we have conducted a comparative case study about prediction methods and horizon lengths using real-world driving data.

The main research contributions of this thesis are summarized as follows:

- *Real-time optimization-based torque split strategy for P0+P4 MHEVs*

  To address a complex problem of optimizing the torque split among the two electric machines and the ICE considering realistic operational constraints and longitudinal load transfer, this dissertation proposes a real-time-implementable strategy using an approximated adaptive equivalent consumption minimization strategy (AA-ECMS) and a suboptimal braking force distribution strategy from dynamic programming (DP) analysis. The simulation results reveal that the proposed strategy can achieve about 97.7 % of global optimality in terms of fuel economy under validation driving cycles, as compared to the results by the DP. As a comparison, the rule-based strategy, as a benchmark, achieves 94.5 % under the same drive cycles. Considering the architecture complexity of 48V P0+P4 MHEV, existing literature only design the controller with rule-based methods. The proposed torque split strategy for a 48V P0+P4 MHEV has been published at:

  - [64] He, Y., Kwak, K.H., Kim, Y., Jung, D., Lee, J.H. and Ha, J., 2021. "Real-time Torque-split Strategy for P0+ P4 Mild Hybrid Vehicles with eAWD Capability," IEEE Transactions on Transportation Electrification, 8(1), pp.1401-1413.

- *Deep reinforcement learning-based torque split strategy for P0+P4 MHEVs with prioritized exploration and experience replay*

  State-of-art TD3 requires a critic network to generate a predicted Q value for state-action pairs for updating the policy network. However, the critic network may struggle with predicting Q values at certain states when Q values of these states are sensitive to action selection. To address this issue, this dissertation proposes a prioritized exploration technique that encourages the agent to visit action-sensitive states more frequently in the application of HEV energy management. Based on this expert twin-delayed deep deterministic policy gradient with prioritized exploration and experience replay (TD3-PEER), a novel energy management strategy is proposed for a 48V P0+P4 MHEV. Simulation results demonstrate that, with expert knowledge considered for all learning-based methods, the proposed TD3-PEER outperforms other RL-based energy management strategies including DDPG-PER and DQN by averagely 2.3% and 3.74% over the training and validation cycles, respectively. This work has been submitted to:

- He, Y., and Kim, Y. "Energy Management Strategy for 48V MHEVs Based on Expert Twin Delayed Deep Deterministic Policy Gradient Algorithm with Prioritized Exploration and Experience Replay," Submitted to 2023 American Control Conference.

- *Defensive Ecological Adaptive Cruise Control Considering Neighboring Vehicles' Blind-Spot Zones*

  This dissertation proposes a defensive ecological adaptive cruise control (DEco-ACC) algorithm that is capable of reducing an ego vehicle's dwelling time in the blind spot zones (BSZs) of its neighboring vehicles. To this end, a model predictive control is applied in the use of information such as speed, position, and blind spot zones about preceding and neighboring vehicles. The cost function of the DEco-ACC consists of tracking performance, control effort, and dwelling time in BSZs. Specifically, a continuous and one-time differentiable penalty function is introduced to handle the constraints regarding the BSZs. The simulation results from 100 cases demonstrate that on average, the DEco-ACC with optimized weighting factors can reduce the dwelling time in the neighboring vehicles' BSZs by 46.3% without significant deterioration of fuel consumption (0.04% increase in average fuel consumption) and drivability, as compared to the Eco-ACC, whose primary objective is the minimization of fuel consumption during safe car-following. The proposed DEco-ACC approach has been published at:

  - [65] He, Y., Kim, Y., Lee, D.Y. and Kim, S.H., 2021. Defensive ecological adaptive cruise control considering neighboring Vehicles' blind-spot zones. IEEE Access, 9, pp.152275-152287.

- *Personalized One-pedal Driving for Electric Vehicles by Learning-based Model Predictive Control*

  This dissertation proposes an advanced personalized one-pedal-driving (POPD) algorithm for electrified vehicles by learning-based model predictive control (MPC), capable of learning a driver's braking behaviors from collected data. Within the POPD, a simple but effective driver braking characteristic model is proposed, describing a specific driver's desired braking profile. In addition, Chapter 5 proposes a learning framework that updates the MPC weights for a specific driver while guaranteeing safety, inspired by [66]. Two constraints that are tightly related to drivers' personal preferences are derived by analyzing 450 drivers' real-world on-road data. The MPC-based controller possesses predictability of car following dynamics, which can calculate the upcoming collision event and prevent it happen. Both open-loop and closed-loop (human-in-the-loop) simulation results demonstrate the efficacy of the proposed POPD method as compared to a benchmark control method, the desired

relative distance-based personalized braking (DRD-PB) algorithm. Specifically, the human-in-the-loop results from two drivers show that brake pedal use can be reduced on a specific route by around 80%. This work has been submitted to:

– He, Y., Kwak, K.H., Kim, Y., Fan, S.. Personalized One-pedal Driving for Electric Vehicles by Learning-based Model Predictive Control. Submitted to IEEE Transactions on Systems, Man, and Cybernetics: Systems.

CHAPTER 2

Real-time Torque-split Strategy for P0+P4 Mild Hybrid Vehicles with eAWD Capability

2.1 Introduction

As introduced in Chapter 1, the torque split strategy is essential to the fuel consumption of hybrid electric vehicles, especially dual-motor HEV. The rule-based strategy has the advantage of low computation cost. However, it suffers from low-level of optimality [16]. This chapter describes an optimization-based approximated A-ECMS method for P0+P4 MHEV, which improve the fuel efficiency of this P0+P4 MHEV. At first, a P0+P4 MHEV model is laid out, including realistic constraints such as nonlinear tire, load transfer effects and braking force distribution constraints. Then, a dynamic programming analysis is conducted to determine the optimal fuel consumption and ideal torque split behaviors. Next, a modified logistic function, which captures this ideal torque split behavior, is combined with A-ECMS to develop a three-power-source torque-split algorithm. Finally, a comparison was made to the existing rule-based torque split strategy [14].

The main contributions of this chapter are summarized as follows:

- A realistic P0+P4 MHEV model is presented, with the effects of longitudinal load transfer, nonlinear tire effects, and braking force distribution for vehicle safety included in the optimization problem.
- The fuel economy potential of the 48 V P0+P4 MHEV is investigated, and useful features for developing a real-time control problem are derived from observations of the DP analysis.
- A real-time-implementable torque-split strategy for a P0+P4 MHEV is proposed using an approximated A-ECMS and a suboptimal braking force distribution strategy. Fuel economy results with the proposed strategy are compared with the results from DP and a rule-based control strategy in the existing literature.

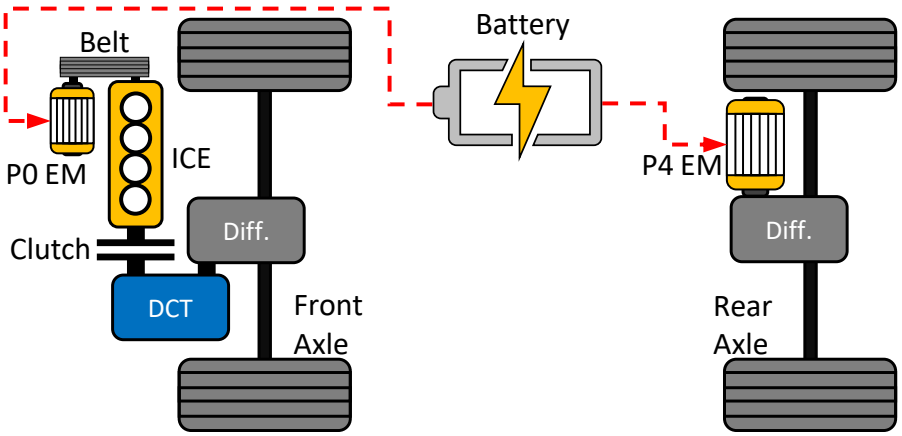

Figure 2.1: The layout of the considered 48 V P0+P4 MHEV.

The rest of this chapter is organized in the following order: The vehicle/powertrain modeling and the fuel consumption minimization problem using the DP algorithm are presented in Sections 2.2 and 2.3, respectively. The development of a real-time-implementable control method and simulation results using the proposed method are presented and discussed in Section 2.4. Finally, a summary are made in Section 2.5.

2.2 Vehicle and Powertrain Model

Throughout this thesis, a mild hybrid compact sport utility vehicle with a 1.6-liter 4-cylinder turbocharged SI engine and a 7-speed dual-clutch transmission (DCT) is studied. The DCT shifting strategy was optimized beforehand and hence fixed in this study, meaning that gear shift is not considered as an optimization variable, similarly to the existing studies [14, 67–69].

The P0+P4 hybrid powertrain is powered by a 19.4 Ah lithium-ion (li-ion) battery. An 11.5 kW P0 motor is coupled to the IC engine through a belt and pulley setup, while a 23.0 kW P4 motor is integrated on the rear axle of the vehicle through a fixed gear ratio. Figure 2.1 shows the layout of the P0+P4 MHEV powertrain. Note that in this study, a detailed model for the power electronics is not considered as the main focus is on the energy management, which is a typical approach in developing high-level control strategies in the literature [10, 11, 13, 14]. The power loss maps used in this study include inverter loss as well as mechanical loss of electric machines, i.e., P0 and P4 motors.

2.2.1 Longitudinal Vehicle Dynamics Model

The vehicle dynamics model is constructed in consideration of the longitudinal transfer of the load acting on axles. The overall wheel torque, τ_w, is computed by,

$$\tau_w = (4J_w + Mr_w^2)\dot{\omega}_w + F_r r_w, \tag{2.1}$$

where $J_w = 0.51$ kg/m^2 and $M = 1725$ kg are the wheel inertia and the vehicle gross mass, respectively; $r_w = 0.347$ m is the effective rolling radius of the wheel. With longitudinal acceleration of the vehicle, a_x, the angular acceleration of tire can be derived as:

$$\dot{\omega}_w = \frac{a_x}{r_w}, \tag{2.2}$$

and driving resistive force

$$F_r = C_0 + C_1 v + C_2 v^2 + Mgsin\theta, \tag{2.3}$$

The coefficients C_0, C_1 and C_2 were obtained from a vehicle test as 123.88 N, 2.83 N/(m/s) and 0.49 N/(m^2/s^2), respectively. The θ is the road gradient slope.

When tires slip, the angular speed of a wheel is different from the simple quotient of vehicle speed and tire radius. The angular speed, ω_w, of the front and rear wheels is then calculated by considering the slip term as follows:

$$\omega_{w,j} = \frac{v}{r_w}\left(1 + \kappa_j\right), \tag{2.4}$$

where κ is the tire slip and subscript j indicates terms for the front or rear axle/wheel/tire.

The tire slip for each axle is given as a function of the tire force developed on the tires and the normal load on the specific tire as follows:

$$\kappa_j = f\left(F_{x,j}, \frac{W_j}{2}\right), \tag{2.5}$$

where the tire force of the tires on a single axle is defined as,

$$F_{x,j} = \frac{\tau_{w,j} - 2J_w\dot{\omega}_w}{2r_w}, \tag{2.6}$$

The torque at the front wheels, $\tau_{w,front}$, can be described as the difference between total wheel torque, τ_w, and rear wheels torque, $\tau_{w,rear}$, that is,

$$\tau_{w,front} = \tau_w - \tau_{w,rear}. \tag{2.7}$$

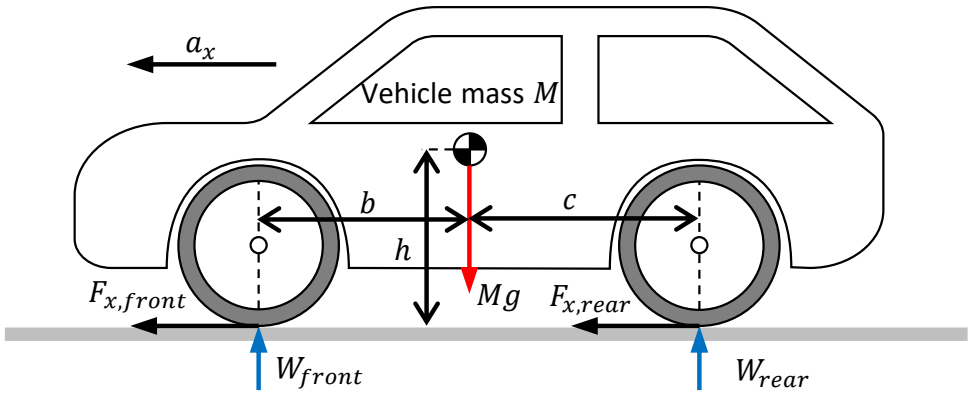

Figure 2.2: Longitudinal load transfer force distribution.

During acceleration or deceleration, inertia force causes vertical axle load transfers from front to rear or vice versa. This changing of vertical axle load influences the maximum braking force that the tires on each axle can handle. The vertical axle load on the front or rear axle is the normal load on the tires in Eq. (2.5), which is expressed as

$$W_{front} = \frac{cM}{b+c}g - \frac{hM}{b+c}a_x, \tag{2.8}$$

$$W_{rear} = \frac{bM}{b+c}g + \frac{hM}{b+c}a_x, \tag{2.9}$$

where b, c, and h measured as 1.15 m, 1.52 m and 0.67 m, are the horizontal distances from the vehicle center of gravity to the front axle and rear axle and the height of the center of gravity, respectively, as illustrated in Fig. 2.2; g denotes the gravitational acceleration of 9.81 m/s^2.

As for the powertrain, the torque and speed at the front axle are computed as follows:

$$\tau_t = \tau_e + \gamma_p \tau_{m1}, \tag{2.10}$$

$$\omega_t = \gamma_{fd}\gamma_t \omega_{w,front}, \tag{2.11}$$

$$\tau_{prop} = \left(\left(\tau_t - \tau_{gb,loss}\right)\gamma_t \eta_t - \tau_{f,loss}\right)\gamma_{fd}, \tag{2.12}$$

where τ_t is the transmission input torque, τ_e is the engine torque, τ_{m1} is the P0 motor torque, and γ_p is the pulley ratio between the engine and the P0 motor; γ_{fd} and γ_t are the final drive ratio and transmission ratio of the gear box, respectively. The transmission efficiency, η_t, is assumed constant for each gear state for simplification. Then, the front wheels torque can be rewritten with propulsion torque to the front axle, τ_{prop}, and the front brake's friction torque, $\tau_{f,front}$, as

$$\tau_{w,front} = \tau_{prop} + \tau_{f,front}. \tag{2.13}$$

Table 2.1: Relations between magic formula coefficients D and normal load.

F_z (N)	4905	9810	19620	29430	39420
D	4384.8	8375	15698	21486	26747

The torque losses by the transmission, $\tau_{gb,loss}$, and the final drive, $\tau_{f,loss}$, are given by:

$$\tau_{gb,loss} = J_{t,in}\dot{\omega}_{t,in} + \frac{J_{t,out}\dot{\omega}_{fd,in}}{\gamma_t \eta_t}, \tag{2.14}$$

$$\tau_{f,loss} = J_{fd,in}\dot{\omega}_{fd,in} + \frac{J_{fd,out}\dot{\omega}_{fd,out}}{\gamma_{fd}}, \tag{2.15}$$

where $J_{t,in}$ and $J_{t,out}$ are the inertia of transmission components and $J_{fd,in}$ and $J_{fd,out}$ are the inertia of final drive components, respectively; $\dot{\omega}_{t,in}$, $\dot{\omega}_{fd,in}$, and $\dot{\omega}_{fd,out}$ are the angular acceleration at input of transmission, input of final drive, and output of final drive, respectively.

Since the P4 motor is mounted on the rear axle, the rear wheel torque, $\tau_{w,rear}$, is expressed as

$$\tau_{w,rear} = \tau_{m2}\gamma_{m2} + \tau_{f,rear}, \tag{2.16}$$

where τ_{m2} and $\tau_{f,rear}$ are the P4 motor torque and the rear brakes friction torque, respectively. The gear ratio for the P4 motor drive is denoted by γ_{m2}.

2.2.2 Nonlinear Tire Model

The longitudinal tire force generated during acceleration and deceleration may cause tire slip. To capture the dynamic effect of realistic maximum allowable tire force, a nonlinear tire model, the Magic Formula Tire Model [70], is used:

$$F_{x,j} = D \sin\left(C \tan^{-1}\left(B(1-E)\kappa_j + E \tan^{-1}(B\kappa_j)\right)\right). \tag{2.17}$$

Since only longitudinal dynamics is considered in this study, the force coupling between longitudinal slip and lateral slip is ignored. Therefore, B, C, and E are dimensionless constants determined as 0.0735, 1.8704 and 0.686, respectively, using tire data from the commercial software CARSIM. The coefficient D is a function of instant normal load on this tire, and the relations of them are described in Table 2.1.

Figure 2.3 shows the relationship of tire force vs. tire slip generated by the nonlinear tire model used in this study. In general, the tire slips when torque is applied to the wheel. The generated

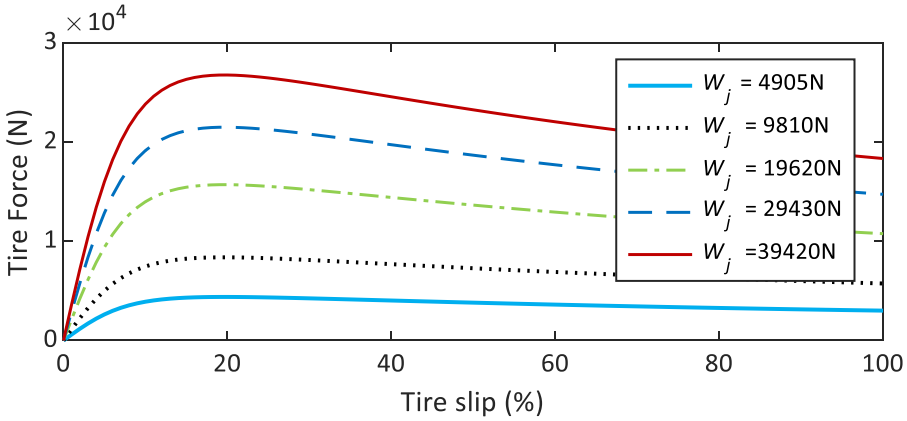

Figure 2.3: Tire force vs. tire slip modeled with the Magic Formula Tire Model. The tire force is a function of both the normal load on the tire and the tire slip.

tire force is greater with a higher normal load to the tire. At first, the generated tire force increases monotonically as slip increases; then, the force declines as the tire keeps slipping. In the figure, the tire performance reaches its peak value at around 20 % tire slip. After this point, increased tire slip does not contribute to developing better tire force. Therefore, operating the tire with the tire slip under 20 % is preferred. Under this condition, the tire force has a monotonic relationship with the tire slip, allowing for a simple 2-D lookup table. It should be noted that the vehicle lateral dynamics are not considered, and hence the coupling between longitudinal and lateral tire forces is ignored.

2.2.3 Braking Force Distribution Constraints

Even though the maximum allowable tire force is determined by the tire slip and the normal load to the tire, the force distribution between front and rear wheels needs to be determined in consideration of vehicle safety. Based on Regulation 13 of the UNECE, requirement 3.1.2 [71], the friction utilization of the front axle, k_{front}, should be greater than or equal to that of the rear axle, k_{rear},

$$k_{front} \geq k_{rear}. \tag{2.18}$$

where the friction utilization at each axle is defined as the ratio of braking force $F_{x,j}$ to the maximum braking force $F_{x,j}^{max}$ at the axle:

$$k_j = \frac{F_{x,j}}{F_{x,j}^{max}}. \tag{2.19}$$

The friction utilization and braking intensity is regulated per requirement 3.1.1 of UNECE [71], which demands the friction utilization of a given axle of between 0.2 and 0.8 and the braking

intensity, z, in the following inequality:

$$z \geq 0.1 + 0.85(k_j - 0.2), \tag{2.20}$$

with

$$z = \frac{F_{x,total}}{Mg} = -\frac{a_x}{g}. \tag{2.21}$$

2.2.4 Battery and Motor Power

The state-of-charge (SOC) dynamics of the Li-ion battery is determined by

$$\dot{SOC} = -\frac{I_b}{Q_b}, \tag{2.22}$$

where the battery capacity, Q_b, is 19.4 Ah. The battery current, I_b, is calculated using the open circuit voltage, V_{oc}, the internal resistance, R_b, and the battery power, P_b, as follows:

$$I_b = -\frac{V_{oc} - \sqrt{V_{oc}^2 - 4R_bP_b}}{2R_b}, \tag{2.23}$$

where the internal resistance, R_b, is set to be 9 mΩ. The Li-ion battery provides the power to the two motors as well as to the auxiliary load. Therefore, the battery power is expressed as follows:

$$P_b = P_{m1} + P_{m2} + P_{aux}, \tag{2.24}$$

where P_{m1} and P_{m2} are the electrical power consumption of the P0 and P4 motors, respectively; P_{aux} is the auxiliary power consumption. It should be noted that in this study, the battery temperature is assumed to be well regulated around its target value, and hence the temperature effects are ignored.

The electrical power at each motor is calculated by

$$P_{m_1/m_2} = \omega_{m_1/m_2}\tau_{m_1/m_2} + P_{m_1/m_2,loss}. \tag{2.25}$$

where the power loss of each motor, $P_{m,loss}$, is a function of the motor speed and torque.

2.2.5 Engine Fuel Consumption

In this study, the fuel consumption of the IC engine is determined from a static fuel consumption map as a function of engine speed and torque. The maximum engine brake torque is limited by the engine wide open throttle (WOT) torque. The minimum torque is restricted by the engine

motoring curve, which is the zero fueling torque curve. The minimum fuel curve can influence fuel consumption during the regenerative braking events of an MHEV [72].

2.3 Optimal Torque Split Control Strategy

2.3.1 Optimal Torque-Split Problem

Unlike a conventional HEV with a single electric motor, the P0+P4 MHEV architecture adds an additional degree of freedom to torque-split control; that is, two variables need to be determined for the torque split among the engine and the two motors. This increased problem complexity requires a new torque-split algorithm for P0/P4 HEVs. To that end, an optimal torque-split problem is first formulated and solved with full knowledge of the entire driving cycle in advance. Despite being non-causal, it still can provide an unbeatable solution to extract important rules and to evaluate the performance of new strategies. As the goal is to maximize the fuel economy of the P0+P4 MHEV, the following optimization problem is considered:

$$\begin{aligned} \min \quad & \mathcal{J} = \sum_{k=0}^{N-1} \dot{m}_{f,k} \Delta t \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_{1,k}, u_{2,k}) \\ & x_0 = SOC_0, \quad x_f = SOC_f \\ & x_k \in \mathcal{X}, \quad u_k = [u_{1,k}, u_{2,k}]^T \in \mathcal{U} \end{aligned} \tag{2.26}$$

where the cost, $\mathcal{J}$, is the total fuel consumption over a driving cycle; the fueling rate, $\dot{m}_{f,k}$, is a function of the engine speed, ω_e, and the engine torque, τ_{eng}; and x is the SOC of the battery. The subscripts 0 and f indicate the initial and final values, respectively. The control variables u_1 and u_2 represent the P0 motor torque and the P4 motor torque, respectively. The feasible sets of state and control variables are denoted by $\mathcal{X}$ and $\mathcal{U}$, respectively.

A global solution to the problem given in Eq. (2.26) is numerically obtained by using dynamic programming with the **dpm** function implemented in the MATLAB environment [73]. It is noted that the dynamic equations with state and control variables need to be properly discretized with consideration for the accuracy of the solution and the computation time. The level of discretization is as follows:

- time: 1 s,
- SOC: 2×10^{-3},
- τ_{m1} and τ_{m2}: 2 Nm.

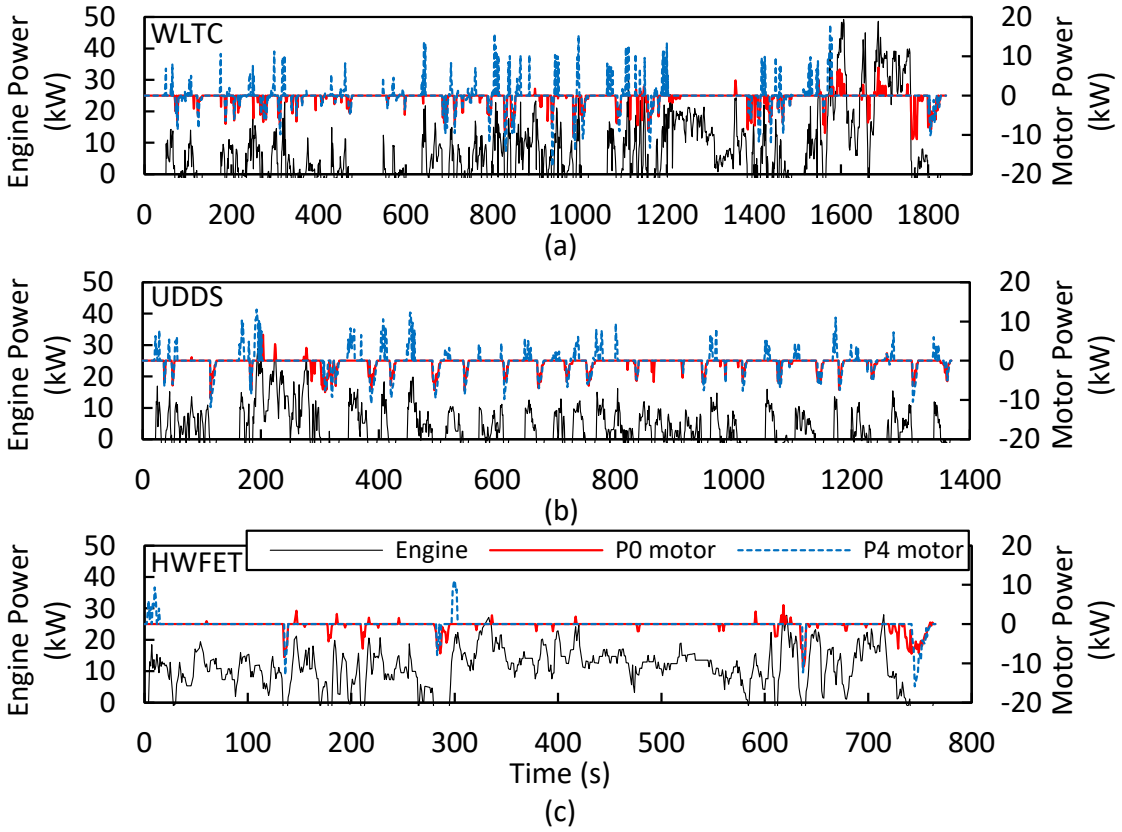

Figure 2.4: Power distribution by the DP algorithm under (a) the WLTC (b) the UDDS and (c) the HWFET.

2.3.2 Dynamic Programming Results and Analysis

2.3.2.1 Energy Consumption and Regeneration

Three regulatory driving cycles are considered in this study: the World harmonized Light-duty Test Cycle (WLTC), the Urban Dynamometer Driving Schedule (UDDS), and the Highway Fuel Economy Test Cycle (HWFET). Figure 2.4 shows the power distribution by the DP algorithm of the IC engine and the two motors with globally optimized torque-split control under the three driving cycles. In the figures, the engine provides the most propulsion power, especially when the vehicle is traveling at higher speeds, such as in the later stage of the WLTC driving cycles in Fig. 2.4 (a) or the HWFET driving cycle in Fig. 2.4 (c). This preference for using the engine is because the sizes of the P0 and P4 motors are much smaller compared to the engine in the 48 V MHEV system. Figures 2.5 (a) and (b) show the results of longitudinal load transfer and the corresponding slip ratio of front and rear axle transfer under WLTC. Since the variation in slip ratio is relatively small (-1.69 % to 4.07 % for WLTC, -1.29 % to 3.55 % for UDDS, and -1.43 % to 2.77 % for HWFET), the common rolling assumption can be applied in the development of torque-split strategies.

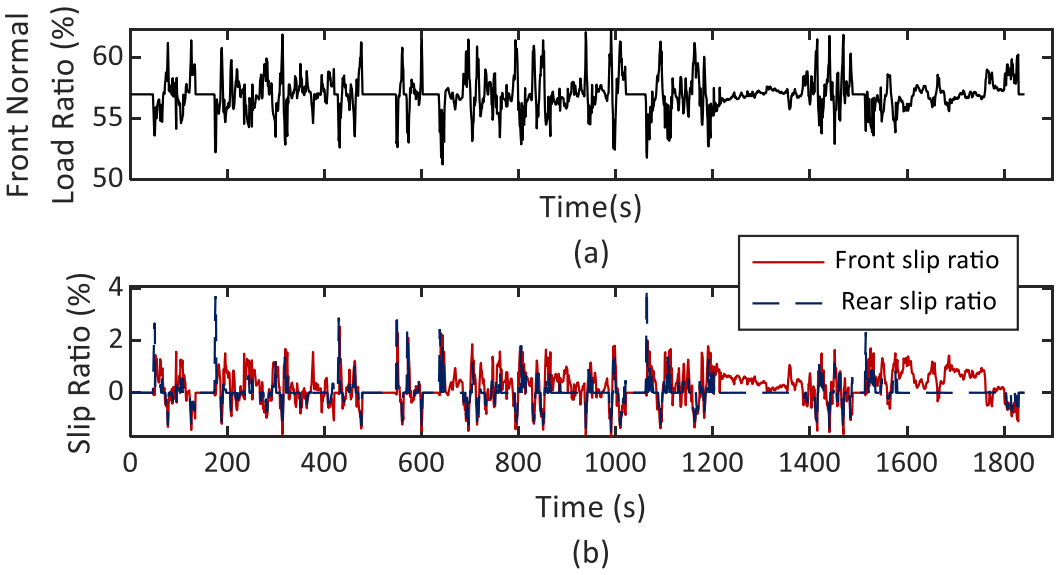

Figure 2.5: Simulation results of (a) longitudinal load transfer, (b) slip ratio and under the WLTC (DP).

To compare overall energy distribution over the driving cycles, energy consumption and regeneration of the engine and the two motors are compared in Fig. 2.6. The overall energy consumption also confirms that most energy is consumed by the engine in this 48 V MHEV. For propulsion, it is clear that the P0 motor is rarely used in all three cycles as it uses less than 1 % of the engine energy consumption. The P0 motor is mechanically coupled with the engine through the belt-pulley system which has lower transition efficiency than the geared P4 motor. Therefore, the utilization of the P4 motor for the torque assist is preferred.

The P4 motor is used for propulsion much more in the WLTC or the UDDS cases than in the HWFET. Under the WLTC or the UDDS cycles, frequent stop-and-go operations, where the IC engine efficiency is relatively low, make the use of the P4 motor preferable for assisting vehicle acceleration. Thus, the P4 motor uses about 8 % and 14 % of the total engine energy consumption under the WLTC and the UDDS cycles, respectively. In the HWFET case, the vehicle cruises at higher speeds, where the engine can operate more fuel-efficiently, and the benefits of having P0 and P4 motors are small due to the less frequent opportunity for torque assisting in acceleration and battery regeneration in deceleration.

Notably, the amount of regeneration energy by the P4 motor is similar to that of the P0 motor despite its larger size. This is because the braking force is constrained by Eqs. (2.18) and (2.21), so that the rear brakes are utilized less than the front brakes and the amount of energy regenerated by the P4 motor is limited. More specifically, the P0 motor captures breaking energy by 14 % and 10 % of the total engine energy consumption under the UDDS and WLTC, respectively. The P4

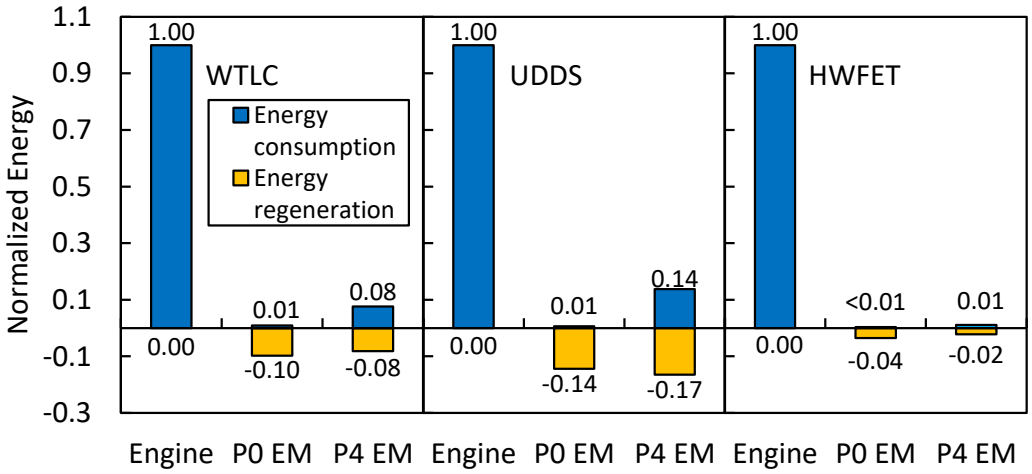

Figure 2.6: Energy distribution in each component (DP).

motor captures 17 % and 8 % under the same driving cycles. In the HWFET case, regeneration is minimal due to the fact that the vehicle is mostly cruising at higher speeds.

Since the P0 motor is small and hardly used for assisting the torque to the front axle, the P0 motor can be used to provide additional propulsion and regeneration power when the torque demand is limited to two cases: (1) In the case of the torque demand exceeding the limit of the engine and the P4 motor, the P0 torque is determined as the torque required to fulfill the demand. (2) In case of the torque demand being less than the limit of the engine motoring torque, the P0 torque helps regenerate energy only.

Under these two cases, the P0 motor torque is calculated without including P0 torque control in the optimization problem, especially for the propulsion. Therefore, a real-time-implementable optimal torque-split controller of a P0+P4 MHEV at a reduced scale that retains performance close to the globally optimal solution is feasible. For deceleration, the ratio of brake utilization between the front and the rear wheels needs to be determined first, before the calculation of P4 motor torque.

2.3.2.2 Braking Distribution Analysis

Figure 2.7 shows the scatter plots of the ratio of front tire force to the total braking force under the three driving cycles. The braking force ratio constraints are calculated based on Eqs. (2.18) and (2.21), and the derivation of the upper and lower limits can be found in [74].

The braking force distribution in the figures shows front-biased braking per the UNECE regulation. The braking force distribution affects the regeneration of the P0 and P4 motors. In a mild braking condition with deceleration lower than about 0.5 m/s^2, braking is done mostly at the front axle; however, in a hard braking condition, the rear brake is utilized as much as possible, biasing the braking force ratio toward the lower limit to maximize the energy recuperation. If the braking

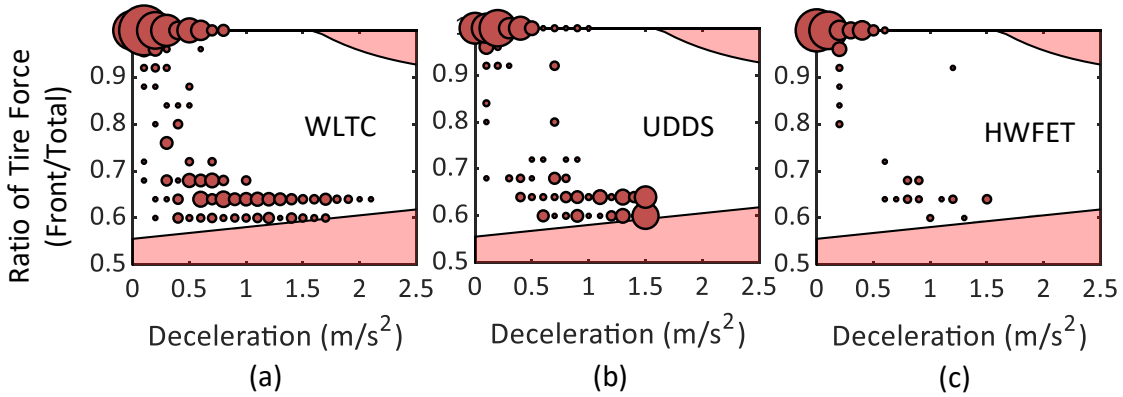

Figure 2.7: DP results of braking distribution under (a) WTLC, (b) UDDS, and (C) HWFET driving cycle. The size of bubble indicates frequency.

force distribution is predetermined, then the P4 motor torque can be calculated without solving the optimization problem, as can the torque of the P0 motor. This strategy gives the additional benefit of reducing the computational load for the real-time implementable torque-split controller.

2.4 Real-time Torque-split Strategy

This section presents a real-time torque-split strategy for the P0+P4 48 MHEV, as illustrated in Fig. 2.8, inspired by these observations from the DP results: (i) the P0 motor is hardly used for propulsion, and (ii) the ratio of tire force (front-to-total) is highly related to the deceleration.

2.4.1 Approximated Adaptive Equivalent Consumption Strategy for Propulsion

When the vehicle torque demand is positive, the torque split is determined between the front axle torque, $\tilde{\tau}_e$, and the P4 motor torque, τ_{m2}, where

$$\tilde{\tau}_e = \tau_e + \tau_{m1} \tag{2.27}$$

with

$$\tau_{m1} = \begin{cases} \tilde{\tau}_e - \tau_{e,\max} & \text{if } \tilde{\tau}_e > \tau_e \\ 0 & \text{otherwise} \end{cases}. \tag{2.28}$$

This simplification reduces the number of control variables, allowing for use of the adaptive equivalent consumption minimization strategy (A-ECMS) [67], which is widely used for real-time torque-split control of HEVs because of its performance in terms of good optimality and low

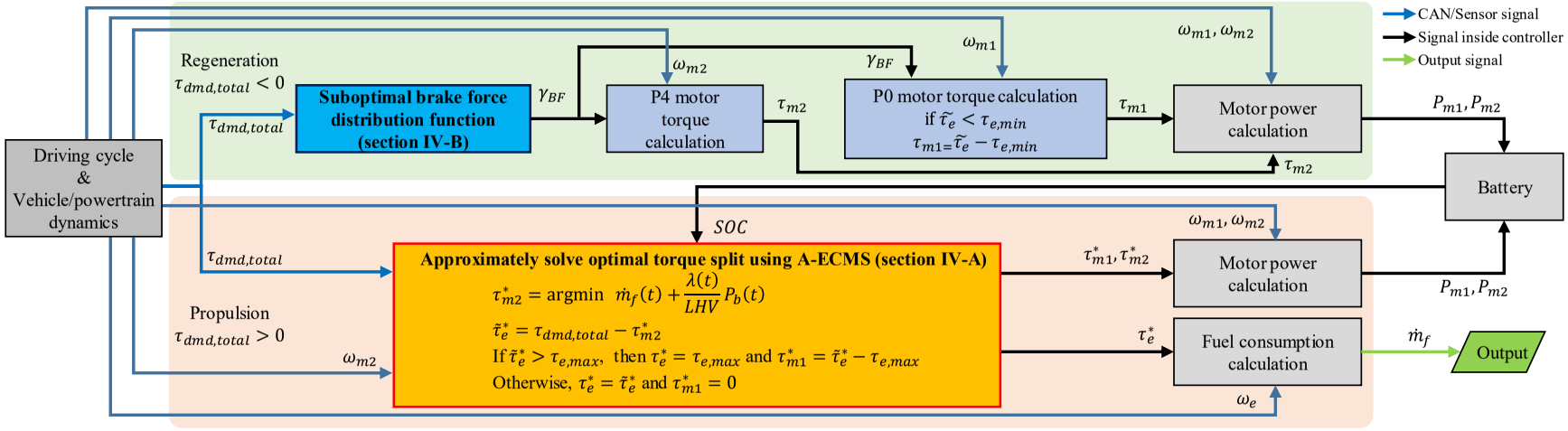

Figure 2.8: The proposed torque-split strategy for the P0+P4 MHEV: Approximated A-ECMS and suboptimal brake force distribution function are used for propulsion and braking, respectively.

computational expense. The formulation of the approximated A-ECMS is given as follows:

$$\begin{aligned} \min_{u_{2,k}} \quad & \dot{m}_{f,k} + \frac{\lambda_k}{LHV} P_{b,k} \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_k) \\ & x_k = SOC_k, \quad u_k = \tau_{m2,k} \\ & x_k \in \mathcal{X}, \quad u_k \in \mathcal{U}, \quad \phi(x_k, u_k) \in \Omega \end{aligned} \tag{2.29}$$

where λ and LHV are equivalence factor and the lower heating value of the fuel. The state and control constraints are denoted by $\mathcal{X}$ and $\mathcal{U}$, respectively. The nonlinear function $\phi(x_k, u_k)$ represents the vehicle and powertrain dynamics, accounting for the constraints on the IC engine, the P0 motor, and the battery, which are denoted by Ω.

In this formulation, P_{m1} provides torque assist only if the front axle power demand exceeds the engine capability. Thus, P_{m1} is the dependent variable, and the only independent variable is P_{m2}.

2.4.2 Suboptimal Braking Force Distribution Function for Regeneration

As seen from Fig. 2.7, the optimal brake force distribution of all three driving cycles is highly correlated with vehicle deceleration. Thus, a simple and practical approach is proposed for the deterministic computation of braking force distribution, γ_{BF}. When vehicle deceleration is lower than 0.5 m/s^2, γ_{BF} is almost unity. Then, the ratio decreases sharply closer to the lower bound. A sigmoid function such as the logistics function can describe an S-shape curve empirically. However, the logistics function approaches a fixed minimum point, where the lower bound of γ_{BF} is not a constant but a function of deceleration and vehicle parameters. Therefore, to contain γ_{BF} within the bounds, the logistics function is modified as follows:

$$\begin{aligned} \gamma_{BF} &= \frac{F_{x,front}}{F_{x,total}} \\ &\approx \underbrace{\frac{A_1}{1 + \exp(-A_2(-a_x - A_3))} + A_4}_{\text{scaled and shifted logistics function}} + \underbrace{\frac{c + h(-a_x/g)}{b + c}}_{\text{lower bound of } \gamma_{BF}} \end{aligned} \tag{2.30}$$

where A_1, A_2, A_3, and A_4 are the coefficients determined by the least squares method using DP results of three driving cycles. It is noted that these coefficients may need to be updated in different driving conditions, which can be one of the directions for future work. This optimized regression model is compared with the data from the DP results in Fig. 2.9. With this model, the negative torque demand can be simply split and provided by the P0 and the P4 motors, and hence the computational expense can be dramatically reduced.

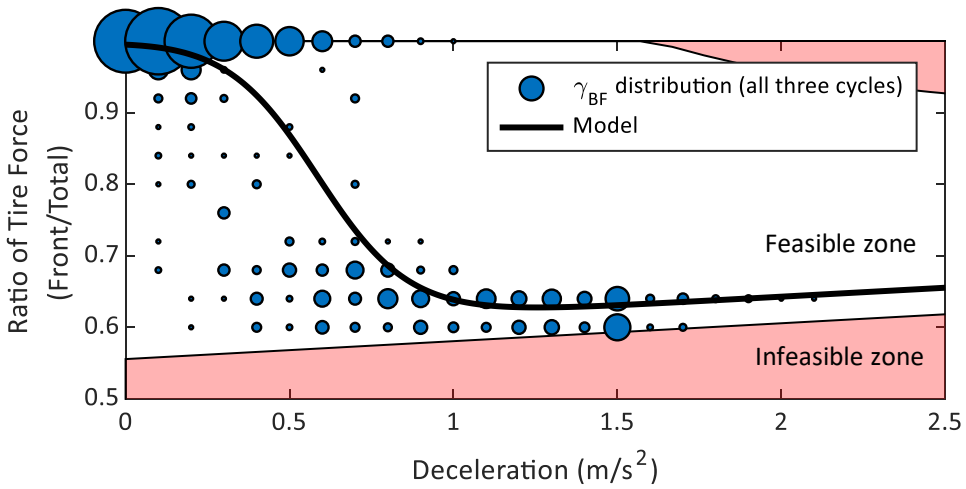

Figure 2.9: Suboptimal braking force ratio function compared to γ_{BF} distribution of all three cycles combined. The size of the bubble indicates frequency.

In order to maximize braking energy recuperation, the P4 motor is used first, and then the rear friction brake is used when the torque demand at the rear axle exceeds the P4 motor torque capacity. At the front axle, the P0 motor is used for regeneration when the demanded braking torque is greater than the engine motoring torque.

2.4.3 Adaptation of Different Driving Scenarios: A Parametric Study

The equivalence factor, λ, plays an important role in the fuel economy performance of the P0+P4 MHEV. When λ is too high or low, the controller would charge or drain the battery quickly. It should be noted that the work by the authors in [68] has shown that the balance between immediate fuel rate and SOC level in A-ECMS is coherent with the immediate reward and future reward in reinforcement learning (RL). In such a causal framework, the equivalence factor should be periodically updated to sustain the battery SOC level. Thus, various techniques have been considered to adequately adapt the equivalence factor λ. There are mainly three approaches to adapting equivalence factor λ [75] with different driving conditions and a trade-off between chemical and electric power:

1. Adaptation based on driving cycle prediction;
2. Adaptation based on driving pattern recognition;
3. Adaptation based exclusively on feedback from SOC.

These three approaches are not stand alone from each other, and their combinations have also been proposed. Especially, experimental results from [69] show that a parallel HEV could achieve

near-optimal fuel economy even with a sole adaptation of the approach 3).

As suggested by [68] and [69], in considering the memory of car ECU and computation time, adaptation approach 3) is applied in this study. To guarantee the robustness of the time varying equivalence factor λ_k, the relationship between λ_k and the current SOC, SOC_k, is given by:

$$\lambda_k = K_p(SOC_{ref} - SOC_k) + \lambda_0, \tag{2.31}$$

where the reference SOC, SOC_{ref}, is set to be 60 %. The parameters K_p and λ_0 are constant values to be determined to maximize the overall fuel economy of the P0+P4 MHEV in consideration of charge sustainability. Equation (31) forces the IC engine to operate by increasing the λ_k when the SOC is low, and vice versa.

To properly select K_p and λ_0 for the proposed strategy, 1736 cases of parameters for K_p and λ_0 were evaluated over the WLTC, the UDDS, and the HWFET cycles in considering fuel consumption and the terminal SOC deviation to the initial SOC. The electric power associated with the terminal SOC deviation for each case is converted into equivalent fuel and added to the total fuel consumption, noted as corrected fuel. Each driving cycle is run with ECMS algorithm [76] and several different equivalence factors. With enough runs on the same cycle, the change of fuel regarding SOC deviation, $\frac{\Delta(fuel)}{\Delta(SOC)}$, is obtained as a near-constant value and used for fuel consumption correction. At current Architecture, with K_p and λ_0 being as 8500 and 10000, the proposed strategy can achieve the best overall fuel economy among all cases with a reasonable terminal SOC deviation.

2.4.4 A Rule-based Real-time Control Algorithm

With current existing researches, there is no available instantaneous optimization-based algorithm for P0+P4 48V MHEVs. However, one research has developed a rule-based algorithm for a similar P0+P4 architecture but equipped with a much stronger battery and electric machines [14]. Their rule-based algorithm cannot be applied to this P0+P4 48V architecture directly due to the low capability of electric machines in MHEVs, but it still inspires us to design a similar rule-based algorithm for benchmark purposes.

In this rule-based algorithm, the possible demand power is divided into six zones as shown in Fig. 2.11. In the figure, the boundaries of each zone are defined as follows: the green curve denotes the optimal engine power level at each speed[1]; the red curve denotes the EV mode threshold; the yellow curve denotes the boundary of positive demand power and negative demand power. The threshold for the EV mode is included to accommodate the algorithm in [14] for P0+P4 48V MHEVs.

[1]The optimal engine operating line are determined from the minimum BSFC points.

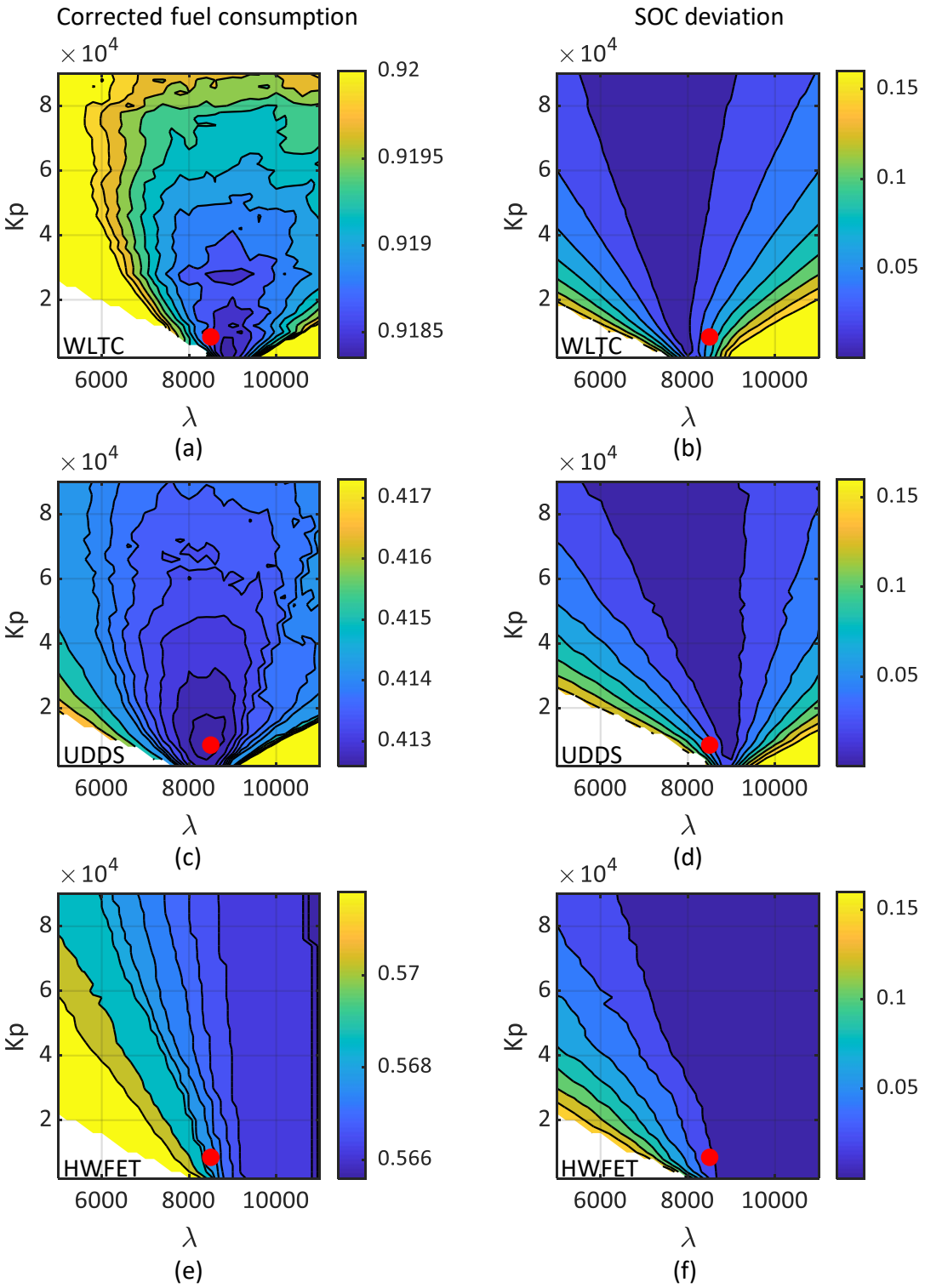

Figure 2.10: Parameter study of K_p and λ_0 for the WLTC, the UDDS and the HWFET driving cycles, left column (a, c, e) represents corrected fuel, right column (b, d, f) represents SOC deviation. Highlighted point is the selected to conduct following research.

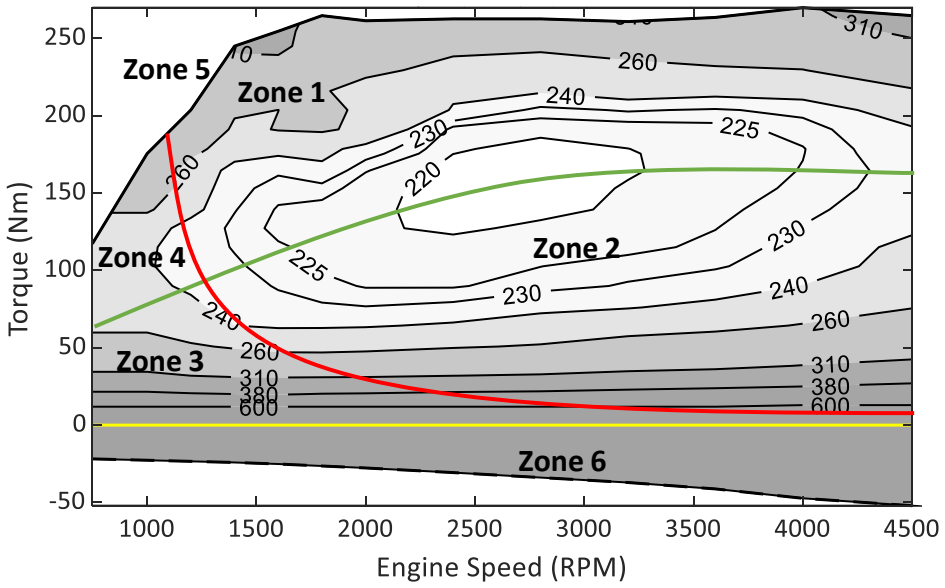

Figure 2.11: Demand power zone on engine brake specific fuel consumption (BSFC) map. The green curve denotes engine optimal power level. the red curve denotes EV mode on/off power. The yellow curve denotes positive/negative power boundary.

The engine and two motors operate based on which zone the current demand power falls into:

- Zone 1: when the battery SOC is low, use the engine solely; when the battery SOC is high, operate the engine at optimal power level and use the P4 motor to satisfy the total demand power.
- Zone 2: when the battery SOC is high, use the engine solely; when the battery SOC is low, operate the engine at optimal power level to charge the battery with excess power.
- Zone 3: when the battery SOC is high, use the P4 motor solely; when the battery SOC is low, operate the engine at optimal power level to charge the battery with excess power.
- Zone 4: when the battery SOC is high, use the P4 motor solely; when the battery SOC is low, use the engine solely.
- Zone 5: operates the engine at the maximum power level, and the P0 and P4 motors assist the engine.
- Zone 6: the engine provides motoring power; the P4 and P0 motors recapture braking energy based on the method described in Fig. 2.9.

The thresholds for the EV mode and low/high states of the battery SOC are carefully calibrated for the best performance using the WLTC, the UDDS, and the HWFET driving cycles.

2.4.5 Performance of Real-time Control Strategies

The proposed torque-split algorithm is implemented in the P0+P4 MHEV model and evaluated under various driving cycles by comparing the results with the DP and the rule-based strategy. The WLTC, the UDDS, and the HWFET cycles are used to determine the optimal parameters of the proposed strategy shown in Fig. 2.8, and the LA92 and the US06 are used for performance validation.[2]

Table 2.2: Fuel consumption comparison between DP, proposed and rule-based algorithm. Corrected fuel consumption is compared with DP.

Type	Driving cycle	Fuel consumption (kg)			End SOC deviation from initial (%)		Corrected fuel consumption	
		DP	proposed	Rule-based	proposed	Rule-based	proposed	Rule-based
Training	WLTC	0.9104	0.9107 (+0.03%)	0.9906 (+8.81%)	−4.64	0.75	0.9186 (+0.9%)	0.9893 (+8.67%)
Training	UDDS	0.4069	0.4220 (+3.71%)	0.4296 (+5.58%)	5.67	0.59	0.4126 (+1.4%)	0.4282 (+5.23%)
Training	HWFET	0.5649	0.5630 (−0.34%)	0.5946 (+5.26%)	−2.06	0.35	0.5678 (+0.51%)	0.594 (+5.15%)
Validation	LA92	0.6403	0.6575 (+2.68%)	0.6862 (+7.17%)	−1.47	2.91	0.6598 (+3.05%)	0.6816 (+6.45%)
Validation	US06	0.6337	0.6375 (+0.6%)	0.6982 (+10.18%)	−3.67	9.75	0.644 (+1.63%)	0.681 (+7.46%)

Simulation results from the dynamic programming, the proposed strategy and the rule-based strategy for all five drive cycles are summarized in Table 2.2. As shown in the table, the terminal SOC values from the proposed strategy and the rule-based strategy deviate from the initial SOC. Therefore, for a fair comparison, the total fuel consumption is corrected by considering the mismatch of terminal SOC. Each driving cycle is run again by the proposed strategy with several different equivalence factors. With enough runs on the same cycle, the change of fuel regarding SOC deviation, $\frac{\Delta(fuel)}{\Delta(SOC)}$, is obtained and used for fuel consumption correction.

Compared to the DP results, the corrected fuel consumption values from the proposed strategy are 0.9 %, 1.4 %, 0.51 %, 3.05 % and 1.63 % higher under the WLTC, the UDDS, the HWFET, the LA92 and the US065 driving cycles, respectively. The rule-based strategy underperforms compared to the proposed strategy; the corrected fuel consumption values are 8.67 %, 5.23 %, 5.15 %, 6.45 % and 7.46 % higher than the DP results under the same driving cycles. On average, the fuel consumption performance is 93.6 % of global optimality in the training cycles and 94.5 % of global optimality in the validation cycles with the rule-based strategy. The performance of the

[2]For a detailed discussion, only the results from the WLTC are presented in the paper; however, the fuel economy results over all driving cycles are reported in Table 2.2.

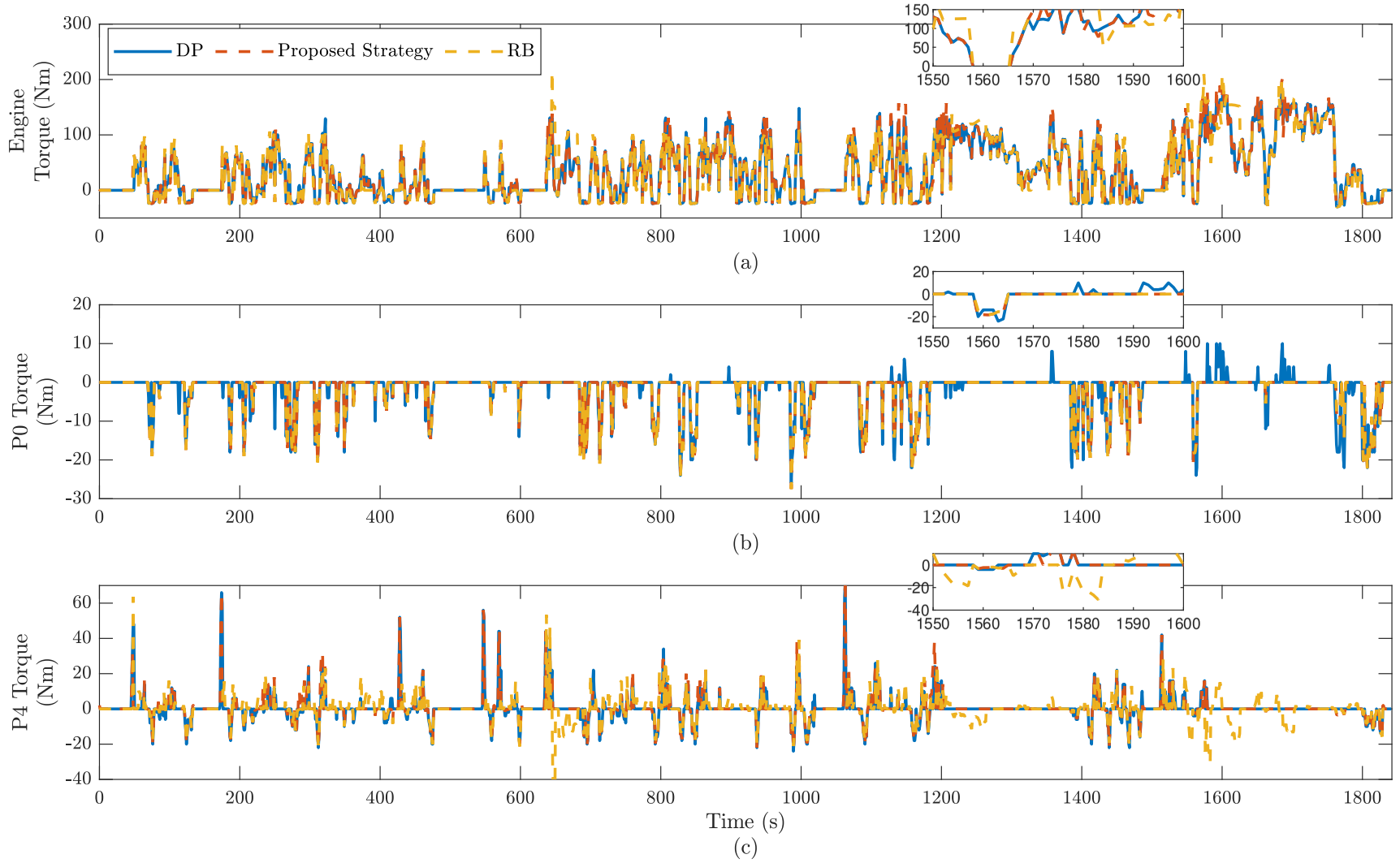

Figure 2.12: Torque trajectory of the DP, the proposed strategy and rule-based (RB) results under the WLTC cycle for (a) Engine, (b) P0 motor, and (c) P4 motor.

proposed strategy is superior as an average of 99.1 % of global optimality in the training cycles and an average of 97.7 % of global optimality in the validation cycles are achieved.

Figure 2.12 shows the torque trajectories of the engine, the P0 motor, and the P4 motor with three strategies under the WLTC cycle. As shown in the figure, the proposed strategy controls the torque split similar to the DP. The P0 motor rarely operates for the propulsion by the DP as shown in Fig. 2.12 (b), the P0 motor torque trajectory from the proposed strategy mostly matches to that of the DP as well. In contrast, the rule-based strategy's torque split notably differs from those by the other strategies. Sometimes the engine of the rule-based strategy provides higher power than the others and the P4 motor captures the exceeding power to charge the battery (the zoomed-in portion in Fig. 2.12 (a) and (c)). This double energy conversion makes the rule-based strategy to be inefficient.

Figure 2.13 shows the SOC trajectories from the DP, the proposed strategy and the rule-based strategy under all five drive cycles ((a) the WLTC, (b) the UDDS, (c) the HWFET, (d) the LA92 and (e) the US06). As discussed, the rule-based strategy pays an extra cost (double energy conversion) to maintain the SOC close to the reference SOC. Therefore, the resulting SOC trajectory differs from those obtained by the proposed strategy and the DP. For example, in Fig. 2.13 (c), unlike other two strategies that drain the battery SOC from 300 s to 600 s, the rule-based strategy charge the battery around 420 s because the battery SOC falls below the low SOC threshold. Fig. 2.13 shows that the overall SOC trends of the proposed strategy are similar to the those from the DP results in all five cycles, which can be explained by the similarity of the powertrain operations between two strategies as shown in the Fig 2.14.

Figure 2.14 compares the operating points of the engine and the two motors with the DP, the proposed strategy and the rule-based strategy under the WLTC. It can be observed that the engine (Fig. 2.14 (a) and (d)) and the P4 motor (Fig. 2.14 (c) and (f)) operate very similarly under the DP and the proposed strategy. The frequency and range of the visited operating points are almost identical except for a few high-load points in the engine operation. On the other hand, the P0 motor operations are slightly different, as seen from Fig. 2.14 (b) and (e), which is due to the fact that the P0 motor is used mostly for regenerative braking under the proposed strategy. It is noted that since the P0 motor is attached to the engine, the speed range of the P0 motor operation is very similar between the two strategies. The engine and the P4 motor operation of the rule-based strategy (Fig. 2.14 (g) and (f)), however, are very different from those with the DP and the proposed strategy. The engine (Fig. 2.14 (g)) of the rule-based algorithm tends to operate at the optimal power level that is defined in Fig. 2.11. As discussed earlier, for maintaining the battery SOC level, the P4 motor operation with the rule-based strategy is also different from those with the DP and the proposed strategy as shown in Fig. 2.14 (f). However, the P0 motor operation of the rule-based strategy is similar to the proposed strategy, which is due to the fact that both

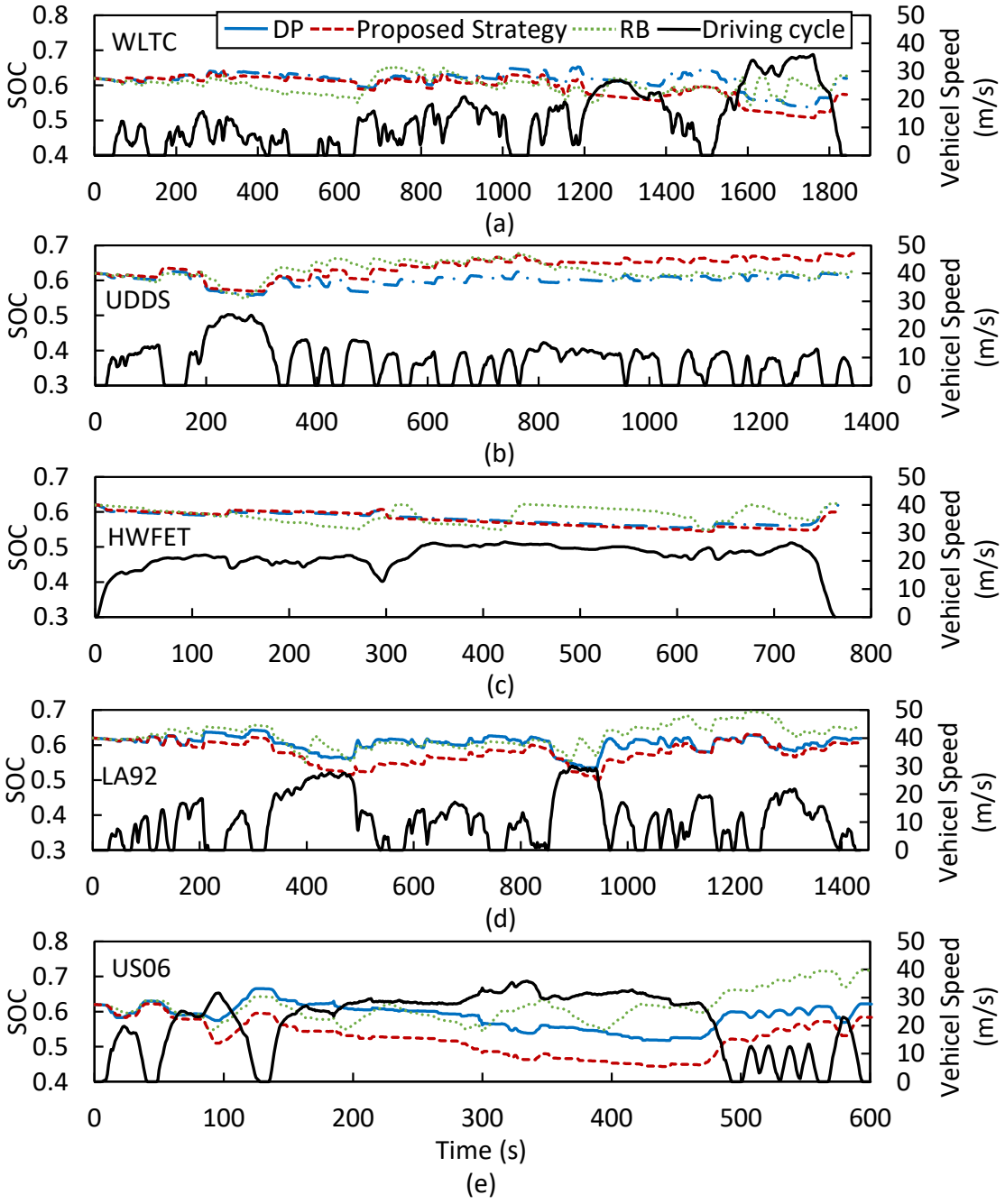

Figure 2.13: SOC trajectories of the DP and the proposed strategy results under the five driving cycles: (a) the WLTC, (b) the UDDS, (c) the HWFET, (d) the LA92 and (e) the US06

strategies adopt similar braking rules: sub-optimal braking force function is utilized to find the force distribution between two axles and the P0 motor only recuperate braking energy when the braking demand power is larger than the engine motoring power.

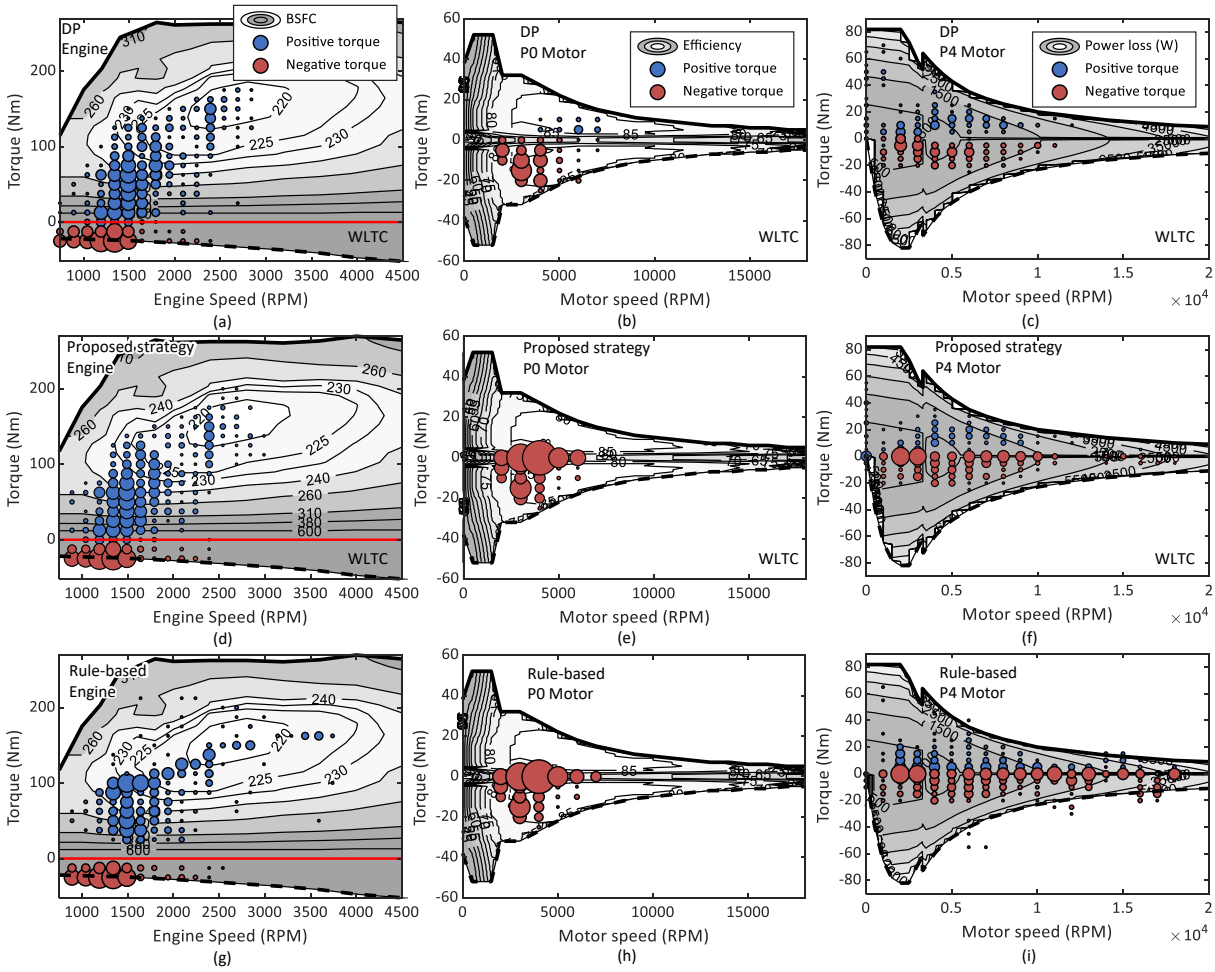

Figure 2.14: Comparison of operation points distribution under the WLTC driving schedule from the DP results: (a) the engine, (b) the P0 motor, and (c) the P4 motor, from the proposed strategy results: (d) the engine (e) the P0 motor and (f) the P4 motor and from the rule-based strategy results: (g) the engine (h) the P0 motor and (i) the P4 motor. The size of the bubble indicates frequency.

2.5 Summary

A real-time-implementable torque-split strategy for minimizing the fuel consumption of a P0+P4 MHEV is proposed in this chapter. Since the optimal torque-split among the IC engine and the P0 and P4 motors is complicated and computationally demanding, reducing the size of the optimization problem is desired. In this chapter, the optimal torque-split problem is formulated with a detailed modeling approach, including longitudinal load transfer, non-linear tire model with tire slip, and brake distribution regulation, and then solved with dynamic programming. The DP results reveal that (i) the commonly used rolling assumption is applicable, (ii) the P0 motor is rarely used for propulsion, and (iii) the ratio of tire force (front-to-total) is highly related to deceleration. Based

on these observations, the proposed strategy combines an approximated A-ECMS and a suboptimal braking force distribution function for vehicle propulsion and regeneration, respectively. The simulation results show that the proposed strategy for the considered P0+P4 MHEV can achieve more than 99.1% and 97.7% of global optimality compared with the DP results in both training and validation cycles, and also it is capable to adapt other drive cycles that is not been exposed to. In contrast, a rule-based strategy only achieves 93.6% and 94.5% of global optimality under same drive cycles.

CHAPTER 3

HEV Energy Management Strategy Based on TD3 with Prioritized Exploration and Experience Replay

3.1 Introduction

Thanks to reinforcement learning's astonishing adaptability to challenging problems, researchers have developed several torque-split controllers for single motor HEV with RL methods: recurrent Q-learning [27], DDPG [29] and TD3 [36]. However, to the authors' knowledge, there is no existing literature about the DRL EMS for the P0+P4 MHEV system. Also, existing literature does not encourage the DRL agent to explore action-sensitive states during the training. To address the issues of the aforementioned RL-based HEV energy management strategies in section 1.2.2, this paper proposes a prioritized exploration and experience replay (PEER) technique as an add-on to TD3-PER for the energy management of HEVs. During the early stage of the training, the PEER encourages the agent to explore high-complexity regions in the transition space more frequently. As prior art in [29] has proven benefits of expert knowledge to DRL training speed and converged returns, our proposed TD3 framework will incorporate the expert experience from dynamic programming analysis [64] from a previous study. The main contributions of this work are threefold:

- a non-linear mapping between actor output and motor power is constructed, which condenses the on/off state of the motor and the motor power into a single variable.
- an expert-interposing DRL method is developed based on the state-of-art algorithm TD3 for P0+P4 HEVs.
- a novel exploration method for TD3 is proposed to encourage agents to explore complex dynamics region of the system, and its performance is compared with DDPG-PER and DQN.

The later sections organize as follows: The detailed reinforcement learning framework is in section 3.2. The convergence performance of the proposed methods during training is in section

3.3. Section 3.4 analyzes the importance of expert knowledge to the proposed method in this P0+P4 MHEV architecture. Section 3.5 investigates the improvements between the proposed methods and existing DRL methods. Section 3.6 draws the summary and future work.

3.2 Deep Reinforcement Learning with Expert Knowledge

3.2.1 Optimal Torque Split Problem for P0+P4 MHEV

Compared to a conventional HEV, which has only one electric machine, the P0+P4 MHEV requires determining the power split among three power sources. Moreover, the increase in power sources introduces an additional control variable that makes applying the traditional torque split algorithm from HEV difficult, e.g., ECMS [67]. Hence, the optimal torque split problem should be formulated and solved from scratch. Furthermore, because there is no plug-in charging port for MHEVs, the terminal battery SOC is desired not to deviate too much compared to the initial SOC. In the end, an optimal torque split problem among three power sources with SOC level constraints can be formulated as follows:

$$\begin{aligned} \min \quad & \mathcal{J} = \sum_{k=0}^{N-1} \dot{m}_{f,k} \Delta t \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_{1,k}, u_{2,k}) \\ & x_0 = SOC_0, \quad x_f = SOC_f \\ & x_k \in \mathcal{X}, \quad u_k = [u_{1,k}, u_{2,k}]^T \in \mathcal{U} \end{aligned} \tag{3.1}$$

The cost $\mathcal{J}$ is the total fuel consumption over the trip. The instantaneous fuel consumption $\dot{m}_{f,k}$ depends on engine speed, ω_e, and the engine torque, τ_{eng}. u_1 and u_2 represent the control variables of the P0 motor torque and the P4 motor torque, respectively. x represents the battery SOC, and its initial and terminal values are defined as SOC_0 and SOC_f, respectively. A global solution to the problem in equation (3.1) is obtained through a numerical approach of dynamic programming with dpm.m in MATLAB [73]. Unfortunately, the dynamic programming method is computationally expensive and non-causal, hence cannot be converted into a real-time control strategy. However, the control sequence from the global optimal solution usually contains a substantial trend that can be taken advantage of while developing a real-time control strategy. The level of discretization is chosen to be:

- time: 1 s,
- SOC: 2 $\times 10^{-3}$,

- τ_{m1} and τ_{m2}: 2 Nm.

3.2.2 Expert Knowledge from Dynamic Programming

In our previous study [64] and chapter 2, the dynamic programming (DP) results revealed that the P0 motor operation should be avoided except when the P4 motor and the engine cannot fulfill the propulsion/braking demanding torque. The power generated/captured from the P0 motor suffers losses due to the belt/pulley system and the transmission system. Moreover, since the P0 motor is coupled with the engine, it is inherently disabled when the clutch is disengaged. With this expert knowledge from the DP, the control of the P0 motor can be replaced with a simple rule. In a later section, we will show that with the simplified P0 motor operation, the learning of TD3 can be faster and more stable, with guaranteed fuel economy performance.

Thus, the optimization problem can be reduced to the torque split between the engine at the front axle $\tilde{\tau}_e$ and the P4 motor at the rear axle τ_{m2}, as given by,

$$\tilde{\tau}_e = \tau_e + \tau_{m1}, \tag{3.2}$$

and the further torque split between the engine and the P0 motor is controlled by a rule-based logic:

$$\tau_{m1} = \begin{cases} \tilde{\tau}_e - \tau_{e,\max} & \text{if } \tilde{\tau}_e > \tau_e \\ 0 & \text{otherwise} \end{cases}, \tag{3.3}$$

where $\tau_{e,\max}$ is the maximum engine torque. This simplified torque split problem is then solved with twin-delayed deep deterministic policy gradient (TD3) methods.

3.2.3 Twin-delayed Deep Deterministic Policy Gradient

The twin-delayed deep deterministic policy gradient is a reinforcement learning technique inspired by the Markov decision process (MDP) [77]. The MDP provides a mathematical framework for control decision-making. The MDP contains an agent and an environment that interact with each other. In this study, the torque split controller is defined as the agent, and the HEV model described in Section 2.2 is defined as the environment. For each time step t, the agent located at the state s_t will apply an action a_t to the system. The system will be perturbed to a new state s_{t+1} and will give the agent a reward r_t. The benefit of MDP is that the s_{t+1} is dependent on only (s_t, a_t) and is conditionally independent of any previous states. Therefore, the transition of $(s_t, a_t) \xrightarrow{r_t} s_{t+1}$ contains the knowledge of the system dynamics at the current state. By design, the agent should with a sufficient exploration of the environment be able to take the energy-efficient torque-split action that leads to the largest possible Q for the current state. In the method of TD3, there are three

main networks. The main actor network maps each state s to action a. Two main critic networks map transitions to their estimated value $Q(s, a)$, respectively. This actor-critic architecture aims to select an action that leads to the largest possible $Q(s, a)$.

Based on the MDP, the actions and states for the considered HEV power management problem are chosen as

- State: SOC, gear, v_e, acceleration, r_{tp}
- Action: P_{m2} with on/off

The trip ratio r_{tp} is defined as the traveled distance divided by the total trip distance. The P4 motor power P_{m2} and its on/off are combined into a single action output of the actor-network. The reward function is constructed as follows:

$$\begin{aligned} r_t = &- (c_1 \dot{m}_{f,t} + c_2 (SOC_{t+1} - SOC_0)^2 + \\ &c_3 I_f + c_4 (\max(0, SOC_{lb} - SOC_{t+1}) + \\ &\max(0, SOC_{t+1} - SOC_{ub}))), \end{aligned} \tag{3.4}$$

where $\dot{m}_{f,t}$ is the fuel rate. The lower and upper bounds of the battery SOC are denoted by SOC_{lb} and SOC_{ub}, respectively, and the c_is represent weighting factors for the terms in the reward function.

In each iteration, the action network will generate action with policy π, current state s, and a certain amount of random exploration ϵ:

$$a_t \sim \pi_\phi(s_t) + \epsilon, \tag{3.5}$$

where ϵ follows the normal distribution with a mean of 0 and a standard deviation of σ,

$$\epsilon \sim N(0, \sigma), \tag{3.6}$$

The observed reward r_t and next state s_{t+1} also will be recorded into a tuple (s_t, a_t, r_t, s_{t+1}) for experience replay and the evaluation of the two critic networks.

3.2.4 P4 Motor Power Control with on/off

The P4 electric machine's on/off and power command are combined into a single control variable (action). First, an activation threshold ζ for the P4 motor is introduced. If the absolute value of action is less than ζ, the P4 motor is classified as deactivated and should output zero power. If the action value is larger than ζ, then the normalized power output for this motor is calculated as

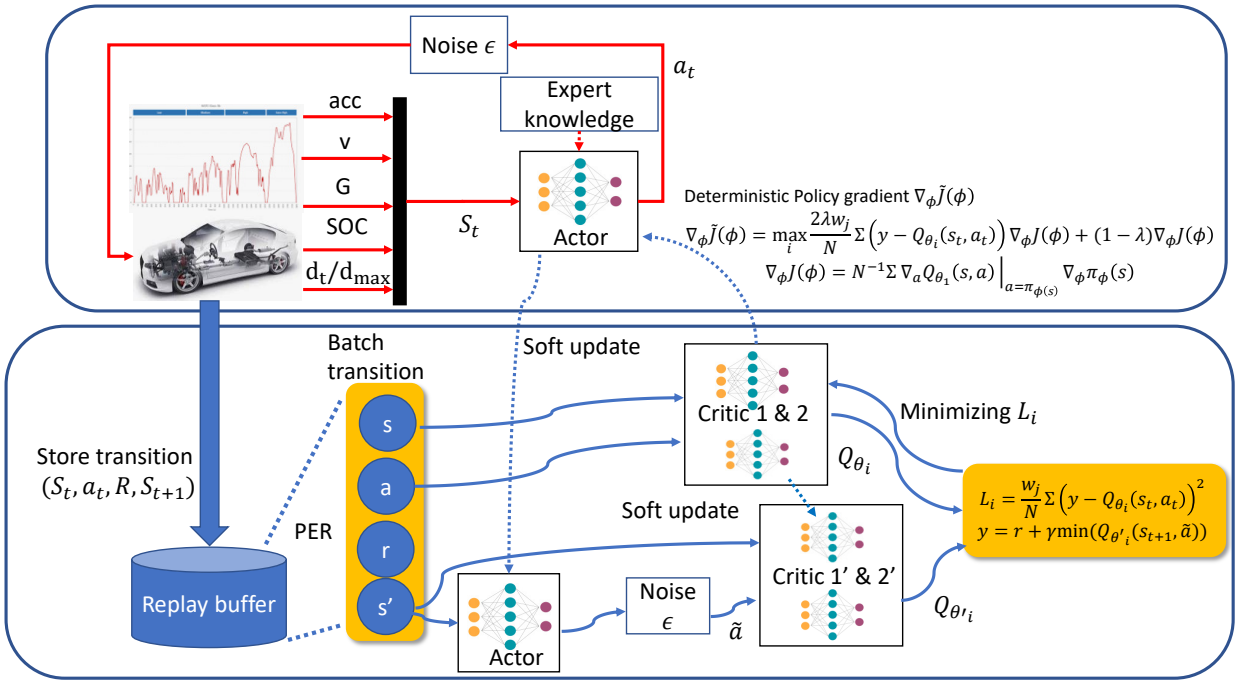

Figure 3.1: The proposed power-split strategy for the P0+P4 MHEV: structure of expert TD3 with prioritized experience replay and prioritized exploration.

shown in Fig. 3.2 and the equation below:

$$\hat{P}_{m2} = \begin{cases} \frac{a-\zeta}{1-\zeta} & \text{if } a > \zeta \\ \frac{a+\zeta}{1-\zeta} & \text{if } a < -\zeta \\ 0 & \text{otherwise} \end{cases}, \tag{3.7}$$

where $a \in [1, -1]$ is the actor-network output and $\hat{P}_{m2} \in [-1, 1]$ is the normalized P4 motor power. The motor torque automatically adjusts itself within the torque max/min limits at different motor speeds by controlling the motor power. In addition, the control of the P4 motor on/off is merged into the same power control variable, which simplifies the control problem.

3.2.5 Networks Updating Rule

After each transition is completed, the networks will update themselves with a batch of N samples selected from the experience buffer. The critic networks can update their parameters with the temporal difference (TD) target:

$$y = r_t + \gamma \min_{i=1,2} Q_{\theta'_i}(s_{t+1}, \tilde{a}|_{\tilde{a}=\pi_{\phi'}(s_{t+1})+\epsilon}), \tag{3.8}$$

Algorithm 1 TD3 with prioritized exploration and experience replay

initialization: critic network and actor network with weights θ_i and ϕ
copy target net $\theta_i' \leftarrow \theta_i$, $\phi_i' \leftarrow \phi_i$
Initialize Replay Buffer and random process for action exploration.
for episode 1:M **do**
    get initial states: SOC, gear, v_e, acceleration, r_{tp}
    for t=1:T **do**
        Select action $a_t \sim \pi_\phi(s_t) + \epsilon$, according to the current policy and exploration noise
        Execute action a_t , observe reward r_t and new states s_{t+1}
        Store transition (s_t, a_t, r, s_{t+1}) in Replay buffer
        sample a mini-batch of N transitions (s_t, a_t, r, s_{t+1}) from Replay buffer
        Set $y = r + \gamma min_{i=1,2} Q_{\theta_i'}(s', \tilde{a})$
        Update critic parameters θ_i by minimizing the loss: $L = w_i(Q_i - y_i)^2$
        if t mod d **then**
            Update the actor policy using the deterministic policy gradient: $\nabla_\phi \tilde{J}(\phi)$
            Update the target networks: $\theta_i' \leftarrow \tau\theta_i + (1+\tau)\theta_i'$, $\phi_i' \leftarrow \tau\phi_i + (1+\tau)\phi_i'$,

where γ is the discount factor for the future predicted value, usually chosen as less than 1; $\tilde{a}$ is the action generated by the target action network based on the state s_{t+1} and its policy parameter ϕ' with a Gaussian-distributed but clipped random exploration noise:

$$\tilde{a} \sim \pi_{\phi'}(s_{t+1}) + \epsilon, \tag{3.9}$$

$$\epsilon \sim clip(N(0, \tilde{\sigma}), -c, c). \tag{3.10}$$

The purpose of having both critic networks is to avoid overestimation of the Q value. Each main critic network parameter θ_i is updated with the cost function, defined as the square of the TD error:

$$L_i = \frac{1}{N}\sum_{m=1}^{N}(y_m - Q_{\theta_i}(s_{m,t}, a_{m,t}))^2, \tag{3.11}$$

To ensure the stability of the training process, a traditional TD3 adopts a delayed update policy for the actor network and all other target networks. For every d time steps, the parameter ϕ in the actor network is updated with a deterministic policy gradient, given by

$$\nabla_\phi J(\phi) = N^{-1}\sum_{m=1}^{N}\nabla_{a_m} Q_{\theta_1}(s_m, a_m)|_{a_m=\pi_\phi(s_m)}\nabla_\phi \pi_\phi(s_m), \tag{3.12}$$

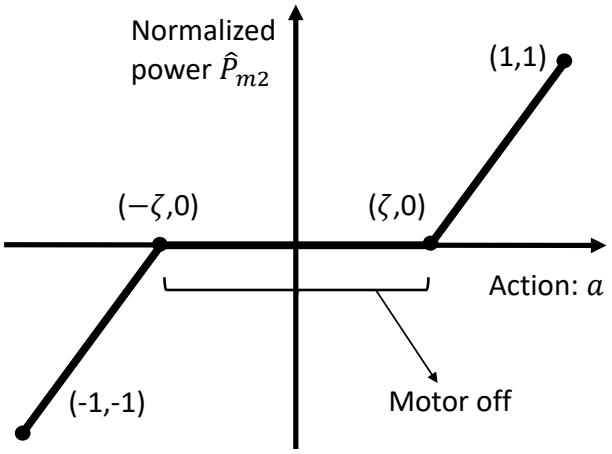

Figure 3.2: Combined control of motor activation and motor power: the relationship between motor normalized power and actor network output.

and the target networks are updated with:

$$\theta_i' \leftarrow \tau\theta_i + (1-\theta_i)\theta_i' \tag{3.13}$$

$$\phi_i' \leftarrow \tau\phi_i + (1-\phi_i)\phi_i' \tag{3.14}$$

3.2.6 Prioritized Experience Replay

The original TD3 algorithm randomly samples experience from the experience buffer at each time step. However, during the training, the critic networks do not fit the Q value of each transition equally well. For a particular experience, the TD error between the original prediction and the TD target is already low; thus, sampling them at a lower rate is acceptable. On the other hand, the TD error for some other experiences is still high and thus should be sampled more frequently. The idea of sampling experience with different priorities is named prioritized experience replay (PER).

In [78], the authors have proposed two variants of PER, proportional prioritization and rank-based prioritization. The authors in [35] have validated that both variants have a computation complexity of $O(log(N))$ and have task-dependent advantages compared with each other. In this study, we adopt the variant of proportional prioritization, where the priority of each transition is established based on the TD-error δ:

$$p_j = |\delta| + \epsilon_p, \tag{3.15}$$

where ϵ_p is a small value that prevents a zero chance of sampling for certain transitions. The chance

of each transition being sampled is given by

$$P(j) = \frac{p_j^\alpha}{\Sigma_k p_k^\alpha} \tag{3.16}$$

The hyper-parameter α balances between greedy search and random search. When α is equal to 0, this PER method becomes a random search. During the sampling, the [0,$\Sigma_k p_k^\alpha$] is divided into k intervals, and one transition is sampled from each interval with the method of sum tree method [79]. The sum tree is a structured binary tree, where each parent node equals the sum of its two children. The weighted priority of each transition p_j^α is stored in a distinct leaf node. The top node contains the information of $\Sigma_k p_k^\alpha$. When any random number is generated within the interval [0,$\Sigma_k p_k^\alpha$], the sum tree allows the agent to search and locate the leaf and transition with the complexity of $O(log(N))$.

The PER method increases the utilization of high-priority transition. However, high utilization also leads to a biased update of the neural network parameters compared with random-sampled mini-batch training. To eliminate the bias caused by high utilization, an importance sampling (IS) weight is introduced to the updated rule:

$$\bar{w_j} = \left(\frac{1}{N}\frac{1}{P(j)}\right)^\beta \tag{3.17}$$

where β determines the bias correction and is usually adjusted from β_0 to 1 as training evolves. The intuition of selecting β_0 is between 0 and 1. The philosophy of $\bar{w_j}$ is to utilize high-priority transition experience more frequently but to contribute less to the update each time. To guarantee the stability of the update, $\bar{w_j}$ is normalized with $\max_j(\bar{w_j})$ between 0 and 1. After simplification, the normalized form of IS weight is

$$w_j = \frac{P(j)^{-\beta}}{\max(P(j)^{-\beta})} = \frac{P(j)^{-\beta}}{\min(P(j))^{-\beta}} \tag{3.18}$$

and the update rule in (3.11) is adjusted as

$$L_{i=1,2} = \frac{w_j}{N}\sum_{m=1}^{N}(y_m - Q_{\theta_i}(s_{m,t}, a_{m,t}))^2, \tag{3.19}$$

During each time step, a new transition will be stored into the experience replay buffer. A high initial priority value p_{init} will be assigned to this transition to guarantee that it will be replayed at least once in the future. When the replay buffer reaches the maximum size k, the earliest experience will be discarded.

3.2.7 Prioritized Exploration

The cost function of the actor network in a traditional TD3 is selected such that the policy π will take an action that maximizes Q at the current state, and, in (3.12), the back-propagation updating rule of ϕ relies on the prediction of Q from the critic networks. However, during the cold start of the first few training episodes, the critic networks may not be able to predict $Q(s, a)$ precisely. Therefore, updating the actor network in the first few episodes may not lead to the optimal solution. There are usually two reasons that critic networks predict Q of a transition poorly:

- The experience of transition (s, a) is replayed few times; hence the estimated Q is not yet converged,
- The estimation of $Q(s, a)$ is sensitive to the action selection; hence, more interaction between agent and environment in a similar state is needed to learn the correct dynamic.

The prioritized experience replay allows the replay buffer to select a transition with bad prediction more frequently, which solves the issue of slow convergence. However, replaying the same transition does not provide more information to those states which are sensitive to the decision of action. For example, in a 2 dimensional state-space, the slope of state A in each direction varies much larger than state B, which means the cost of transition of (s, a) is more sensitive to the selection of action. The difficulty of predicting transition from state A is much larger than from state B. Therefore, to resolve the issue of insufficient exploration, this paper proposes *prioritized exploration* that actively explores the system's region with complex action dynamics.

During the TD target calculation in (3.8), the target action $\tilde{a}$ is perturbed by a random exploration noise ϵ. If a certain state's Q is sensitive to the selection of action, the TD error δ of that transition is likely to be high. Therefore, in prioritized exploration, the cost function of the actor network J is dynamically adjusted such that the actor is encouraged to actively explore the transitions that critic networks do not predict well in the first few epochs. As explained earlier, these bad predictions are mostly due to the current state being highly sensitive to action selection. This active exploration is achieved by including the mean square error between $Q(s, a)$ and the better prediction generated from (3.8) into the cost of actor networks J. After the critic networks capture the dynamics of the environment, the policy shifts its focus to the maximization of Q. The cost function of the actor network in prioritized exploration is designated as

$$\tilde{J} = -\lambda L - (1 - \lambda)Q, \tag{3.20}$$

where λ is a hyper-parameter that starts between $(0, 1)$ and eventually decays to 0. Similar to a traditional TD3, only the first critic network is used to generate the gradient of the actor network. The

actor parameters are then updated with gradient descent through a deterministic policy gradient:

$$\begin{aligned}\nabla_\phi \tilde{J}(\phi) = &\max_i \frac{2\lambda w_j}{N} \sum_{m=1}^{N} (y_m - Q_{\theta_i}(s_{m,t}, a_{m,t})) \nabla_\phi J(\phi) \\ &+ (1-\lambda) \nabla_\phi J(\phi)\end{aligned} \tag{3.21}$$

where

$$\nabla_\phi J(\phi) = N^{-1} \sum_{m=1}^{N} \nabla_{a_m} Q_{\theta_1}(s_m, a_m)|_{a_m = \pi_{\phi(s_m)}} \nabla_\phi \pi_\phi(s_m), \tag{3.22}$$

Note that in (3.21), L and Q may not be in the same order of magnitude. Therefore, the first term needs to be properly scaled. Also, note that in (3.8), the prediction of Q is coupled with current policy π_ϕ. Therefore, the initial λ should be selected such that it does not overly perturb the goal of the actor network. The overall algorithm structure and pseudo code of TD3 with prioritized exploration and experience replay (PEER) are listed as figure 3.1 and Algorithm 0.

3.3 Learning Performance

Three standard test cycles are combined into a training set: the WLTC, the UDDS, and the HWFET. In addition, the agent of TD3-PEER was trained with three different random seeds to prove its robustness under all random initialization. Figure 3.3 (a) shows the accumulative reward of every episode. With different initialization, all three agents show a similar trend on the accumulative reward as episodes evolve.

During the training, the agents focus on exploration for the first few episodes. At this stage, all agents tend to explore actions that critics cannot predict well. The behaviors at this stage may lead to overall bad accumulative rewards; however, they also provide valuable transitions for experience replay, which helps critic networks capture complex dynamics. Figure 3.3 (b) shows the L of each step for a selected agent. With around 25000 steps of prioritized exploration, the L of both critic networks rapidly converged into a small interval for the rest of the training. The convergence of L indicates that both critics capture the dynamics of the environment and can provide a sufficiently precise prediction of $Q(s, a)$ for the deterministic policy gradient. All three agents are able to learn sub-optimal control policies for this P0+P4 HEV architecture, as their accumulative reward stabilized to around -1×10^4, as shown in 3.3 (a).

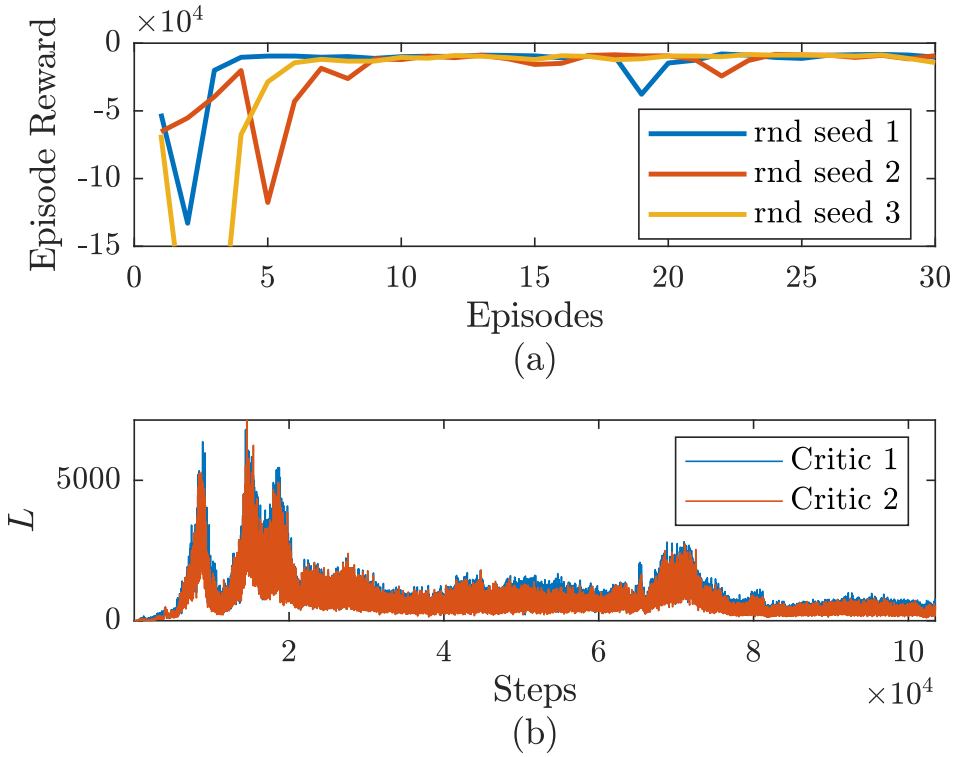

Figure 3.3: (a): Training process of three TD3-PEER agents with different initial random seed. (b): The L of both critic networks of a selected agent.

3.4 Importance Study of Expert Knowledge

Through analyzing the control behavior from the dynamic programming results, the operation trend of the front axle was extracted and can be applied for all the cases of learning-based method study. In this section, the importance of such expert knowledge is evaluated by comparing the the proposed method with and without expert knowledge.

The P0 motor operation should be considered a stand-alone action without considering expert knowledge. This additional action brings the dimensional changes of the actor network and critic networks. The on/off of both motors are still combined with power output optimization, as shown in the table 3.1. After the training is done with three standard test cycles, both cases are validated with two additional cycles: LA92 and US06 cycles.

The well-trained agents under both TD3 algorithms perform a greedy run for all five drive cycles. Their SOC trajectory and the optimal global solution obtained from dynamic programming are listed in figure 3.4. As shown in figure 3.4, Both TD3-PEER managed to learn the capability of regulating the SOC in a charge-sustaining mode. Their battery SOC is maintained in a reasonable bound for all five cycles compared to their initial SOC.

Although the case without expert knowledge successfully learned the charge sustaining control

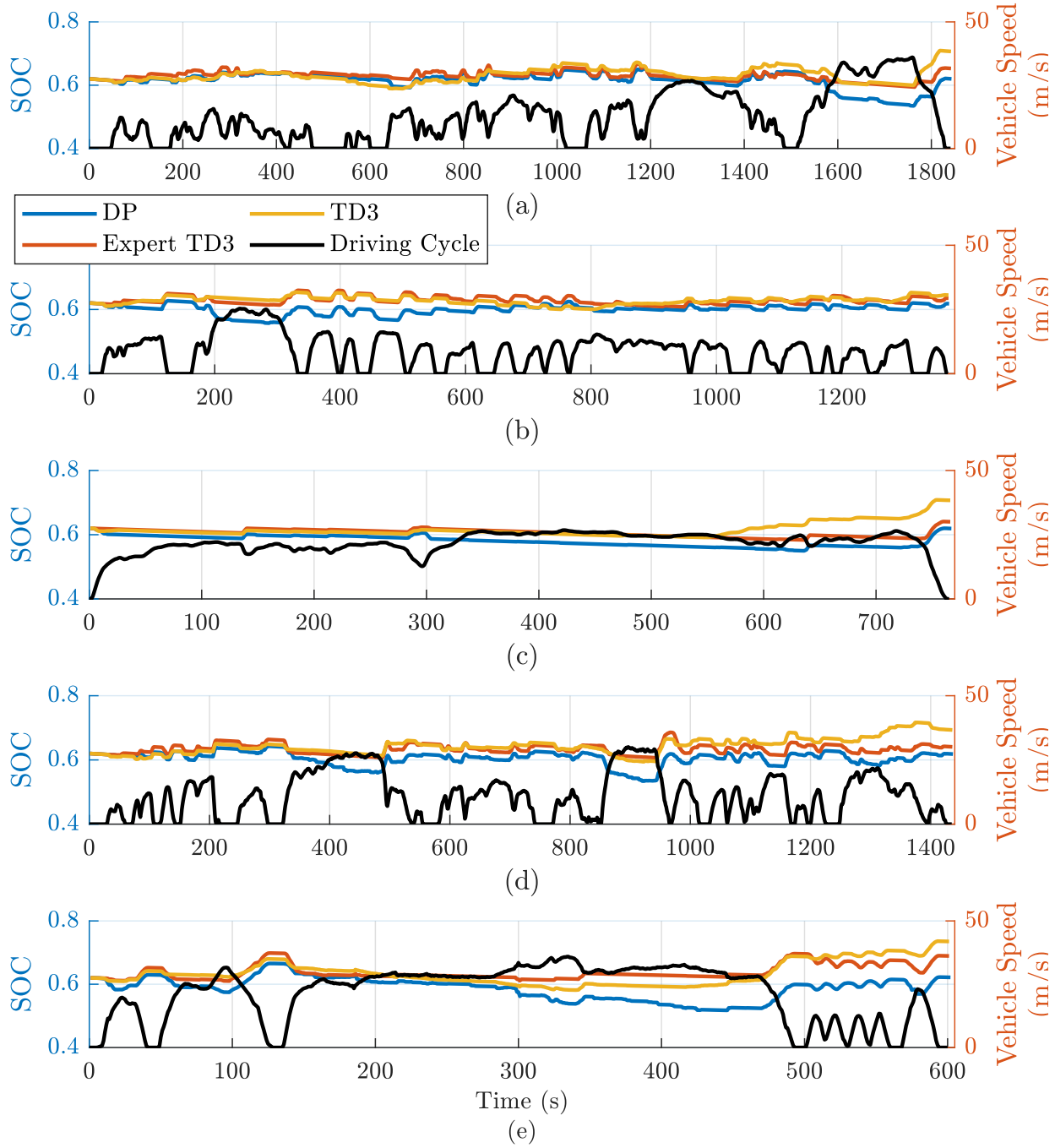

Figure 3.4: SOC trajectory of the DP and the proposed strategy results under the five driving cycles: (a) the WLTC, (b) the UDDS, (c) the HWFET, (d) the LA92 and (e) the US06

policy, the learned operations are obviously different from those suggested by the dynamic pro-

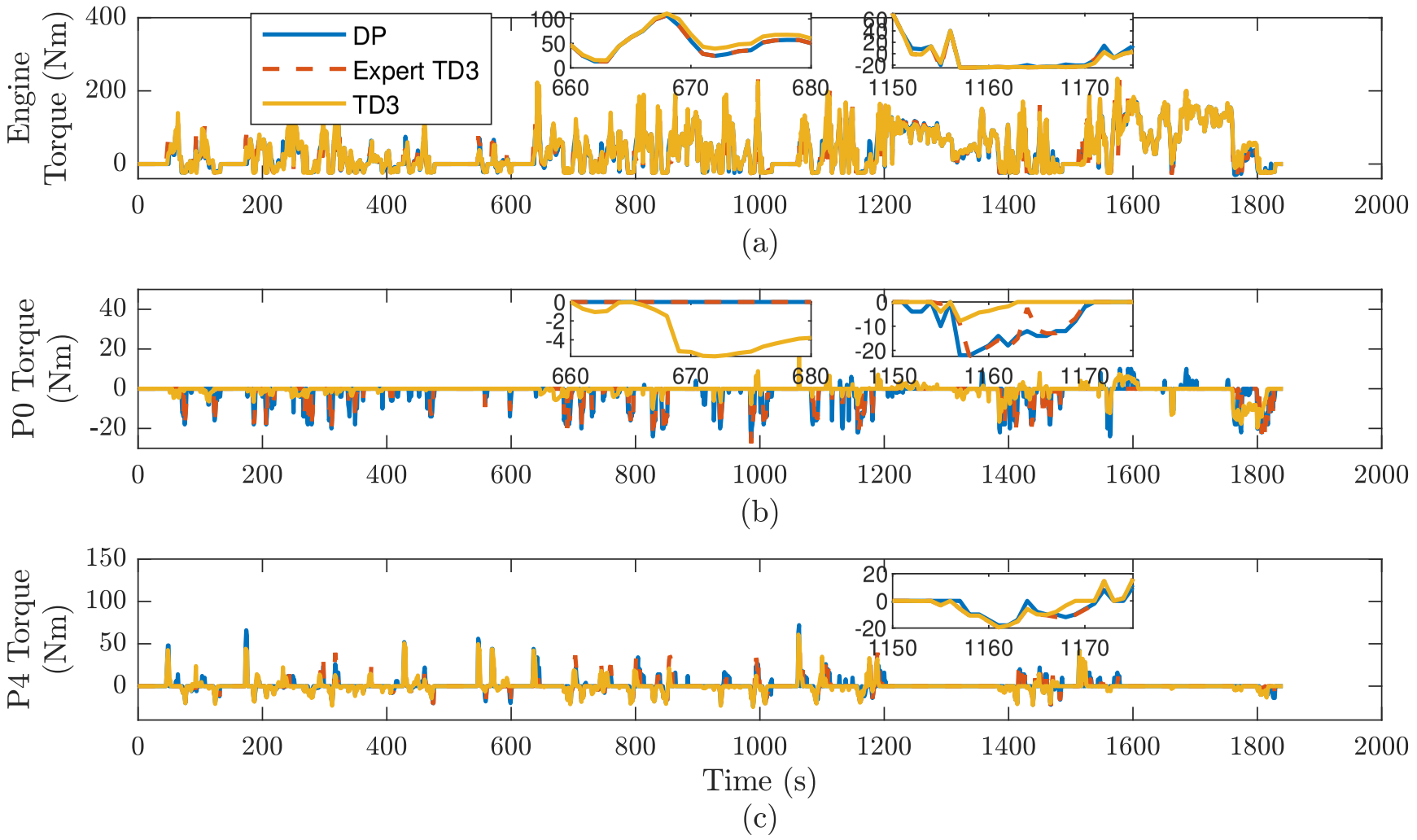

Figure 3.5: Greedy-run of both TD3-PEER agents and the DP results over the WLTC cycle. (a): The engine torque over the time. (b): the P0 motor torque over the time. (c): the P4 motor torque over the time.

gramming. For example, from time 660 seconds to 680 seconds of figure 3.5, The P0 motor of non-expert TD3 is recapturing energy while the engine is generating propulsion torque. As indicated from the same plot, the expert knowledge from dynamic programming does not suggest this double energy conversion as it brings unnecessary energy consumption. From 1150 to 1175 seconds, both Expert TD and the dynamic programming prefer to use P0 and P4 motors to recover energy in a similar trend. However, The non-expert TD3 only learns a sub-optimal solution, which behaves differently to the aforementioned cases. The detailed fuel consumption statistics and SOC deviation for each cycle and each case are listed in table 3.2.

Considering that the terminal SOC of each agent deviates slightly from the initial SOC, the fuel consumption is corrected based on the SOC deviation. The change in fuel in response to the SOC deviation $\frac{\Delta(fuel)}{\Delta(SOC)}$ is obtained through the case study from chapter 2 and our previous work about a 48V P0+P4 MHEV [64].

3.5 Comparison with Other Learning-based Methods

The proposed TD3 method is then compared with state-of-the-art learning-based methods, namely, DDPG+PER [29], and DQN [80]. Since these two methods were developed for single-motor HEVs, expert knowledge is also interposed to reduce the number of control variables. The simulation results of all energy management strategies among five cycles with expert knowledge are listed in table 3.3. And, the SOC trajectory of each method during each cycle is labeled in figure 3.6. Although the state-of-art methods also learned the charge sustaining EMS after the training, they achieve a different level of optimality compared to the proposed method.

The DQN suffers from the discretized action space, hence struggling with learning a near-optimal control policy. Table 3.3 indicates that the proposed TD3-PEER outperforms DQN by reducing fuel consumption by 3.27% and 4.44% over the training and validation cycles, respectively. Proposed by [29], the continuous action space of DDPG-PER allows the actor network to

Table 3.1: States and action defines in the proposed method.

state (S)	action (A)
SOC	P_{m1} with on/off
gear	P_{m2} with on/off
v_e	
acceleration	
r_{tp}	

Table 3.2: Fuel consumption comparison between DP, the proposed TD3 algorithm with and without expert knowledge.

Type	Driving cycle	Fuel consumption (kg)			SOC deviation (%)		Corrected fuel consumption	
		DP	Expert TD3	TD3	Expert TD3	TD3	Expert TD3	TD3
Training	WLTC	0.9104	0.9394 (+3.18%)	0.9770 (+7.31%)	+3.3	8.7	0.9339 (+2.58%)	0.9622 (+5.69%)
Training	UDDS	0.4069	0.4254 (+4.55%)	0.4534 (+11.4%)	1.5	0.5	0.4229 (+3.93%)	0.4493 (+10.42%)
Training	HWFET	0.5649	0.5706 (+1.01%)	0.5882 (+4.12%)	2.0	8.7	0.5659 (+0.18%)	0.5679 (+0.53%)
Validation	LA92	0.6403	0.6710 (+4.79%)	0.7054 (+10.16%)	2.1	7.3	0.6677 (+4.28%)	0.6940 (+8.39%)
Validation	US06	0.6337	0.6679 (+5.40%)	0.6854 (+8.16%)	6.9	11.5	0.6556 (+3.46%)	0.6650 (+3.36%)

Table 3.3: Fuel consumption comparison the proposed method and two state of art method: DDPG-PER and DQN.

Type	Driving cycle	Corrected fuel consumption (kg)		
		TD3-PEER	DDPG-PER	DQN
Tra.	WLTC	0.9339	0.9493(+1.65%)	0.9633(+3.15%)
Tra.	UDDS	0.4229	0.4368(+3.29%)	0.4426(+4.66%)
Tra.	HWFET	0.5659	0.5672(+0.23%)	0.5773(+2.01%)
Val.	LA92	0.6677	0.6906(+3.43%)	0.7024(+5.20%)
Val.	US06	0.6556	0.6748(+2.93%)	0.6797(+3.68%)

better fit the optimal control policy. Prioritized experience replay also improves sample utilization, speeding up the training process. With the method of DDPG-PER, the agent spends roughly 1.72% and 3.18% more fuels over the training and validation cycles as compared with the proposed TD3-PEER.

Benefiting from the dual critic networks setup, the agent of TD3-PEER is less likely to be distracted by overestimated actions during the training. The trick of delayed policy updates and gradient clip improves the stability of the training. The prioritized exploration mechanism encourages the actor to actively collect low-confidence transitions for the critics. Then, the critic networks can provide a more precise gradient for policy training. As shown in Fig. 3.7 (b) and (c), the P0 and P4 motor operation of TD3-PEER from 182 to 190 seconds is much closer to the DP's operation, as compared with the DQN and DDPG-PER. In Fig. 3.7 (c), the P4 motor operation from 1150 to 1170 seconds also demonstrates that the TD3-PEER agent has already learned a near-optimal policy. Overall, the agent of TD3-PEER achieves fuel consumption closest to the DP results in

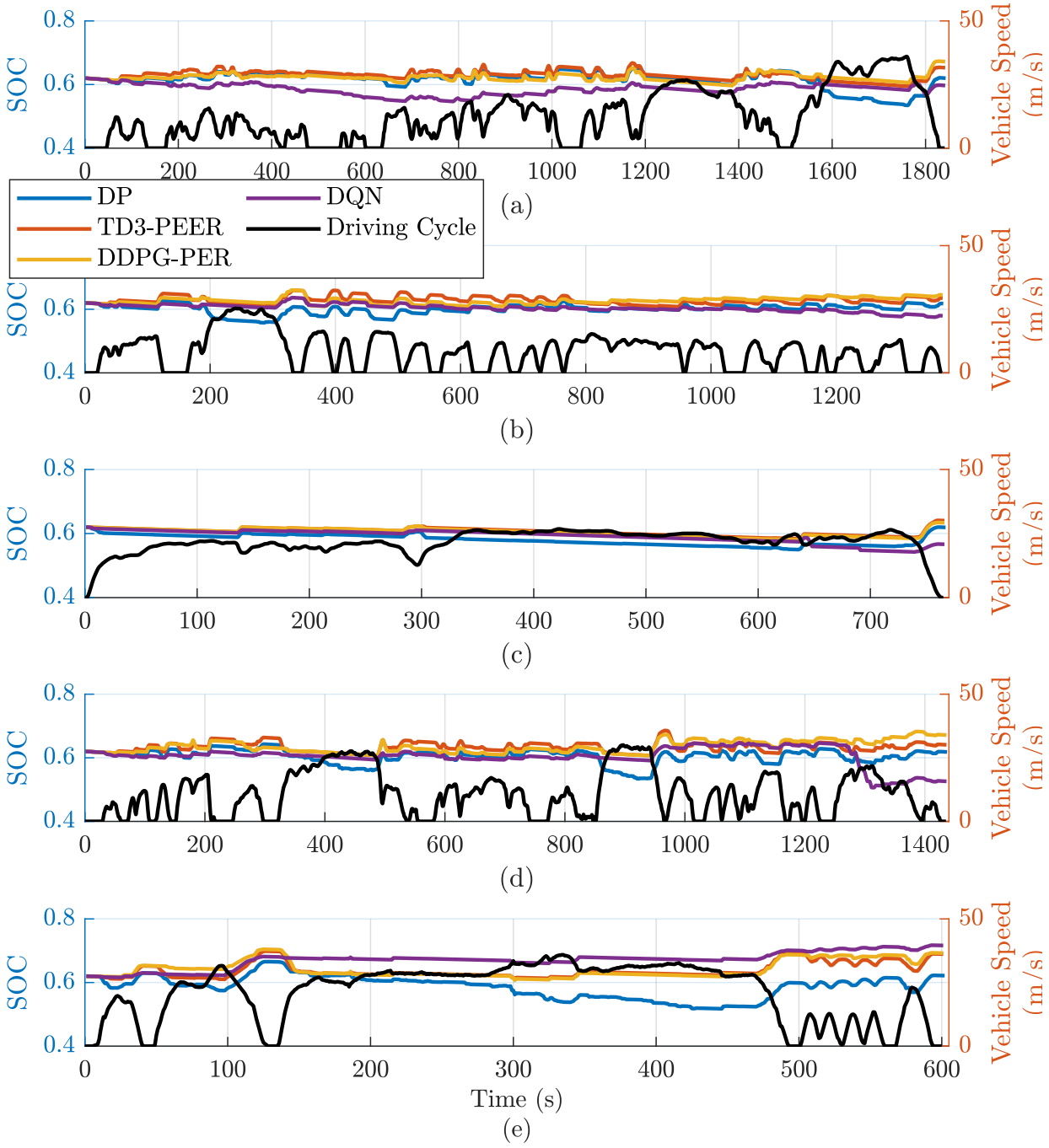

Figure 3.6: SOC trajectories of DP and several learning-based methods results under the five driving cycles: (a) the WLTC, (b) the UDDS, (c) the HWFET, (d) the LA92 and (e) the US06

both the training and the validation cycles among all three methods.

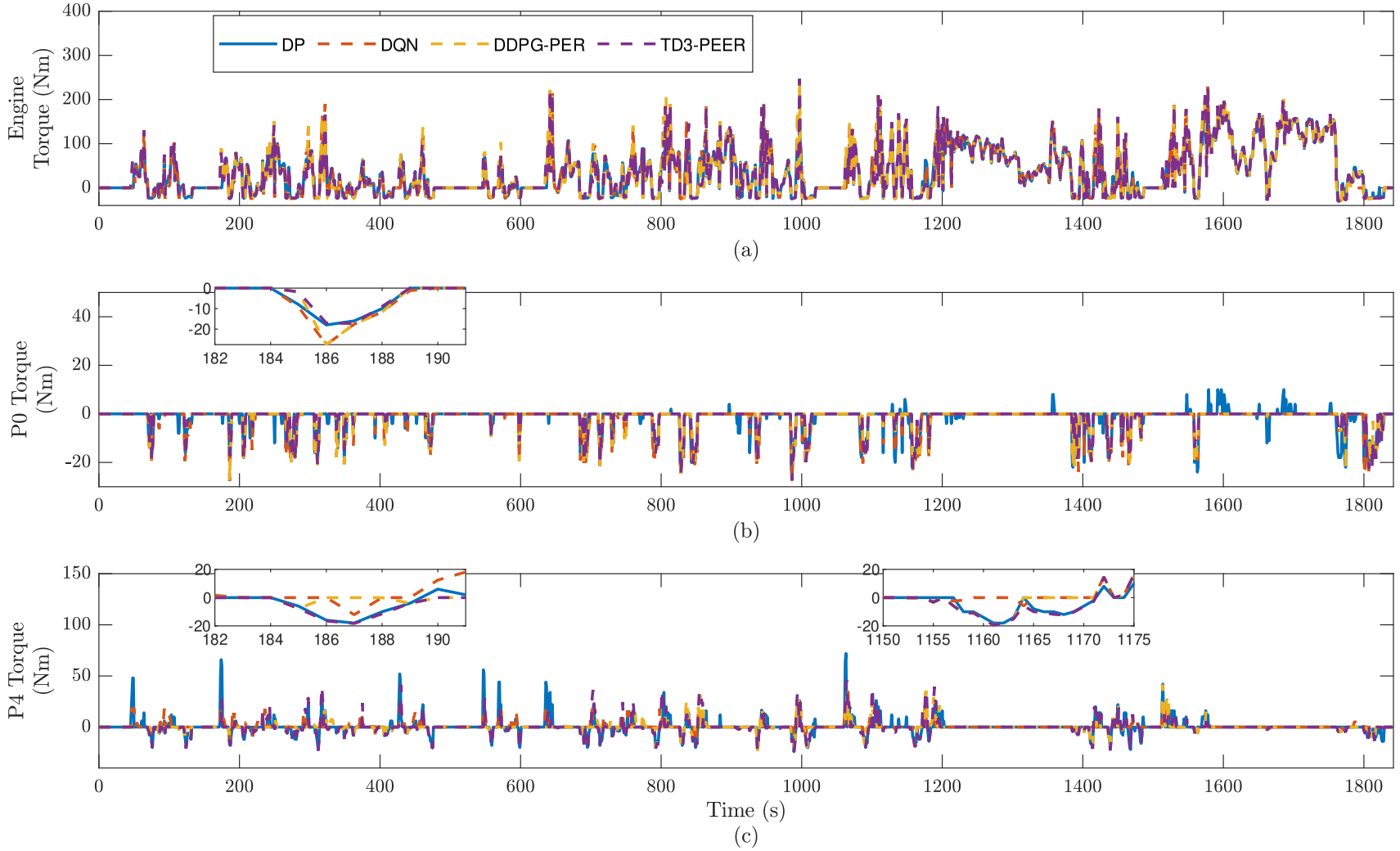

Figure 3.7: Greedy run of learning-based methods and the DP results over the WLTC cycle. (a): The engine torque over the time. (b): The P0 motor torque over the time. (c): The P4 motor torque over the time.

3.6 Summary

This chapter presents a novel energy management strategy for P0+P4 HEVs that is based on an expert twin-delayed deep deterministic policy gradient with prioritized exploration and experience replay (TD3-PEER). To address the issue that the critic network in the state-of-the-art TD3 may struggle with predicting Q value, this paper proposes prioritized exploration that encourages the agent to visit action-sensitive states more frequently. The proposed algorithm is tested and validated on a P0+P4 HEV model, including consideration of realistic operational constraints such as nonlinear tire effects, braking force distribution for safety, and motor/engine limitations are considered. To simplify the control design, the P4 motor's on/off control and the power control are condensed into a single variable by introducing a motor activation threshold into the final layer of the agent's actor. In addition, the dynamic programming results are incorporated into the training of TD3, helping the agent avoid inefficient operations. The results from the case study with random seeds show that the method is well stabilized during training, and all agents converge to a similar level of optimal solutions after around 10 episodes. With expert knowledge known for all methods, the proposed TD3-PEER outperforms DDPG-PER and DQN, reducing fuel consumption over the training and validation sets by an average of 2.3% and 3.74%, respectively.

CHAPTER 4

Defensive Ecological Adaptive Cruise Control Considering Neighboring Vehicles' Blind-spot Zones

4.1 Introduction

Aside from the torque-split optimization described in chapter 2 and chapter 3, optimizing a vehicle's longitudinal motion also possesses a huge impact on fuel economy, safety and driving comfort to the P0+P4 MHEV. Existing literature [43] have developed an adaptive cruise controller (ACC) with advanced V2V/V2I technology that reduces driver's efforts during highway scenario and lead to 17% fuel consumption reduction from traditional cruise control. However, no existing literature considers the potential threat from the adjacent lane. Dwelling within the blind spot of a neighboring vehicle blind spot can increase the risk of lane-change-collision. In this chapter, a defensive eco-logical adaptive cruise control (DEco-ACC) method for P0+P4 MHEV, which is capable of predicting and avoiding neighboring vehicles' blind spots, is proposed. The main contribution of this chapter is twofold:

1. The BSZs of the vehicles in adjacent lanes are considered and described mathematically to develop a DEco-ACC algorithm.

2. MPC is exploited to systematically implement state constraints related to the neighboring vehicles' BSZs and to reduce dwelling time in the BSZs.

The remainder of this chapter is organized as follows: Section 4.2 briefs the definition and computation of the BSZs of neighboring vehicles to design constraints used in the considered control problem. Section 4.3 provides the formulation of the proposed DEco-ACC based on model predictive control. Section 4.4 presents simulation results to demonstrate the performance of the proposed controller in comparison with a traditional ACC and Eco-ACC from other literature. Finally, a summary and directions for future work are presented in Section 4.5.

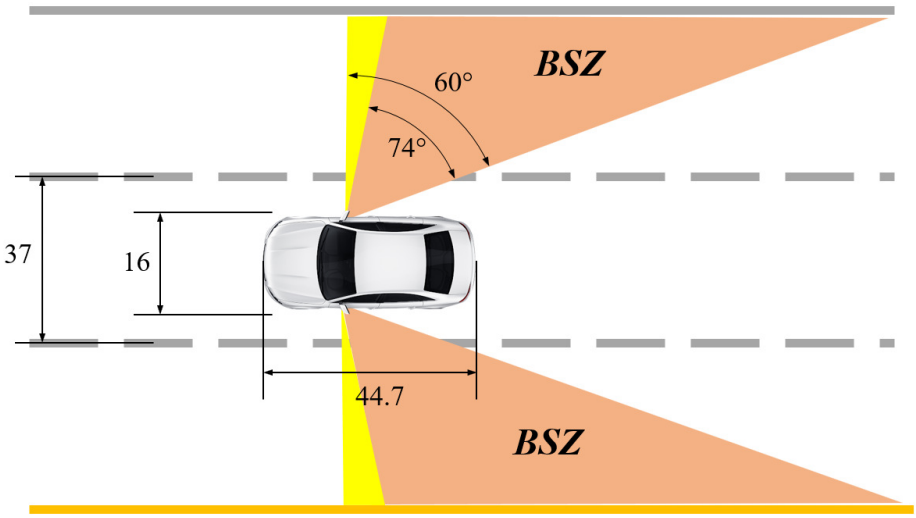

Figure 4.1: An example diagram of blind spot zones of a sedan in orange color; visible region by head tilt in yellow

4.2 Driving Conditions Based on Blind Spot Zone

This section briefly describes the approach used to compute the blind spot zone (BSZ) of a vehicle, and discusses driving conditions based on a neighboring vehicle' BSZ to be used in the DEco-ACC later.

4.2.1 Computation of Blind Spot Zone

As per the seating position of the driver in a vehicle, there is a field of view (FOV), which is defined as the total 360 degree span of area that is visible to the driver directly or indirectly around his or her seating position. The FOV of the driver can be determined based on the ray method [81, 82] in consideration of head, neck, and eye movements as well as mirror fields described in the SAE J941, SAE J1050, and FMVSS 111 standards [83–85].

Based on the driver's seating position, the direct view includes the portion that is visible to the driver through daylight openings (DLO) such as the windshield glass. Two important areas are presented in Fig. 4.1. The yellow region is the area visible by head tilt. The portion that cannot be seen by the driver through side view mirrors is represented by the orange color. To be conservative in control design, the blind spot zone is defined as a union of the yellow and orange regions, and the blind spot angles of sample vehicles are provided in Table 4.1. This information will be used in the vehicle simulation presented in Section 4.4.

Table 4.1: Blind spot angle of sample vehicles obtained by ray method

Vehicle Class	Length [m]	Breadth [m]	Blind Spot Angle [°]
Compact Car	3.57	1.6	73
Mid-size Sedan	4.47	1.6	74
Minivan	5.17	2.0	76

Table 4.2: The symbols and corresponding definitions used in this paper

Symbol	Definition
L_{BSZ}	Blind spot zone's length
L_E	Ego vehicle's bumper-to-bumper distance
L_N	Neighboring vehicle's rear bumper to a side mirror distance
a	Ego vehicle's acceleration
Δa	Ego vehicle's acceleration change
v_e	Ego vehicle's speed
v_n	Neighboring vehicle's speed
v_p	Preceding vehicle's speed
y_e	Ego vehicle's displacement
y_n	Neighboring vehicle's displacement
y_p	Preceding vehicle's displacement
$Y^{\Delta NE}$	$y_n - y_e$
$Y^{\Delta PE}$	$y_p - y_e$
$Y^{\Delta PN}$	$y_p - y_n$

4.2.2 Blind Spots Formulation

The ultimate goal of this study is to develop an ACC algorithm that can help an ego vehicle avoid BSZs of its neighboring vehicles as much as possible during eco-driving. Thus, distance/velocity information about the preceding and neighboring vehicles is required. It is assumed that this information is available via sensors and/or V2V communication so that the distance from the preceding vehicle and the location of the neighboring vehicles' BSZs can be easily determined, as shown in Fig. 4.2. The symbols used in this paper are summarized in Table 4.2.

In a conventional adaptive cruising mode, the distance between the ego vehicle and the preced-

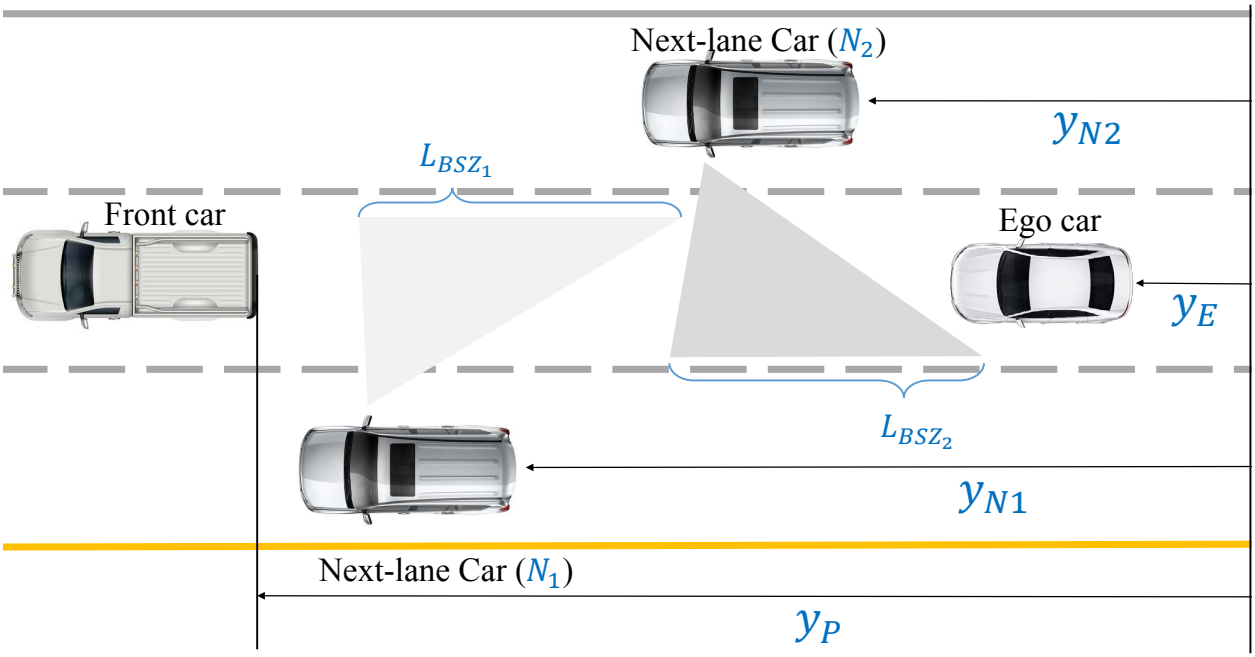

Figure 4.2: A concept of car-following in consideration of the BSZs of neighboring vehicles

ing vehicle $Y^{\Delta PE}$ should be within a proper range:

$$Y_{\min}^{\Delta PE}(t) \leq Y^{\Delta PE} \leq Y_{\max}^{\Delta PE}(t), \tag{4.1}$$

where $Y_{\min}^{\Delta PE}$ and $Y_{\max}^{\Delta PE}$ are the minimum safety distance and the maximum comfort distance, respectively. The minimum safety distance to the preceding vehicle is formulated as a constant time headway policy [40],

$$Y_{\min}^{\Delta PE}(t) = Y_{\min,0}^{\Delta PE} + t_h v_e, \tag{4.2}$$

where $Y_{\min}^{\Delta PE}$ consists of the constant time-gap t_h, the ego vehicle's speed v_e, and the constant distance headway to the preceding vehicle $Y_{\min,0}^{\Delta PE}$. A constant distance d_{com} is introduced to determine $Y_{\max}^{\Delta PE}$ for comfortable following as well as for preventing a neighboring vehicle's cutting in [86]:

$$Y_{\max}^{\Delta PE}(t) = Y_{\min}^{\Delta PE}(t) + d_{com}. \tag{4.3}$$

The spacing policy introduced in (4.2) and (4.3) provides the ego vehicle with room for minimizing both fuel consumption and dwelling time in the neighboring vehicles' blind spots.

For simplicity, let us consider a single neighboring vehicle case. To avoid entering the BSZ, the

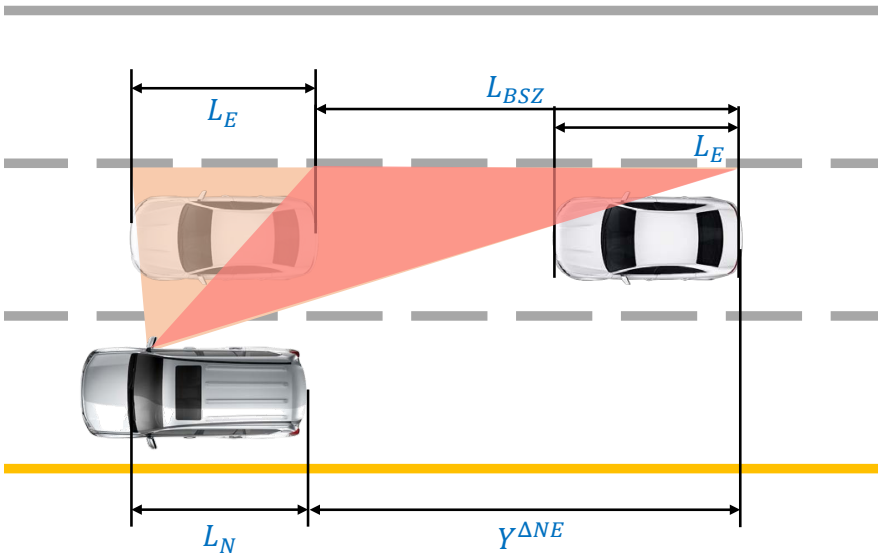

Figure 4.3: Graphical demonstration of the constraints to avoid the BSZ of the neighboring vehicle.

ego vehicle should satisfy the following two constraints:

$$L_E + \left(-Y^{\Delta NE}\right) \geq L_N, \tag{4.4}$$

$$L_{BSZ} + L_E \leq L_N + Y^{\Delta NE}. \tag{4.5}$$

As shown in Fig. 4.3, constraint (4.4) requires that the ego vehicle stays in front of the neighboring vehicle's BSZ; on the other hand, constraint (4.5) requires that the ego vehicle stays behind the BSZ of the neighboring vehicle. Since these constraints (4.4) and (4.5) cannot be satisfied at the same time, the midpoint of the actual region in which the ego vehicle is completely hidden is introduced as follows:

$$l = -L_N + L_E + \frac{L_{BSZ}}{2}. \tag{4.6}$$

Then, constraints (4.4) and (4.5) can be given by

$$|Y^{\Delta NE*}| \geq \frac{1}{2} L_{BSZ} \tag{4.7}$$

where

$$Y^{\Delta NE*} = Y^{\Delta NE} - l. \tag{4.8}$$

Constraint (4.7) will be penalized with a slack variable to resolve feasibility issues when the ego vehicle needs to pass the blind spots of the neighboring vehicles.

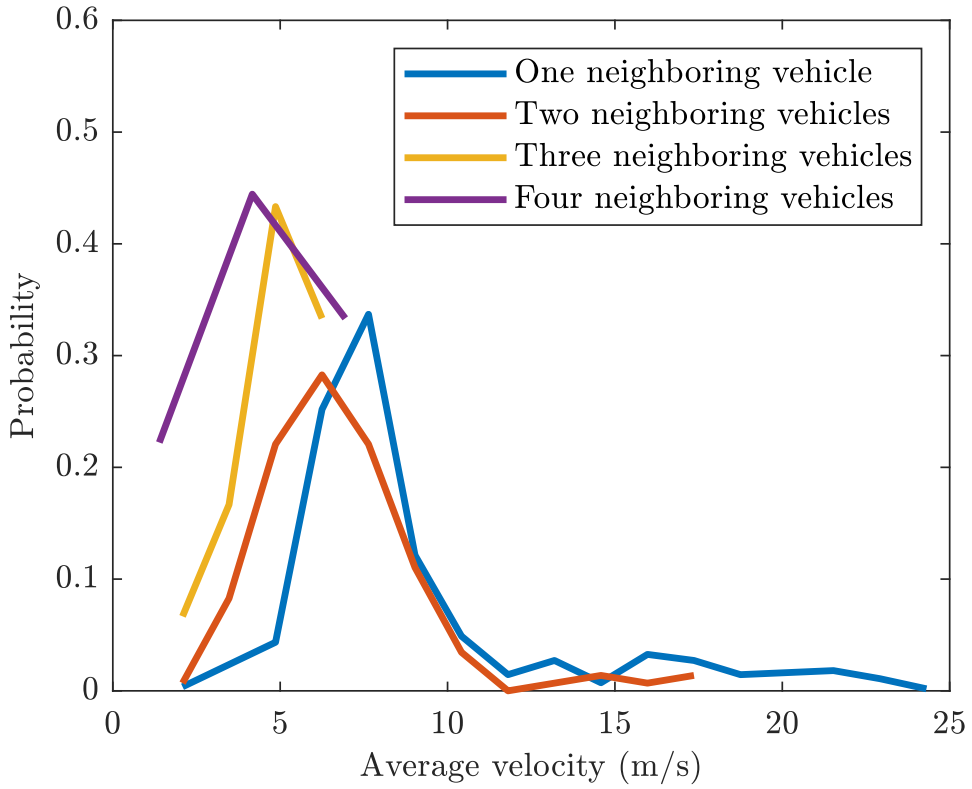

Figure 4.4: The average occurrence probability of NVs scenarios using 2403 vehicles from NGSIM data.

4.2.3 N-many Neighboring Vehicles Scenarios

During car-following, the number of neighboring vehicles near the ego vehicle varies depending on traffic conditions; especially, one of the factors is vehicle speed.

To determine the most probable car-following scenario, we have analyzed 2403 human-driven driving data collected through the Next Generation Simulation (NGSIM) project [87]. This NGSIM data has been widely used in various traffic simulation studies [88–91].

The N neighboring vehicles (N-NVs) scenarios are defined as follows: for instance, when there exists a neighboring vehicle and the midpoint of its blind spot defined in equation (4.8) is closer to the ego vehicle than its blind spot length, this case is classified as 1-NV scenario. When there are two, three, or four neighboring vehicles that satisfy this condition simultaneously, this scenario is classified as a 2-NVs, 3-NVs, or 4-NVs scenario, respectively.

Figure 4.4 shows the average probability of N-NVs scenario with $1 \leq N \leq 4$ for all 2403 vehicles over their total trip. It is found that the cases with more than 4 neighboring vehicles do not exist in the NGSIM data and hence it is assumed that 4-NVs scenario is sufficient to cover a

daily highway driving scenario. As it can be seen, 3-NVs and 4-NVs scenarios are not observed when the speed of the ego vehicle is above 8 m/s, meaning that a 2-NVs scenario could be sufficient to cover normal (high-speed) highway driving, which is the main focus of this paper. It is noted that the formulation of the Deco-ACC for 4-NVs scenarios and simulation results are presented in the Appendix.

4.3 MPC Formulation for DEco-ACC

This section describes the vehicle dynamics of interest and the mathematical formulation of the proposed DEco-ACC based on model predictive control. It is noted that the control problem is formulated for the case of 2-NVs scenario based on the observation in Section 4.2.3.

4.3.1 Modeling of Vehicle Longitudinal Dynamics

In this study, the ego vehicle is assumed to drive on a single lane, and hence only longitudinal dynamics are considered. The vehicle dynamics are expressed in the discrete-time domain with a sampling time of T_s. As mentioned before, the distance/velocity information about the preceding and neighboring vehicles is assumed to be available via sensors and/or V2V communication. Thus, the relative distances to the two nearest neighboring vehicles (2-NVs scenario) in adjacent lanes are considered at the same time as given by,

$$x_{k+1} = Ax_k + Bu_k + d_k, \tag{4.9}$$

with

$$A = \begin{bmatrix} 1 & 0 & 0 & -T_s & -\frac{1}{2}T_s^2 \\ 0 & 1 & 0 & -T_s & -\frac{1}{2}T_s^2 \\ 0 & 0 & 1 & -T_s & -\frac{1}{2}T_s^2 \\ 0 & 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.10}$$

$$B^T = \begin{bmatrix} -\frac{T_s^3}{6} & -\frac{T_s^3}{6} & -\frac{T_s^3}{6} & \frac{T_s^2}{2} & T_s \end{bmatrix}^T, \tag{4.11}$$

$$d_k^T = \begin{bmatrix} T_s v_p & T_s v_{n1} & T_s v_{n2} & 0 & 0 \end{bmatrix}^T, \tag{4.12}$$

where the state and control vectors are defined by

$$x_k = \begin{bmatrix} Y^{\Delta PE} \\ Y^{\Delta NE1*} \\ Y^{\Delta NE2*} \\ v_e \\ a_e \end{bmatrix}_k, \; u_k = \dot{a}_k. \tag{4.13}$$

Note that term (4.12) includes velocity information about the neighboring vehicles and the preceding vehicle; thus, in MPC, future velocity and displacement are treated as known disturbances. It is also noted that the jerk $\dot{a}_k$ is used as a control input to directly penalize for ride comfort and to achieve zero-offset tracking performance.

4.3.2 Optimal Control Problem Formulation

For optimally controlling the ego vehicle's speed with consideration of safety, ride comfort, and energy efficiency over the prediction horizon, the following cost function to be minimized at each time instant is considered:

$$\begin{aligned} J = \sum_{k=0}^{N-1} & u_k^2 P_1 + (a_{e,k})^2 P_2 + (v_{e,k} - v_{p,k})^2 P_3 \\ & + (\delta_{slack,1} \times mode_1 + \delta_{slack,2} \times mode_2) P_4 \end{aligned} \tag{4.14}$$

where P_i's are weighting factors for penalizing jerk, acceleration, velocity difference to the preceding vehicle, and the ego vehicle's dwelling in BSZs, respectively. In (4.14), δ_{slack1} and δ_{slack2} are slack variables for implementing constraint (4.7) as two soft constraints for two neighboring vehicles and are defined as follows:

$$\delta_{slack,i} = \cos\left(\max\left\{\min\left(\frac{2Y^{\Delta NE^*,i}\pi}{L_{BSZ,i}}, \pi\right), -\pi\right\}\right) \tag{4.15}$$

In particular, $mode_1$ and $mode_2$ are boolean signals that are defined as follows: if the neighboring vehicle i approaches from behind, $mode_i$ is set to be 0; if the neighboring vehicle i approaches from the front, $mode_i$ is set to be 1. When a neighboring vehicle approaches the ego vehicle from behind, it is unlikely that the neighboring vehicle will make a lane change toward the ego vehicle causing any danger because the neighboring vehicle's driver can see the ego vehicle through DLO. Thus, at each time instant, the controller determines whether to turn on/off the penalty on a neighboring vehicle's BSZ using $mode_1$ and $mode_2$. Note that $mode_i$ is fixed as the same value as the current one in the prediction horizon.

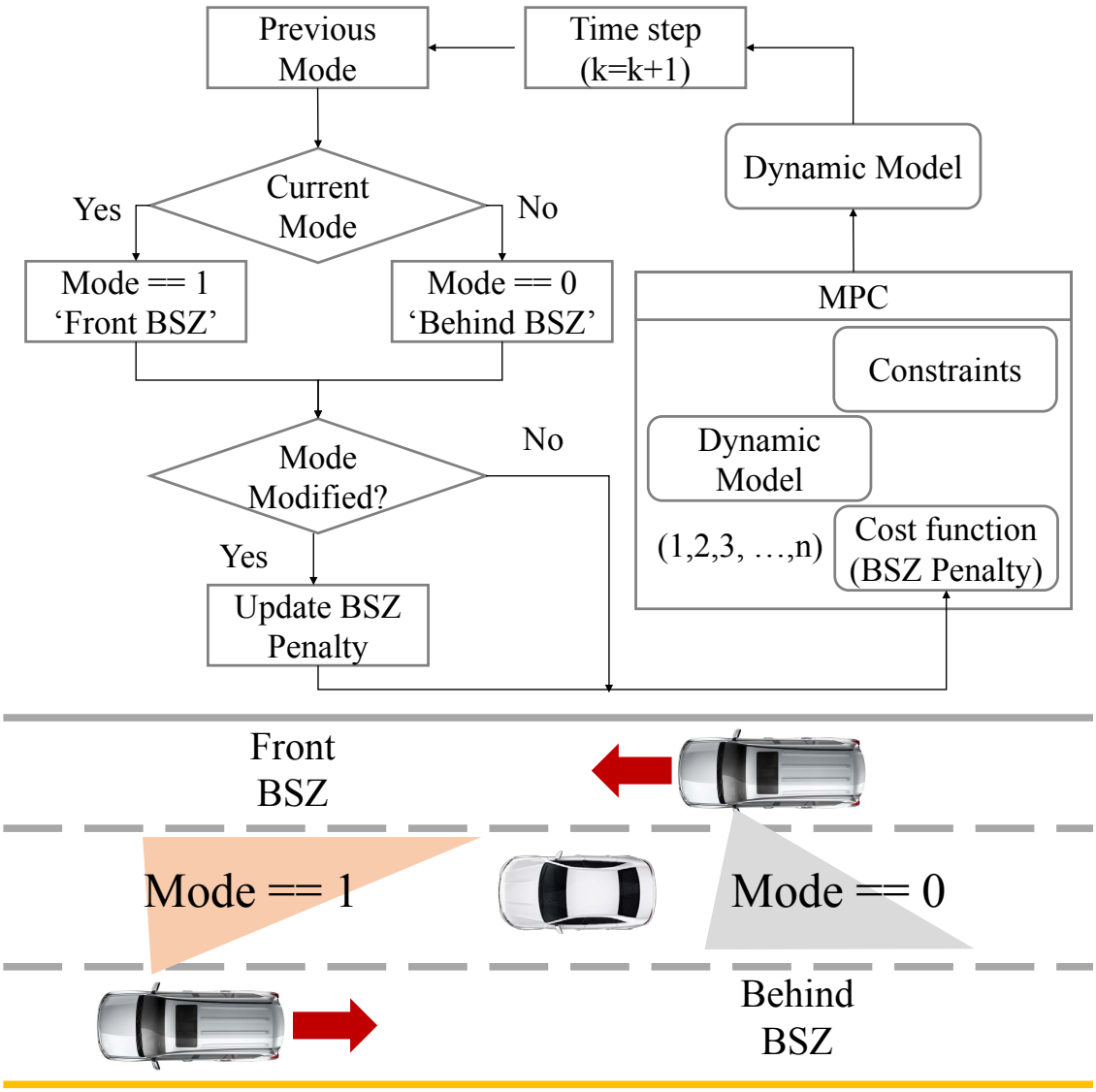

Figure 4.5: The proposed algorithm that determines when to activate penalty of blind spots.

As shown in Fig. 4.6, the distance of L_{BSZ} about its midpoint is mapped into $[-\pi, \pi]$. When the ego vehicle enters the BSZ of the neighboring vehicle, the term becomes a positive value; otherwise, it is zero. Moreover, this function is continuous and one-time differentiable.

State and input constraints for the DEco-ACC formulation are summarized as follows:

$$Y_{\min,k}^{\Delta PE} \leq Y_k^{\Delta PE} \leq Y_{\max,k}^{\Delta PE} \tag{4.16}$$

$$v_{e,\min} \leq v_{e,k} \leq v_{e,\max} \tag{4.17}$$

$$a_{e,\min} \leq a_{e,k} \leq a_{e,\max} \tag{4.18}$$

$$u_{\min} \leq u_k \leq u_{\max}. \tag{4.19}$$

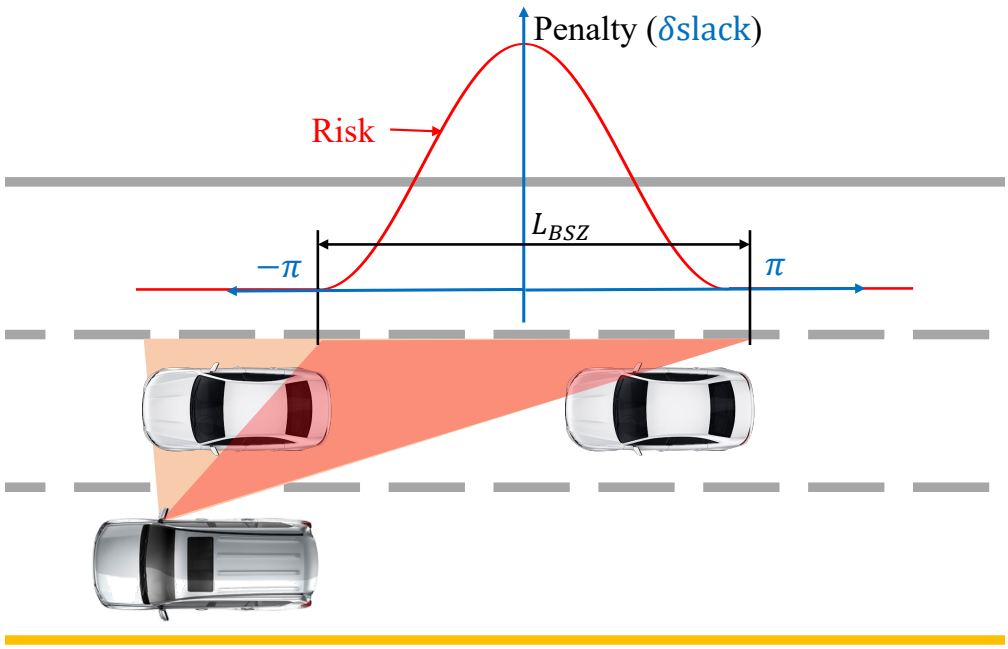

Figure 4.6: Concept diagram of the penalty function for the normalized BSZ, which will be used to formulate the slack variable when the ego vehicle enters the blind spot.

The formulated optimization problem is solved with MPC solver CasADi [92] via the mpctools interface [93] in the Matlab environment.

4.4 Simulation and Results

This section presents the simulation results to demonstrate the effectiveness of the proposed DEco-ACC approach compared to that of baseline Eco-ACC strategy: Eco-ACC is formulated as an acceleration minimization problem, a surrogate optimization problem of energy minimization.

4.4.1 Simulation Setup

In the simulation, two neighboring vehicles and one preceding vehicle are considered. The ego vehicle performs adaptive cruising behind the preceding vehicle with both ACC strategies while it also seeks to minimize the dwelling time between two neighboring vehicles as well with DEco-ACC.

As shown in Fig. 4.7, the vehicle located at the foremost part of the middle lane represents the preceding vehicle, and the vehicle behind it represents the ego vehicle. Two neighboring vehicles

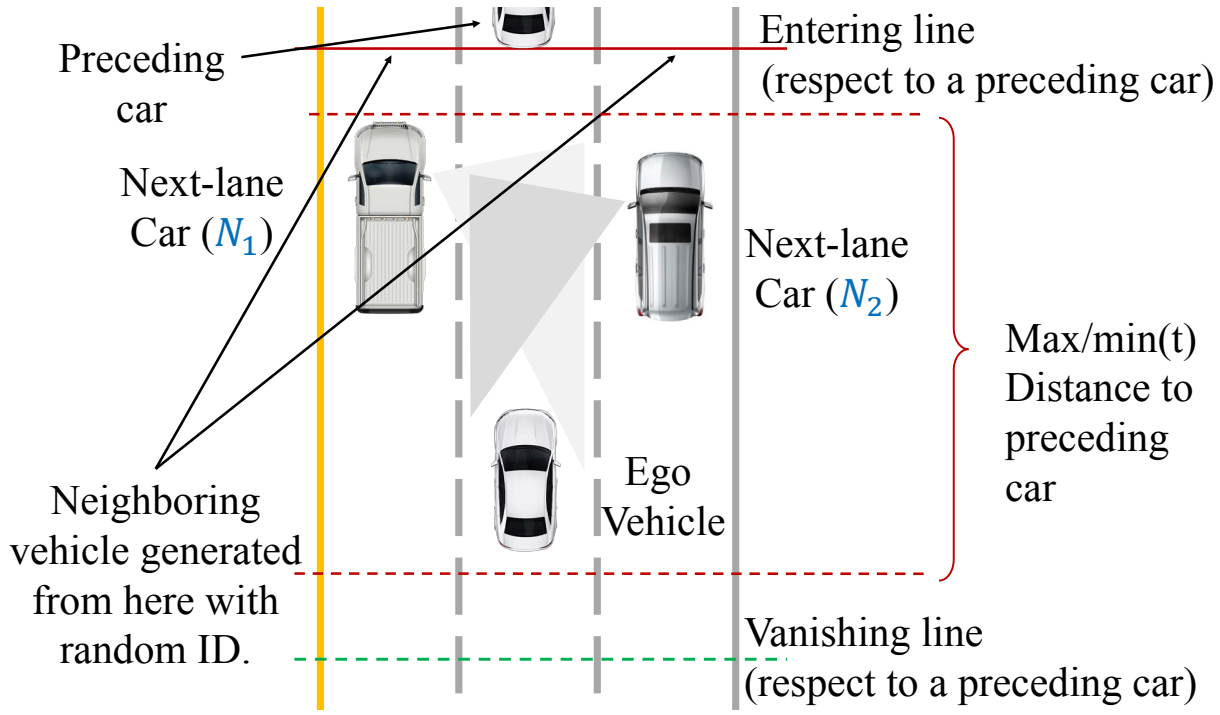

Figure 4.7: Car-following simulation setup for a 2-NVs scenario.

are located in adjacent lanes. Suppose a neighboring vehicle has a larger average speed than the preceding vehicle. In that case, this neighboring vehicle will catch up with the ego vehicle from the rear and drive away to the front of the ego vehicle. Therefore, the neighboring vehicle driver can notice the ego vehicle from the front windshield during driving, and the DEco-ACC is not necessary in this case. For this reason, the neighboring vehicle speed profiles are selected such that all neighboring vehicles are generally slower than the preceding vehicle in terms of average speed. There are two lines defined in Fig. 4.7: an entering line (solid red) and a vanishing line (green dashed) of neighboring vehicles with respect to the preceding vehicle. Once a neighboring vehicle falls behind the vanishing line, a new neighboring vehicle that is randomly chosen from neighboring vehicle candidates will be assigned at the entering line. Two red dashed lines in Fig. 4.7 represent the lower bound, $Y_{\min}^{\Delta PE}$, and upper bound, $Y_{\max}^{\Delta PE}$, of $Y^{\Delta PE}$, respectively. As discussed earlier, the $Y_{\min}^{\Delta PE}$ in (4.2) changes dynamically with t_h and v_e.

The speed profiles of the considered neighboring vehicles candidates and the preceding vehicle are obtained from the data collected through the Next Generation Simulation (NGSIM) project [87]. Among all 2403 vehicles from the NGSIM data, 21 vehicles are selected based on their average speed values. The detailed information of the preceding and neighboring vehicles are listed in Table 4.3.

In vehicle simulation, a sampling time of $T_s = 0.5$ second is used, and the prediction horizon N

Table 4.3: Statistics of the preceding vehicle (ID: 0) and neighboring vehicles (ID: 1-20) speed trajectories candidates used in this study, units are in [m/s].

Vehicle ID	Average speed	Max. speed	Min. speed
0	23.9	26.8	21.4
1	23.7	28.3	17.8
2	23.6	26.6	20.9
3	23.7	29.0	21.0
4	23.2	26.4	16.3
5	23.6	27.7	20.9
6	23.2	27.1	20.2
7	22.9	25.5	21.4
8	22.3	28.6	16.8
9	22.7	28.7	20.1
10	21.4	27.8	15.8
11	22.2	25.0	18.5
12	21.3	24.8	18.7
13	21.8	25.8	18.8
14	21.5	25.1	19.1
15	20.1	27.1	15.2
16	20.3	26.1	17.4
17	20.3	29.0	0.7
18	20.1	24.0	16.3
19	19.5	27.6	1.2
20	22.5	27.0	19.8

is set to be 20, which corresponds to 10 seconds; that is, the accurate speed profiles of the preceding and neighboring vehicles are available for the next 10 seconds.[1] The car length of each vehicle is assumed to be 4.5 m for simplicity, and hence the BSZ values of all vehicles are the same. The road width and the blind spot angle for determining the length of BSZ are set to be 3.7 m and 74°, respectively. The road width is a typical value considered in the United States. Thus, the projection of the blind spot zone of the neighboring vehicle to the ego vehicle lane is $3.7\tan(74^\circ) \approx 12.9$ meters, resulting in the BSZ length of 8.4 meter. The ego vehicle can be seen by the neighboring vehicle as long as the ego vehicle is not completely in the BSZ.

The information about the upper and lower limits of the constraints are summarized in Table 4.4. It is noted that the acceleration and jerk limits are chosen with consideration of passenger comfort. Specifically, the acceleration limit is bounded by ± 0.5 m/s^2 based on the analysis in [95], and the

[1]Many approaches to forecasting a vehicle's future speed have shown that 10-second prediction is reasonable in consideration of the state-of-art vehicle communication technologies (e.g., [94]).

Table 4.4: The minimum and maximum values for the considered constraints

Parameter	Value	Unit
$u_{\min}$	-2.5	m/s^3
$u_{\max}$	2.5	m/s^3
$a_{\min}$	-0.5	m/s^2
$a_{\max}$	0.5	m/s^2
$v_{\min}$	0	m/s
$v_{\max}$	33	m/s
$Y_{\min,0}^{\Delta PE}$	2	m
t_h	2	s
d_{com}	30	m

jerk u is bounded by $\pm$2.5 m/s^3 smaller than the maximum limit to retain the passenger-comfortable 2.94m/s^3 as suggested in [96]. In a typical car-following scenario, the vehicle velocity cannot be negative, and hence v_{min} is set as 0 m/s. The maximum speed v_{max} is set as 33 m/s (118.8km/h or 74 mph), which covers the highway speed limit in the United States. Regarding the minimum safe distance, $Y_{\min,0}^{\Delta PE}$ is set to 2 m. The time headway t_h of 2 seconds is considered in this study, which allows the ego vehicle to stay at a safe distance from the preceding vehicle when the fleet speed is significant, as suggested in [40], and d_{com} is set to 30 m for the purpose of preventing the neighboring vehicle cutting in [86].

As the desired goal is to reduce the BSZ dwelling time without significant sacrifice in fuel consumption, two metrics are considered: (i) dwelling time in BSZs, T_{BSZ}, and (ii) fuel consumption, m_f. The computation of T_{BSZ} is performed by integrating an indicator function that determines the status of the ego vehicle as follows:

$$T_{\text{BSZ}} = \sum_{k=1}^{N} I(k) T_s$$

with

$$I(k) = \begin{cases} 0 & \text{if (4.7) is satisfied OR both modes are zero,} \\ 1 & \text{otherwise.} \end{cases} \tag{4.20}$$

The fuel consumption of the ego vehicle is computed by the integration of engine fueling rate $\dot{m}_f$.

$$m_f = \sum_{k=1}^{N} \dot{m}_f(k) T_s. \tag{4.21}$$

Since the target vehicle is a P0+P4 MHEV, the fuel consumption is corrected with terminal SOC deviation and $\frac{\Delta(fuel)}{\Delta(SOC)}$ obtained from section 2.4.3.

4.4.2 A Parametric Study

For achieving the best performance in terms of tracking, ride comfort, energy efficiency, and BSZ avoidance, the weighting factors of the cost function (4.14) need to be optimally selected. To this end, a parametric study is conducted with consideration of realistic traffic and driving conditions. Since the objective of this parametric study is to investigate the influence of each weighting factor on the performance, a fixed scenario is used; that is, a preceding vehicle and a sequence of neighboring vehicles are fixed. It should be noted that this scenario (i.e., the sequence of assigned neighboring vehicles) is randomly selected first and then kept fixed throughout the parametric study. For each parameter case, the ego vehicle follows the preceding vehicle described in Table 4.3 for 480 seconds, and the neighboring vehicles are added/removed when they fall behind the vanishing line.

The weighting factors P_1 and P_2 are normalized with the maximum bounds of the constraints on states. The factor P_3 is normalized based on the worst-case speed difference between the ego vehicle and its preceding vehicle obtained from various simulations. These weighting factors are sampled by the Latin Hypercube Sampling (LHS) method [97]. The benefit of LHS is that it can generate a near-random sample of parameter values that are near-evenly distributed from a multi-dimensional space. With a given amount of sampling points, LHS allows us to capture features in parameter space with the best efficiency. On the other hand, the blind spots penalty P_4 is unevenly discretized between 0 to 10. The detailed information about the parameters is provided in Table 4.5.

A total of 3000 sets of weighting factors (P_1, P_2, P_3, P_4) are investigated over the 480-second car-following scenario. The resulting performance of fuel consumption and BSZ dwelling time are shown in Fig. 4.8. It is worth noting that for each set of parameters (P_1, P_2, P_3), the minimum achievable fuel increases with P_4. Without including P_4, the average corrected fuel consumption

Table 4.5: Weight normalization and sampling method

	Lower/Upper bounds	Method	Number of points	Normalization value
P1	[0 10]			2.5^2
P2	[0 10]	LHS	500	0.5^2
P3	[0 10]			4^2
P4	[0 10]	Discrete	6	-

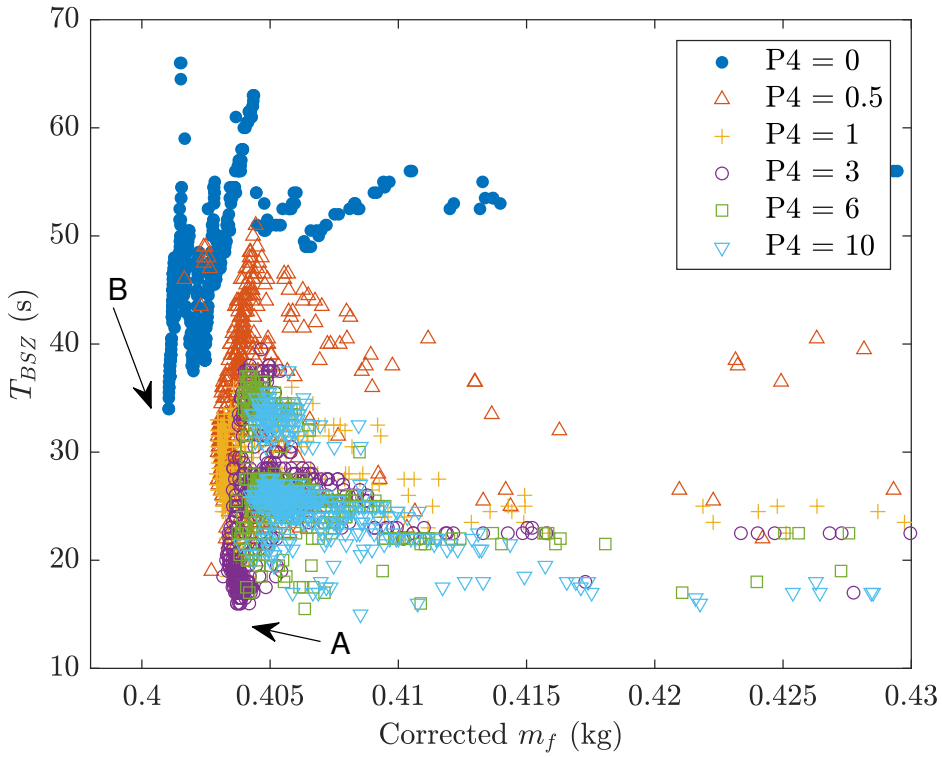

Figure 4.8: Parameter study results for 3000 cases.

among all parameter sets is approximately 0.4034 kg and the BSZ dwelling time is around 47.4 seconds. When P_4 is introduced, by slightly sacrificing the average fuel consumption to 0.4052 kg ($+0.45\%$), the average BSZ dwelling time can be significantly reduced to 31.3 s (-33.97%).

Among all the cases, Case A and Case B are of interest: Case B emphasizes the minimization of fuel consumption but does not care much about the ego vehicle's dwelling time in the BSZs of neighboring vehicles, which can be classified as Eco-ACC in the literature (e.g., [38, 44, 98]). Case A, however, sacrifices the fuel consumption by a little but gains a dramatic reduction in the BSZ dwelling time, which possibly decreases the risk of collision with neighboring vehicles. Case

Table 4.6: Comparison of different ACC methods

Approach	Fuel [kg]	Dwelling time [s]	Weighting factors P1/2/3/4
Eco-ACC	0.4010	34	2.45/5.75/1.95/ 0
DEco-ACC	0.4037	16	7.27/6.87/5.45/3

B (or Eco-ACC) leads to the least fuel consumption: 0.4010 kg during the 480-second trip. As can be seen from the weighting factors in Table 4.6, the Eco-ACC penalizes only jerk, acceleration and velocity difference to the preceding vehicle but not BSZ dwelling time. In comparison to the Eco-ACC, Case A, which is a good candidate of DEco-ACC, sacrifices only 0.67% of fuel consumption and reduces its BSZ dwelling time by 52.9%. Since the DEco-ACC considers minimizing both BSZ dwelling time and fuel consumption, it penalizes jerk, acceleration, tracking error, and dwelling time in the blind spots, as shown in the Table 4.6.

It is noted that the average computation time of solving the MPC problem at each time step is 0.015 second. Therefore, in consideration of the time step of 0.5 second, the proposed controller is real time implementable.

4.4.3 A Case Study

To statistically investigate the performance of the proposed DEco-ACC, a case study was conducted with the weighting factors of Case A selected from the parametric study for 100 different driving scenarios whose trip periods are identical at 960 seconds. When a neighboring vehicle falls behind the vanishing line, a new neighboring vehicle is assigned randomly to the entering line. This randomness is introduced to assess the averaged performance of the proposed DEco-ACC over various car-following cases.

The results of the 100 simulation cases are summarized in Fig. 4.9. On average, the DEco-ACC results in 0.804 kg of fuel consumption, which is a similar level of fuel consumption to that of the Eco-ACC, 0.801 kg, while it further reduces the BSZ dwelling time by 29.5% as compared to the Eco-ACC. It is noted that the speed profile of the preceding vehicle is fixed and hence the corrected fuel consumption with the Eco-ACC does not change as $P_4 = 0$. However, the BSZ dwelling time varies depending on neighboring vehicles.

Figure 4.10 shows the performance of the the Eco-ACC and the proposed DEco-ACC during a particular trip. More specifically, the ego vehicle's relative distance and the blind spots of neighboring vehicles trajectories to the preceding vehicle along with position constraints are compared in Fig. 4.10(a), and the ego vehicle's velocity, acceleration, and jerk trajectories are compared in Fig. 4.10(b). It is worth noting that in Fig. 4.10(a), items are plotted relative to the preceding vehicle. Therefore, the trajectories of Eco-ACC and DEco-ACC are plotted as $-Y^{\Delta PE}$ in each case; moreover, the trajectories of $Y_{\min}^{\Delta PE}$ and $Y_{\min}^{\Delta PE}$ appear to be negative because the preceding vehicle is located at 0 m in this relative coordinate. The filled areas are the trajectories of the BSZs associated with N1 and N2 constraints Eq. (4.7).

Based on the constraints (4.16), during car-following, the ego vehicle with the Eco-ACC maintains the relative distance to the preceding vehicle within the maximum relative distance limit, as

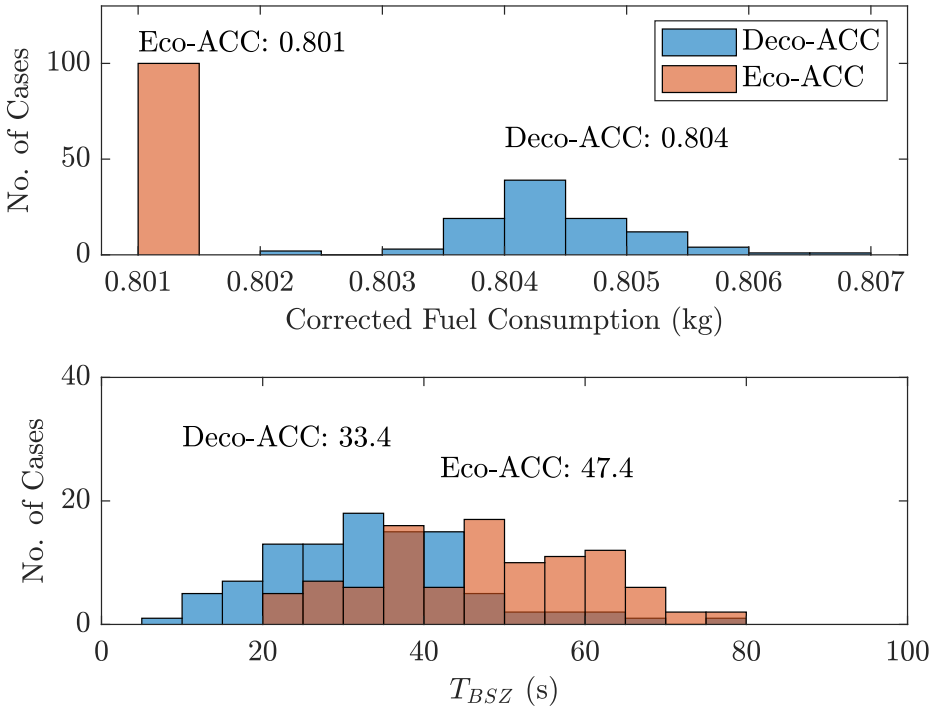

Figure 4.9: A histogram shows 100-case fuel consumption and dwelling time of DEco-ACC and Eco-ACC.

shown in Fig. 4.10(a). The weighting factors P_1 and P_2 in Eco-ACC make the ego vehicle follow the preceding vehicle with reduced acceleration and jerk, which satisfies the comfort needs of ACC and improves energy efficiency. In summary, the operation of the ego vehicle with Eco-ACC is not influenced by the motion of the neighboring vehicles.

On the other hand, the ego vehicle with the DEco-ACC behaves defensively and avoids blind spots when passing neighboring vehicles. The ego vehicle slows down to avoid entering the BSZ of the neighboring vehicle if possible, e.g., 180 seconds. When passing the BSZ, the ego vehicle accelerates to minimize the BSZ dwelling time, for example, 450 seconds and 710 seconds. It should also be noticed that the DEco-ACC decides to pass by using the best knowledge within this prediction horizon and satisfies all kinds of constraints. As can be seen from Fig. 4.10(b), all states (acceleration and velocity) and input (jerk) constraints are well satisfied, guaranteeing vehicle safety and comfort. Compared to the Eco-ACC algorithm, the DEco-ACC actively changes the ego vehicle's relative displacement to avoid the neighboring vehicle's blind spots, this change is reflected in velocity, acceleration, jerk trajectories as shown in Fig. 4.10(b). Because of the proactive avoidance behavior, the DEco-ACC has a slower velocity than Eco-ACC at 180 seconds and faster velocity at 450 and 710 seconds. As the price of reducing the BSZ dwelling time, the

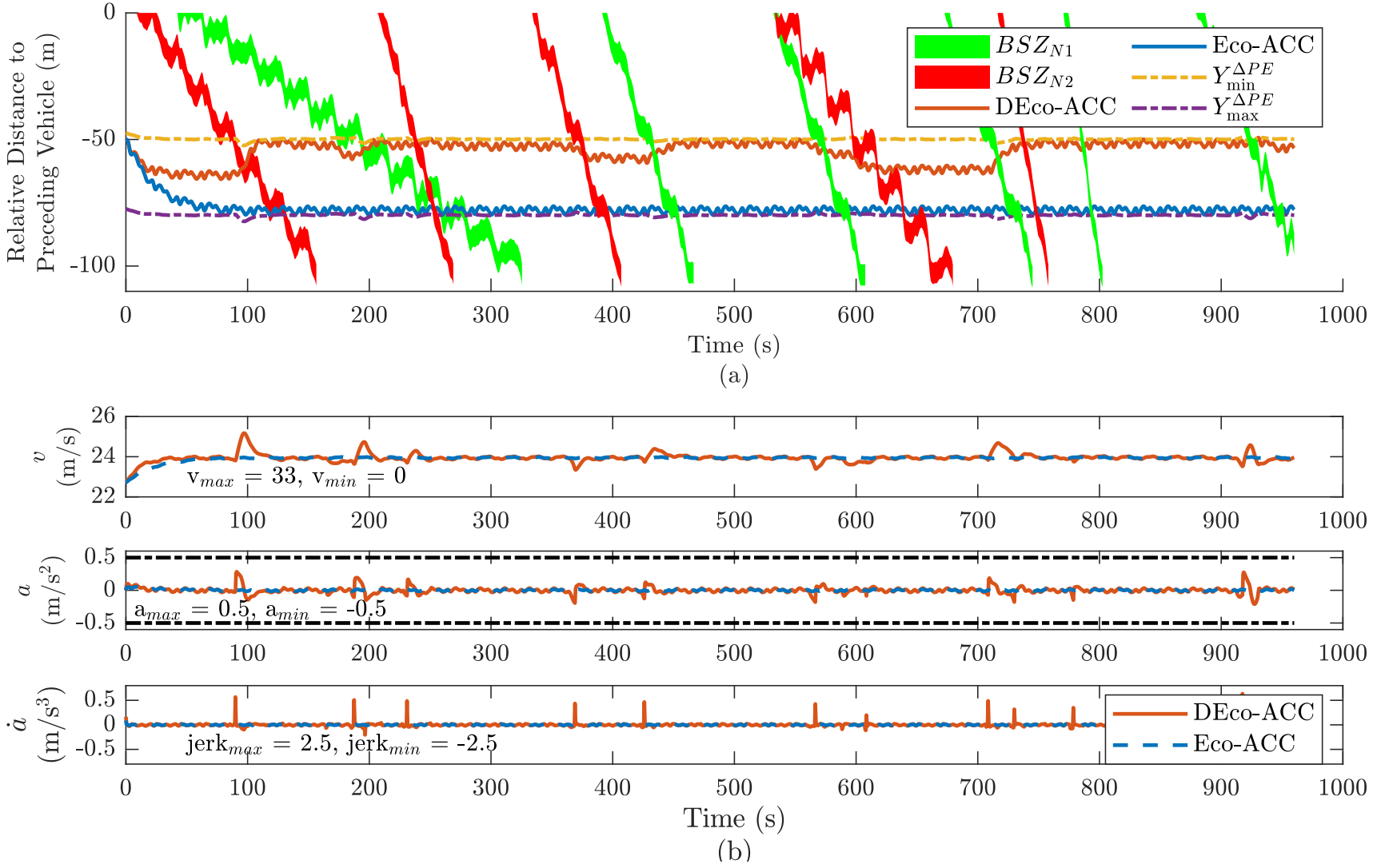

Figure 4.10: Comparison of trajectories with DEco-ACC and eco-ACC: (a) displacement of each vehicle, relative to the preceding vehicle, (b) velocity, acceleration, and jerk.

magnitude of jerk and acceleration with the DEco-ACC is slightly increased as compared to that with Eco-ACC; however, their magnitude values are sufficiently smaller than the maximum limits.

Figures 4.11 and 4.12 show the comparison of acceleration and jerk distribution, respectively, among various driving cycles. Particularly, the first three plots are obtained from three federal driving cycles: (a) HWFET, (b) WLTC, and (c) US06. As a human driver drove these cycles, the results are assumed to be representative of human behavior. The other three plots are obtained from the trip shown in Fig. 4.10 with the preceding vehicle and two different car-following methods: (d) the preceding vehicle itself (PV), (e) Eco-ACC, and (f) DEco-ACC. The preceding vehicle from the trip of Fig. 4.10, which serves as a baseline for comparison, was human-driven and followed by Eco-ACC and DEco-ACC. As shown in Figs. 4.11(a)–(d), the maximum acceleration values by human driving are close to or greater than ± 2 m/s^2. With regard to jerk performance, as shown in Figs. 4.12(b)–(d), human driving, except for the HWFET case, results in relatively high jerk values. In contrast, driving with the Eco-ACC and the DEco-ACC results in considerably milder operation since both controllers penalize acceleration and jerk with non-zero values of P_1 and P_2.

Figures 4.11(e) and (f) show that during more than 90% of the time, the acceleration of those two controllers is less than $0.3m/s^2$. Figure 4.12 (e) shows that the jerk distribution of the Eco-ACC and the DEco-ACC are similar to the HWFET cycle and much milder than the aggressive cycles of WLTC and US06.

The detailed acceleration and jerk statistics of the driving cycles are summarized in Table 4.7. As can be seen from Table 4.7, the maximum and minimum acceleration values from driving with the Eco-ACC are 0.05 m/s^2 and -0.02 m/s^2, respectively. Its average acceleration, $0.01m/s^2$, and average deceleration, $-0.01m/s^2$, are the mildest among all driving cases. This low magnitude of average acceleration and deceleration allows the Eco-ACC controller to achieve the best fuel economy, as discussed earlier. The lowest magnitude of the maximum jerk, $0.09m/s^3$, and the

Table 4.7: Acceleration statistics for different cycles; average acceleration is denoted as acc(+) and average deceleration is denoted as acc(-). Units for a are in $[m/s^2]$. Units for $\dot{a}$ are in $[m/s^3]$

Cycle	$\max a$	$\min a$	$\bar{a}_+$	$\bar{a}_-$	$\max \dot{a}$	$\min \dot{a}$
HWFET	1.43	-1.47	0.194	-0.221	0.89	-0.72
WLTC	2.26	-2.09	0.396	-0.432	1.93	-1.66
US06	3.76	-3.08	0.67	-0.73	3.40	-3.64
PV	2.03	-1.91	0.65	-0.70	3.04	-3.32
Eco-ACC	0.05	-0.02	0.01	-0.01	0.09	-0.01
DEco-ACC	0.28	-0.2	0.03	-0.03	0.64	-0.20

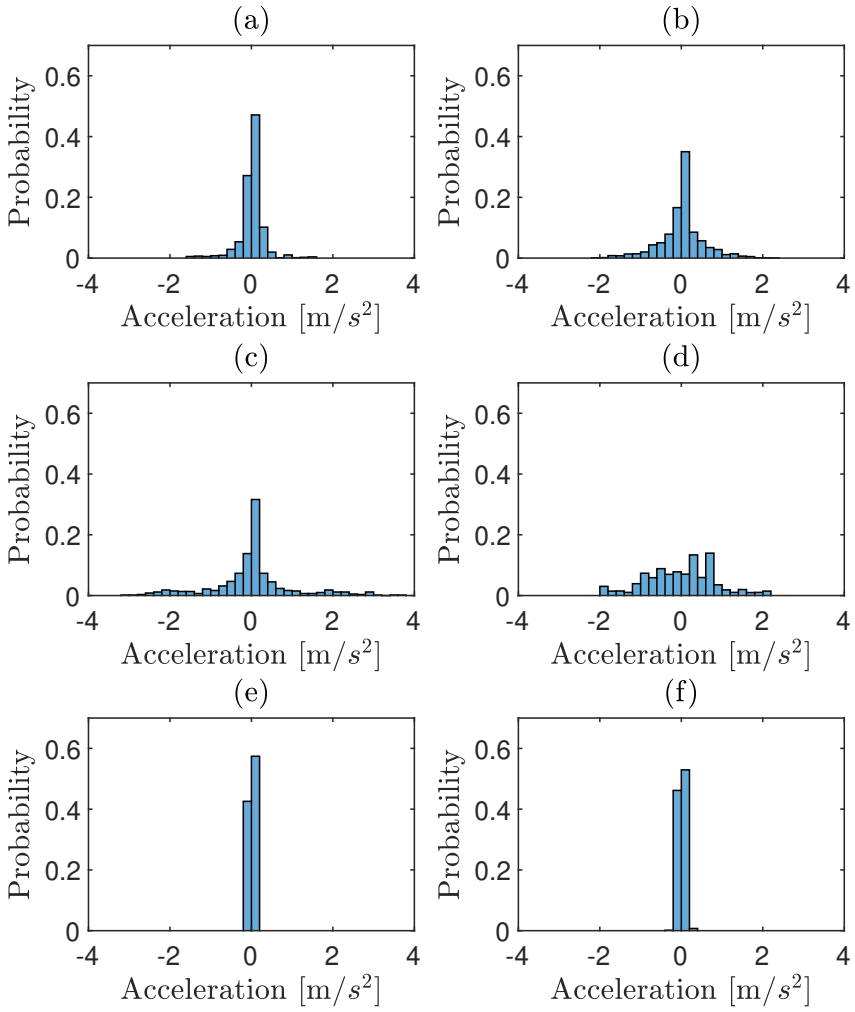

Figure 4.11: Acceleration distribution for different driving cycles. (a): HWFET. (b): WLTC. (c): US06. (d): Preceding vehicle. (e): eco-ACC. (f): DEco-ACC.

minimum jerk, $-0.01m/s^3$, as shown in Table 4.7, allows the Eco-ACC controller to offer the best ride comfort to the passengers.

The magnitudes of $\max a$, $\min a$, $\bar{a}_+$, and $\bar{a}_-$ with the DEco-ACC are greater than those with the Eco-ACC, demonstrating that the DEco-ACC controller sacrifices fuel economy to avoid the neighboring vehicle's BSZ. However, Fig. 4.9 shows that the sacrifice of fuel economy is insignificant despite considerable reduction in BSZ dwelling time. With regard to ride comfort, the maximum jerk, $0.64m/s^3$, and the minimum jerk, $-0.2m/s^3$, of the DEco-ACC indicate that the magnitude is still smaller than the maximum limit to retain passenger comfort, $2.94m/s^3$. Therefore, the DEco-ACC operation is acceptable from both the fuel economy and comfort perspectives [44].

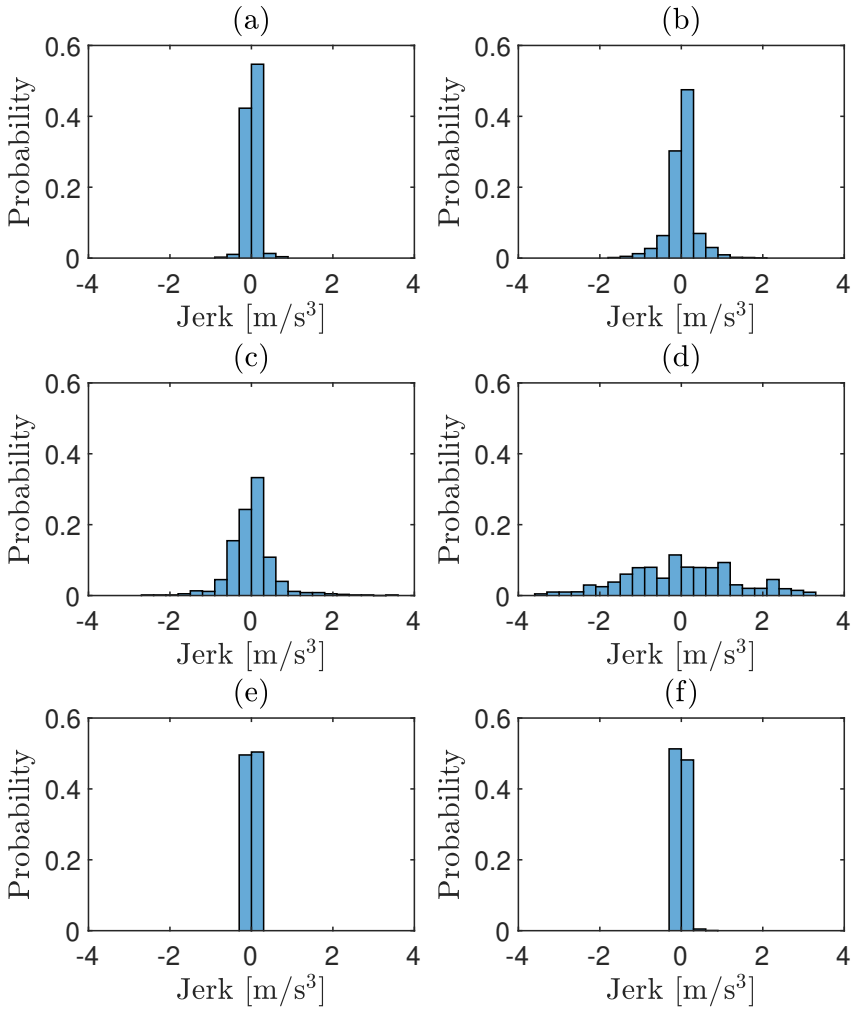

Figure 4.12: Jerk distribution for different driving cycles. (a): HWFET. (b): WLTC. (c): US06. (d): Preceding vehicle. (e): eco-ACC. (f): DEco-ACC.

4.5 Summary

This chapter proposes a defensive ecological adaptive cruise control (DEco-ACC) method to proactively avoid the neighboring vehicles' BSZs during car-following. Unlike the existing Eco-ACC utilizing information about the preceding vehicle only, the proposed DEco-ACC utilizes the neighboring vehicles' speeds and positions and their BSZs. The DEco-ACC is formulated in model predictive control such that car-following can be performed with consideration of energy consumption, ride comfort, and vehicle safety. Specifically, a penalty function that is continuous and one-time differentiable is introduced to handle the constraints about the neighboring vehicles' BSZs.

The impact of the weighting factors of the proposed DEco-ACC is comprehensively evaluated

through a realistic car-following scenario using real-world traffic data from Next Generation Simulation (NGSIM) with two neighboring vehicles located in adjacent lanes. Optimal weighting factors for DEco-ACC are selected based on the results, with 3000 sets generated with the combination of LHS (P_1, $P2$, and $P3$) and unevenly discretized P_4 values. In order to statistically investigate the performance of the proposed DEco-ACC, 100 different driving scenarios are simulated, and the results show that on average, the proposed DEco-ACC can further reduce, by 29.5%, the dwelling time in the BSZs of the neighboring vehicles without significant fuel penalty (0.4% increase in fuel consumption) and deterioration of ride comfort compared to the Eco-ACC and can successfully follow the preceding vehicle without violating safety-related constraints.

In this study, exact information about the nearest surrounding vehicles' future speed is assumed to be known for the MPC controllers. Since uncertainties could lead to inaccuracy in speed prediction, investigating their impact on the proposed DEco-ACC's performance is an important direction for future research.

CHAPTER 5

Personalized One-pedal Driving for Electric Vehicles by Learning-based Model Predictive Control

5.1 Introduction

As proposed in chapter 4, the defensive ecological adaptive cruise control (DEco-ACC) for P0+P4 MHEV can reduce driver's effort during car-following, minimize the threat from neighboring lanes and maintains a similar level of fuel economy compared to Eco-ACC. However, such type of ACC feature has limited application outside the highway scenarios, especially in an urban area with many traffic signals and stop signs.

In order to maximize the range of electrified vehicles, the area of optimal regenerative braking is being actively studied. Furthermore, to cope with the scenarios with many stop-and-go events, researchers have developed another feature, one-pedal-driving (OPD), which automatically starts the energy regeneration once the driver begins to release the acceleration pedal. This feature is soon implemented onto many on-market vehicles such as Nissan Leaf, BMW i3, and Tesla Model S. However, [59] reports that the transition from two-pedal-driving (TPD) to OPD still confuses drivers. And, [57, 58] showed that existing OPD does not sufficiently consider individual driver's behaviors when performing brake action. Hence, this chapter proposes a personalized one-pedal-driving algorithm (POPD), which possesses a learning framework to adapt the individual driver's braking behavior. The POPD only performs braking action when an upcoming braking event is predicted through the MPC controller. Therefore, it requires less effort for the driver to adapt this POPD feature. Furthermore, thanks to MPC's predictability of car-following dynamics, all upcoming collision events from experiments are prevented from happening. The main contributions of this chapter are threefold:

- Constraints related to an individual driver's preference during braking are designed based on the analysis of real-world driving data.

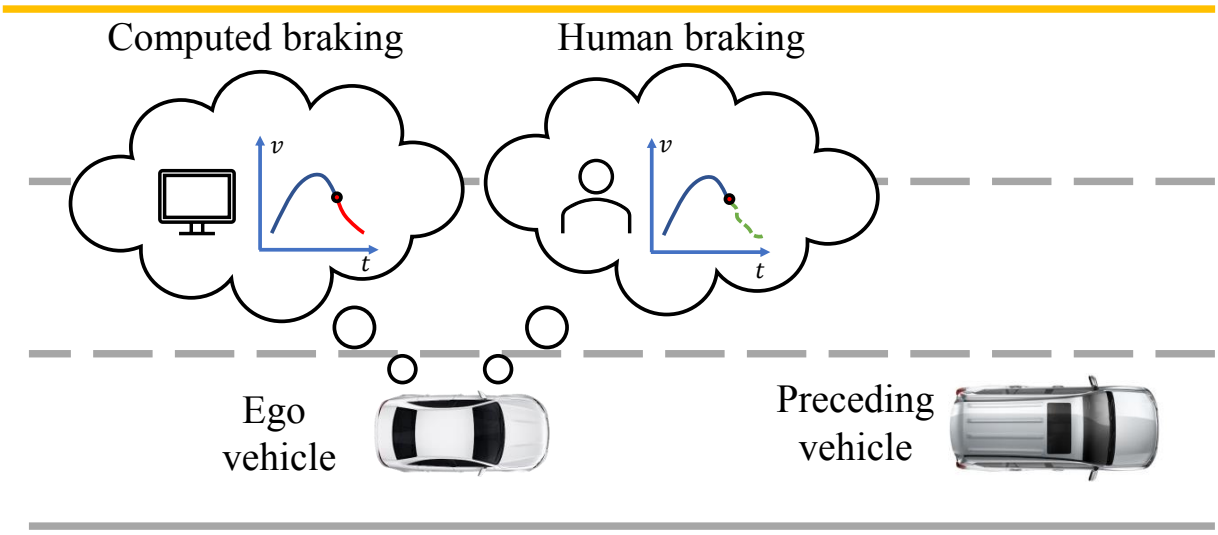

Figure 5.1: Personalized One-pedal driving: algorithm generate human-like deceleration before the driver takes any action. The driver only needs to control the acceleration pedal most of the time.

- A personalized one-pedal driving (POPD) method is proposed using model predictive control with a learning framework that determines optimal weights of the cost function.
- The performance of the proposed POPD is tested and validated in open-loop simulations and human-in-the-loop co-simulations.

The remainder of this chapter is organized as follows: First, two constraints associated with the driver's characteristics are introduced in Section 5.2. Then, the identification of the constraints is detailed with real-world driving data in Section 5.3. Next, the proposed POPD algorithm and its learning method are presented in Section 5.4 and Section 5.5, respectively. The simulation and experimental validation of the POPD are discussed in Section 5.6 and Section 5.7. Finally, Section 5.8 presents the Summary and future work.

5.2 Driving Behavior Analysis

When following a preceding vehicle with a TPD system, a human driver regulates the relative distance to the preceding vehicle within a comfortable distance interval using acceleration and brake pedals. In order to mimic this human behavior in the control design, this study considers two types of constraints that mainly influence a driver's desired distance interval during car following: (i) time headway constraints based on a micro-level traffic flow analysis [65, 99] and (ii) perceptual constraints based on Wiedemann's car-following model [100].

5.2.1 Time Headway Constraints

The time headway, defined as the time interval between the front bumpers of two successive vehicles, has been considered a useful indicator for safety evaluation [101]. Thus, the comfortable distance range is often described with the time headway [65, 99] as follows:

$$d_{rel} \geq \underline{t}_h v_e + \underline{d}_h, \tag{5.1}$$

$$d_{rel} \leq \bar{t}_h v_e + \bar{d}_h. \tag{5.2}$$

where v_e is the ego vehicle's speed and d_{rel} is the distance from the rear bumper of the preceding vehicle to the front bumper of the ego vehicle. The maximum and the minimum relative distances are determined with the maximum and the minimum time headway values $\bar{t}_h$ and $\underline{t}_h$ and constants $\bar{d}_h$ and $\underline{d}_h$, respectively. Note that these time headway values and constants reflect a driver's preference during car following; therefore, these parameters could be driver-specific and need to be determined for an individual driver.

5.2.2 Perceptual Constraint

In [100], Wiedemann has proposed a perception-based car-following model capable of capturing human behavior with a set of parameters by considering drivers' physical and mental aspects. According to the model, the actions of accelerating and braking are triggered by perception thresholds, as defined by Wiedemann.

Figure 5.2 shows how a human drives when the ego vehicle approaches its preceding vehicle in consideration of thresholds described with the desired relative distance d_{rel} and the relative speed $\Delta v = v_p - v_e$, where v_p represents the preceding vehicle's velocity. Two artificial reaction zones with perceptual thresholds are introduced to demonstrate the driver's behavior when following a preceding vehicle. Suppose the ego vehicle approaches the preceding vehicle with a faster speed; then, the Δv becomes negative, and d_{rel} decreases until the ego vehicle enters the SDV shown in Fig. 5.2. By definition, the driver in the perceptual threshold will press the brake pedal to leave the SDV and keep the vehicle from entering another threshold (CLDV). During this braking period, the ego vehicle slows down, and the magnitude of Δv decreases until the ego vehicle leaves the reaction zone. Between two reaction zones, the ego vehicle continues to follow the preceding vehicle unconsciously as long as SDV, SDX, and OPDV bound its operation.

During this Approaching-Braking-Following process, the reaction zones play a role of constraints. The perceptual thresholds divide reaction zone into soft constraints and hard constraints, which can be slightly violated and should be strictly enforced, respectively. In this study, a soft linear constraint is applied to capture the SDV at first. Then, the later results show that by adequately

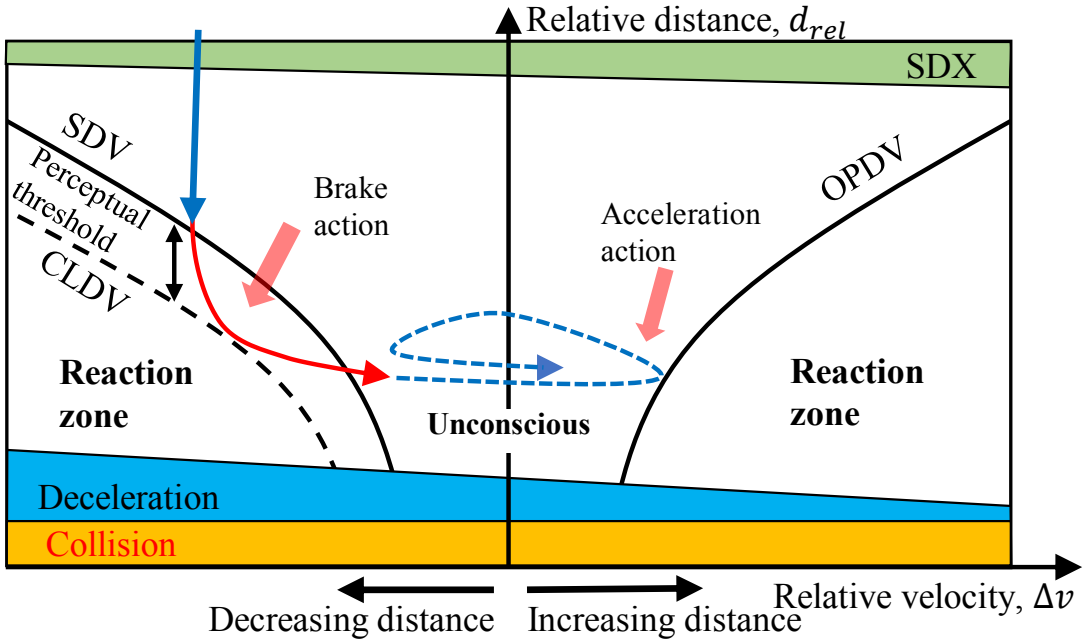

Figure 5.2: Wiedemann's car following model describes the relationship between the relative distance and the relative velocity. SDV, OPDV, and SDX represent brake threshold, acceleration pedal threshold, and max follow distance threshold, respectively.

penalizing the SDV, the CLDV may not be violated and hence can be excluded from the formulation. Note that this study focuses on developing a personalized braking algorithm for one-pedal driving; the constraints of OPDV and SDX in Fig. 5.2 are handled with the human acceleration pedal position. Thus, the case with negative Δv is considered only.

Knowing the constraint-like property of the reaction zone in the Wiedemann's car-following model, the boundary line between the reaction zone and unconscious zone, as shown in Fig. 5.2, can be simplified as a linear constraint, noting as

$$d_{rel} \geq k\Delta v + b. \tag{5.3}$$

where k and b are slope and bias values for the fitted constraints, which varied depending on drivers. Hence, they need to be determined through an identification method described in the later section.

5.3 Driving Data Analysis

According to the studies in [102], and [103], calibrated parameters of the Wiedemann's model can drastically differ from one driver to another. Therefore, to guarantee that the proposed algorithm

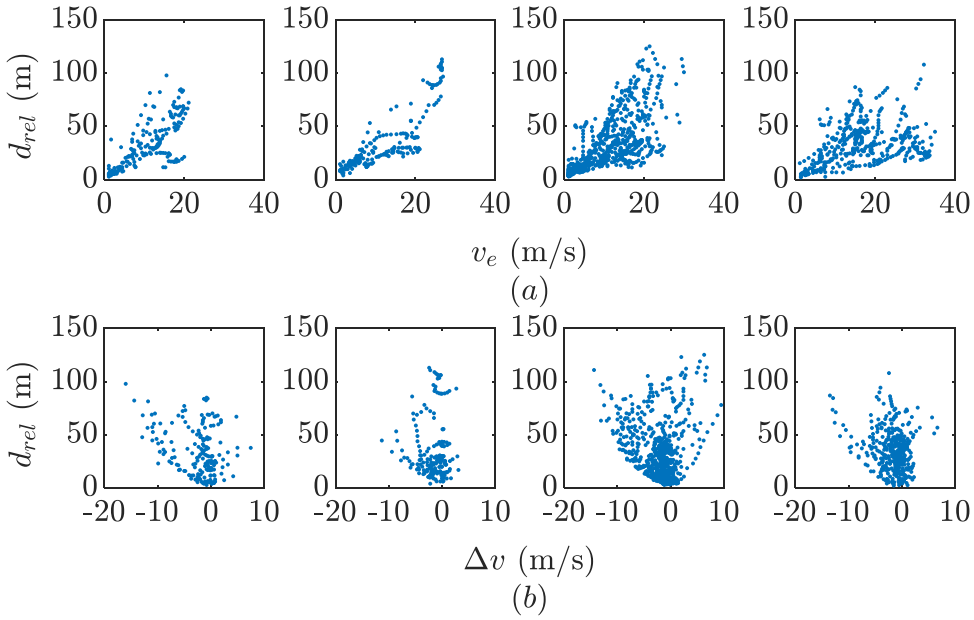

Figure 5.3: Data of four selected drivers, used for identifying time headway constraints (a) and for identifying perceptual constraints (b).

mimics a specific driver's behavior, the constraints in Eqs. (5.1), (5.2) and (5.3) need to be identified for an individual driver. This section proposes a constraints identification method based on linear regression, which can automatically identify headway constraints and perceptual constraints with a small amount of pre-collected driving data from a specific driver.

5.3.1 Real-world Driving Data

Real-world driving data of 450 drivers from their daily driving are collected and used to analyze the ego vehicle's braking behavior, particularly when a driver encounters a preceding vehicle. The recorded signals from the real-world vehicles include velocity, acceleration, brake on/off, acceleration pedal position, relative velocity and relative distance to a preceding vehicle. The constraints identification and weight learning methods are developed based on all 450 drivers' behavior analyses. After then, The developed personalized one-pedal driving algorithm is validated with all cycles driven by the 450 drivers.

Figures 5.3 (a) and (b) show examples of the d_{rel} vs. v_e plot and d_{rel} vs. Δv plot from four selected drivers. The boundaries of these data points can be approximately identified as the constraints in Eqs. (5.1), (5.2) and (5.3).

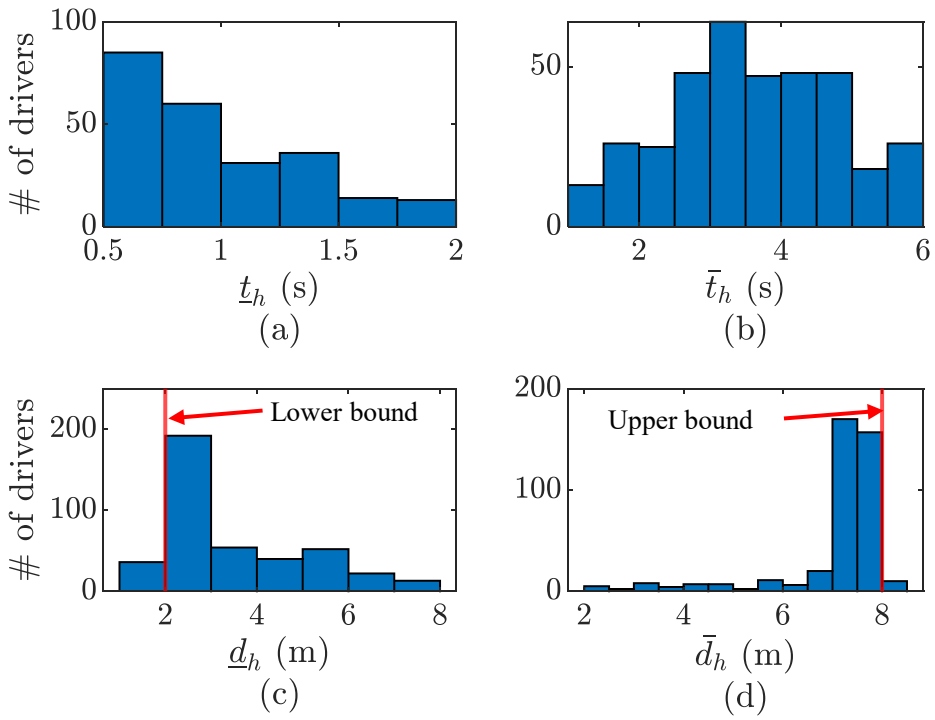

Figure 5.4: Headway constraints and perceptual constraints fitting of all drivers' data: (a) Minimum time headway, (b) Maximum time headway, (c) Minimum distance headway and (d) Maximum distance headway.

5.3.2 Identification of Headway Constraints

Through analyzing the 450 driver's driving data, it is observed that most drivers share a similar preference for distance headway, which is bounded between 2 m and 8 m as shown in Figure 5.4 (c) and (d), but with various time headway preferences, Figure 5.4 (a) and (b). Thus, in this study, fixed distance headway values of 2 m and 8 m are considered for the lower and upper bounds, respectively. However, the time headway values need to be determined for different drivers. The lower and upper bounds of distance headway are marked on to Figure 5.4 (c) and (d) as vertical red lines.

The collected driver's data is post-processed to determine the comfort time headway of different drivers, first divided into intervals with different velocities as shown in Fig. 5.5 (a). In each interval, the data points with the maximum and the minimum relative distances are identified. Then, the velocity and relative distance of these identified data points are collected into vectors $\boldsymbol{X}_{\max}$, $\boldsymbol{X}_{\min}$, $\boldsymbol{Y}_{\max}$, and $\boldsymbol{Y}_{\min}$, respectively. The upper and lower bounds of the time headway, $\bar{t}_h$ and $\underline{t}_h$, are

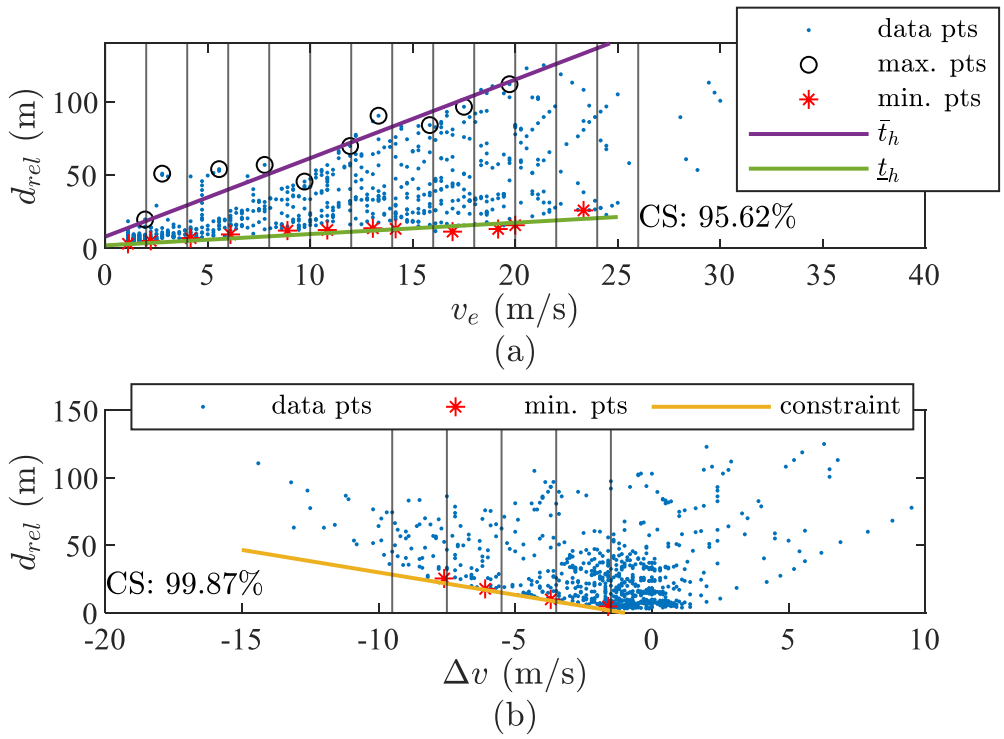

Figure 5.5: A fitted constraint function for a selected driver.

defined by

$$\bar{t}_h = (\boldsymbol{X}_{\max}^\top \boldsymbol{X}_{\max})^{-1} \boldsymbol{X}_{\max}^\top (\boldsymbol{Y}_{\max} - \bar{d}_h), \tag{5.4}$$

$$\underline{t}_h = (\boldsymbol{X}_{\min}^\top \boldsymbol{X}_{\min})^{-1} \boldsymbol{X}_{\min}^\top (\boldsymbol{Y}_{\min} - \underline{d}_h). \tag{5.5}$$

As an example, the fitted constraints boundaries of (5.1) and (5.2) of a selected driver are shown in Fig. 5.5 (a). Note that the constraints satisfaction (CS) rate of these headway constraints is 0.95, meaning that over 95% of the driver's braking action can be captured by the constraints (5.1) and (5.2) despite human drivers' nonlinear behavior. As we admit the limitation of the constraints (5.1) and (5.2) in capturing human behavior, these constraints will be handled as soft constraints in the control problem.

5.3.3 Identification of Perceptual Constraint

The perceptual constraints are identified similarly compared to the headway constraints. The data analysis shows that most drivers share a similar slope k in Figure 5.6 (a), but with different bias b preferences of (5.3) as shown in Figure 5.6 (b). Thus, in this study, the average slope k is used. Collected driving data are divided into different intervals based on relative velocity Δv as shown

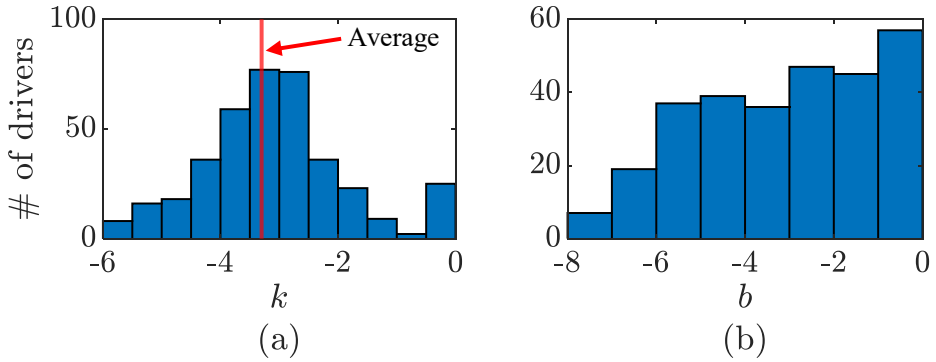

Figure 5.6: Headway constraints and perceptual constraints fitting of all drivers' data: (a) Slope of the perceptual constraint and (b) Bias of the perceptual constraint.

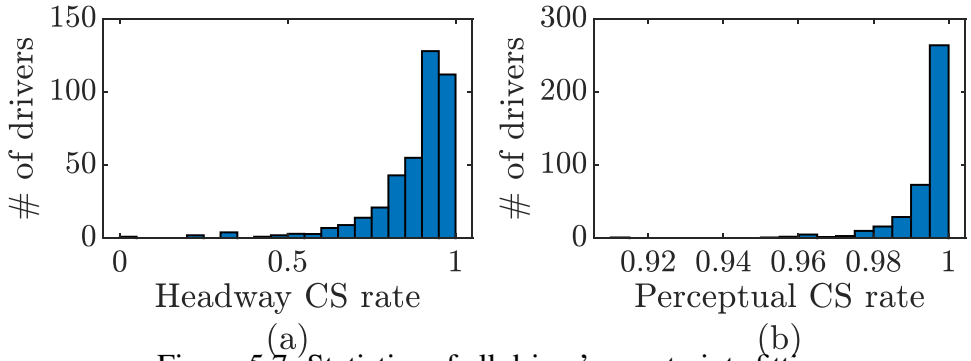

Figure 5.7: Statistics of all driver's constraints fitting.

in Fig. 5.5(b). In each interval, the data point with the minimum relative distance is identified. Then, these identified data points are collected into vectors $\boldsymbol{X}_p$ and $\boldsymbol{Y}_p$. The bias term b in (5.3) is identified with

$$b = \frac{\sum_{j=1}^{n} (\boldsymbol{Y}_{p,j} - k\boldsymbol{X}_{p,j})}{n} \tag{5.6}$$

where n is the number of the identified data points.

Figure 5.5(b) shows the identified perceptual constraint of an example driver and his/her braking data in a d_{rel} vs. Δ_v plot. Since this study focuses on braking rather than acceleration, only the negative Δv side constraint is considered. It should be noted that this perceptual constraint is a simplified substitute for the left reaction zone in Fig. 5.2. The CS of the perceptual constraint in Fig. 5.5(b) is 0.998, meaning that most of the driver's braking behavior is captured by this constraint. Similar to the headway constraints, the non-linearity from the reaction zone in Fig. 5.2 cannot be captured with a linear constraint. Therefore, this perceptual constraint is treated as a soft constraint in the control design and allowed to be slightly violated.

5.3.4 Performance of Constraints Fitting

The CS statistics of all 450 drivers' time headway and perceptual constraints are shown in Fig. 5.7. Fig. 5.7(a) shows that most of 450 drivers have their headway CS higher than 0.7, and Fig. 5.7(b) demonstrates almost all of drivers have their perceptual CS higher than 0.98. It can be concluded that both the proposed time headway and perceptual fitting methods describe drivers' behavior reasonably well.

5.4 Personalized One-Pedal-Driving Algorithm

This section describes the proposed personalized braking algorithm based on model predictive control (MPC). The constraints described in Section 5.2 are identified and considered within a learning framework.

5.4.1 Vehicle Longitudinal Dynamics

The current algorithm assumes that the ego vehicle drives on a single lane; therefore, only longitudinal dynamics are considered. It is also assumed that the preceding vehicle's velocity information within the prediction horizon is available through specific speed prediction algorithms [104]. The car-following dynamics are expressed as a discrete-time state-space model with a sampling time T_s:

$$x_{k+1} = \boldsymbol{A}x_k + \boldsymbol{B}u_k + \boldsymbol{d}_k, \tag{5.7}$$

with

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 0 \\ T_s & 1 & 0 \\ -\frac{1}{2}T_s^2 & -T_s & 1 \end{bmatrix}, \tag{5.8}$$

$$\boldsymbol{B} = \begin{bmatrix} T_s & \frac{T_s^2}{2} & -\frac{T_s^3}{6} \end{bmatrix}^\top, \tag{5.9}$$

$$\boldsymbol{d}_k = \begin{bmatrix} 0 & 0 & \frac{T_s}{2}v_{p,k} + \frac{T_s}{2}v_{p,k+1} \end{bmatrix}^\top, \tag{5.10}$$

where the state and control variables are defined as

$$x_k = \begin{bmatrix} a_e & v_e & d_{rel} \end{bmatrix}_k^\top,\ u_k = \dot{a}_k. \tag{5.11}$$

It is noted that the rate of acceleration is used as a control input in order to penalize a jerk for enhanced ride comfort.

5.4.2 MPC Formulation

The objective function in the MPC-based POPD is formulated as the weighted sum of the square of acceleration a, rate of acceleration u, and velocity difference Δv over the prediction horizon N, with two additional costs associated with slack variables:

$$J(u, \delta_1, \delta_2) = \sum_{k=0}^{N-1} (a_{e,k})^2 P_1 + u_k^2 P_2 + \Delta v_k^2 P_3 + \delta_1 P_4 + \delta_2 P_5. \tag{5.12}$$

where P_i's are the weighting factors for penalizing acceleration, rate of acceleration and velocity difference to the preceding vehicle. The slack variables δ_1 and δ_2 are introduced to implement the constraints (5.1), (5.2) and (5.3) as soft constraints. Other state and control constraints are summarized as follows:

$$v_{e,\min} \leq v_{e,k} \leq v_{e,\max} \tag{5.13}$$

$$a_{e,\min} \leq a_{e,k} \leq a_{e,\max} \tag{5.14}$$

$$u_{\min} \leq u_k \leq u_{\max}. \tag{5.15}$$

The proposed algorithm activates itself when the driver releases the acceleration pedal in the presence of a preceding vehicle, and the algorithm terminates when the driver presses the acceleration pedal again or in the absence of a preceding vehicle/stop sign. During the activation of the proposed algorithm, a Proportional-Integrator (PI) controller converts the optimized $\dot{a}$ into the desired brake pedal position. The driver, therefore, only needs to control the acceleration pedal most of the time. If the optimized deceleration does not meet the driver's requirement, the driver can still press the brake pedal to override the brake signal generated by the proposed algorithm. This override action will be recorded and used for learning MPC weighting factors to be described in the later section. When there are no preceding vehicle/traffic stops, the algorithm allows the vehicle to coast and avoid double energy conversion.

5.5 MPC Weights Learning

To guarantee the proposed algorithm matches the specific driver's behavior, the weights P_i's in the MPC cost function (5.12) need to be determined for each driver. This section proposes an optimal weights learning method based on particle swarm optimization, which can automatically identify P_i's with a small amount of pre-collected driving data from a specific driver.

5.5.1 Optimal Weight Learning

[105] trains a controller to capture human's behavior with inverse reinforcement learning and a performance evaluation metric consisting of mean square error between the controller's action and historical human action.

In order to judge the similarity between our controller's action and the human driver's action, a similar performance metric L is considered:

$$L = \frac{1}{T}\sum_{j=1}^{T}\left((a_{e,j} - \tilde{a}_{e,j})^2 + (\Delta v_j - \Delta\tilde{v}_j)^2 + (\dot{a}_j - \tilde{\dot{a}}_j)^2\right) \tag{5.16}$$

where $\tilde{a}_e$, $\Delta\tilde{v}$ and $\tilde{\dot{a}}$ represents acceleration, relative velocity and rate of acceleration of human driver's action during braking. The variable T denotes the total samples used for optimal weight learning. The performance metric L depends on the weights $P_i's$ in the cost function (5.12), and a driver with different driving behavior may have different optimal weights.

Finding $P_i's$ that minimizes L for a specific driver can be considered an optimization problem. However, there is no guarantee of differentiability and convexity on such an optimization problem. Thus, the gradient-based optimization technique is challenging to find an optimal solution. Therefore, in this study, Particle swarm optimization (PSO) is applied to optimize the weights of the cost function (5.12) for a specific driver. PSO is one of the most popular meta-heuristics [106, 107], inspired by group collaboration behavior from nature, such as a flock of birds. At the beginning of the optimization, PSO randomly generates a certain number of particles within the search space.

In this study, P_1 to P_3 are optimized with PSO, that is, the max dimension $j_{max} = 3$. Each particle in the swarm represents a set of weights $[P_1, P_2, P_3]$ that leads to a certain evaluation metric L. The swarm size i_{max} is chosen to be 50, and maximum iteration t_{max} is 20. In (5.12), P_4 and P_5 are associated with slack variables and only contribute to the cost term if the constraints in Fig. 5.5 are violated. Since the constraint in Fig. 5.5 are fitted to be feasible in most of the scenarios, P_4 and P_5 are chosen to be universal constants for improving computational efficiency. The research in [107] has shown that in a low-dimension search space, the PSO is less likely to get trapped into a local optimal. The upper and lower bounds of the searching parameter space are chosen as $P_1 \in [0.01\ 500]$, $P_2 \in [0.01\ 100]$, and $P_3 \in [0.01\ 100]$.

5.5.2 Prediction Method Selection

The state-space model shown in Section 5.4.1 requires the future operation of the preceding vehicle to predict the car-following dynamics in a short horizon. However, the future operation of the preceding vehicle is hard to gather and prone to errors. For those preceding vehicles without

autonomous driving and V2V technology, the prediction of the future motion can only be made by collecting its previous operations. In the literature, [94, 108, 109], various methods capable of predicting the preceding vehicle's speed have been proposed with a relatively small error through a polynomial fitting of previous information. However, our study observed that a relatively simple constant acceleration (CA) prediction [110] is sufficient to predict braking operation. The CA method predicts the preceding vehicle's l-step future speed $v_{p,k+l}$ by assuming that it maintains the same acceleration as the current step k as follows:

$$v_{p,k+l} = v_{p,k} + T_s a_{p,k} l. \tag{5.17}$$

Although the CA prediction in a long horizon possesses errors due to the change of acceleration in the preceding vehicle's acceleration, this prediction error in a long horizon has little impact on the performance of MPC [108]. The reason is, in each time step, the MPC only adopts the first control action from the control sequence generated within the prediction horizon and re-calculates the control sequence when it reaches the next time step [111]. Thus, the MPC performance can be guaranteed if the v_p prediction in a short horizon is precise enough.

To assess the performance of the CA method in the current MPC formulation, a comparative case study is conducted between the CA method and the perfect information prediction where the preceding vehicle's future speed v_p can be acquired with no error. In addition, the impact of the horizon length can affect the performance of the proposed POPD, and hence different prediction lengths ranging from 3 to 10 are investigated.

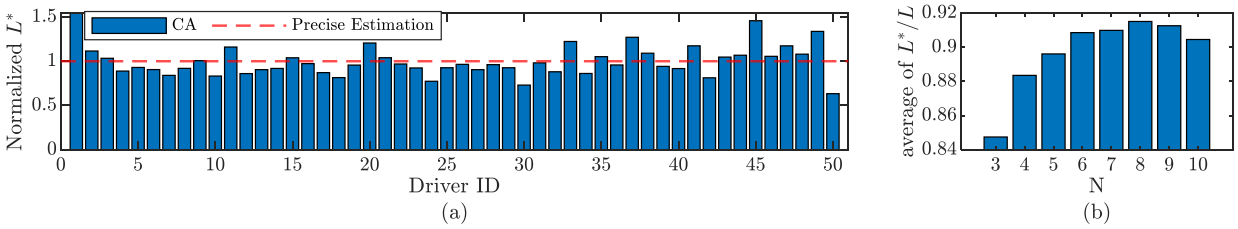

Figure 5.8: (a) The comparison of L^* between CA and perfect information prediction. L^* of CA is normalized base on L^* of the perfect information prediction method.(b) Averaged $\frac{L^*}{L}$ of 50 driver. $N \in [7, 9]$ results highest performance.

Figure 5.8 (a) shows the performance comparison between the CA and perfect prediction among 50 selected driving cycles. The best performance metric L among all prediction horizons (from $N = 3$ to $N = 10$) is denoted by L^* in a specific cycle. Note that L^* of the CA is normalized based on L^* of the perfect prediction. The average normalized L^* of the CA method among all drivers is 0.997, which means the CA method does not lead to severe performance degradation.

The study shows that the optimal prediction length N varies depending on different drive cycles.

For simplification, the prediction horizon is selected to lead to the best performance for most of the drivers. To determine the optimal prediction horizon, a normalized metric $\frac{L^*}{L}$ is introduced to assess how different N performs in a specific cycle. The best N in a specific cycle leads to $\frac{L^*}{L} = 1$, and other N's lead to $\frac{L^*}{L} < 1$. Through averaging $\frac{L^*}{L}$ among all 50 drivers, the best N that fits for all drivers, in general, can be found. The relationship between averaged $\frac{L^*}{L}$ and N are shown in Fig. 5.8 (b). As it can be seen, the prediction length $N = 8$ yields the highest $\frac{L^*}{L}$; thus, in the rest part of this paper, the optimal horizon of $N = 8$ is used.

5.6 Performance Evaluation

In order to statistically investigate the performance of the proposed POPD algorithm, this section presents the simulation results of the proposed algorithm under 450 driving cycles compared with those of a benchmark braking algorithm, the desired-relative-distance personalized braking (DRD-PB) algorithm.

5.6.1 Simulation Setup

In vehicle simulation, the proposed algorithm controls the ego vehicle's braking action when the acceleration pedal is released until the acceleration pedal is pressed again. The driving cycles used in the simulation are recorded from real-world human drivers, and hence the algorithm-generated braking actions can be compared with the human drivers' actions for performance evaluation purposes.

Four hundred driving cycles driven by different drivers are considered. Each drive cycle is divided into two parts, a training cycle for MPC weight learning and a validation cycle for assessing the performance of matching human behavior, namely.

5.6.2 Desired Relative Distance-based Personalized Braking

For the purpose of performance comparison, this section describes a desired relative distance-based personalized braking (DRD-PB) algorithm [112] that performs a function similar to the proposed POPD method. The DRD-PB tracks the desired relative distance d_{des}, which captures relative distance from human driving data during active braking scenarios in a function of the v_e and Δv as follows:

$$d_{des} = d_s + \tau v_e + b_{des}\Delta v^2. \tag{5.18}$$

The desired distance headway d_s, desired time-headway τ, and personalized factor b_{des} are identified to capture the driver's behavior in active braking. Since DRD-PB is also designed to perform

braking only, Δv^2 term in Eq. (5.18) does not interfere with the braking performance of the algorithm. When the preceding vehicle speeds up, Δv^2 term increases so does d_{des}. In this case, the DRD-PB will not result in acceleration and will wait for the driver to operate the acceleration pedal.

To accommodate the stochastic behavior of the human driver's braking action, the desired distance in Eq. (5.18) is corrected by a factor c_f expressed as follows:

$$c_f = \min\left(1, \hat{d}_{rel}/\hat{d}_{des}\right), \tag{5.19}$$

where $\hat{d}_{rel}$ and $\hat{d}_{des}$ are the relative distance and computed desired relative distance at the moment of acceleration pedal released. The correction factor is kept the same for continuous braking and will reset when the driver press and release the acceleration pedal in the next event. When the driver releases the acceleration pedal late so that d_{rel} becomes shorter than computed d_{des} initially, the correction factor will reduce d_{des} to avoid harsh braking caused by a large initial relative distance error. In the DRD-PB, a PI controller is used to track the corrected desired relative distance,

$$\tilde{d}_{des} = c_f d_{des}. \tag{5.20}$$

Since this DRD-PB method only focuses on braking control, there will be no positive acceleration generated from the controller if d_{rel} is larger than $\tilde{d}_{des}$; Instead, the ego vehicle maintains a coasting mode. Much detailed information about DRD-PB can be found in [112].

5.6.3 Performance Comparison

The histogram in Fig. 5.9 shows the performance comparison over 400 driving cycles with the proposed POPD and the DRD-PB methods. The performance metric L of the POPD method is normalized with that of the DRD-PB. In most cases, the normalized L of the proposed algorithm is less than 0.5, meaning that the proposed POPD method outperforms the DRD-PB by over 50%.

A time-series comparison of a selected driving cycle segment between both algorithms is shown in Fig. 5.10. Particularly, Fig. 5.10(a)–(d) compare the velocity, acceleration, relative distance, and relative velocity among the human driver and two braking algorithms, respectively. Note that in Fig. 5.10(c), the value of d_{rel} is shown only if there exists a preceding vehicle within a distance of $150\ m$. Figure 5.10 (e) shows the driver's historical acceleration pedal position and brake signals. For the acceleration pedal, 1 and 0 represent completely pressed and released states, respectively, and an intermediate position is also recorded. On the other hand, the brake signal is a Boolean-type signal, and hence it only records whether the driver presses the brake. Both control algorithms activate when the driver completely releases the acceleration pedal and terminate if the driver

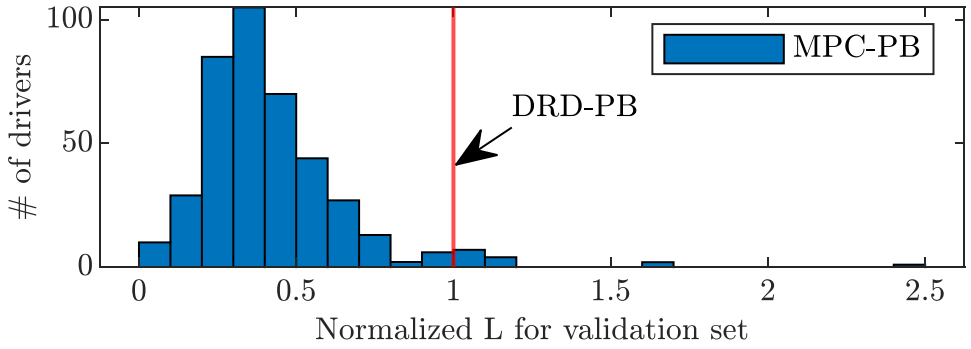

Figure 5.9: Histogram comparison between POPD with DRD-PB: the L of POPD is normalized based on DRD-PB.

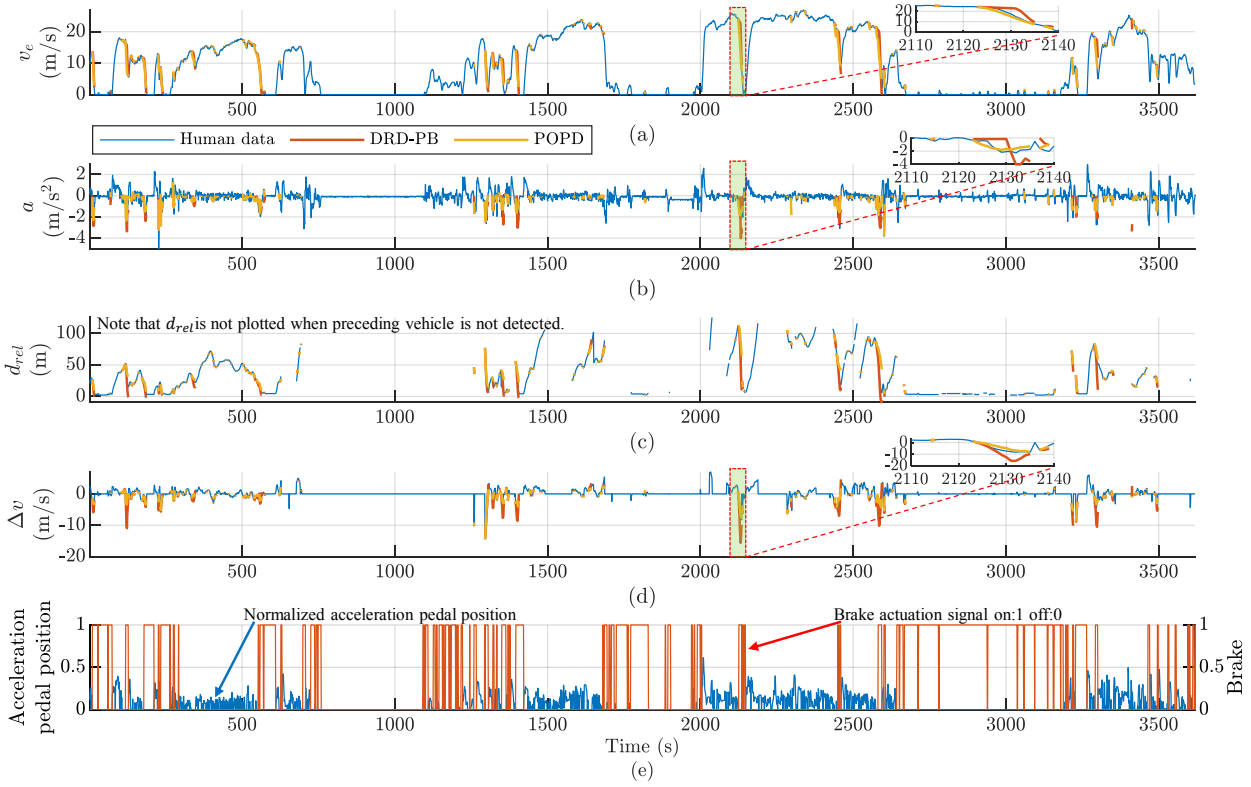

Figure 5.10: Simulated time series comparison between POPD and DRD-PB method of a selected driver. (a): ego vehicle velocity over the time. (b): ego vehicle acceleration over the time. (c): relative distance between ego vehicle and the preceding vehicle over the time. (d): relative velocity between ego vehicle and the preceding vehicle. (e): acceleration pedal position and brake pedal actuation signal.

presses the acceleration pedal again.

Overall, both methods can properly slow down the ego vehicle when the driver releases the

acceleration pedal and with an existence of a preceding vehicle, as shown in Fig. 5.10(a). However, Fig. 5.10(b) indicates that the peak deceleration generated from the DRD-PB is much larger than the human data and the POPD at around 100 seconds, 1400 seconds, and 2100 seconds. The zoom-in portion of Fig. 5.10(a), (b) and (d) show that the ego vehicle encounters a preceding vehicle between 2110 and 2140 seconds. In this case, the time headway is small, and there exists a risk of collision. The DRD-PB is not able to predict the upcoming collision event until two vehicles are extremely close. Therefore, The DRD-PB lets the vehicle coast for a while and then applies a relatively harsh brake, for which the driver's ride comfort can deteriorate. The shape of this acceleration profile in Fig. 5.10(b) mismatches with the driver's desired action, which can lead to a feeling of heterogeneity. When it comes to the proposed POPD algorithm, benefiting from the learned Wiedemann's constraints, headway constraints, and personalized weights from the driver's historical data and the predictive model, the POPD can realize the collision before it happens. As shown in Fig. 5.10(a), (b), and (d), the proposed POPD generates the braking action that matches the human's behavior in a, Δv and d_{rel}. Moreover, the change of acceleration is smooth, which eliminates the jerky operation during the braking.

To statistically analyze the behavior of the DRD-PB and the POPD in each component of L, the difference of $\dot{a}$, a and v_{rel} to the human behavior from both methods are plotted into histograms as shown in Fig. 5.11. Because the driver's behavior contains stochasticity and is also influenced by every day's mood and purpose of driving, which cannot be quantified easily. Therefore, the errors in each component of L can only be reduced but not eliminated. The blue and red solid lines in each subplot of Fig. 5.11 represent the probability density function (PDF) of the DRD-PB and the POPD in the validation set. The standard deviation $\sigma = [\sigma_{\dot{a}}, \sigma_a, \sigma_v]$ of POPD is $[0.467, 0.413, 0.891]$, which is smaller than $[0.479, 0.425, 0.949]$ of the DRD-PB in every perspective. Therefore, it can be concluded that with the POPD, braking operation becomes more similar to human behavior than that with the DRD-PB. The statistical analysis of three additional drivers is listed in Table 5.1, clearly indicating that the POPD outperforms the DRD-PB in terms of σ for every independent driver.

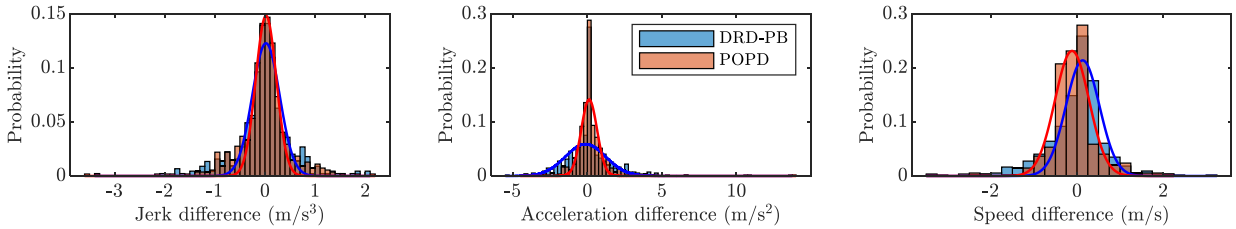

Figure 5.11: Probability distribution comparison between DRD-PB and POPD to the human driver: the probability density function shows the brake action generated from POPD is more similar to humans than the DRD-PB. The mean and standard deviation are listed as driver #3 in Table 5.1.

Table 5.1: Statistics of three selected drivers:

Driver	Method	$\dot{a}$ error (m/s^3)		a error (m/s^2)		Δv error (m/s)	
		μ	σ	μ	σ	μ	σ
1	DRD-PB	0.032	0.479	-0.001	0.425	0.132	0.949
	POPD	-0.023	0.467	-0.088	0.413	0.192	0.891
2	DRD-PB	0.025	0.494	-0.038	0.617	-0.103	2.00
	POPD	-0.029	0.374	-0.027	0.295	-0.224	0.767
3	DRD-PB	-0.017	0.560	-0.023	0.706	0.093	1.626
	POPD	-0.023	0.522	-0.088	0.545	0.171	1.181

5.7 Experimental Validation

In order to investigate the real-time performance of the proposed POPD algorithm within a complex driving environment with multiple drive-lanes, other vehicles surrounding, and a driver's intervention, a human-in-loop test is conducted on the virtual framework developed in [113, 114].

The state-space model in Eq. (5.7), the convex control problem in Eq. (5.12) and its constraints are converted into their stack-state representation and solved in a quadratic programming manner. The quadratic programming solver used in current code generation is MATLAB® quadprog.m. The average computation time for each step is around 3×10^{-4} second handled by a desktop with 16 GB memory and Intel Core i9 9900K 3.6 GHz CPU. This computation time is fast enough to support a real-time implementation.

5.7.1 Driver Simulator Setup

This virtual framework contains an urban city route that is duplicated from Ann Arbor, Michigan, as shown in Fig. 5.12(a). Every road segment contains multiple traffic lanes, traffic signals, stop signs, and elevation profiles extracted from the real world. The visualization of the aforementioned road conditions is achieved by CarMaker® [115], then displayed on three high-resolution TVs to mimic the driver's front windshield view and side window views in Fig. 5.12(b) and (c), respectively.

As shown in Fig. 5.12 (b), every component of the cockpit is from an actual production vehicle factory. The spring and damper of the gas/brake pedal are adequately tuned to mimic the force feedback from the actual vehicle's pedal. An electric motor properly controls the resistive force of the steering wheel to mimic the sense of steering in an actual vehicle.

During the test, the driver controls the ego vehicle's direction with the steering wheel and the

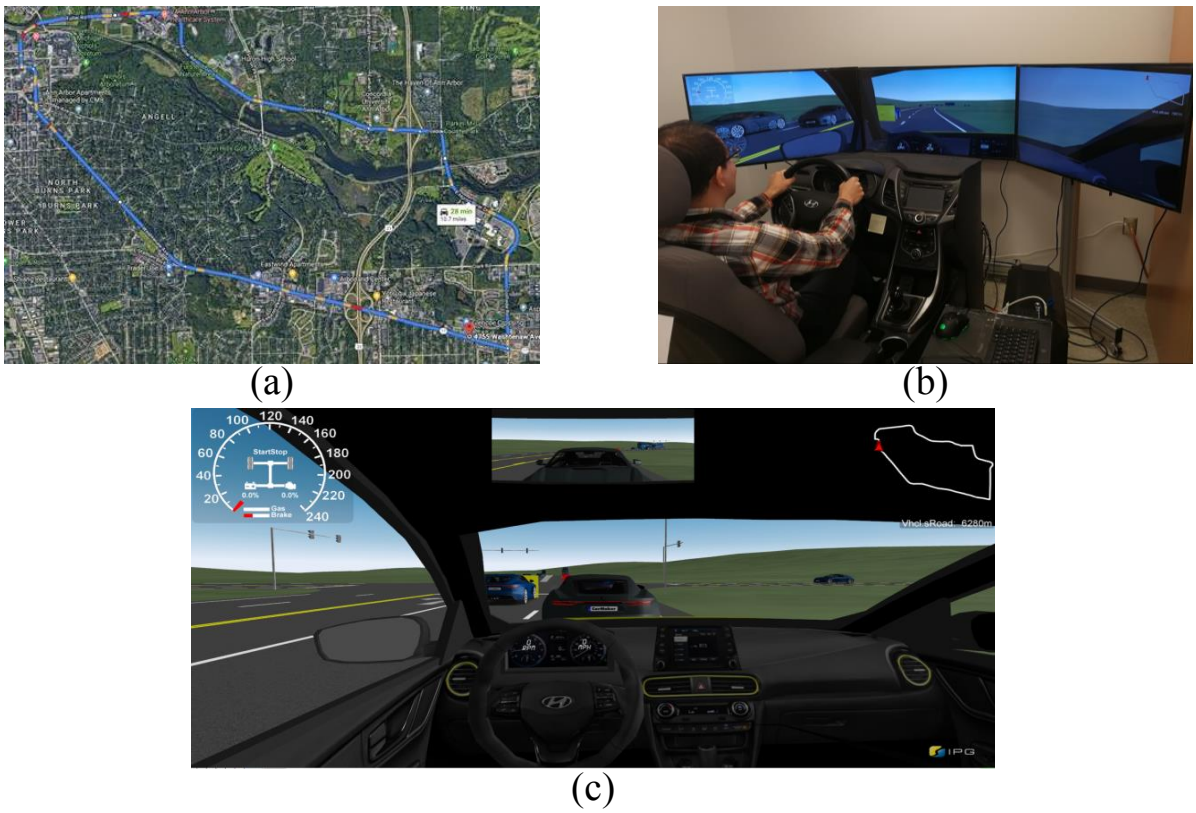

Figure 5.12: Human-in-the-loop co-simulation environment: (a) driving route from Ann Arbor area, (b) simulator setup (c) simulator interface

acceleration/deceleration with the acceleration and brake pedals, like driving a real-world vehicle. These input signals are collected by CarMaker® and then passed to GT-suite® [116] for powertrain and vehicle dynamics simulation. It is noted that a lane shift is made by the driver's decision when it is needed. In the meantime, PTV VISSIM® [117] generates random numbers of robot sedans and robot buses to the CarMaker® environment that share the same road with the test vehicle. These robot vehicles are programmed by VISSIM® and follow a similar behavior to real-world drivers. Robot vehicles make lane shifts as they desire and wait for stop signs and red traffic signals. The traffic volume of randomly generated robot vehicles is obtained through the Michigan department of transportation website [118]. There is a possibility that neighboring lane vehicles will cut in with high relative speed, which will be a chance to test the safety level of the proposed POPD. The POPD algorithm is compiled from Matlab/SIMULINK® to C-code and is called by the CarMaker® master environment.

5.7.2 Experimental Results

During the experiment, two different drivers are involved. Each driver performs two test drives on the same route, with and without the proposed algorithm implemented. In order to show the full potential of the POPD, the drivers' behavior is fully learned before the test drives. The overall trip time, braking time by human and braking time by the algorithm of each driver are summarized in Table 5.2.

As shown in Table 5.2, because of the driving style difference, each driver has completed the test drive with different trip times. In addition, the randomness in the traffic simulation (e.g., neighboring vehicles and traffic signals) makes even the same driver complete the same route with different trip times. Without the POPD algorithm, the overall braking time occupies 31.3% of driver A's overall trip time and 16.7% of driver B's trip time. However, with POPD included, the human braking time is reduced to 5.25% and 3.56% of the total trip time, respectively, which significantly reduces the driver's brake pedal efforts. Note that the current POPD algorithm does not use any navigation information, which means that the human driver is responsible for braking events caused by traffic signals, stop signs, and slowing down for turning left/right.

The time-series response of driver A's ego vehicle velocity and the preceding vehicle velocity is shown in Fig. 5.13 (a). The desire and actual acceleration, relative distance to the preceding vehicle, and relative speed to the preceding vehicle are listed in Fig. 5.13 (b), (c), and (d), respectively. The algorithm-generated braking signal, human override signal, and algorithm activation indicator over time are shown in Fig. 5.13 (e). Overall, Fig. 5.13 (a) shows that driver A can track the preceding vehicle's speed relatively well with the POPD implemented onto the ego vehicle. Fig. 5.13 (b) demonstrates that the actual acceleration always matches the desired acceleration if the driver does not override the algorithm. It is worth noting that the d_{rel} in Fig. 5.13 (c) is set to 0 if the preceding vehicle does not exists. Otherwise, the constraints of the relative distance can be satisfied most of the time.

From the time 1000 to 1200 seconds, there are several consecutive stop signs, and human driver A would have to press the brake pedal by himself in Fig. 5.13(e). With navigation information

Table 5.2: Comparison of simulator results with/without the POPD

	Driver A		Driver B	
	w/o POPD	w/ POPD	w/o POPD	w/ POPD
Trip time (s)	1345	1383	1635	1623
Brake time (s)	425.6	72.7	273.0	57.8
POPD usage (s)	-	540.9	-	633

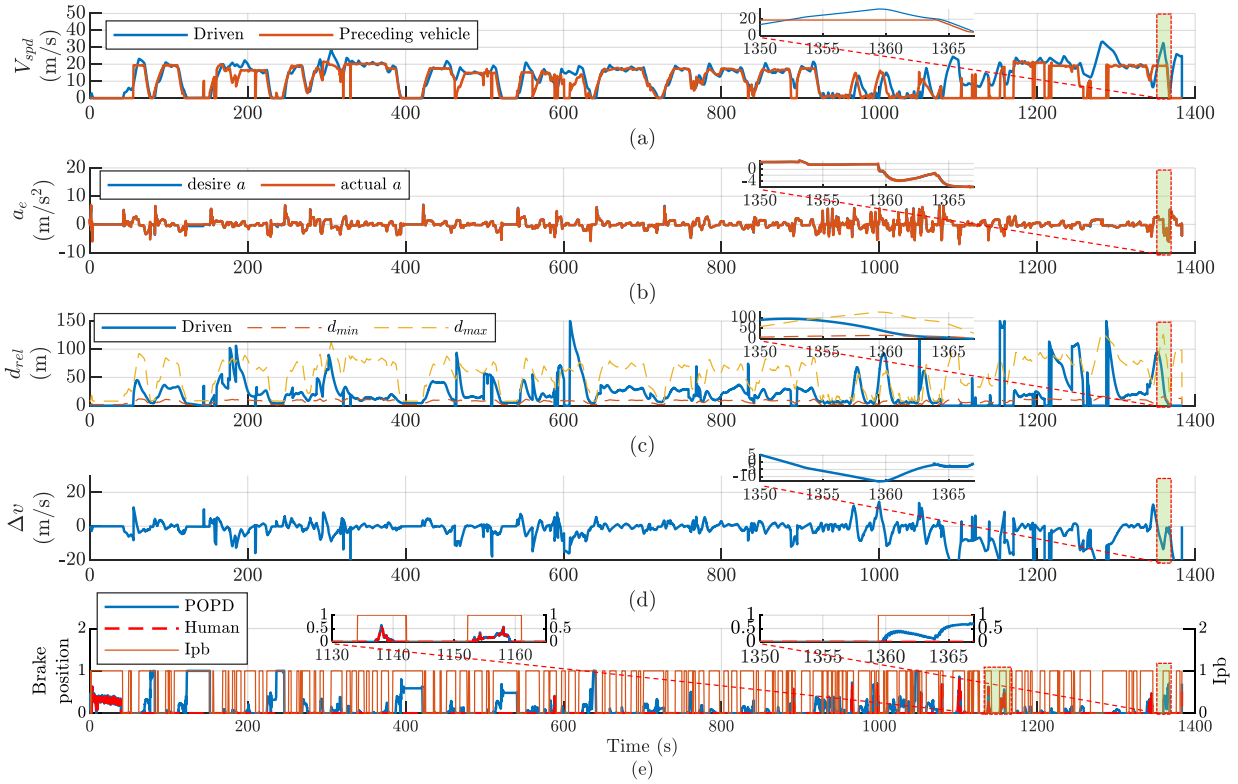

Figure 5.13: Human-in-loop experimental results: (a) the ego vehicle velocity compared to the preceding vehicle, (b) desired and actual acceleration, (c) relative distance between the ego vehicle and the preceding vehicle and constraints (d) relative velocity between the ego vehicle and the preceding vehicle, (e) brake pedal position from POPD, brake pedal position from human driver, algorithm activation indicator (Ipb).

included in the future, the human brake usage in the POPD case can be further decreased from 5.25% (or 3.56%) to an even lower value. During driving, the human driver can override the operation from the POPD with the gas/brake pedal when feeling uncomfortable. As Table 5.2 indicates, the driver merely overrides the POPD action except for traffic/ stop signs, meaning that the driver accepts the generated deceleration most of the time. Throughout the entire trip, there was no collision observed, and the test driver did not feel heterogeneity when POPD engaged, as shown in Fig. 5.13.

During the time 1350 seconds to 1360 seconds in Fig. 5.13(a) and (d), driver A intentionally pushes the acceleration pedal, approaches the preceding vehicle at high speed, and releases the acceleration pedal when two vehicles are extremely close. At the time 1358 seconds when POPD was activated, it successfully recognized the danger of collision and reduced the vehicle's speed with more consideration for safety and less consideration of comfort, as shown in Fig. 5.13 (b), (c) and (e).

5.8 Summary

This study proposes a personalized one-pedal-driving (POPD) algorithm using learning-based MPC. To capture a driver's unique characteristics, two personalized constraints, namely time headway and perceptual constraints, are identified based on the analysis of real-world on-road driving data. Learning a driver's braking behavior is realized through the optimization of the weighting factors in the MPC cost function with particle swarm optimization using historical driver data. Specifically, an evaluation metric is introduced to judge the similarity between the controller's and the human driver's actions. With this evaluation metric, optimal weights learning is conducted to ensure the controller's action matches the individual driver's expectation.

A comparative case study shows that predicting a preceding vehicle's speed with a constant acceleration assumption leads to a similar level of control performance compared to the case with a precise estimation of the preceding vehicle's speed. The comparative study also reveals that the optimal prediction horizon for this car-following control problem is 8 seconds. An open-loop simulation with 450 cycles from different drivers was conducted to test the proposed POPD's capability of matching driver's braking behavior. The simulation results show that, on average, the proposed POPD algorithm outperforms our earlier developed desired relative distance model-based personalized braking algorithm by over 50% in terms of similarity to the individual human drivers.

In addition, a human-in-loop experiment was conducted under a GT-Suite®/CarMaker®/SIMULINK®/VISSIM® co-simulation environment. Over this simulator test with two different drivers, the proposed POPD algorithm proved its effectiveness and real-time implementability in a real-life driving environment. Furthermore, by implementing the POPD algorithm, both drivers can finish the same drive cycle with only one pedal control in most scenarios, which dramatically reduces the deceleration pedal's effort. Meanwhile, both drivers are satisfied with the deceleration generated from the POPD as the driver merely overrides the brake signals.

CHAPTER 6

Conclusion

6.1 Summary of Contribution

Due to the increasing transportation demand and strict greenhouse gas emissions regulations, vehicle industries have shifted their focus from conventional ICE vehicles to electrified vehicles. A battery electric vehicle (BEV), which avoids tailpipe emission completely, is considered the ultimate solution to transportation emissions. However, due to charging speed and capacity limitations on battery, currently, EV users are facing the problem of range anxiety and the lack of charging stations. Therefore, hybrid electric vehicles (HEVs), which possess the advantages of both conventional vehicles and BEVs, appear to be a viable solution to cope with such strict emission regulations while mitigating range anxiety and allowing long haul travel.

Among all variations of HEVs, a 48V mild hybrid electric vehicle have brawn much attention. The increased voltage from 12 to 48 volts allows more powerful electronics implementation and the system to exploit the advantage of hybrid architecture. Moreover, a 48V system still avoids the extra expense of wiring and electric shock protection. To promote the utilization of a 48 V system, a P0+P4 dual motor configuration is considered. A less expensive P0 motor connected to the engine crankshaft can help with engine start. This P0 motor also provides power assist when the requested torque on the front axle exceeds the wide-open-throttle torque of the engine. Furthermore, the P0 motor captures energy when the rear stronger P4 motor regenerative braking capability is limited due to vehicle stability constraints. The overall P0+P4 configuration allows the driver to choose among different modes: front drive, rear drive, and electric all-wheel-drive. In the mean time, recent advances in connected and automated vehicle technologies have created vast opportunities to improve the efficiency of electrified vehicles. Thus, this dissertation aims at optimizing powertrain and vehicle dynamics to achieve the full potential of fuel economy of the P0+P4 system while satisfying safety and ride comfort requirements. When a certain level of passenger comfort is required, co-optimizing powertrain and vehicle dynamics may not achieve significant improvements in energy efficiency; therefore, this dissertation adopts a less computa-

tionally expensive hierarchical control approach to vehicle power-split and longitudinal motion in a sequential manner.

Despite the P0+P4 MHEV's advantages, the three power source structure brings extra difficulties in designing a real-time torque split strategy. In this dissertation, the safety, ride comfort, and fuel economy of a P0+P4 MHEV are optimized in a sequential manner: torque split optimization and vehicle longitudinal motion optimization. Moreover, personalized behaviors of an individual driver are considered in control design to further improve the drivability and ride comfort through a one-pedal-driving feature.

To begin with, the detailed P0+P4 MHEV model considers realistic constraints are developed. Those realistic constraints include nonlinear tire, longitudinal load transfer and brake constraints due to vehicle safety. Then, the dynamic programming analysis of the P0+P4 MHEV is conducted on three standard test cycles such as the WLTC, the UDDS and the HWFET. Finally, through analyzing the global optimal solution of energy consumption and torque distribution, two important features are derived from the DP results:

1. The P0 motor is hardly used for propulsion unless the demanding torque at the front axle exceeds the engine's maximum torque.

2. In the case of braking, the torque distribution at the front and the rear axle has a clear relationship with the current deceleration.

Due to the first feature, the real-time torque split strategy can exclude the P0 motor from the control variables. Hence, the power split during the propulsion is solved with one of the well-known optimization-based techniques, A-ECMS. The second feature is captured by a modified logistic function in equation (2.30). During the braking event, the torque distribution can be determined through the diagram shown in figure 2.8. The overall control framework is named approximated A-ECMS for P0+P4 MHEVs.

In addition to the optimization-based strategy, this dissertation proposes another learning-based torque-split algorithm using a twin delayed deep deterministic policy gradient algorithm with prioritized exploration and experience replay (TD3-PEER). Due to the data-driven characteristic, this TD3-PEER can be easily migrated to a similar hybrid architecture or a BEV with multiple different-sized motors. The simulated training process with different seeds demonstrated the stability of the proposed TD3 algorithm. The case study also proves that the expert knowledge from DP results helps the agent achieve much better fuel consumption. The proposed TD3 method outperforms state-of-art DDPG+PER and DQN in fuel economy among all five standard test cycles.

Regarding the vehicle longitudinal motion, this dissertation proposes a defensive ecological adaptive cruise control strategy (DEco-ACC). The DEco-ACC formulates the multi-lane car-following system into MPC problem, and blind spot zone of neighboring vehicles as a one-time dif-

ferentiable continuous penalty function. Unlike the traditional ACC or Eco-ACC, this DEco-ACC allows the vehicle proactively avoid the blind spot zone of the neighboring vehicles within its prediction horizon. The simulation results show that with two neighboring vehicles present simultaneously, the DEco-ACC reduces 29.5% of dwelling time in the blind spot and only scarifies 0.4% of fuel consumption. The case study about four neighboring vehicles during car following is presented in the appendix.

In complex urban driving scenarios, the DEco-ACC may be disabled due to many stop signs and traffic signals. To cope with such situation, this dissertation proposes a personalized one-pedal-driving algorithm (POPD) based on a learning-based model predictive controller whose parameters are trained with the driver's historical braking data. With the POPD, braking actions will be performed automatically during car-following. During the driving, the driver may press the brake/acceleration pedal to override the actions done by the POPD controller temporarily. Such overriding behavior will be recorded for future parameter learning improvements. The proposed POPD algorithm is validated through a Human-in-the-loop co-simulation with complex driving scenarios duplicated from the Ann Arbor area in Michigan, including multiple traffic lanes, stop signs, roundabouts, and traffic signals. The experiment shows that the brake pedal usage is reduced from 31.3% to 5.25% for human driver A and from 16.7 to 3.56% for human driver B.

6.2 Possible Future Extension

This dissertation proposes four novel algorithms in terms of power-split and longitudinal motion control for a P0+P4 MHEV. These proposed algorithms are proven effective in improving energy efficiency, safety, and ride comfort with the human factor sufficiently considered. Nevertheless, the following can be considered opportunities for future research to advance the presented work.

6.2.1 MHEV Power-split Combined with Trip Information

The power-split strategy for an MHEV is often designed as a charge-sustaining mode that does not require the MHEV's battery to be physically plugged in for charging. However, the battery SOC in charge-sustaining mode should be well regulated around a reference level to cope with the motor's potential regenerative braking or power assist events. Such regulation on battery SOC highly restricts the current power-split policy and may conflict with a potentially more energy-optimal solution over the entire trip [22, 119–122].

Due to the rapid development of vehicle connectivity technologies, future trip information may be available before or during the trip, such as road gradient and traffic volume in a certain area. With this trip information available, an improved torque-split algorithm may reach even lower

energy consumption over the trip. The studies in [123–132] have demonstrated the potential of such combinations. The energy efficiency of the proposed AA-ECMS and TD3 may be enhanced after the trip information included. Also, future work will include fuel economy and dynamic performance of the proposed strategy through vehicle testing for experimental verification.

6.2.2 Thermal System Integrated Control

This dissertation optimizes the power-split and longitudinal motion of a P0+P4 MHEV in a hierarchical manner. And, it has been proved in [12] that the integrated control brings extra computation complexity but only improves little energy efficiency. During this study, the engine, electric machines and battery temperature are assumed to be well-regulated. However, it is known that the control of a cooling system within HEVs and EVs are also an active research area [133–141], as the cooling effort influences to the overall energy consumption as well. An optimized control action in power-split or vehicle longitudinal motion may lead to extra cooling effort. Thus, the influence of cooling effects on the decision of power-split and vehicle longitudinal motion is worth studying. [142–145] Future work will investigate the impact of thermal effects, such as cabin and battery heating/cooling, on the power management problem. In addition, the performance when combining an off-line RL technique with dynamic programming results will also be investigated.

6.2.3 Impact of Different Risk Penalty Functions on DEco-ACC Performance

To ensure the real-time implementability of DEco-ACC, it is desired that the penalty function of blind spots considered in Chapter 4 is a continuous, differentiable and symmetrical function, where the mid-point of the blind spot zone has the highest penalty/risk. However, for several types of neighboring vehicles, the projections of their lengths to the ego traffic lane do not overlap with the mid-point of their blind spots. Therefore, it would be more rigorous if the highest risk could be assigned to the overlapped region of neighboring vehicles' length projection and blind spots.

Future work will investigate an improved version of the risk function setup that eliminates the dwelling time within both blind spots and the neighboring vehicles' length projection.

6.2.4 Enable DEco-ACC to Learn from Human

The study from Chapter 5 demonstrates that different drivers have different preferences on car-following time headway, distance headway, and time-to-collision. Therefore, personalized driving behavior from the individual driver can be considered further to stretch the ride comfort of DEco-ACC. However, unlike the POPD, which can frequently collect a driver's gas pedal and

brake pedal position data, a driver usually does not override actions from the ACC controller unless they turn off the ACC mode. Therefore, a novel data collection approach should be developed. Several existing pieces of literature about personalized ACC can be a reference [105, 146–152].

Moreover, Chapter 4 assumes that all states required by DEco-ACC can be observed accurately. However, in real-world operation, states acquired from sensors may be subject to measurement error and noise. It is worth exploring the performance of DEco-ACC with the existence of state estimation/observation error. The implementation of robust MPC on DEco-ACC can be a potential solution to this future direction.

6.2.5 POPD Dealing with Traffic Signals and Stop Signs

The current POPD algorithm proposed in Chapter 5 considers the motion of preceding vehicles, which allows it to perform personalized braking if a preceding vehicle exists. However, traffic signals and stop signs in urban driving can also cause vehicle braking events. With the current formulation, the driver still needs to press the brake pedal if traffic signals or stop signs cause the brake event. Hence, a possible extension to the POPD algorithm is to include the traffic signal and stop signs in the MPC formulation [153–155]. The information on traffic signals and stop signs can be either collected from the vehicle onboard sensors [156–162] or by vehicle-to-infrastructure communications [163–168]. Moreover, to validate the adaptability of the POPD in complex real-world driving environments, a real vehicle experiment is needed. And, the energy efficiency improvement of POPD can be rigorously studied and compared with the rule-based OPD.

APPENDIX

Deco-ACC with Four Neighboring Vehicles

The simulation shown in section 4.4.3 is mainly focused on the controller's performance at high-speed operation with two neighboring vehicles (2NVs). When a traffic condition changes (e.g., heavy traffic or urban driving), the number of neighboring vehicles could increases. Therefore, in this Appendix, the DEco-ACC is modified to handle a 4-NVs scenario and its performance is presented through low speed car-following scenarios with more neighboring vehicles surrounded from the NGSIM data. More specifically, 10 cases of 4-NVs scenarios are extracted from real-world NGSIM data.

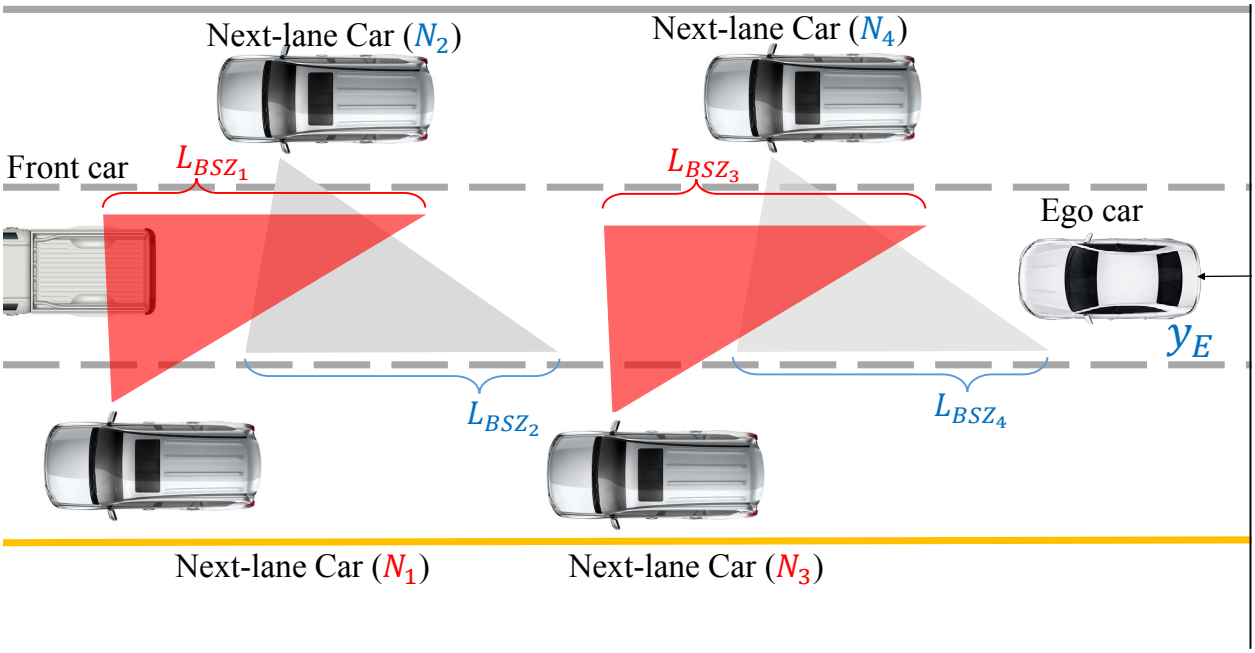

Figure A.1: A concept of car-following in consideration of the BSZs of four neighboring vehicles.

In a 4-NVs scenario, the state-space matrices in Eq. (4.9) are modified as follows:

$$A = \begin{bmatrix} 1 & 0 & 0 & -T_s & -\frac{1}{2}T_s^2 & 0 & 0 \\ 0 & 1 & 0 & -T_s & -\frac{1}{2}T_s^2 & 0 & 0 \\ 0 & 0 & 1 & -T_s & -\frac{1}{2}T_s^2 & 0 & 0 \\ 0 & 0 & 0 & 1 & T_s & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -T_s & -\frac{1}{2}T_s^2 & 1 & 0 \\ 0 & 0 & 0 & -T_s & -\frac{1}{2}T_s^2 & 0 & 1 \end{bmatrix},$$

$$B^T = \begin{bmatrix} -\frac{T_s^3}{6} & -\frac{T_s^3}{6} & -\frac{T_s^3}{6} & \frac{T_s^2}{2} & T_s & -\frac{T_s^3}{6} & -\frac{T_s^3}{6} \end{bmatrix}^T,$$

$$d_k^T = \begin{bmatrix} T_s v_p & T_s v_{n1} & T_s v_{n2} & 0 & 0 & T_s v_{n3} & T_s v_{n4} \end{bmatrix}^T,$$

with state x and input u as

$$x_k = \begin{bmatrix} Y^{\Delta PE} \\ Y^{\Delta NE1*} \\ Y^{\Delta NE2*} \\ v_e \\ a_e \\ Y^{\Delta NE3*} \\ Y^{\Delta NE4*} \end{bmatrix}_k, \; u_k = \dot{a}_k.$$

where $Y^{\Delta NE3*}$ and $Y^{\Delta NE4*}$ are blind spots constraints of neighboring vehicle 3 and neighboring vehicle 4 which are defined in a similar way shown in Eq. (4.8). The following constraints are included to address the neighboring vehicles 3 and 4:

$$|Y^{\Delta NE3*}| \geq \frac{1}{2} L_{BSZ3},$$
$$|Y^{\Delta NE4*}| \geq \frac{1}{2} L_{BSZ4}.$$

After including the slack variable and mode signal for neighboring vehicles 3 and 4, the cost function of this incremental Deco-ACC controller is modified as follows:

$$\begin{aligned} J = \sum_{k=0}^{N-1} & u_k^2 P_1 + (a_{e,k})^2 P_2 + (v_{e,k} - v_{p,k})^2 P_3 \\ & + (\delta_{slack,1} \times mode_1 + \delta_{slack,2} \times mode_2 \\ & + \delta_{slack,3} \times mode_3 + \delta_{slack,4} \times mode_4) P_4 \end{aligned}$$

Table A.1: Time span and average preceding vehicle velocity of each car following scenario with four neighboring vehicles.

Scenario #	Length in second	Average speed v_p (m/s)
1	45	5.36
2	45	6.17
3	50	6.66
4	30	7.47
5	30	8.14
6	20	6.31
7	25	8.58
8	35	9.14
9	30	9.69
10	35	8.89

As shown in Table A.1, a total of 10 real driving scenarios are extracted with a preceding vehicle's velocity ranging from 5.36 m/s to 8.89 m/s which is much lower than 23.9 m/s from the case study in Section 4.4.3. Since the ego vehicle is rarely surrounded by four neighboring vehicles simultaneously, each realistic scenario is relatively short, ranging from 20 seconds to 50 seconds.

The comparison of both controllers among all 10 cases are summarized in Fig. A.2. The fuel consumption and dwelling time of DEco-ACC controller are normalized with respect to the values obtained with the Eco-ACC controller. It can be seen that the DEco-ACC controller can reduce the BSZ dwelling time in the most cases, and that as discussed in Section 4.4, an additional fuel consumption could be led by the DEco-ACC.

Figure A.3 compares the operations of the DEco-ACC and Eco-ACC for a specific 4-NVs scenario in detail. In Fig. A.3(a), the filled areas show the trajectory of the BSZ associated with each neighboring vehicle. From Fig. A.3(a), 2 s to 8 s, the DEco-ACC controller commands the ego vehicle to drive at a lower speed in order to avoid the blind spot of neighboring vehicle N4. However, the ego vehicle with Eco-ACC controller ignores the blind spot and drives into it. A similar behavior can be observed from 19 seconds to 21 seconds, the DEco-ACC controller maintains the ego vehicle outside of the blind spots of N4 and visible to other neighboring vehicle drivers which could improve the safety from lane-shifting-collision. However, the ego vehicle with Eco-ACC controller drives directly into the blind spots from time 19 to 21 seconds.

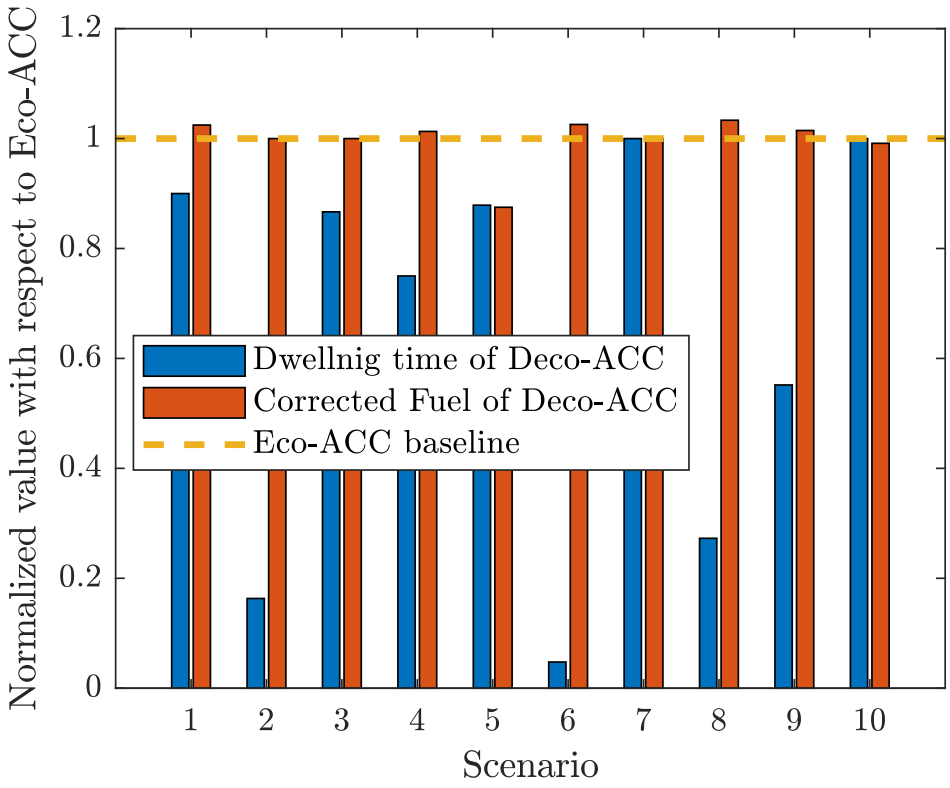

Figure A.2: Performance comparison of the Deco-ACC in 10 real driving scenarios including four neighboring vehicles.

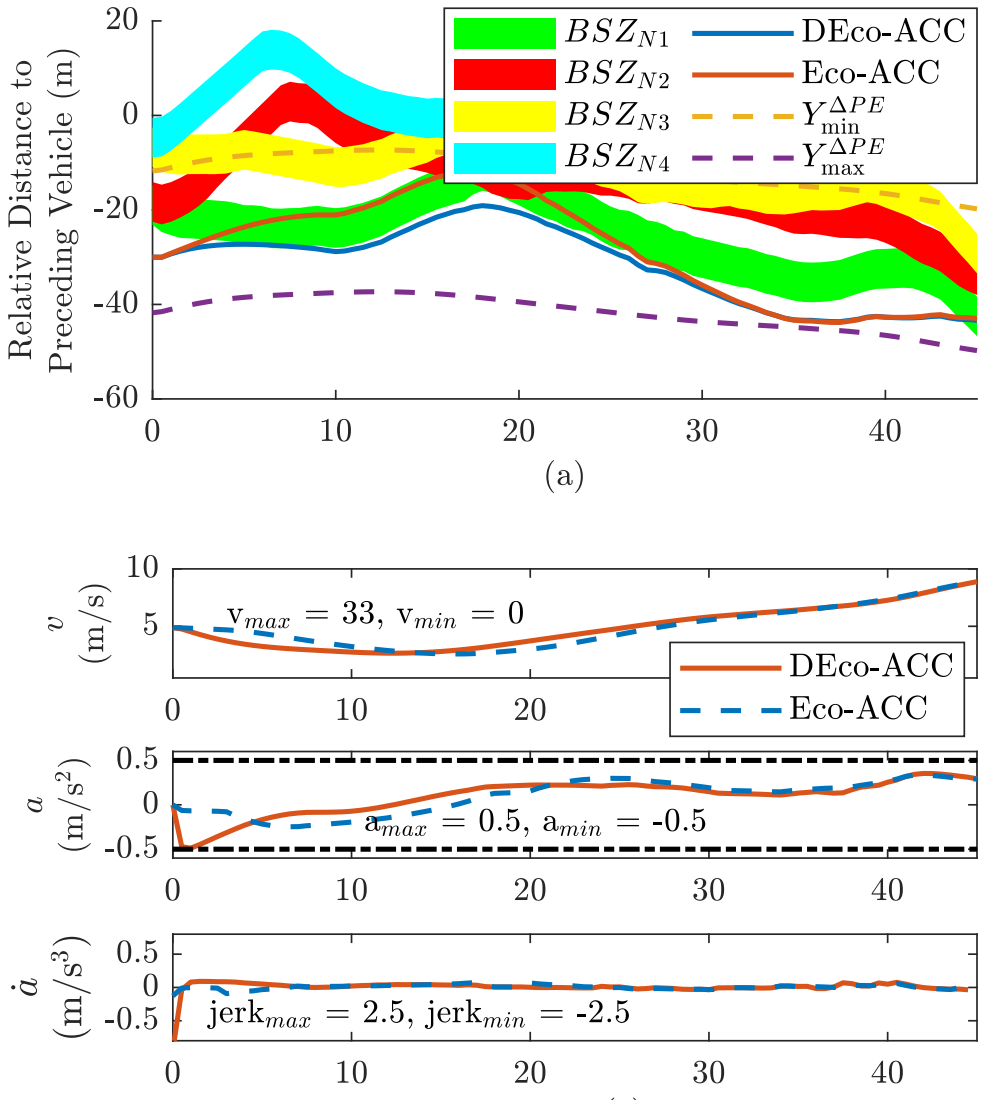

Figure A.3: Comparison of trajectories with DEco-ACC and Eco-ACC at lower speed operation when there are four neighboring vehicles: (a) displacement of each vehicle, relative to the preceding vehicle, (b) velocity, acceleration, and jerk

BIBLIOGRAPHY

[1] "Energy and the environment explained outlook for future emissions," https://www.eia.gov/energyexplained/energy-and-the-environment/outlook-for-future-emissions.php, accessed: 2022-08-31.

[2] "Final rule to revise existing national ghg emissions standards for passenger cars and light trucks through model year 2026," https://www.epa.gov/regulations-emissions-vehicles-and-engines/final-rule-revise-existing-national-ghg-emissions, accessed: 2022-08-31.

[3] "Types of mild hybrid electric vehicles (mhev)," https://x-engineer.org/mild-hybrid-electric-vehicles-mhev-types/, accessed: 2022-10-18.

[4] National Highway Traffic Safety Administration Environmental Protection Agency, "2017 and later model year light-duty vehicle greenhouse gas emissions and corporate average fuel economy standards; final rule," 2017.

[5] Z. Liu, A. Ivanco, and Z. S. Filipi, "Impacts of real-world driving and driver aggressiveness on fuel consumption of 48V mild hybrid vehicle," *SAE International Journal of Alternative Powertrains*, vol. 5, no. 2, pp. 249–258, 2016.

[6] E. Song, L. Fan, G. Liu, and W. Long, "Numerical simulation of combination engine hev on fuel economy," in *2010 WASE International Conference on Information Engineering*, vol. 4. IEEE, 2010, pp. 244–249.

[7] M. Kuypers, "Application of 48 volt for mild hybrid vehicles and high power loads," in *SAE Technical Paper*, 2014, no. 2014-01-1790.

[8] "Mild hybrids—a multi-billion euro growth opportunity alongside e-mobility," https://www.automotiveworld.com/articles/mild-hybrids-a-multi-billion-euro-growth-opportunity-alongside-e-mobility/, accessed: 2020-08-10.

[9] P. Biswas, "Adapting SUV AWD powertrain to P0/P2/P4 hybrid EV architecture: Integrative packaging and capability study," in *2017 IEEE Transportation Electrification Conference (ITEC-India)*. IEEE, 2017, pp. 1–5.

[10] S. Lee, J. Cherry, M. Safoutin, A. Neam, J. McDonald, and K. Newman, "Modeling and controls development of 48 V mild hybrid electric vehicles," in *SAE Technical Paper*, 2018, no. 2018-01-0413.

[11] M. Werra, A. Sturm, and F. Küçükay, "Optimal and prototype dimensioning of 48 V P0+P4 hybrid drivetrains," *Automotive and Engine Technology*, pp. 1–14, 2020.

[12] D. Chen, M. Huang, A. G. Stefanopoulou, and Y. Kim, "Co-optimization of velocity and charge-depletion for plug-in hybrid electric vehicles: Accounting for acceleration and jerk constraints," *Journal of Dynamic Systems, Measurement, and Control*, vol. 144, no. 1, 2022.

[13] D. Lodaya, J. Zeman, M. Okarmus, S. Mohon, P. Keller, J. Shutty, and N. Kondipati, "Optimization of fuel economy using optimal controls on regulatory and real-world driving cycles," in *SAE Technical Paper*, 2020, no. 2020-01-1007.

[14] K. Kuwabara, J. Karl-DeFrain, S. Midlam-Mohler, M. K. Satra, and A. Narasimhan Ramakrishnan, "Model-based design of a hybrid powertrain architecture with connected and automated technologies for fuel economy improvements," in *SAE Technical Paper*, 2020, no. 2020-01-1438.

[15] T. Hofman, M. Steinbuch, R. Van Druten, and A. Serrarens, "Rule-based energy management strategies for hybrid vehicles," *International Journal of Electric and Hybrid Vehicles*, vol. 1, no. 1, pp. 71–94, 2007.

[16] A. M. Ali and D. Söffker, "Towards optimal power management of hybrid electric vehicles in real-time: A review on methods, challenges, and state-of-the-art solutions," *Energies*, vol. 11, no. 3, p. 476, 2018.

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

[18] S. G. Li, S. M. Sharkh, F. C. Walsh, and C.-N. Zhang, "Energy and battery management of a plug-in series hybrid electric vehicle using fuzzy logic," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 8, pp. 3571–3585, 2011.

[19] G. Jinquan, H. Hongwen, P. Jiankun, and Z. Nana, "A novel mpc-based adaptive energy management strategy in plug-in hybrid electric vehicles," *Energy*, vol. 175, pp. 378–392, 2019.

[20] Y. Wang, H. Tan, Y. Wu, and J. Peng, "Hybrid electric vehicle energy management with computer vision and deep reinforcement learning," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 3857–3868, 2020.

[21] Y. Hu, W. Li, K. Xu, T. Zahid, F. Qin, and C. Li, "Energy management strategy for a hybrid electric vehicle based on deep reinforcement learning," *Applied Sciences*, vol. 8, no. 2, p. 187, 2018.

[22] Y. Li, H. He, J. Peng, and H. Wang, "Deep reinforcement learning-based energy management for a series hybrid electric vehicle enabled by history cumulative trip information," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7416–7430, 2019.

[23] M. Volodymyr, K. Koray, S. David, G. Alex, A. Ioannnis, W. Daan, and R. Martin, "Playing atari with deep reinforcement learning. arxiv 2013," *arXiv preprint arXiv:1312.5602*.

[24] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *International conference on machine learning*. PMLR, 2015, pp. 1889–1897.

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

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

[27] M. Sun, P. Zhao, and X. Lin, "Power management in hybrid electric vehicles using deep recurrent reinforcement learning," *Electrical Engineering*, vol. 104, no. 3, pp. 1459–1471, 2022.

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

[29] R. Lian, J. Peng, Y. Wu, H. Tan, and H. Zhang, "Rule-interposing deep reinforcement learning based energy management strategy for power-split hybrid electric vehicle," *Energy*, vol. 197, p. 117297, 2020.

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

[31] R. Huang, H. He, X. Meng, Y. Wang, R. Lian, and Y. Wei, "Energy management strategy for plug-in hybrid electric bus based on improved deep deterministic policy gradient algorithm with prioritized replay," in *2021 IEEE Vehicle Power and Propulsion Conference (VPPC)*. IEEE, 2021, pp. 1–6.

[32] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*. PMLR, 2018, pp. 1861–1870.

[33] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *International conference on machine learning*. PMLR, 2018, pp. 1587–1596.

[34] J. Wu, Z. Wei, W. Li, Y. Wang, Y. Li, and D. U. Sauer, "Battery thermal-and health-constrained energy management for hybrid electric bus based on soft actor-critic drl algorithm," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 3751–3761, 2020.

[35] J. Zhou, S. Xue, Y. Xue, Y. Liao, J. Liu, and W. Zhao, "A novel energy management strategy of hybrid electric vehicle via an improved td3 deep reinforcement learning," *Energy*, vol. 224, p. 120118, 2021.

[36] F. Pardo, "Tonic: A deep reinforcement learning library for fast prototyping and benchmarking," *arXiv preprint arXiv:2011.07537*, 2020.

[37] A. Sciarretta and A. Vahidi, "Energy-efficient speed profiles (eco-driving)," in *Energy-Efficient Driving of Road Vehicles*. Springer, 2020, pp. 131–178.

[38] A. K. Madhusudhanan and X. Na, "Effect of a traffic speed based cruise control on an electric vehicle's performance and an energy consumption model of an electric vehicle," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 386–394, March 2020.

[39] Y. Jia, R. Jibrin, and D. Gorges, "Energy-optimal adaptive cruise control for electric vehicles based on nonlinear model predictive control," in *2019 IEEE Vehicle Power and Propulsion Conference (VPPC)*, Oct 2019, pp. 1–7.

[40] Y. Zhu, D. Zhao, and H. He, "Synthesis of cooperative adaptive cruise control with feedforward strategies," *IEEE Transactions on Vehicular Technology*, pp. 1–1, 2020.

[41] Y. He, Q. Zhou, M. Makridis, K. Mattas, J. Li, H. Williams, and H. Xu, "Multi-objective co-optimization of cooperative adaptive cruise control and energy management strategy for phevs," *IEEE Transactions on Transportation Electrification*, pp. 1–1, 2020.

[42] M. I. Miftakhudin, A. Subiantoro, and F. Yusivar, "Adaptive cruise control by considering control decision as multistage mpc constraints," in *2019 IEEE Conference on Energy Conversion (CENCON)*, Oct 2019, pp. 171–176.

[43] B. Sakhdari, M. Vajedi, and N. L. Azad, "Ecological adaptive cruise control of a plug-in hybrid electric vehicle for urban driving," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 1739–1744.

[44] D. He, W. He, and X. Song, "Efficient predictive cruise control of autonomous vehicles with improving ride comfort and safety," *Measurement and control*, vol. 53, no. 1-2, pp. 18–28, 2020.

[45] M. Á. Sotelo and J. Barriga, "Blind spot detection using vision for automotive applications," *Journal of Zhejiang University-Science A*, vol. 9, no. 10, pp. 1369–1372, 2008.

[46] J. Kim and D. Kum, "Collision risk assessment algorithm via lane-based probabilistic motion prediction of surrounding vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 9, pp. 2965–2976, 2017.

[47] D. Kim, J. S. Eo, and K.-K. K. Kim, "Parameterized Energy-Optimal Regenerative Braking Strategy for Connected and Autonomous Electrified Vehicles: A Real-Time Dynamic Programming Approach," *IEEE Access*, vol. 9, pp. 103 167–103 183, 2021.

[48] S. Zhang and X. Zhuan, "Study on adaptive cruise control strategy for battery electric vehicle," *Math. Probl. Eng.*, vol. 2019, 2019.

[49] J. Guo, W. Li, J. Wang, Y. Luo, and K. Li, "Safe and energy-efficient car-following control strategy for intelligent electric vehicles considering regenerative braking," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 7070–7081, 2021.

[50] S. P. Deligianni, M. Quddus, A. Morris, A. Anvuur, and S. Reed, "Analyzing and modeling drivers' deceleration behavior from normal driving," *Transp. Res. Rec.*, vol. 2663, no. 1, pp. 134–141, 2017.

[51] S. G. Dehkordi, M. E. Cholette, G. S. Larue, A. Rakotonirainy, and S. Glaser, "Energy Efficient and Safe Control Strategy for Electric Vehicles Including Driver Preference," *IEEE Access*, vol. 9, pp. 11 109–11 122, 2021.

[52] C. Wei, E. Paschalidis, N. Merat, A. Solernou, F. Hajiseyedjavadi, and R. Romano, "Human-like Decision Making and Motion Control for Smooth and Natural Car Following," *IEEE Trans. Intell. Veh.*, pp. 1–1, 2021.

[53] K. Min, G. Sim, S. Ahn, M. Sunwoo, and K. Jo, "Vehicle Deceleration Prediction Model to Reflect Individual Driver Characteristics by Online Parameter Learning for Autonomous Regenerative Braking of Electric Vehicles," *Sensors*, vol. 19, no. 19, p. 4171, Sep. 2019.

[54] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, no. 2, pp. 1805–1824, Aug. 2000.

[55] M. U. Cuma, Ç. D. Ünal, and M. M. Savrun, "Design and implementation of algorithms for one pedal driving in electric buses," *Eng. Sci. Technol. Int. J.*, vol. 24, no. 1, pp. 138–144, 2021.

[56] J. Wang, I. Besselink, J. van Boekel, and H. Nijmeijer, "Evaluating the energy efficiency of a one pedal driving algorithm," in *2015 European Battery, Hybrid and Fuel Cell Electric Vehicle Congress (EEVC 2015)*, 2015.

[57] S. Yang, Z. Su, and P. Chen, "Robust inter-vehicle spacing control for battery electric vehicles with one-pedal-driving feature," in *2021 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2021, pp. 259–264.

[58] D. Schafer, M. Lamantia, and P. Chen, "Modeling and spacing control for an electric vehicle with one-pedal-driving feature," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 166–171.

[59] J. Van Boekel, I. Besselink, and H. Nijmeijer, "Design and realization of a one-pedal-driving algorithm for the tu/e lupo el," *World Electr. Veh. J.l*, vol. 7, no. 2, pp. 226–237, 2015.

[60] Y. Saito and P. Raksincharoensak, "Risk predictive haptic guidance: Driver assistance with one-pedal speed control interface," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2017, pp. 111–116.

[61] L.-W. Chen and G.-L. Wang, "Risk-aware and collision-preventive cooperative fleet cruise control based on vehicular sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 1, pp. 179–191, 2021.

[62] D. Lang, T. Stanger, and L. del Re, "Opportunities on fuel economy utilizing v2v based drive systems," SAE Technical Paper, Tech. Rep., 2013.

[63] S. Darbha, S. Konduri, and P. R. Pagilla, "Effects of v2v communication on time headway for autonomous vehicles," in *2017 American control conference (ACC)*. IEEE, 2017, pp. 2002–2007.

[64] Y. He, K. H. Kwak, Y. Kim, D. Jung, J. H. Lee, and J. Ha, "Real-time torque-split strategy for p0+ p4 mild hybrid vehicles with eawd capability," *IEEE Transactions on Transportation Electrification*, vol. 8, no. 1, pp. 1401–1413, 2021.

[65] Y. He, Y. Kim, D. Y. Lee, and S.-H. Kim, "Defensive ecological adaptive cruise control considering neighboring vehicles' blind-spot zones," *IEEE Access*, vol. 9, pp. 152 275–152 287, 2021.

[66] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger, "Learning-based model predictive control: Toward safe learning in control," *Annu. Rev. Control Robot. Auton. Syst.*, vol. 3, pp. 269–296, 2020.

[67] C. Musardo, G. Rizzoni, Y. Guezennec, and B. Staccia, "A-ECMS: An adaptive algorithm for hybrid electric vehicle energy management," *European Journal of Control*, vol. 11, no. 4-5, pp. 509–524, 2005.

[68] Z. Zhu, Y. Liu, and M. Canova, "Energy management of hybrid electric vehicles via deep Q-networks," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 3077–3082.

[69] A. Chasse, A. Sciarretta, and J. Chauvin, "Online optimal control of a parallel hybrid with costate adaptation rule," *IFAC proceedings volumes*, vol. 43, no. 7, pp. 99–104, 2010.

[70] H. B. Pacejka and E. Bakker, "The magic formula tyre model," *Vehicle System Dynamics*, vol. 21, no. sup001, pp. 1–18, 1992.

[71] United Nation Economic Commission for Europe, "Uniform provisions concerning the approval of vehicles of categories M, N and O with regard to braking. Addendumc 12: Regulation No. 13," On the WWW, Mar 2014, uRL `https://www.unece.org/`.

[72] P. Dekraker, D. Barba, A. Moskalik, and K. Butters, "Constructing engine maps for full vehicle simulation modeling," in *SAE Technical Paper*, 2018, no. 2018-01-1412.

[73] O. Sundstrom and L. Guzzella, "A generic dynamic programming matlab function," in *2009 IEEE Control Applications, (CCA) Intelligent Control, (ISIC)*, 2009, pp. 1625–1630.

[74] A. Pennycott, L. D. Novellis, P. Gruber, and A. Sorniotti, "Optimal braking force allocation for a four-wheel drive fully electric vehicle:," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 2014.

[75] S. Onori and L. Serrao, "On adaptive-ECMS strategies for hybrid electric vehicles," in *Proceedings of the international scientific conference on hybrid and electric vehicles, Malmaison, France*, vol. 67, 2011.

[76] L. Serrao, S. Onori, and G. Rizzoni, "ECMS as a realization of Pontryagin's minimum principle for hev control," in *2009 American control conference*. IEEE, 2009, pp. 3964–3969.

[77] R. Bellman, "A markovian decision process," *Journal of mathematics and mechanics*, pp. 679–684, 1957.

[78] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," *arXiv preprint arXiv:1511.05952*, 2015.

[79] D. Horgan, J. Quan, D. Budden, G. Barth-Maron, M. Hessel, H. Van Hasselt, and D. Silver, "Distributed prioritized experience replay," *arXiv preprint arXiv:1803.00933*, 2018.

[80] B. Xu, X. Tang, X. Hu, X. Lin, H. Li, D. Rathod, and Z. Wang, "Q-learning-based supervisory control adaptability investigation for hybrid electric vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[81] A. Schuster, *An Introduction to the Theory of Optics*. E. Arnold, 1094.

[82] J. E. Greivenkamp, *Field Guide to Geometrical Optics*. SPIE Press, 2004.

[83] SAE J941, "Motor vehicle drivers' eye locations," SAE International, Standard, 2008.

[84] SAE J1050, "Describing and measuring the driver's field of view," SAE International, Standard, 2001.

[85] Laboratory Test Procedure For FMVSS 111 Rear Visibility, "Rear visibility (other than school buses)," National Highway Traffic Safety Administration, Standard, 2018.

[86] R. Schmied, H. Waschl, R. Quirynen, M. Diehl, and L. del Re, "Nonlinear mpc for emission efficient cooperative adaptive cruise control," *IFAC-papersonline*, vol. 48, no. 23, pp. 160–165, 2015.

[87] V. Kovali, V. Alexiadis, and L. Zhang, "Video-based vehicle trajectory data collection," in *Proceedings of the 86th annual meeting of the TRB*, 2007.

[88] M. Montanino and V. Punzo, "Making ngsim data usable for studies on traffic flow theory: Multistep method for vehicle trajectory reconstruction," *Transportation Research Record*, vol. 2390, no. 1, pp. 99–111, 2013.

[89] V. Punzo, M. T. Borzacchiello, and B. Ciuffo, "On the assessment of vehicle trajectory data accuracy and application to the next generation simulation (ngsim) program data," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 6, pp. 1243–1262, 2011.

[90] X.-Y. Lu and A. Skabardonis, "Freeway traffic shockwave analysis: exploring the ngsim trajectory data," in *86th Annual Meeting of the Transportation Research Board, Washington, DC*. Citeseer, 2007.

[91] B. Coifman and L. Li, "A critical evaluation of the next generation simulation (ngsim) vehicle trajectory dataset," *Transportation Research Part B: Methodological*, vol. 105, pp. 362–377, 2017.

[92] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, pp. 1–36, 2019.

[93] M. Risbeck and J. Rawlings, "Mpctools: Nonlinear model predictive control tools for casadi," 2016.

[94] E. Hyeon, Y. Kim, N. Prakash, and A. G. Stefanopoulou, "Short-term speed forecasting using vehicle wireless communications," in *Proceedings of the 2019 American Control Conference (ACC)*. IEEE, 2019, pp. 736–741.

[95] Y. Du, C. Liu, and Y. Li, "Velocity control strategies to improve automated vehicle driving comfort," *IEEE Intelligent transportation systems magazine*, vol. 10, no. 1, pp. 8–18, 2018.

[96] G. A. Hubbard and K. Youcef-Toumi, "System level control of a hybrid-electric vehicle drivetrain," in *Proceedings of the 1997 American Control Conference (Cat. No. 97CH36041)*, vol. 1. IEEE, 1997, pp. 641–645.

[97] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 42, no. 1, pp. 55–61, 2000.

[98] S. Cheng, L. Li, M. Mei, Y. Nie, and L. Zhao, "Multiple-objective adaptive cruise control system integrated with dyc," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4550–4559, 2019.

[99] Y. Zhu, D. Zhao, and H. He, "Synthesis of cooperative adaptive cruise control with feedforward strategies," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 3615–3627, 2020.

[100] R. Wiedemann, "SIMULATION DES STRASSENVERKEHRSFLUSSES." in *Proceedings of the Schriftenreihe des tnstituts fir Verkehrswesen der Universitiit Karlsruhe*, Germany, 1974.

[101] I. W. Suweda, "Time headway analysis to determine the road capacity," *Jurnal Spektran*, vol. 4, no. 2, 2016.

[102] H. Kaths, A. Keler, and K. Bogenberger, "Calibrating the wiedemann 99 car-following model for bicycle traffic," *Sustainability*, vol. 13, no. 6, p. 3487, 2021.

[103] U. Durrani, C. Lee, and H. Maoh, "Calibrating the wiedemann's vehicle-following model using mixed vehicle-pair interactions," *Transp. Res. C: Emerg. Technol.*, vol. 67, pp. 227–242, 2016.

[104] Z. Zhou, Z. Yang, Y. Zhang, Y. Huang, H. Chen, and Z. Yu, "A comprehensive study of speed prediction in transportation system: From vehicle to traffic," *iScience*, p. 103909, 2022.

[105] M. F. Ozkan and Y. Ma, "Personalized adaptive cruise control and impacts on mixed traffic," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 412–417.

[106] Y. Fan, P. Wang, A. A. Heidari, H. Chen, M. Mafarja *et al.*, "Random reselection particle swarm optimization for optimal design of solar photovoltaic modules," *Energy*, vol. 239, p. 121865, 2022.

[107] W. Zhao, L. Wang, and S. Mirjalili, "Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications," *Comput. Methods Appl. Mech. Eng.*, vol. 388, p. 114194, 2022.

[108] E. Hyeon, Y. Kim, T. Ersal, and A. Stefanopoulou, "Data-driven forgetting and discount factors for vehicle speed forecasting in ecological adaptive cruise control," *J. Dyn. Syst. Meas. Control*, vol. 144, no. 1, 2022.

[109] W.-K. Lai, T.-H. Kuo, and C.-H. Chen, "Vehicle speed estimation and forecasting methods based on cellular floating vehicle data," *Applied Sciences*, vol. 6, no. 2, p. 47, 2016.

[110] Y. Zhang, J. Lv, and W. Wang, "Evaluation of vehicle acceleration models for emission estimation at an intersection," *Transp. Res. D: Transp. Environ.*, vol. 18, pp. 46–50, 2013.

[111] E. F. Camacho and C. B. Alba, *Model predictive control*. Springer science & business media, 2013.

[112] K. H. Kwak, Y. He, Y. Kim, Y. M. Chen, S. Fan, J. Holmer, and J. H. Lee, "(Accepted) Desired relative distance model-based personalized braking algorithm for one-pedal driving of electric vehicles," in *2022 Modeling, Estimation, and Control Conference (MECC)*, 2022.

[113] S. Fan, Y. Sun, J. H. Lee, and J. Ha, "A co-simulation platform for powertrain controls development," in *SAE Technical Paper, No. 2020-01-0265*, 2020.

[114] S. Fan, J. Lee, Y. Sun, J. Ha, and J. Harber, "Virtual platform development for new control logic concept test and validation," in *SAE Technical Paper, No. 2021-01-1143*, 2021.

[115] S. Ziegler and R. Höpler, "Extending the ipg carmaker by fmi compliant units," in *Proceedings of the 8th International Modelica Conference; March 20th-22nd; Technical University; Dresden; Germany*, no. 063, 2011, pp. 779–783.

[116] C. Chen, "The development of hybrid electric vehicle control strategy based on gt-suite and simulink," in *2015 International Conference on Intelligent Systems Research and Mechatronics Engineering. Atlantis Press*, 2015.

[117] V. Zeidler, H. S. Buck, L. Kautzsch, P. Vortisch, and C. M. Weyland, "Simulation of autonomous vehicles based on wiedemann's car following model in ptv vissim," in *Transportation Research Board 98th Annual Meeting*, 2019, p. 12.

[118] Michigan Department of Transportation, "Mdot, traffic volumes map," https://lrs.state.mi.us/portal/apps/webappviewer/, accessed: 2022-08-31.

[119] Y. Choi, J. Guanetti, S. Moura, and F. Borrelli, "Data-driven energy management strategy for plug-in hybrid electric vehicles with real-world trip information," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 14 224–14 229, 2020.

[120] M. Vajedi, "Real-time optimal control of a plug-in hybrid electric vehicle using trip information," 2016.

[121] A. B. Patel, N. M. Waters, I. E. Blanchard, C. J. Doig, and W. A. Ghali, "A validation of ground ambulance pre-hospital times modeled using geographic information systems," *International journal of health geographics*, vol. 11, no. 1, pp. 1–10, 2012.

[122] A. Rajagopalan and G. Washington, "Intelligent control of hybrid electric vehicles using gps information," SAE Technical Paper, Tech. Rep., 2002.

[123] F. Tianheng, Y. Lin, G. Qing, H. Yanqing, Y. Ting, and Y. Bin, "A supervisory control strategy for plug-in hybrid electric vehicles based on energy demand prediction and route preview," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 5, pp. 1691–1700, 2014.

[124] L. C. Fang, G. Xu, T. L. Li, and K. M. Zhu, "Real-time optimal power management for hybrid electric vehicle based on prediction of trip information," in *Applied Mechanics and Materials*, vol. 321. Trans Tech Publ, 2013, pp. 1539–1547.

[125] X. Zeng and J. Wang, "Optimizing the energy management strategy for plug-in hybrid electric vehicles with multiple frequent routes," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 1, pp. 394–400, 2017.

[126] Y. Ma and J. Wang, "Integrated power management and aftertreatment system control for hybrid electric vehicles with road grade preview," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 12, pp. 10 935–10 945, 2017.

[127] T. S. Kim, C. Manzie, and R. Sharma, "Two-stage optimal control of a parallel hybrid vehicle with traffic preview," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 2115–2120, 2011.

[128] L. Guo, H. Chen, B. Gao, and Q. Liu, "Energy management of hevs based on velocity profile optimization." *Sci. China Inf. Sci.*, vol. 62, no. 8, pp. 89 203–1, 2019.

[129] C. J. Mansour, "Trip-based optimization methodology for a rule-based energy management strategy using a global optimization routine: the case of the prius plug-in hybrid electric vehicle," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 230, no. 11, pp. 1529–1545, 2016.

[130] M. Pourabdollah, V. Larsson, L. Johannesson, and B. Egardt, "Phev energy management: A comparison of two levels of trip information," in *SAE World Congress*, 2012.

[131] X. Zeng and J. Wang, "A parallel hybrid electric vehicle energy management strategy using stochastic model predictive control with road grade preview," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 6, pp. 2416–2423, 2015.

[132] Z. Yang, H. Chen, S. Dong, Q. Liu, and F. Xu, "Energy management strategy of hybrid electric vehicle with consideration of road gradient," in *2020 Chinese Control And Decision Conference (CCDC)*. IEEE, 2020, pp. 2879–2885.

[133] J. Lin, X. Liu, S. Li, C. Zhang, and S. Yang, "A review on recent progress, challenges and perspective of battery thermal management system," *International Journal of Heat and Mass Transfer*, vol. 167, p. 120834, 2021.

[134] A. Wei, J. Qu, H. Qiu, C. Wang, and G. Cao, "Heat transfer characteristics of plug-in oscillating heat pipe with binary-fluid mixtures for electric vehicle battery thermal management," *International Journal of Heat and Mass Transfer*, vol. 135, pp. 746–760, 2019.

[135] A. H. Akinlabi and D. Solyali, "Configuration, design, and optimization of air-cooled battery thermal management system for electric vehicles: A review," *Renewable and Sustainable Energy Reviews*, vol. 125, p. 109815, 2020.

[136] S. Arora, "Selection of thermal management system for modular battery packs of electric vehicles: A review of existing and emerging technologies," *Journal of Power Sources*, vol. 400, pp. 621–640, 2018.

[137] M. Akbarzadeh, J. Jaguemont, T. Kalogiannis, D. Karimi, J. He, L. Jin, P. Xie, J. Van Mierlo, and M. Berecibar, "A novel liquid cooling plate concept for thermal management of lithium-ion batteries in electric vehicles," *Energy Conversion and Management*, vol. 231, p. 113862, 2021.

[138] S. Wiriyasart, C. Hommalee, S. Sirikasemsuk, R. Prurapark, and P. Naphon, "Thermal management system with nanofluids for electric vehicle battery cooling modules," *Case Studies in Thermal Engineering*, vol. 18, p. 100583, 2020.

[139] A. Verma, S. Shashidhara, and D. Rakshit, "A comparative study on battery thermal management using phase change material (pcm)," *Thermal Science and Engineering Progress*, vol. 11, pp. 74–83, 2019.

[140] M. R. Amini, I. Kolmanovsky, and J. Sun, "Hierarchical mpc for robust eco-cooling of connected and automated vehicles and its application to electric vehicle battery thermal management," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 1, pp. 316–328, 2020.

[141] J. Gou and W. Liu, "Feasibility study on a novel 3d vapor chamber used for li-ion battery thermal management system of electric vehicle," *Applied Thermal Engineering*, vol. 152, pp. 362–369, 2019.

[142] J. Han, H. Shu, X. Tang, X. Lin, C. Liu, and X. Hu, "Predictive energy management for plug-in hybrid electric vehicles considering electric motor thermal dynamics," *Energy Conversion and Management*, vol. 251, p. 115022, 2022.

[143] G. Caramia, N. Cavina, A. Capancioni, M. Caggiano, and S. Patassa, "Combined optimization of energy and battery thermal management control for a plug-in hev," SAE Technical Paper, Tech. Rep., 2019.

[144] T. J. Boehme, M. Schori, B. Frank, M. Schultalbers, and B. Lampe, "Solution of a hybrid optimal control problem for parallel hybrid vehicles subject to thermal constraints," in *52nd IEEE conference on decision and control*. IEEE, 2013, pp. 2220–2226.

[145] Q. Hu, M. R. Amini, H. Wang, I. Kolmanovsky, and J. Sun, "Integrated power and thermal management of connected hevs via multi-horizon mpc," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 3053–3058.

[146] B. Zhu, Y. Jiang, J. Zhao, R. He, N. Bian, and W. Deng, "Typical-driving-style-oriented personalized adaptive cruise control design based on human driving data," *Transportation research part C: emerging technologies*, vol. 100, pp. 274–288, 2019.

[147] B. Gao, K. Cai, T. Qu, Y. Hu, and H. Chen, "Personalized adaptive cruise control based on online driving style recognition technology and model predictive control," *IEEE transactions on vehicular technology*, vol. 69, no. 11, pp. 12 482–12 496, 2020.

[148] Y. Wang, Z. Wang, K. Han, P. Tiwari, and D. B. Work, "Personalized adaptive cruise control via gaussian process regression," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 1496–1502.

[149] C. Su, W. Deng, R. He, J. Wu, and Y. Jiang, "Personalized adaptive cruise control considering drivers' characteristics," SAE Technical Paper, Tech. Rep., 2018.

[150] A. P. Bolduc, L. Guo, and Y. Jia, "Multimodel approach to personalized autonomous adaptive cruise control," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 2, pp. 321–330, 2019.

[151] J. Jiang, F. Ding, Y. Zhou, J. Wu, and H. Tan, "A personalized human drivers' risk sensitive characteristics depicting stochastic optimal control algorithm for adaptive cruise control," *IEEE Access*, vol. 8, pp. 145 056–145 066, 2020.

[152] Y. Liu, J. xiang Qin, and M. di Liao, "Analysis and design of personalized adaptive cruise system," SAE Technical Paper, Tech. Rep., 2020.

[153] Q. Xin, R. Fu, W. Yuan, Q. Liu, and S. Yu, "Predictive intelligent driver model for eco-driving using upcoming traffic signal information," *Physica A: Statistical Mechanics and its Applications*, vol. 508, pp. 806–823, 2018.

[154] C. Sun, X. Shen, and S. Moura, "Robust optimal eco-driving control with uncertain traffic signal timing," in *2018 annual American control conference (ACC)*. IEEE, 2018, pp. 5548–5553.

[155] S. Bae, Y. Choi, Y. Kim, J. Guanetti, F. Borrelli, and S. Moura, "Real-time ecological velocity planning for plug-in hybrid vehicles with partial communication to traffic lights," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 1279–1285.

[156] Y. Lu, J. Lu, S. Zhang, and P. Hall, "Traffic signal detection and classification in street views using an attention model," *Computational Visual Media*, vol. 4, no. 3, pp. 253–266, 2018.

[157] A. Salaymeh, "Machine learning techniques for automated traffic signal detection and timing," Ph.D. dissertation, Wayne State University, 2021.

[158] K. Anirudh, M. S. Dhanoosh, A. Vamsi, and S. Latha, "Driver assisting feature for collision avoidance, sign and traffic signal detection."

[159] R. Zhang, A. Ishikawa, W. Wang, B. Striner, and O. K. Tonguz, "Using reinforcement learning with partial vehicle detection for intelligent traffic signal control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 1, pp. 404–415, 2020.

[160] R. J. Franklin *et al.*, "Traffic signal violation detection using artificial intelligence and deep learning," in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*. IEEE, 2020, pp. 839–844.

[161] Y. Xiang, W. Niu, E. Tong, Y. Li, B. Jia, Y. Wu, J. Liu, L. Chang, and G. Li, "Congestion attack detection in intelligent traffic signal system: combining empirical and analytical methods," *Security and Communication Networks*, vol. 2021, 2021.

[162] Z. Shi, Y. Huang, Z. Hu, and T. Li, "Design of traffic-signal condition detection system based on intelligence," in *2019 4th International Conference on Intelligent Green Building and Smart Grid (IGBSG)*. IEEE, 2019, pp. 260–263.

[163] S. Jones, N. Wikström, A. F. Parrilla, R. Patil, E. Kural, A. Massoner, and A. Grauers, "Energy-efficient cooperative adaptive cruise control strategy using v2i," in *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, 2019, pp. 1420–1425.

[164] N. Wikström, A. F. Parrilla, S. J. Jones, and A. Grauers, "Energy-efficient cooperative adaptive cruise control with receding horizon of traffic, route topology, and traffic light information," *SAE International Journal of Connected and Automated Vehicles*, vol. 2, no. 12-02-02-0006, 2019.

[165] F. Ma, Y. Yang, J. Wang, X. Li, G. Wu, Y. Zhao, L. Wu, B. Aksun-Guvenc, and L. Guvenc, "Eco-driving-based cooperative adaptive cruise control of connected vehicles platoon at signalized intersections," *Transportation Research Part D: Transport and Environment*, vol. 92, p. 102746, 2021.

[166] S. Coskun, C. Huang, and F. Zhang, "Quadratic programming-based cooperative adaptive cruise control under uncertainty via receding horizon strategy," *Transactions of the Institute of Measurement and Control*, vol. 43, no. 13, pp. 2899–2911, 2021.

[167] C. Pan, A. Huang, L. Chen, Y. Cai, L. Chen, J. Liang, and W. Zhou, "A review of the development trend of adaptive cruise control for ecological driving," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 236, no. 9, pp. 1931–1948, 2022.

[168] L. Zhu, F. Tao, Z. Fu, N. Wang, B. Ji, and Y. Dong, "Optimization based adaptive cruise control and energy management strategy for connected and automated fchev," *IEEE Transactions on Intelligent Transportation Systems*, 2022.