Speed Forecasting Strategies for the Energy-Optimal Car-Following of Connected and Automated Vehicles

by

Eunjeong Hyeon

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

Doctoral Committee:

Associate Research Scientist Tulga Ersal, Co-Chair
Professor Anna Stefanopoulou, Co-Chair
Dominik Karbowski, Argonne National Laboratory
Assistant Professor Youngki Kim
Professor Ilya Kolmanovsky
To my grandmother, for her unconditional love.
ACKNOWLEDGEMENTS

This dissertation and its underlying research were not possible without the support of countless people. First and foremost, I thank my advisors Prof. Anna Stefanopoulou and Dr. Tulga Ersal. I appreciate their trust in me and guidance through my Ph.D. journey. I would not be able to go this far without them.

I am thankful to the committee members, Prof. Ilya Kolmanovsky, Prof. Youngki Kim, and Dominik Karbowski, for taking their time and interest in evaluating my work and providing constructive feedback. I genuinely believe that their insightful comments have led this dissertation to be more thorough and complete.

I appreciate all the financial support for this research provided by multiple funding sources, including the National Science Foundation for its financial support under Grant Number 1646019; Division of Computer and Network Systems, U.S. Department of Energy (DOE) Advanced Research Projects Agency-Energy (ARPA-E) under Award Number DE-AR0000837, also known as NEXTCAR, and U.S. DOE Vehicle Technologies Office under the Systems and Modeling for Accelerated Research in Transportation (SMART) Mobility Laboratory Consortium.

I thank my collaborators in the Vehicle and Mobility Systems group at Argonne National Laboratory, including Dr. Jihun Han, Dr. Daliang Shen, Namdoo Kim, Julien Grave, and Aymeric Rousseau for their help and enthusiastic discussion during my internships. I also want to thank Southwest Research Institute and Toyota Motor North America for all our discussions for the ARPA-E NEXTCAR project.

It has been a great pleasure working with all my colleagues in the Powertrain Control Laboratory. I thank their company —Shima Nazari, Niket Prakash, Bryan Maldonado, Saravanan Duraiarasan, Miriam Figueroa-Santos, Di Chen, Chunan Huang, Ting Cai, Suhak Lee, Peyman Mohat, Joe Drallmeier, Omar Ahmed, Sravan Pannala, Mitchell Bieniek, Robert Middleton, Jason Siegel, and Rasoul Salehi.

Last but not least, I am thankful to my parents and sisters for their support and faith in me, making me keep moving forward. Finally, I want to thank Yeonjoon. Without his love and support, I would not have completed this journey.
# TABLE OF CONTENTS

DEDICATION ................................................................. ii

ACKNOWLEDGEMENTS .................................................. iii

LIST OF FIGURES ......................................................... viii

LIST OF TABLES .......................................................... xiv

ABSTRACT ........................................................................ xv

CHAPTER

I. Introduction ............................................................... 1

1.1 Background ............................................................... 1

  1.1.1 Vehicle Connectivity and Automation Technologies .......... 2

  1.1.2 Optimal Control for Vehicle Energy Efficiency ............. 3

1.2 Optimizing Longitudinal Motions in Car-Following Scenarios .... 4

  1.2.1 Objective Functions ............................................. 5

  1.2.2 Constraints ....................................................... 6

1.3 Vehicle Trajectory Forecasting ........................................ 6

  1.3.1 Model-based Prediction Approaches ........................... 7

  1.3.2 Data-driven Prediction Approaches ............................ 7

1.4 Contributions and Organization ....................................... 9

  1.4.1 Driver Models and Control Formulations .................... 10

II. Forecasting Short-term Speed Trajectories of a Preceding Vehicle using V2V Information ................................................. 14

  2.1 Introduction ........................................................... 14

  2.2 Prediction Algorithm ............................................... 16

    2.2.1 Processing Vehicle-to-Vehicle Basic Safety Messages ... 16

    2.2.2 Calculation of Future Speed Trajectories ................. 17

  2.3 Simulation Settings ................................................ 20

    2.3.1 Car-following Traffic Scenarios ........................... 20
III. Influence of Vehicle Speed Forecasting on Ecological Adaptive Cruise Control Performance .......................... 28

3.1 Introduction .................................................. 28
3.2 Model Description ............................................ 30
    3.2.1 Vehicle Longitudinal Dynamics Model .................. 31
3.3 Ecological Adaptive Cruise Control (QP-HC) .................. 32
3.4 Simulations .................................................. 33
    3.4.1 Traffic Scenarios and Baseline Model .................. 33
    3.4.2 Speed and Position Prediction Results .................. 34
    3.4.3 Control Results ........................................ 36
    3.4.4 Fuel Economy Assessment .............................. 38
    3.4.5 Comparison with Various Powertrain Types ............... 41
3.5 Summary ................................................... 42

IV. Forecasting Mid-Length Speed Trajectories of a Preceding Vehicle using V2V and V2I Information .......................... 44

4.1 Introduction .................................................. 44
4.2 Speed Forecasting Methodology .............................. 45
    4.2.1 Prediction Algorithm .................................. 46
    4.2.2 Design Matrix .......................................... 46
    4.2.3 Weight Matrix .......................................... 48
    4.2.4 Speed Forecasting with V2I Information ................. 49
4.3 Eco-Driving Control Strategy (PMP-Pwt) ....................... 52
4.4 Simulation Setup ............................................. 53
    4.4.1 Traffic Scenarios ....................................... 53
4.5 Simulation Results .......................................... 54
    4.5.1 Target Vehicle Speed Forecasting Results ............... 55
    4.5.2 Ego Vehicle Eco-Driving Control Results ............... 57
4.6 Summary ................................................... 60

V. Extended Case Study on the Impacts of Vehicle Connectivity and Traffic Conditions for Eco-Driving Control and Prediction .......................... 61

5.1 Introduction .................................................. 61
5.2 Heterogeneous Traffic Generation ............................ 63
5.3 Eco-Driving Controller (PMP-Spd) ............................ 63
5.4 Parameter Study ................................................. 64
  5.4.1 Mean and Standard Deviation of Time Gap ......... 65
  5.4.2 Prediction Horizon Length ......................... 67
  5.4.3 Number of Connected Vehicles .................... 68
5.5 Validation based on Software-in-the-Loop Testing ............. 70
  5.5.1 Simulation Setting .................................. 70
  5.5.2 Results and Discussion ............................. 70
  5.5.3 Comparison between Polynomial Regression and LWPR Predictors ........................................... 72
5.6 Summary .......................................................... 74

VI. Data-Driven Parameter Tuning Methodology for Vehicle Speed Forecasting based on Polynomial Regression ....................... 75
  6.1 Introduction ................................................. 75
  6.2 Tuning Input Weights using Real-World Driving Records .... 77
    6.2.1 Data ................................................. 78
    6.2.2 Information from Past Speed Measurements ....... 80
    6.2.3 Look-ahead Information from V2V Measurements .... 80
  6.3 Ecological Adaptive Cruise Control (QP-SC) .................. 83
  6.4 Simulation Study ........................................... 85
    6.4.1 Simulation Set-up .................................. 85
    6.4.2 Prediction Horizon Length Selection .............. 87
    6.4.3 Baselines ........................................... 87
    6.4.4 Simulation Results with the Real-World Traffic Data .... 88
    6.4.5 Simulation Results under Various Simulated Traffic Conditions ........................................... 91
    6.4.6 Comparison with Various Powertrains .................. 94
  6.5 Summary .......................................................... 95

VII. Loss Function Design of Data-Driven Vehicle Speed Predictors for Enhancing Energy Efficiency ............................... 96
  7.1 Introduction ................................................. 96
  7.2 Designing the Loss Function of a Vehicle Speed Predictor to Improve Eco-ACC Energy Saving .................. 98
    7.2.1 Derivation of Artificial Forecasting Uncertainty .... 99
    7.2.2 Eco-ACC Simulation with Forecasting Uncertainty ... 100
    7.2.3 Quantifying the Influence of Prediction Uncertainty on the Eco-ACC Energy Saving Performance ......... 101
  7.3 Example Vehicle Speed Predictors .......................... 103
    7.3.1 Predictor Input Features and Outputs ............... 103
    7.3.2 Polynomial Regression ................................ 104
    7.3.3 Artificial Neural Network ............................ 105
    7.3.4 Long Short-Term Memory Network .................... 106
7.3.5 Training and Validation Data ........................................... 107
7.4 Validation ............................................................................. 109
  7.4.1 Simulation Settings .......................................................... 109
  7.4.2 Tested Loss Functions ....................................................... 110
  7.4.3 Prediction Results ............................................................ 111
  7.4.4 Control Results ............................................................... 113
  7.4.5 Tests with Various Powertrain Configuration .................... 116
7.5 Summary ............................................................................... 118

VIII. Conclusions and Future Work .......................................... 119
  8.1 Results and Conclusions ..................................................... 119
  8.2 Future Work ....................................................................... 120

BIBLIOGRAPHY ......................................................................... 123
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Share of U.S. GHG emissions by sectors and sources in 2019 [1].</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>Schematic overview of typical sensors used in automated vehicles and their functions [2].</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>Conceptual flowchart of an eco-ACC system implemented in a MPC framework. This diagram is illustrated by referring to the figure in [3].</td>
<td>5</td>
</tr>
<tr>
<td>1.4</td>
<td>Dissertation organization and summary of contributions</td>
<td>9</td>
</tr>
<tr>
<td>2.1</td>
<td>Schematic overview of the speed forecasting method. The “target” is a vehicle from which the “ego” (or controlled) vehicle needs information on how it will drive over a prediction horizon. Multiple other preceding vehicles driving in front of the target are indicated as “preceding vehicles” in this chapter. We assume that the information of the target and the preceding vehicles’ speed and location can be transmitted to the ego vehicle through V2V communication.</td>
<td>15</td>
</tr>
<tr>
<td>2.2</td>
<td>An exponential weighting function as proposed in this chapter. The weighting values for past speed measurements are determined according to the blue shaded region. Preceding vehicles’ speed measurement are weighted according to the function in the red shaded region. The coefficients on the power terms of exponential functions are $\alpha = 1$ and $\gamma = 0.3$ for the past and future time, respectively.</td>
<td>19</td>
</tr>
<tr>
<td>2.3</td>
<td>Examples of traffic simulations and validation based on US06 and IDM. In the top subplot, the blue line shows the distance from the closest preceding vehicle to the target vehicle, and the black dotted line is the safety limit defined in (2.11). The middle subplot presents the distance from the farthest preceding vehicle to the target vehicle. Since the first preceding vehicle exceeds the communication range from around 180 s to 480 s, the ego vehicle loses the first preceding vehicle’s signals during this period. The time-varying number of preceding vehicles connected to the ego vehicle is shown in the bottom subplot.</td>
<td>23</td>
</tr>
<tr>
<td>2.4</td>
<td>The performance metric $J$ defined in (2.12) is computed with different $\alpha$ and $\gamma$ using US06 and shown in the left subplot. The red cross notates the minimum $J$ value, where the optimal weighting parameters are $\alpha^* = 1$ and $\gamma^* = 0.3$. In the right subplot, MAE is compared for the cases without weighting (blue) and with optimal weighting (red).</td>
<td>24</td>
</tr>
</tbody>
</table>
2.5 Demonstration of the proposed speed predictor on US06 with the RDM traffic model. In the top result, the predictor uses only the past speed trajectories of the target for prediction; thus, the linear polynomial regression is applied. In the bottom subplot, the predictor uses both the past speed and the preceding vehicles’ speed, and the quadratic polynomial regression is applied. .............................. 25

2.6 Comparing the quadratic polynomial regression results of (1) using only the past speed of the target vehicle (left) and (2) using the past speed of the target and preceding vehicles’ speed data (right). With only the past measurements, prediction is produced based on extrapolation. By including the preceding vehicles’ speed information in the regression, prediction can be computed based on interpolation, which can reduce prediction error significantly. ....................................................... 26

3.1 Schematic overview of the eco-ACC system studied in this chapter. The ego (following) vehicle predicts the future speed of the target vehicle by using V2V information from the multiple preceding vehicles (PV 1 to PV N in the figure) and the target vehicles. ....................... 29

3.2 Example of speed prediction for 10-second prediction horizon. The blue solid line presents the true UDDS cycle which the target vehicle drives, and the red lines are predicted speed generated at every second. The number of connected preceding vehicles is five in this example. ...... 34

3.3 Mean absolute errors (MAE) and root-mean-square errors (RMSE) of speed prediction (top) and position prediction (bottom) from different speed forecasting approaches: constant acceleration model (red), and V2V-based prediction introduced in Section 2.2 with one (blue) and five connected preceding vehicles (green). The UDDS is used for traffic generation. .............................................................. 35

3.4 Root-mean-square errors (RMSE) of the speed predictions with the different numbers of the preceding vehicle data. In the legend, “i” indicates a time step after the current time as used in (3.10). The UDDS is used for traffic generation. ......................................................... 35

3.5 Simulation results from the eco-ACC with the V2V-based speed predictor over the UDDS. The red solid lines present the target vehicle’s trajectories and the blue solid lines shows the trajectories of the ego or controlled vehicle. ............................................................... 37

3.6 Smoothed traces of acceleration and speed traces by the eco-ACC. The entire drive cycle is shown in Figure 3.5. ......................... 38

3.7 Histograms of longitudinal acceleration from the IDM-driven and the eco-ACC driven cycles over the UDDS and the LA92. .................. 38

3.8 Comparison of fuel economy from the UDDS and the LA92, IDM, and the eco-ACC while following each original cycle with different numbers of connected vehicles. Similar study with PHEV can be found in [4] (see Figure 8.5). .......................................................... 40

3.9 Average position prediction root-mean-squared errors (RMSE) and fuel economy improvement compared to the original UDDS and LA92 cycles. 41
3.10 Energy consumption of the eco-ACC trajectories compared to the IDM over various powertrain types. 42

4.1 Schematic overview of this chapter. The dark blue “ego” vehicle is the controlled vehicle, and the red “target” vehicle is the nearest car in front of the ego vehicle. The ego vehicle predicts the target vehicle’s future speed trajectories and uses the prediction to optimize its powertrain operations. 45

4.2 Illustration of the LWPR-based prediction method proposed in this chapter. 47

4.3 The proposed predictor can distinguish six scenarios with V2I information. Three traffic light statuses are considered in this chapter. The six cases represent the combinations of the current status (on the left) of the next traffic light and its estimated upcoming status (on the right) at the arrival of the target vehicle. 50

4.4 Schematic overview of the two simulation designs used in this chapter. 54

4.5 Drive cycles used for the simulations in this chapter. 55

4.6 Example of speed prediction when the target vehicle approaches a stop sign. 56

4.7 Snapshots of speed prediction using both V2V and V2I information. 57

4.8 Speed prediction root-mean-squared errors (RMSE) from the proposed predictor over various driving cycles. 57

4.9 Energy consumption (top) and travel times (bottom) of the proposed method compared to the baseline. 58

5.1 Overview of this chapter focusing on single-lane scenarios. The ego vehicle is the CAV controlled by the PMP-Spd controller introduced in Section 5.3; the target vehicle is the vehicle immediately in front of the ego vehicle and the prediction target of the ego vehicle, and the preceding vehicles are all the vehicles driving in front of the target vehicle. 62

5.2 Speed and distance gap trajectories of 100 human-driven vehicles. The heavier blue lines indicate the 60th vehicle’s trajectory. 64

5.3 Speed profiles of the lead vehicle (black solid line), 99 preceding vehicles (black dotted lines), and target vehicle (blue solid line) for the time gap mean values of 0.5 s (top) and 1.5 s (bottom). 66

5.4 Comparison of acceleration energy and corresponding final position accuracy resulted from different mean and standard deviation of the A2ODM time gap parameter. 66

5.5 Comparison of acceleration energy and prediction accuracy over various prediction horizon lengths applied to the predictors and controller. 68

5.6 Trajectories of the V2V ego vehicle with the various lengths of prediction horizons. 69

5.7 Comparison of prediction accuracy and acceleration energy using different numbers of connected vehicles. 69

5.8 Comparison of energy consumption across the various system parameters. The UDDS cycle is used to generate the A2ODM traffic. 70

5.9 Speed prediction results of the V2V predictor and the ground-truth speed of the target vehicle, given $\mu_T=0.5$ s. 71
5.10 Speed trajectories of the target and the ego vehicles using the V2V predictor, given $\mu_t=0.5\text{ s}$. 71
5.11 Energy consumption increase compared to the perfect preview case over various powertrain configurations. In the simulations, the time gap mean and standard deviation, horizon length, and the number of connected vehicles vary with the same parameter sets used in Section 5.4. 72
5.12 Comparison between the polynomial regression-based predictor and the LWPR-based predictor. 73
6.1 Overview of this chapter. 76
6.2 Flowcharts of the processes to obtain the weighting parameters, the forgetting factor (top) and the discount factor (bottom), respectively. Detailed explanation of these processes are presented in Section 6.2.2 and 6.2.3. 77
6.3 An example of routes traveled by one of the human drivers in the RWD-UM dataset. 78
6.4 The route driven by the three vehicles in the RWD-AA dataset. The map shows the Ann Arbor area in Michigan. 79
6.5 Speed profiles of the three vehicles in the RWD-AA data. 79
6.6 Accelerations with different time gaps at low-speed sections in the real-world driving records traveled by the UMTRI driver 1. 81
6.7 Accelerations with different time gaps at high-speed sections in the real-world driving records traveled by the UMTRI driver 1. 81
6.8 Calculations between correlation and coefficient of determination of linear fit between accelerations with time differences. Ten human drivers' driving records from the RWD-UM datasets are included in this analysis. The solid lines are the fitted curves of results from real data that are plotted in the lines with the dot markers. 81
6.9 Inverse standard deviations of the V2V errors with different time headway between preceding and following vehicles from the RWD-AA data. The data are used for designing the discount factor ($\gamma$). 82
6.10 Standard deviations of acceleration (left) and electricity consumption (right) with different prediction horizon lengths in the eco-ACC. 86
6.11 Comparison between the trajectories driven by the real human driver in the Ann Arbor area (RWD-AA) and the eco-ACC with the predictor based on WLS. 88
6.12 Cumulative probability distributions of prediction errors from the entire prediction horizon (left) and the first prediction step (right). 89
6.13 Comparison of eco-ACC performance over the real traffic data (RWD-AA) with different prediction methods. 89
6.14 Comparison of root-mean-square errors (RMSE) of prediction accuracy by prediction steps. The shaded areas are results from the variation in traffic simulation parameters: the number of preceding vehicles from 1 to 10, and the time headway from 1 to 4 s. The lines with the diamond and triangle markers indicate the medians of simulation results with traffic variations. In RWD-AA, the number of preceding vehicles is 1 and the average time headway is 2 s. ........................................... 91

6.15 Comparison of the eco-ACC performance with different speed prediction methods. The clouds of black dots show results from various traffic simulation parameters (1 ≤ N_{PV} ≤ 10 and 1 ≤ T_h ≤ 4 s). ........................................... 92

6.16 Relative values of electricity consumption by using the weighted-least-square method compared to the least-square method. .................. 92

6.17 Real-world driving data with six different vehicle powertrain types. .... 93

6.18 Energy saving of WLS compared to the LS over different powertrain types (Source codes for the plots: [5]). .......................... 94

7.1 Flowchart of the proposed methodology for designing a loss function for eco-ACC performance. ........................................... 97

7.2 The uncertainty bounds of the target vehicle’s future speed trajectories over a prediction horizon. ........................................... 99

7.3 Electricity consumption from the eco-ACC trips with the artificial forecasting uncertainty at each prediction step are plotted compared to the eco-ACC with accurate previews. The unit of the y-axis is a percentage [%]. ........................................... 100

7.4 Scaled values of electricity consumption increase with respect to the eco-ACC with perfect preview, and the fitted curves based on various probability density functions. ........................................... 102

7.5 Coefficient of determination (R^2) and the number of parameters of fitted functions based on various probability density functions. ........ 103

7.6 Training and validation root-mean-squared errors (RMSE) with the various number of hidden layers in the ANN. .................. 105

7.7 Training and validation root-mean-squared errors (RMSE) with the various number of hidden units in the LSTM. .................. 106

7.8 Mean of speed and acceleration magnitudes of the real-world driving dataset and the standard drive cycles. .................. 108

7.9 Example training data generated with US06. .................. 109

7.10 Loss function weights for the different types of loss functions over prediction time. ........................................... 109

7.11 Mean absolute errors (MAE) of the polynomial regression, ANN, and LSTM network. ........................................... 111

7.12 Cumulative distributions of absolute prediction errors at the different prediction steps from using the polynomial regression over the US06. 111

7.13 Snapshots of prediction results over the US06 cycle using the ANN predictor trained by the MSE loss function and the Rayleigh loss function. 112

7.14 Trajectories of the target vehicle and the eco-ACC using the MSE and the Rayleigh loss functions. ........................................... 113
Standard deviations of longitudinal acceleration from using the exponential and Rayleigh loss functions compared to the MSE loss function. The polynomial regression (left), ANN (middle), and the LSTM (right) predictors are tested. .......................................................... 114
Electricity consumption from using the exponential and Rayleigh loss functions compared to the MSE loss function. The polynomial regression (left), ANN (middle), and LSTM (right) predictors are tested with three different loss functions. .......................................................... 115
Comparison of acceleration and electricity consumption with the inexperienced trip (SC03). .......................................................... 115
Increase of the standard deviations of acceleration demanded by the eco-ACC with the artificial forecasting uncertainty compared to the eco-ACC with accurate previews. .......................................................... 116
Energy consumption compared to the IDM-driven trips over UDDS, US06, LA92, WLTP, and SC03 with the three different predictors: PR, ANN, and LSTM (Source codes for the plots: [5]). ................. 117
LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Traffic models and eco-driving control formulations used in this dissertation.</td>
<td>11</td>
</tr>
<tr>
<td>2.1</td>
<td>Definitions of IDM Parameters</td>
<td>21</td>
</tr>
<tr>
<td>2.2</td>
<td>RMSE of Prediction Errors [m/s]</td>
<td>27</td>
</tr>
<tr>
<td>2.3</td>
<td>The 90\textsuperscript{th} Percentile of Prediction Errors [m/s]</td>
<td>27</td>
</tr>
<tr>
<td>3.1</td>
<td>Vehicle Parameters</td>
<td>31</td>
</tr>
<tr>
<td>3.2</td>
<td>IDM Parameter Values</td>
<td>34</td>
</tr>
<tr>
<td>3.3</td>
<td>Vehicle Model Configuration for Energy Consumption Evaluation</td>
<td>39</td>
</tr>
<tr>
<td>3.4</td>
<td>Fuel Economy Comparison [MPG]</td>
<td>40</td>
</tr>
<tr>
<td>3.5</td>
<td>Vehicle Model Parameters</td>
<td>42</td>
</tr>
<tr>
<td>4.1</td>
<td>Summary of energy consumption and travel time with different prediction methods. The numbers in the parenthesis indicate relative values in percentage compared to the baseline.</td>
<td>59</td>
</tr>
<tr>
<td>6.1</td>
<td>Summary of Prediction Parameter Tuning</td>
<td>82</td>
</tr>
<tr>
<td>6.2</td>
<td>Parameters in MPC Formulation</td>
<td>84</td>
</tr>
<tr>
<td>6.3</td>
<td>Vehicle Configuration Parameters</td>
<td>86</td>
</tr>
<tr>
<td>6.4</td>
<td>IDM Parameter Values</td>
<td>87</td>
</tr>
<tr>
<td>7.1</td>
<td>Usage of the drive cycles in this chapter.</td>
<td>98</td>
</tr>
</tbody>
</table>
ABSTRACT

Accurate previews of the preceding vehicle’s future trajectories are essential for automated driving in car-following scenarios to minimize energy consumption. This dissertation proposes strategies for forecasting the preceding vehicle’s speed for the eco-driving control system of connected and automated vehicles. First, accurate and economical predictors are developed based on linear-regression-based algorithms targeting short and mid-length horizons. The proposed predictors avoid the need for an exhaustive training process by enabling instant learning, and can provide high accuracy by leveraging information obtained by vehicle communication. Energy benefits from using the proposed predictors are evaluated by applying them to powertrain-agnostic eco-driving controllers minimizing longitudinal acceleration across various powertrain configurations, namely, an internal combustion engine vehicle, a battery electric vehicle, and a hybrid electric vehicle. Simulation results show that the eco-driving control using the proposed predictor reduces energy consumption by up to 6-14% compared to the human-driver following by predicting 10-s horizons with the full penetration of vehicle-to-vehicle communication.

The novel methodologies for designing data-driven predictors are then developed. First, unlike advanced data-driven predictors, simple prediction methods such as polynomial regression are not given any dedicated tools for determining weights in their input layer. Thus, the input weight function is derived, where its parameters are tuned based on the driving characteristics extracted from real-world driving records. By using the polynomial-regression-based predictor with the proposed input weights, energy saving obtained by the eco-driving controller is added by 1% on average compared to using the same predictor without input weights. For more complex data-driven predictors, a new loss function is designed to increase the eco-driving control performance. Unlike conventional loss functions defined as averaged error metrics, the proposed loss function is formulated as a weighted-mean-squared error, where its weights are determined based on the impact of forecasting uncertainty on the eco-driving control performance. The efficacy of this method is evaluated by training various types of vehicle speed prediction algorithms with the proposed loss function and applying the predictors in the eco-driving control system. The proposed loss function achieves an additional energy-saving of 3% compared to the mean-squared-error loss function, commonly used in the state-of-the-art.
CHAPTER I

Introduction

1.1 Background

The transportation sector is one of the most significant contributors to the U.S. greenhouse gas (GHG) emissions. According to a report by the U.S. Environmental Protection Agency (EPA), transportation accounted for the most considerable portion (29%) of total U.S. GHG emissions in 2019, where automobiles are responsible for 82% of GHG emissions in the transportation sector [1], as shown in Figure 1.1.

Figure 1.1: Share of U.S. GHG emissions by sectors and sources in 2019 [1].

Consequently, much effort has been made in the automobile industry to improve vehicle energy efficiency. One of the strong trends is the shift toward battery electric vehicles. However, electricity generation accounts for 25% of the U.S. GHG emissions [1], the second-largest contributor. Therefore, reducing overall trip energy consumption regardless of energy types is essential for clean and sustainable transportation. Much effort
has been made to reduce vehicle energy consumption in automotive control fields. Thanks to the recent advances in connected automated vehicle (CAV) technologies, these control applications have become more promising for improving vehicle energy efficiency [6].

1.1.1 Vehicle Connectivity and Automation Technologies

Vehicle automation technologies have evolved with the high demands on safer and more efficient vehicles in the automotive industry. The levels of automation are categorized from Level 0 to 5 [7], where Level 0 represents no automation while Level 5 indicates fully automated driving. Levels 1 and 2 can be represented by advanced driver assistance systems (ADAS). The applications of ADAS include emergency braking, lane-keeping assist, and blind-spot detection. In addition, automated driving systems (ADS) aim to reduce human inputs even further by controlling both steering and brake/acceleration under some or all circumstances (Level 3-5).

Onboard sensing technologies have advanced, along with the need for vehicle automation. For example, radars, light detection and ranging (LiDAR), and ultrasound sensors can allow modern vehicles to detect their surroundings faster and more accurately when the human driver’s vision is limited due to the driving environment, such as blind spots and weather. Navigation sensors such as an inertial measurement unit (IMU) and global navigation satellite systems (GNSS) play an important role in automated driving by accurately locating a vehicle on a map. An embedded vision system with camera sensors has been actively studied with advanced artificial intelligence functions and sensor fusion techniques. These advances in sensor technologies achieve accurate perception and localization within centimeter accuracy. The typical automated vehicles’ sensors and their functions are illustrated in Figure 1.2.

On a different front, vehicle communication has been actively studied by governments, industries, and academics for developing intelligent transportation systems. Vehicle connectivity can increase the monitoring range even further out of sensors’ line-of-sight. For example, vehicle-to-vehicle (V2V) communications allow vehicles directly connected to other vehicles on the road to share their states such as position, speed, and heading in a basic safety message format [8, 9]. In addition, signal phase and timing (SPaT) messages broadcasting via vehicle-to-infrastructure (V2I) communication allow vehicles to be aware of upcoming traffic lights at signalized intersections [8]. Terrain information such as road grades can be obtained by online platforms providing digital maps collected by geographic information systems (GIS). In addition, by accessing an online network through vehicle-to-network (V2N) channels, vehicles can receive real-time traffic information and avoid traffic congestion [10]. The recent development of the fifth-generation (5G) or even
more advanced generation communication technologies leads to faster and more reliable cellular-based vehicle-to-everything (C-V2X) networks connecting vehicles to personal devices possessed by pedestrians and riders [11].

With the increased availability of information and monitoring, CAVs have the potential to bring many benefits to future transportation in various aspects, including driving safety, personal mobility, and energy efficiency [12, 3, 13]. The focus of this dissertation is energy benefits brought by CAV technologies. The following section summarizes efforts in earlier research on energy-efficient driving and recent opportunities for improving vehicle efficiency by leveraging CAV technologies.

1.1.2 Optimal Control for Vehicle Energy Efficiency

Early efforts for energy-efficient driving, or namely eco-driving, were training human drivers to manually adopt energy-efficient driving styles such as pressing an accelerator pedal mildly and avoiding aggressive stop and go [14]. One way to reduce energy consumption is to drive so that the engine can spend as much time as possible at its most efficient operating points, which typically means high load and moderate speed [15]. Another is to minimize repeated braking-acceleration cycles because braking represents wasted energy [16].

Nowadays, control engineers have been developing optimal control systems to adopt
energy-efficient driving manners automatically [17]. In addition, control systems enable energy-optimal driving realized in lower operation levels. For example, gear shifting [18, 19], regenerative braking [20], and power-split in hybrid vehicles [21] can be optimized to enhance energy efficiency while vehicles drive.

Moreover, with the recent advancement in vehicle connectivity and autonomy, the anticipation of achieving greater energy efficiency by eco-driving control has been rising. For example, energy-optimal platooning strategies have been proposed by exploiting V2V information [22] and sensor measurements [23]. In addition, knowing traffic light cycles in advance allows a vehicle to reduce idling at signalized intersections while obeying traffic rules. These strategies, often referred to as the “eco-approach and departure” approaches have been intensively studied [24, 25, 26, 27, 28, 29]. In addition, map information such as road grades can help optimize gear shifting [30] of vehicles and the power-split strategy of hybrid vehicles [31]. Comprehensive studies on the potential impact of CAVs on energy efficiency are available in the literature [32, 15, 33].

However, regardless of driving situations, if a preceding vehicle exists in the range of considerably close distance, following vehicle’s maneuvers would be highly affected by the preceding vehicle’s actions. It is important to develop eco-driving controllers considering the presence of a preceding vehicle, considering realistic traffic conditions and road safety. Hence, control strategies for energy-efficient car-following have attracted considerable attention in the previous literature [34, 35]. In this context, the main concern of this dissertation is energy-optimal driving while following a preceding vehicle. The following sections explain key technologies in this research area, along with the gaps in the state of the art that this dissertation aims to address.

### 1.2 Optimizing Longitudinal Motions in Car-Following Scenarios

In the modern automobile industry, cruise control and adaptive cruise control (ACC) are well-established technologies focusing on driving safety by automatically adjusting vehicle speed to keep a safe gap from the vehicle ahead. An ecological adaptive cruise control (eco-ACC) system is a variation of cruise controllers that optimizes the following vehicle’s reactions to enhance energy efficiency while maintaining safety [36, 34, 35].

Typically, eco-ACC systems solve an optimization problem over a finite prediction horizon at each time. Over this horizon, the disturbances and constraints acting on the controlled vehicle are predicted based on the vehicle dynamics and surrounding information given at that instant. This approach is referred to as model predictive control (MPC) or receding horizon control. Many researchers have developed eco-ACC systems using
MPC formulations and evaluated their potential in energy-saving [3]. Moreover, it was shown that if the prediction horizon length reaches a certain threshold (e.g., 20 sec), the MPC formulations for eco-ACC can achieve dynamic programming (DP) level of performance, which is considered as a benchmark in optimal control [37]. Figure 1.3 illustrates the concept of eco-ACC implemented in an MPC framework.

### 1.2.1 Objective Functions

The objective functions of eco-ACC problems can be formulated in various ways. A natural choice for the objective function of the eco-ACC is energy expenditure. For instance, for internal combustion engine vehicles (ICEVs), optimization problems are generally formulated as the minimization of the sum of fuel rate or fuel power over a horizon [38, 39]. For battery electric vehicles (BEVs), the objective functions may include a term for battery power or the derivative of battery state of charge (SOC) [40, 41, 42]. Finally, the objective functions of hybrid electric vehicles (HEVs) may consider minimizing energy from both power sources, but mainly focus on minimizing fuel use due to efficiency and pollution considerations [43, 44, 45, 46, 47]. For diesel engine vehicles such as heavy-duty trucks, emission-related terms are also included in the objective function of eco-ACC [48, 49, 50].

For all types of powertrain, the energy demanded at the wheels, or simply the longitudinal acceleration/deceleration [37, 51], the wheel torque [52], or the traction force [53] can be minimized. The “powertrain-agnostic” formulations are rather surrogate optimization problems of direct energy minimization but still play an important role in determining energy consumption [54] and are universally applicable to various powertrains.
Since eco-ACC aims for safe car-following simultaneously, the objective functions often encompass a term for tracking reference, such as desirable time headway gap from the immediately preceding vehicle. Previous literature proposed hierarchical control structures, where the top layer provides an optimal speed profile, while the bottom layer works like a conventional ACC tracking the optimization results [52]. On the other hand, co-optimization of vehicle dynamics and powertrain is another research stream, which can minimize the potential performance degradation due to serving sub-optimal solutions in the layered approaches by jointly optimizing multiple objectives [55, 21].

1.2.2 Constraints

One of the reasons that an MPC is considered a powerful implementation tool for safety-critical systems is because it can impose and update constraints in a forward-looking manner in real-time. The constraints that need to be addressed in typical road environments include road characteristics, traffic signals, and a preceding vehicle. Such constraints can be taken into account in MPC formulations in a forward-looking manner if the information is available through the CAV technologies. For example, road curvatures and grades could be retrieved from digital geographic maps if online or offline map services are available. Many studies have applied road previews to planning fuel-efficient vehicle speed profiles under varying terrains [56, 57, 58, 59]. Traffic signals, for instance, traffic lights, stop signs, and speed limits could be available through dedicated communication channels in the near future [60].

Lastly, the most significant constraint is imposed by the presence of adjacent vehicles [61, 62]. The relative position and speed of an immediately preceding vehicle can be detected by ADAS sensors such as radars. However, the preceding vehicle’s future trajectories are unknown; therefore, dedicated prediction algorithms are needed to predict this information using the given measurements. Such prediction is critical because it can strongly affect the effectiveness of the optimization results and vehicle safety. In previous research, it was discovered that the energy-saving through the eco-ACC systems is dependent on the accuracy of the prediction accuracy of the preceding vehicle’s future trajectories [37, 51]. A survey of previous studies on forecasting human drivers’ behaviors is given in the following section.

1.3 Vehicle Trajectory Forecasting

Knowing the future trajectories of on-road vehicles is essential for many applications in modern transportation systems. For example, autonomous vehicles might require the
future states of neighboring objects over a prediction horizon to enable safe driving [63]. Along with improving safety, the speed profiles of autonomous vehicles can be planned to improve vehicle fuel economy when the future trajectories of a preceding vehicle are given [37]. In addition, the benefit of vehicle hybridization/electrification can be maximized by optimal energy management strategies when future speed profiles of an ego vehicle are known [64, 65]. For such applications, speed forecasting accuracy would significantly impact the control performance. In fact, inaccurate speed prediction can lead to worse fuel economy than the baseline system that purely reacts to driver demands instead of predicting and optimizing [66, 67, 51].

1.3.1 Model-based Prediction Approaches

Many studies have been conducted to predict a vehicle’s future trajectories in the literature. Earlier efforts have developed model-based forecasting methods to predict the future speed of a vehicle. The simplest of these are constant speed [68, 69] and constant acceleration models [70, 71, 69] that project a predicted vehicle’s current speed or acceleration over a prediction horizon. Some previous work has modeled human drivers’ actions based on their characteristics in more elaborate versions. For example, human drivers’ braking actions are modeled in [71] to predict a preceding vehicle’s trajectory at signalized intersections. The authors in [46] have modeled human driver’s pedaling motion based on their observations that most drivers applied acceleration with exponentially decaying patterns. An enhanced version of this exponential model is developed by using radar and V2I information in [43]. Other researchers selected a sigmoid function to model human driver’s acceleration demands [72]. In addition, microscopic traffic models such as an intelligent driver model (IDM) [73], and Gipps’ model [74] have been applied to predict human drivers’ behaviors in car-following scenarios. The models have been utilized to generate the estimated trajectory of a preceding vehicle based on the current states [75, 76, 69, 77].

In general, however, the aforementioned model-based techniques have limitations to address various driving styles and environments. This drawback makes the model-based predictors only applicable for instant-length prediction, for example, less than a couple of seconds, due to the limited accuracy for a longer horizon.

1.3.2 Data-driven Prediction Approaches

With the recent improvements in on-road sensors and communication technologies, advanced learning-based or data-driven forecasting approaches have been widely studied for predicting a human-driven vehicle’s future trajectories considering various features in the
driving environment. Conventional time-series forecasting methods that have been used to predict a vehicle’s speed include an auto-regressive integrated moving average model [78, 79], and an auto-regressive model with external input [80, 79]. An artificial neural network (ANN) is one of the most popular prediction techniques and has been studied in vehicle trajectory forecasting applications. For example, road geometry data [81], past speed measurement [82], and vehicle’s speed and distance to intersections [83] were applied to forecasting vehicle speed trajectories. In addition, more complex regression techniques such as recurrent neural networks (RNN) and long short-term memory (LSTM) networks have been widely studied for vehicle trajectory prediction. For instance, in [84], RNN is used to estimate drivers’ future pedaling and steering motions with ADAS sensor measurements and vehicle communication information. In [79], LSTM is developed to predict the future speed of a vehicle based on V2I information and speed measured by sensors. In [85], the authors emphasize that using the sensor data with LSTM layers can yield high accuracy for the short and mid-length of future vehicle speed profiles. In [86], an LSTM network is augmented in a graph neural network to predict the trajectories of surrounding traffic participants.

Another direction in trajectory forecasting is stochastic or probabilistic prediction approaches, such as a Markov chain model based on the past acceleration of the predicted vehicle [87, 88, 79]. The variant of hidden Markov models is used in [89] to predict human-driven vehicles’ future speed profiles. In addition, a conditional linear Gaussian model has also been studied based on measurements obtained by a radar and V2V and vehicle-to-infrastructure (V2I) communications [90, 79]. In [91], a Bayesian network is used to detect and predict the corresponding trajectories. More complex predictors have been explored that combine regression, and probabilistic approaches to enhance prediction accuracy [92, 93, 94]. Several popular deterministic and stochastic prediction methods are developed, and their accuracy is compared in [69, 79, 82]. The influence of trajectory forecasting on eco-driving control is analyzed and compared with various prediction techniques in the previous literature [95, 51, 96].

Using advanced and complex learning-based forecasting methods, data-driven predictors can have high accuracy. However, these advanced forecasting techniques are powerful only when a sufficient amount of training data is available. Training these prediction algorithms to be versatile enough to handle different driving environments and guarantee high accuracy over various traffic scenarios would be extremely expensive since enormous datasets should be collected under various conditions. In addition, tuning hyperparameters of deep-learning predictors is not straightforward in practice, which is not a negligible issue since the performance of deep learning algorithms is closely dependent on their hyperpa-
Considering these challenges in using deep-learning predictors, a few earlier research efforts have developed simple data-driven predictors. For example, in [97], a linear regression is used to predict speed trajectories 2 s ahead by estimating coefficients based on the past measurements only. However, the prediction quality of the simple linear regression is insufficient to address a prediction horizon length demanded by most eco-ACC systems, for example, longer than 10 s.

Moreover, while advanced data-driven methods can learn weights on input features through dedicated algorithms like backpropagation, no schemes are given for the aforementioned simple data-driven predictor with a single layer. No studies present methodologies for assigning weights on input features in vehicle trajectory forecasting.

In addition, most previous studies have focused on developing predictor designs with high accuracy over an entire prediction horizon by minimizing loss functions, such as mean-square errors. However, near-future prediction steps would be more critical than far-future steps to the performance of controllers implemented in a receding-horizon manner. Nonetheless, no research has investigated predictor designs that are the most beneficial for MPC-based eco-ACC systems.

### 1.4 Contributions and Organization

The overall contribution of this dissertation encompasses the design of vehicle speed forecasting methods to achieve better energy efficiency in car-following scenarios and the analysis of potential energy savings by applying the methods to various applications. Accordingly, the organization and contributions of this dissertation are summarized in Figure 1.4.
In the following section, the driver models and control formulations used in this dissertation are summarized.

1.4.1 Driver Models and Control Formulations

The proposed forecasting strategies developed in this dissertation are applied to various eco-driving control formulations:

1. **QP-HC**: Acceleration minimization formulated as a quadratic program (QP) with hard constraints on the ego vehicle’s position. This optimal control problem is explained in detail in Chapter III.

2. **QP-SC**: Acceleration minimization formulated as a QP with soft constraints on the ego vehicle’s position by penalizing slack variables. The optimal control problem formulation is elaborated in Chapter VI.

3. **PMP-Pwt**: Approximate energy minimization formulated based on a Pontryagin’s minimum principle (PMP) [98].

4. **PMP-Spd**: Acceleration minimization formulated based on a PMP [99].

In addition, the developed eco-driving control systems combined with the proposed prediction strategies are implemented in various traffic scenarios by applying different longitudinal human driver models:

1. **Repetitive driver model (RDM)**: The following vehicle drives the same speed profile with its preceding vehicle with a time lag.

2. **Intelligent driver model (IDM)**: a microscopic traffic model describing human car-following dynamics maintaining safe distance gaps [73].

3. **Analytical optimal solution model (AOM)**: Analytical solutions of an optimal control problem are used where the control problem maximizes driving comfort while maintaining a desired time headway without collision [100].

4. **Analytical anticipative optimal drivability model (A2ODM)**: Enhanced version of AOM by improving model accuracy [101].

Table 1.1 summarizes where the different types of driver models and control formulations are used. In the table, “T” indicates the human driver model is used for traffic generation and “B” represents the human driver model is used as a baseline for energy consumption comparison.

Accordingly, the following contributions have been made in this thesis:
• Development of accurate and economic vehicle speed predictors using vehicle wireless communication

In Chapter II, an accurate data-driven speed prediction method is proposed, which does not require expensive data collection and training processes. By exploiting V2V information, the polynomial regression improves vehicle speed prediction accuracy significantly compared to the previous literature [97]. The second-order polynomial coefficients are estimated by a weighted-least-square (WLS) algorithm, and measurement weights are determined by exhaustive search using federal drive cycles. The development of the predictor is explained in Chapter II, and published in the conference proceeding:


The proposed predictor is then applied to an eco-ACC simulation and validated that its accuracy is sufficient to improve energy efficiency of ICEV, BEV, and power-split HEV. Furthermore, the influence of vehicle connectivity on the proposed predictor is studied. These topics are further discussed in Chapter III and published in the following conference article:


The extended version of the aforementioned predictor has been developed based on a locally weighted polynomial regression algorithm and utilizing V2V and V2I in-

Table 1.1: Traffic models and eco-driving control formulations used in this dissertation.

<table>
<thead>
<tr>
<th>Category</th>
<th>Type</th>
<th>Ch. II</th>
<th>Ch. III</th>
<th>Ch. IV</th>
<th>Ch. V</th>
<th>Ch. VI</th>
<th>Ch. VII</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driver Model</td>
<td>RDM</td>
<td>T</td>
<td>T, B</td>
<td></td>
<td>T</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IDM</td>
<td>T</td>
<td>B</td>
<td></td>
<td>B</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AODM</td>
<td></td>
<td>T, B</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>A2ODM</td>
<td></td>
<td>T, B</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control Formulation</td>
<td>QP-HC</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td></td>
<td>QP-SC</td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PMP-Pwt</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PMP-Spd</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>
formation. This predictor enables the elongation of a prediction horizon length with satisfactory accuracy for the eco-driving control applications. The prediction method and validation with the eco-driving controllers are presented in the following articles:


• **Measurement weighting strategy for the vehicle trajectory predictors using onboard sensor and V2V information**

While most data-driven predictors can learn weights on input features through dedicated algorithms like backpropagation, no schemes are given for the aforementioned simple data-driven predictor. This task proposes a method for assigning weights on input features when a predictor does not have any dedicated algorithm for optimal weighting. First, correlations between the input features (sensor measurements and V2V information) and the prediction target (vehicle future speed) are identified based on real-world driving data. Then, the correlations are modeled based on exponential functions with the parameters, a forgetting factor, and a discount factor. Finally, the invented weighting function is applied to the WLS of the polynomial regression and validated by the eco-ACC simulation with various powertrain types. The results serve as a guideline when using data-driven predictors without an optimal tuning technique such as backpropagation. This topic is explained in detail in Chapter VI and published in the following journal:


• **Designing the loss function of data-driven vehicle speed predictors to enhance energy efficiency**

Trajectory predictors developed in previous literature are trained by minimizing averaged accuracy metrics such as mean-squared errors. This metric, or loss function,
is optimally designed in this dissertation to improve vehicle energy efficiency. First, the influence of preview uncertainties for every prediction step is quantified. Then, a loss function is formulated as weighted-mean-square errors reflecting the quantified influence. Finally, the efficacy of the proposed loss function is validated by applying it to various data-driven predictors such as an artificial neural network and long short-term memory network. Simulation results show that the proposed loss function allows data-driven predictors to be more effective and reliable in enhancing energy efficiency in the eco-ACC system. This topic is under preparation for journal publication.
CHAPTER II

Forecasting Short-term Speed Trajectories of a Preceding Vehicle using V2V Information

2.1 Introduction

Knowing the future trajectories of on-road vehicles is essential for many applications in the modern transportation system. For example, autonomous vehicles might require the future states of neighboring objects over a prediction horizon to enable safe driving [63]. Along with improving the safety, the speed profiles of autonomous vehicles can be planned to improve vehicle fuel economy when the future trajectories of a preceding vehicle are given [37]. In addition, the benefit of vehicle hybridization/electrification can be maximized by optimal energy management strategies when future speed profiles of an ego vehicle are known [64, 65]. For the aforementioned applications, the accuracy of speed forecasting would have significant impact on the control performance. In fact, inaccurate speed prediction can lead to worse fuel economy than the baseline system that purely reacts to driver demands instead of predicting and optimizing [66, 67].

Many studies have proposed speed forecasting strategies using various types of algorithms and information. Lefèvre et al. [69] categorized speed forecasting methods into parametric and non-parametric methods and compared the overall performances of different types of algorithms. Since the behavior of human drivers is not deterministic, non-parametric methods, which are data-driven algorithms, can achieve higher prediction accuracy than model-based prediction methods [69]. Hence, much effort has been made to collect data that can well describe driving features and to find effective techniques to develop accurate speed predictors. Without the availability of vehicle communication, the past speed and acceleration measurements of an individual vehicle have been applied to speed prediction with a variety of techniques, such as artificial neural networks [82], Markov chain models [87], and polynomial regressions [97]. However, since vehicle speed is af-
fected not only by its inertial effect but also, more importantly, by exterior factors, it is challenging to predict future driving scenarios by using only the past measurements of an ego vehicle.

As advanced technologies in V2V and V2I communications have been developed, more information is expected to be available for on-road vehicles. Under the assumption of the availability of vehicle communication technologies, many approaches have recently been proposed to speed forecasting by utilizing exogenous information such as the preceding vehicle’s speed [105], traffic signals [90], and traffic flows [106]. Most of the previous works produce prediction based on probability-based or machine learning algorithms, which require vast amounts of data in advance. This would demand excessive cost to construct predictors. Moreover, the previously proposed predictors cannot be implemented when vehicle connectivity is not available. This is not a trivial point, considering the low penetration of vehicle wireless communication.

In this chapter, we propose a novel approach to resolving the aforementioned limita-
tions when forecasting a vehicle’s speed for short horizons (e.g., 10 s). As the schematic overview of the proposed approach shows in Figure 2.1, an “ego” vehicle is a controlled vehicle that needs its preceding vehicle’s, namely, a “target” vehicle’s, short-term future speed profiles. Thus, the proposed predictor in this chapter aims to forecast the target vehicle’s speed trajectories. The instant speed information of multiple vehicles driving in front of the target is used to approximate the future speed trajectory of the target. Polynomial regression based on the weighted least-squares method is applied to the data in order to compute the future speed trajectories of the target. When vehicular communication is not available, speed prediction can be computed by using the speed measurements of the target vehicle, which can be measured by sensors equipped in the ego vehicle. The efficacy of the proposed method has been evaluated in single-lane traffic simulations over the federal drive cycles and the cycles applying car-following dynamics.

The rest of this chapter is organized as follows: In Section 2.2, the proposed prediction approach is explained. Section 2.3 presents a framework to simulate traffic scenarios including car-following models. Simulation results are provided and discussed in Section 2.4. Finally, Section 2.5 draws the conclusions of this study.

2.2 Prediction Algorithm

The proposed forecasting method is composed of two steps, namely, V2V data processing and polynomial regression. In this section, the details of each step, including tuning weighting parameters, are presented. In the proposed method, two types of information are required to predict the target vehicle’s future speed:

1. Target vehicle: past speed, and current speed and location
2. Other preceding vehicles: current speed and location

The near-past speed profiles of the target vehicle are collected in real time using the ego vehicle’s sensor, such as radar. We assume that the location and speed of all the preceding vehicles can be obtained through vehicle wireless communications. Detailed information about how these types of data are utilized in the proposed algorithm are described in the following sections.

2.2.1 Processing Vehicle-to-Vehicle Basic Safety Messages

The main idea of the proposed prediction method is that we approximate the target vehicle’s future speed profile based on the speed of its preceding vehicles. To do this, the
locations of the preceding vehicles are mapped onto the time domain and considered into future time. This process is conducted by defining an estimated time of arrival (ETA) as:

$$\hat{t}_{k,j} = s_k^{<j>} - s_k^{TV} / v_k^{TV}, \quad \forall j \in \{1, \ldots, l_k\}$$  \hspace{1cm} (2.1)

where the subscript $k$ is used to denote a variable that has a value at the current time step $k$. The subscript $j$ represents the indices of the preceding vehicles. The number of preceding vehicles $l_k$ in front of the target can vary over time. The location of the $j$th preceding vehicle and the location of the target vehicle at time $k$ are denoted by $s_k^{<j>}$ and $s_k^{TV}$, respectively. The current speed of the target vehicle is represented by $v_k^{TV}$. This process is performed in real time, and the processed data is used as one of the inputs in forecasting speed trajectories of the target vehicle.

### 2.2.2 Calculation of Future Speed Trajectories

The future speed of the target vehicle is computed based on polynomial regression, which is simple and computationally effective. An estimated future speed at a prediction horizon $p$ is calculated as:

$$\hat{v}_{k+p} = \sum_{i=0}^{n} \beta_{i,k} \cdot [(k + p) - (k)]^i = \sum_{i=0}^{n} \beta_{i,k} \cdot p^i.$$  \hspace{1cm} (2.2)

The coefficient of the $n_{th}$ order polynomials, $\beta_{i,k}$, is estimated based on the WLS method, where the problem is derived as:

$$\beta_k = \arg \min \| W_k^{1/2} (\hat{v}_k - v_k) \|_2^2$$  \hspace{1cm} (2.3)

where $\hat{v}_k$ is a vector composed of estimated speed. The measurement vector $v_k$ has the speed data of the target vehicle and the speed of the other preceding vehicles collected at time $k$, as given by:

$$v_k = \begin{bmatrix} v_k^{TV} \\ v_k^{<1>} \\ \vdots \\ v_k^{<h_k>} \\ \vdots \\ v_k^{<l_k>} \end{bmatrix}^T$$  \hspace{1cm} (2.4)

where the subscript $h_k$ represents the number of the target vehicle’s speed measurements collected at $k$. Considering the performance of regression, we limit the maximum number
of data points for past speed as follows:

\[ h_k = \begin{cases} 
  h_k & \text{if } h_k < h_{\text{max}}, \\
  h_{\text{max}} & \text{otherwise}. 
\end{cases} \]

Particularly, \( h_{\text{max}} \) of 10 is adopted from [97]. When the target vehicle fully stops, the prediction process stops and resets, i.e., \( h_k = 0 \). Once the vehicle restarts driving, data are collected again and \( h_k \) increases until it reaches \( h_{\text{max}} \).

The weighting matrix \( W_k \) in (2.3) is a diagonal and positive definite matrix. The weighting matrix can be defined by users to achieve different desired performances. In this chapter, we introduce a time-decaying weighting function. The weighting matrix is constructed such that the \( i^{\text{th}} \) diagonal element of \( W_k \) is defined as:

\[
\{ W_k \}_i = \begin{cases} 
  \exp \{ \alpha \cdot (-h_k + i) \} & \text{if } 1 \leq i \leq h_k, \\
  \exp \{ -\gamma \cdot (\hat{t}_k,i - h_k) \} & \text{if } h_k + 1 \leq i \leq h_k + l_k. 
\end{cases}
\]  

When a measurement is the past speed of the target vehicle, the input of the weighting function is the time difference between current time and the time when the past measurement was sampled in the past. When a measurement is a preceding vehicle’s speed, the input of the weighting function is ETA, defined in (2.1).

Figure 2.2 visually describes the principle of how a weighting value is determined for each speed measurement. As shown in Figure 2.2, the weighting function in (2.5) is composed of two different exponential functions, and each of them is governed by \( \alpha \) (past information) and \( \gamma \) (future information), respectively. Data points collected close in time are more weighted than other data points. That is, the current speed data are always weighted by 1. On the other hand, weights for other data are exponentially decaying from 1 as the data become old or further from the current time. The decay rates for the past and the future data can be adjusted by the weighting parameters \( \alpha \) and \( \gamma \), respectively. In this example, the \( \alpha \) is set to one and \( \gamma \) is 0.3. Since the weighting function is composed of exponential functions, if \( \alpha \) or \( \gamma \) are zeros, the weighting matrix is an identity matrix and every measurement is weighted equally. More details about how we select these weighting parameters are described in Section 2.4.1.

The analytic solutions of problem (2.3) can be obtained by:

\[
\beta_k = (X_k^T W_k X_k)^{-1} X_k^T W_k v_k \tag{2.6}
\]

where a design matrix \( X_k \) is constructed based on the differences between the current and
Figure 2.2: An exponential weighting function as proposed in this chapter. The weighting values for past speed measurements are determined according to the blue shaded region. Preceding vehicles’ speed measurements are weighted according to the function in the red shaded region. The coefficients on the power terms of exponential functions are $\alpha = 1$ and $\gamma = 0.3$ for the past and future time, respectively.

It is noted that the dimension of $X_k$ can vary over time, since the number of rows is determined by the number of collected data at time $k$. The columns of $X_k$ are determined by the desired order of polynomial fits.
2.3 Simulation Settings

2.3.1 Car-following Traffic Scenarios

The proposed prediction method is implemented in simulations under a single-lane traffic scenario. In this chapter, we assume dedicated short-range communication (DSRC) is available for the vehicles and they share basic safety messages (BSMs). In the simulations, we assume that there are five vehicles in front of the target vehicle and all of them can communicate when they are located within communication range. It is difficult to define the communication range of V2V communication, since it can be affected by various factors such as the line-of-sight (LOS) of signals, climate, and road types [107]. Thus, the V2V communication range is fixed to 300 m in our simulation. In other words, we assume that all the packets transmitted by preceding vehicles driving within 300 m from the target vehicle are successfully delivered to the target vehicle. On the other hand, if a preceding vehicle exceeds 300 m, the packets are dropped.

Since we do not limit the application of our prediction algorithm, the ego vehicle’s drive cycle is not generated in the simulation. Thus, we assume that the ego vehicle is driving near the target vehicle for entire trips and the ego vehicles’ V2V connectivity is the same as the target’s for simplicity. In addition, a following vehicle behind the ego vehicle is not taken into account in the simulations.

The drive cycles of vehicles are generated based on federal standard drive cycles such as HWFET, UDDS, and US06. These cycles are selected to evaluate the performance of the proposed forecasting method in different driving characteristics: non-stop highway driving, medium-speed urban traffic with frequent stop-and-go, and high-speed aggressive urban traffic. Single-lane traffic simulations are produced based on two different models:

1. Repetitive driver model (RDM): the same federal drive cycles are driven by all the preceding and target vehicles, with distance and time gaps enough to avoid crashes.

2. Intelligent driver model (IDM): The first preceding vehicle drives a federal standard drive cycle and the others drive according to IDM-based car-following dynamics.

The first traffic model is designed to evaluate the best performance of the proposed algorithm. The second traffic model could generate more realistic speed profiles. The car-following dynamics model is described in detail in the following section.

\footnote{More detailed information and description about DSRC basic safety messages and signal phase and timing messages can be found in [? ].}
Table 2.1: Definitions of IDM Parameters

<table>
<thead>
<tr>
<th>Definition</th>
<th>HWFET</th>
<th>UDDS</th>
<th>US06</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum acceleration $\bar{a}$ [m/s$^2$]</td>
<td>1.43</td>
<td>1.48</td>
<td>3.76</td>
</tr>
<tr>
<td>Desired speed $\bar{v}$ [m/s]</td>
<td>40</td>
<td>30</td>
<td>40</td>
</tr>
<tr>
<td>Acceleration exponent $\delta$</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Jam distance $d_0$ [m]</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Jam distance $d_1$ [m]</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Safe time headway $T_h$ [s]</td>
<td>2</td>
<td>1.6</td>
<td>2</td>
</tr>
<tr>
<td>Desired deceleration $\bar{b}$ [m/s$^2$]</td>
<td>1.4</td>
<td>1.4</td>
<td>2</td>
</tr>
</tbody>
</table>

2.3.2 Longitudinal Driver Model

In this study, the IDM is selected to generate the speed profiles of a series of following vehicles. The IDM is a microscopic car-following model describing the crash-free responses of following drivers developed by Treiber et al. [73]. According to the IDM, the acceleration of the $j^{th}$ vehicle is:

$$ \frac{dv_j}{dt} = \bar{a} \left[ 1 - \left( \frac{v_j(t)}{\bar{v}} \right) ^{\delta} - \left( \frac{d^*(v_j, \Delta v_j)}{d_j(t)} \right) ^2 \right] $$  \hspace{1cm} (2.7)

where $v_j$ is the current speed of the $j^{th}$ vehicle, $d_j$ is the actual gap between the $j^{th}$ vehicle and the $(j-1)^{th}$ vehicle (a preceding vehicle) such that:

$$ d_j(t) = s_{j-1}(t) - s_j(t) - L $$  \hspace{1cm} (2.8)

where $s_j$ is the one-dimensional location of the $j^{th}$ vehicle and $L$ is the length of a car, which is set to 5 m in our simulation. In (2.7), $\Delta v_j$ is a relative speed between the $j^{th}$ and the $(j-1)^{th}$ vehicles, such that:

$$ \Delta v_j(t) = v_j(t) - v_{j-1}(t). $$  \hspace{1cm} (2.9)

In (2.7), $d^*$ is a desired gap between vehicles, defined as:

$$ d^*(v, \Delta v) = d_0 + d_1 \sqrt{\frac{v}{\bar{v}}} + T_h \cdot v + \frac{v \Delta v}{2\sqrt{\bar{a}\bar{b}}}. $$  \hspace{1cm} (2.10)

The definitions and the values of other IDM parameters in (2.7) and (2.10) that we use for the simulation are presented in Table 2.1.
2.3.3 Validation of Longitudinal Traffic Simulation

The safe driving model proposed in [108] is adopted to validate the safety of the longitudinal traffic model used in this chapter. The safety limits produced by this model suggest the minimum separation between a preceding and a following vehicle to prevent crashes. The minimum distance for safety, \( d_{j,\text{min}} \), is derived by maintaining a one-car-length distance between two vehicles for every 10 mph, such that:

\[
d_{j,\text{min}}(t) = \frac{v_j}{10 \text{ mph}} L + b
\]  

(2.11)

where \( b \) is an acceptable distance between two vehicles at a standstill, which is set to 1 meter in our simulation. The example of the safety limit over the US06 cycle is shown in Figure 2.3. In the top subplot, the blue line represents the actual gaps between the fifth preceding and the target vehicle. The dotted line represents the minimum distance gaps between two vehicles recommended for safety. It is clearly shown that the generated speed trajectory for a vehicle following one that perform a federal test procedure satisfies the safety limits during the entire trip. The same validations are conducted when the preceding vehicle performs the HWFET and the UDDS.

The middle subplot of Figure 2.3 shows the distance gap between the first preceding vehicle and the target vehicle (blue lines). The red line indicates the communication range limit. The vehicles are not connected when a distance between them exceeds the limit, which is presented as dashed lines. As shown in the bottom subplot, the number of preceding vehicles for data communication changes over time.

2.4 Simulation Results and Discussion

2.4.1 Selecting Weighting Parameters

This section describes how we determine the weighting parameters \( \alpha \) and \( \gamma \) to improve the prediction accuracy of the proposed method. Since we do not limit the applications of our algorithm in this chapter, we consider the prediction accuracy at every prediction horizon equally important. Note that various methods can be used to design a weighting matrix for different applications. An example problem of searching optimal weighting parameters can be formulated as follows:

Since we consider every prediction horizon equally important in this chapter, the main focus of the selection process is to find weighting parameters that can minimize overall prediction errors over the whole prediction horizon. To do this, we consider mean absolute
Figure 2.3: Examples of traffic simulations and validation based on US06 and IDM. In the top subplot, the blue line shows the distance from the closest preceding vehicle to the target vehicle, and the black dotted line is the safety limit defined in (2.11). The middle subplot presents the distance from the farthest preceding vehicle to the target vehicle. Since the first preceding vehicle exceeds the communication range from around 180 s to 480 s, the ego vehicle loses the first preceding vehicle’s signals during this period. The time-varying number of preceding vehicles connected to the ego vehicle is shown in the bottom subplot.

Mean Absolute Error (MAE) to be a measure of prediction performance. The performance metric $J$ is defined as:

$$J = \sum_{p=1}^{N_p} \left( \frac{\mu_p - \bar{\mu}_p}{\bar{\sigma}_p} \right)$$

(2.12)

where $\mu_p$ is the MAE of speed prediction at the $p^{th}$ prediction horizon, and $N_p$ is the number of time steps in a prediction horizon, set to 10 in this chapter. MAE values are computed with the different combinations of $\alpha$ and $\gamma$. Since the MAE at the end of the prediction horizon is larger in general, the mean normalization is applied in (2.12) to scale the $\mu_p$ of different prediction horizons. The mean ($\bar{\mu}_p$) and the standard deviation ($\bar{\sigma}_p$) of $\mu_p$ are computed for each prediction horizon $p$. In this example, we restrict the search range of $\alpha$.
The performance metric $J$ defined in (2.12) is computed with different $\alpha$ and $\gamma$ using US06 and shown in the left subplot. The red cross notates the minimum $J$ value, where the optimal weighting parameters are $\alpha^* = 1$ and $\gamma^* = 0.3$. In the right subplot, MAE is compared for the cases without weighting (blue) and with optimal weighting (red).

The example results of the proposed selecting process for weighting parameters are presented in Figure 2.4. The quadratic polynomial fitting and the RDM traffic model over US06 are used for this example. In the left subplot, the contour plot of the performance metric $J$ is presented with different values of $\alpha$ and $\gamma$. The minimum of $J$ is found on $(\alpha^*, \gamma^*) = (1, 0.3)$, marked as a red cross on the contour plot. The MAE of prediction from the optimal weighting parameters are compared with those from no weighting ($\alpha = \gamma = 0$) in the right subplot in Figure 2.4. The results show that by applying the optimal weighting parameters (red), the predictor can have smaller prediction error, in particular, from the first to the seventh prediction horizons. The same search is conducted for other federal drive cycles.

### 2.4.2 Comparison of Predictor Performance

The proposed speed forecasting approach is evaluated using traffic simulations introduced in the previous section. Future speed trajectories are computed for 10-second prediction horizons based on the linear and the quadratic polynomial regression methods to avoid over-fitting. The frequency of data sampling and prediction calculation is 1 Hz. In order to compare the effect of V2V information, speed prediction calculated without V2V data is compared to the proposed strategy.

To highlight the performance of the proposed speed predictor, actual speed (blue) and its prediction (red) over the US06 drive cycle are visually shown in Figure 2.5. The top
Figure 2.5: Demonstration of the proposed speed predictor on US06 with the RDM traffic model. In the top result, the predictor uses only the past speed trajectories of the target for prediction; thus, the linear polynomial regression is applied. In the bottom subplot, the predictor uses both the past speed and the preceding vehicles’ speed, and the quadratic polynomial regression is applied.

and bottom subplots provide results generated by the proposed predictor without V2V and with V2V information, respectively. It can be clearly seen that the prediction results produced by using V2V information (bottom) have significantly smaller deviations from the actual cycle, compared to those produced by using the target’s measurements only (top). This is because when V2V information is absent, the proposed predictor computes future speed based on extrapolation. On the other hand, by including the preceding vehicles’ information as the potential future speed, prediction can be computed based on interpolation, which results in better prediction accuracy. These principles are explained visually in Figure 2.6. Figure 2.6 shows a situation in which the target vehicle cruises at 20 m/s (45 mph) and decelerates much over the next few seconds, which is depicted as the blue lines in both subplots. This time instant is chosen to emphasize the effect of V2V information in the proposed speed prediction method. The left subplot shows a prediction result (red crossed line) from using only measurements from the target vehicle (blue crosses),
and the prediction is the result of extrapolation. Clearly, the prediction error at the end of the prediction horizon is considerably large, which is due to unexpected acceleration and deceleration in the future. On the other hand, the right subplot shows a prediction result from including V2V information (green crosses) in the regression process. In this case, the preceding vehicles’ location information is mapped onto the future time domain by the proposed method explained in Section 2.2, and the future speed trajectory is computed based on interpolation. These types of unexpected acceleration or deceleration in the future cannot be foreseen when the predictor uses only the past speed measurements of the target vehicle. Using V2V data, the predictor can expect deceleration in the near future by observing how the vehicles ahead are acting.

The root-mean-square errors (RMSE) and the 90\textsuperscript{th} percentile of prediction errors from the proposed speed prediction method are summarized in Table 2.2 and Table 2.3, respectively. Overall, a linear polynomial fitting results in better prediction accuracy for the cases without V2V measurements, while a quadratic polynomial fitting results in better prediction accuracy for the cases with V2V measurements. This is because as the data used in regression increase, a high-order regression performs better. In addition, as explained in Figure 2.6, speed profiles having high acceleration or deceleration are more difficult to predict by the proposed predictor. Thus, the prediction accuracy over the HWFET cycle is always better than that over the US06 cycle. Even in the traffic simulations describing a more realistic driving scenario, the proposed prediction strategies can provide improved
Table 2.2: RMSE of Prediction Errors [m/s]

<table>
<thead>
<tr>
<th>Cycle</th>
<th>Traffic Model</th>
<th>No V2V info</th>
<th>Use V2V info</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linear</td>
<td>Quadratic</td>
<td>Linear</td>
</tr>
<tr>
<td>HWFET</td>
<td>RDM 1.31</td>
<td>2.39</td>
<td>0.44</td>
</tr>
<tr>
<td></td>
<td>IDM 1.17</td>
<td>1.90</td>
<td>0.46</td>
</tr>
<tr>
<td>UDDS</td>
<td>RDM 3.00</td>
<td>5.11</td>
<td>1.08</td>
</tr>
<tr>
<td></td>
<td>IDM 4.56</td>
<td>10.08</td>
<td>1.98</td>
</tr>
<tr>
<td>US06</td>
<td>RDM 4.92</td>
<td>6.75</td>
<td>1.82</td>
</tr>
<tr>
<td></td>
<td>IDM 6.30</td>
<td>10.70</td>
<td>2.55</td>
</tr>
</tbody>
</table>

Table 2.3: The 90th Percentile of Prediction Errors [m/s]

<table>
<thead>
<tr>
<th>Cycle</th>
<th>Traffic Model</th>
<th>No V2V info</th>
<th>Use V2V info</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linear</td>
<td>Quadratic</td>
<td>Linear</td>
</tr>
<tr>
<td>HWFET</td>
<td>RDM 1.77</td>
<td>2.95</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>IDM 1.51</td>
<td>2.33</td>
<td>0.53</td>
</tr>
<tr>
<td>UDDS</td>
<td>RDM 4.99</td>
<td>7.30</td>
<td>1.69</td>
</tr>
<tr>
<td></td>
<td>IDM 8.02</td>
<td>14.82</td>
<td>3.28</td>
</tr>
<tr>
<td>US06</td>
<td>RDM 7.51</td>
<td>9.56</td>
<td>2.61</td>
</tr>
<tr>
<td></td>
<td>IDM 10.72</td>
<td>18.25</td>
<td>4.21</td>
</tr>
</tbody>
</table>

prediction accuracy with measurements collected from the preceding vehicles for all the drive cycles. The results show that the proposed assumption that the target vehicle will follow a speed trajectory derived from the preceding vehicles’ state trajectories over short-term future time is valid.

2.5 Summary

This chapter improves the accuracy of a polynomial-regression-based vehicle speed predictor by proposing a strategy leveraging V2V information. Overall, V2V information enhances prediction quality significantly — by 73 % for the best-case scenario and 59 % for the realistic scenario. Moreover, the proposed predictor has high availability; it can be implemented when the vehicle has lost V2V signals and when the driving environment is not experienced before. In the next chapter, the proposed short-horizon predictor is used in an eco-ACC system, and the resultant energy saving is analyzed to evaluate the efficacy of the proposed predictor.
CHAPTER III

Influence of Vehicle Speed Forecasting on Ecological Adaptive Cruise Control Performance

3.1 Introduction

In the previous chapter, the predictor forecasting vehicle speed over short-term horizon is introduced. To assess the efficacy of the predictor, this chapter integrates the predictor in one of the typical eco-driving control applications. This chapter explains the formulation of the eco-driving control application we use, and the influence of speed prediction on energy efficiency achieved by the controller.

The recent progress in vehicle automation and advanced driver-assistance systems present opportunities to achieve better fuel economy in the control aspect [32]. One promising technology is eco-ACC, differentiated from conventional adaptive cruise control in that it aims to enhance fuel economy as well as driving safety. An MPC has been widely studied in developing eco-ACC strategies for its capability of handling constraints and forward-looking. In [39], it is shown that for city driving, eco-ACC based on MPC can reduce fuel consumption by 10% compared to the PID-based ACC on average.

As MPC solves an optimization problem over a predefined prediction horizon in a forward-looking manner, previews on surrounding environment are crucial. The required previews for MPC-based eco-ACC include road grades in front, upcoming traffic signals, and an immediate preceding vehicle’s future trajectory. For example, many studies have applied road previews to planning fuel-efficient vehicle speed profiles under varying terrains [56, 57, 58, 59]. In addition, knowing traffic light cycles in advance allows a vehicle to reduce idling at signalized intersections while obeying traffic rules. These strategies, the so-called “eco-approach and departure,” have been intensively studied [27, 25, 26, 52]. Previews on an immediately front vehicle’s trajectories are beneficial for smoothing a following vehicle’s speed profiles, and as a result, reduce fuel consumption [37].
Figure 3.1: Schematic overview of the eco-ACC system studied in this chapter. The ego (following) vehicle predicts the future speed of the target vehicle by using V2V information from the multiple preceding vehicles (PV 1 to PV N in the figure) and the target vehicles.

Most previous works on eco-ACC based on MPC have focused on controller designs under the assumption that the aforementioned future information is available. This assumption could be reasonable for deterministic information such as road grades and traffic lights. Specifically, road grade data can be provided by online platforms providing digital geographic maps. Traffic signal information would be available through DSRC SPaT messages [8]. However, unlike the road grades and the traffic signals, future states of a human-driven vehicle are difficult to obtain since they are dynamically varying with high uncertainties.

Some previous works have proposed eco-ACC strategies applying the prediction of the immediately front vehicle’s future states. The simplest approach is a constant acceleration model, which propagates current acceleration over a prediction horizon. However, this method could result in large errors at the end of the prediction horizon, and even could degrade the eco-ACC performance [66]. The authors in [44] proposed an exponentially decaying model to describe acceleration patterns by human drivers based on their finding that most drivers apply small acceleration with overall constant speed. Similarly, the
authors in [72], designed an acceleration prediction model based on the sigmoid function. However, these model-based prediction methods have limited accuracy since the models are defined by fixed parameter values which cannot reflect time-varying traffic conditions and driver personalities. Moreover, few studies have assessed the influences of these prediction methods on eco-ACC performance to verify how important the state predictions are for fuel economy improvements.

In this chapter, we present the design of eco-ACC using the data-driven speed predictor proposed in the previous chapter, and investigate the influences of the prediction on the eco-ACC of a following autonomous vehicle. In the following autonomous vehicle (“ego” in Figure 3.1), the speed predictor provides an estimated speed profile of an immediate preceding vehicle (“target” in Figure 3.1) to the eco-ACC controller. The speed predictor utilizes the speed and location of multiple preceding vehicles (“PV 1” to “PV N” in Figure 3.1) to reflect instant traffic conditions ahead. In this chapter, the states of the multiple preceding vehicles are assumed to be available through V2V communication technologies. The eco-ACC problem is formulated based on the MPC framework. The speed prediction of the target vehicle is integrated to position constraint computation, which maintains safe distance from the target vehicle to the ego vehicle to avoid crashes. A simulation study is performed under a single-lane traffic scenario in urban conditions. The influences of speed forecasting on the performance of the eco-ACC are evaluated in terms of acceleration minimization and fuel economy. Furthermore, the relationships between horizon-wise prediction accuracy and vehicle fuel economy are investigated.

The remainder of this chapter is organized as follows. Section 3.2 details the vehicle and longitudinal dynamics model. Section 3.3 introduces the optimal control problem based on an MPC technique. This eco-ACC system is implemented with the vehicle speed predictor introduced in Section 2.2, and the simulation results are presented in Section 3.4. Section 3.5 summarizes the contributions and future work.

3.2 Model Description

In this section, the details of vehicle dynamics model are described. Since the goal of the eco-ACC system is optimizing powertrain control actuators to minimize fuel consumption, the longitudinal dynamics of a vehicle is considered. The considered vehicle is a small SUV equipped with a 2.7L 4-cylinder gasoline engine and 6-speed automatic transmission.
Table 3.1: Vehicle Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m$</td>
<td>kg</td>
<td>1745</td>
</tr>
<tr>
<td>$A_f$</td>
<td>m$^2$</td>
<td>2.841</td>
</tr>
<tr>
<td>$\rho$</td>
<td>kg/m$^3$</td>
<td>1.1985</td>
</tr>
<tr>
<td>$C_r$</td>
<td>-</td>
<td>0.0084</td>
</tr>
<tr>
<td>$C_d$</td>
<td>-</td>
<td>0.356</td>
</tr>
</tbody>
</table>

3.2.1 Vehicle Longitudinal Dynamics Model

Since the main focus of this chapter is energy savings in car-following scenarios, only the longitudinal dynamics of the ego vehicle are considered. For simplicity, we assume that no lane changes occur. Furthermore, the dynamics are simplified with the assumption of negligible road grades and a point-mass vehicle as follows:

\[
\dot{s} = v, \tag{3.1a}
\]

\[
\dot{v} = \frac{1}{m} \left( F_t - mg C_r - \frac{1}{2} \rho A_f C_d v^2 - F_b \right), \tag{3.1b}
\]

where $s$ and $v$ are the longitudinal position and speed of a vehicle, respectively. $m$ denotes the vehicle mass, $F_t$ indicates the traction force, and $g$ is the gravitational acceleration. $C_r$ and $C_d$ represent the rolling resistance coefficient and the air drag coefficient, respectively. $\rho$, $A_f$, and $F_b$ denote the air density, vehicle frontal area, and braking force, respectively. The vehicle parameters used for simulation are extracted from a vehicle energy evaluation software, Autonomie [109]. The parameter values and units are summarized in Table 3.1. The above model is discretized to obtain:

\[
s(k + 1) = s(k) + v(k) \cdot T_s + u_p(k) \cdot T_s^2, \tag{3.2a}
\]

\[
v(k + 1) = v(k) + u_p(k) \cdot T_s, \tag{3.2b}
\]

\[
u_p(k) = u(k) - g C_r - \frac{1}{2m} \rho A_f C_d v(k)^2, \tag{3.2c}
\]

where $u$ is an acceleration command, which is the solution of the optimal control problem described in the next section. The symbol $u$ encompasses deceleration by allowing it to have a negative sign. We use the term “acceleration” to indicate both acceleration and deceleration for simplicity in the rest of the chapter. The equations in (3.2) are used as a plant model in our simulations.
3.3 Ecological Adaptive Cruise Control (QP-HC)

In this chapter, the goal of the eco-ACC controller is to optimize the longitudinal acceleration of the ego vehicle to reduce fuel use with safety guarantees. It is well known that larger acceleration demand results in more fuel consumption [110]. For this reason, we formulate an eco-ACC problem as the minimization of vehicle longitudinal acceleration with desired speed tracking to make the speed profile of the controlled vehicle have less acceleration and deceleration with even speed. The optimization problem is formulated as quadratic programming. The cost function $J$ is constructed as follows:

$$
J = \sum_{i=1}^{N_p} \left( w_a u_{i|k}^2 + w_v (v_{i|k} - v_{des})^2 \right) \tag{3.3}
$$

where $u_{i|k}$ is longitudinal acceleration, $v_{i|k}$ is longitudinal speed at time $k + i$ when the current time is $k$. Two weighting values $w_a$ and $w_v$ are considered for balancing acceleration minimization and speed tracking, respectively. The desired speed $v_{des}$ is the free-flow speed, which is assumed to be constant and equal to the speed limit in each portion of the drive. $N_p$ represents the length of prediction horizons. The system state $x$ predicted by MPC for time $k+i$ is defined as:

$$
x_{i|k} = \begin{bmatrix} s_{i|k} \\ v_{i|k} \end{bmatrix}^T \tag{3.4}
$$

where $s_{i|k}$ and $v_{i|k}$ represent the vehicle longitudinal position and speed of the ego vehicle, respectively. The optimization problem is subject to the vehicle longitudinal dynamics. The vehicle longitudinal dynamics is simplified as a linear time invariant (LTI) model for fast online implementation. The model is discretized with constant sampling time $T_s$ as follows:

$$
x_{i+1|k} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} x_{i|k} + \begin{bmatrix} 0.5T_s^2 \\ T_s \end{bmatrix} u_{i|k} \tag{3.5}
$$

where the control input $u$ is the longitudinal acceleration.

The optimization problem is subject to the vehicle longitudinal dynamics introduced in (3.5) and following constraints:

$$
s_{i|k} \leq s_{i|k}^{target} - L - d_{min}, \quad 0 \leq i \leq N_p, \tag{3.6}
$$

$$
v_{min} \leq v_{i|k} \leq v_{max}, \quad 0 \leq i \leq N_p, \tag{3.7}
$$

$$
u_{min} \leq u_{i|k} \leq u_{max}, \quad 0 \leq i \leq N_p, \tag{3.8}
$$
where \( L \) is one-car length defined as 4.5 meters, \( d_{\text{min}} \) is a standstill distance set to 1.2 m. The minimum and maximum longitudinal speed values are determined based on simulated cycles. For instance, for the urban dynamometer driving schedule (UDDS), the minimum speed is set to zero and the maximum speed is set to 25 m/s. The minimum and maximum longitudinal acceleration, \( u_{\text{min}} \) and \( u_{\text{max}} \), are set to \( \pm 4 \text{ m/s}^2 \). The predicted location of the target vehicle \( s_{\text{target}}^{i|k} \) is generated by integrating predicted speed as follows:

\[
s_{\text{target}}^{i|k} = s_{\text{target}}^{i-1|k} + v_{\text{target}}^{i-1|k} \cdot T_s, \quad 1 \leq i \leq N_p.
\]  

(3.9)

To clarify, the target vehicle is the immediate preceding vehicle from the ego vehicle as described in Figure 3.1. In (3.9), \( v_{\text{target}}^{i-1|k} \) is the \((i - 1)\)-step prediction of the target vehicle. The speed prediction is computed in real time by using the prediction method described in Section 2.2. The acceleration of the preceding vehicle is not considered to avoid additional error accumulation.

Finally, the predicted speed of the target vehicle can be calculated based on the real-time derived polynomial coefficients \( \beta_{m,k} \) where \( m = 0, 1, \) and \( 2 \), as follows:

\[
v_{\text{target}}^{i|k} = \sum_{m=0}^{2} \beta_{m,k} \cdot (k + i - k)^m = \beta_0(k) + \beta_1(k) \cdot i + \beta_2(k) \cdot i^2
\]

(3.10)

where \( i \) is an index for a future time step from the current time \( k \).

### 3.4 Simulations

This section details simulation settings and results of prediction and control performance assessment.

#### 3.4.1 Traffic Scenarios and Baseline Model

The eco-ACC system has been implemented and validated by single-lane traffic simulations. The drive cycles of the preceding and target vehicle are generated by RDM introduced in Chapter 1.4.1 with 2 s time lag and 2 m separation between one another in the beginning of a trip. The RDM is used to eliminate the effect of the car-following dynamics and focus on assessing the influence of prediction and the eco-ACC. Among federal drive cycles, the UDDS and the LA92 are used to assess the eco-ACC in urban driving. It is assumed that all the preceding and the target vehicles can communicate each other through V2V communication technologies. Following vehicles behind the ego vehicle is not taken into account in this analysis.
Table 3.2: IDM Parameter Values

<table>
<thead>
<tr>
<th>Definition</th>
<th>UDDS</th>
<th>LA92</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum acceleration $\ddot{a}$ [m/s$^2$]</td>
<td>1.48</td>
<td>3.08</td>
</tr>
<tr>
<td>Desired speed $\dot{v}$ [m/s]</td>
<td>25</td>
<td>40</td>
</tr>
<tr>
<td>Acceleration exponent $\delta$</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Jam distance $d_0$ [m]</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Jam distance $d_1$ [m]</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Safe time headway $T_h$ [s]</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Desired deceleration $\ddot{b}$ [m/s$^2$]</td>
<td>1.4</td>
<td>1.4</td>
</tr>
</tbody>
</table>

Figure 3.2: Example of speed prediction for 10-second prediction horizon. The blue solid line presents the true UDDS cycle which the target vehicle drives, and the red lines are predicted speed generated at every second. The number of connected preceding vehicles is five in this example.

To compare the eco-ACC driven cycle with a human driver, human driver behaviors are modeled by using the IDM [73]. The IDM parameters used in this chapter are summarized in Table 3.2. The parameter values are chosen to demonstrate a mild driver’s behavior such that the maximum acceleration does not exceed the one in the original cycle.

### 3.4.2 Speed and Position Prediction Results

The V2V-based speed predictor introduced in Section 2.2 is implemented to forecast the target vehicle’s speed over 10-second prediction horizons. In Figure 3.2, the example results of speed prediction are visually shown over the UDDS drive cycle. In this example, we assume that five preceding vehicles can transmit V2V messages to the ego vehicle. The
Figure 3.3: Mean absolute errors (MAE) and root-mean-square errors (RMSE) of speed prediction (top) and position prediction (bottom) from different speed forecasting approaches: constant acceleration model (red), and V2V-based prediction introduced in Section 2.2 with one (blue) and five connected preceding vehicles (green). The UDDS is used for traffic generation.

Figure 3.4: Root-mean-square errors (RMSE) of the speed predictions with the different numbers of the preceding vehicle data. In the legend, “i” indicates a time step after the current time as used in (3.10). The UDDS is used for traffic generation.
predicted speed trajectories are computed in real time and provided to the ego vehicle’s MPC controller. To evaluate prediction performance, three different cases are compared: a constant acceleration model, the V2V-based speed predictor with one preceding vehicle, and five preceding vehicles. The constant acceleration model predicts the target vehicle’s future location by integrating the target vehicle’s acceleration measured at the current time over a prediction horizon with the assumption that the vehicle connectivity is not available.

Horizon-wise accuracy of each method is compared in Figure 3.3 in terms of the MAE and the root-mean-square errors (RMSE) of speed and position predictions. As shown in Figure 3.3, speed predictions based on the constant acceleration method has the largest errors after the second horizon. For the V2V-based prediction, including more preceding vehicle information provides significantly improved prediction accuracy, especially at the end of the prediction horizon. Errors in position predictions rapidly increase at the end of the prediction horizon since the speed prediction errors are accumulated in the position prediction by integration. The prediction accuracy is critical for the eco-ACC since the safety position constraints are based on the predicted positions of the target vehicle. Large position errors might result in infeasibility in the controller or collision from the target vehicle. More discussion on prediction accuracy and fuel economy improvement is introduced in Section 3.4.4.

The accuracy of the V2V-based speed predictor is improved as the number of the preceding vehicle data increases. The speed prediction accuracy with the different numbers of the V2V data is described in Figure 3.4. The left subplot shows the RMSE of speed prediction at relatively short prediction horizons, from the first to the fifth horizons. The right subplot presents RMSE at farther future time steps, which are from the sixth to the tenth horizons. The notation \( i \) is an index for a future time step in a prediction horizon, which corresponds to the definition introduced in (3.10). For the first to third predictions, where \( i = 1, 2, \) and \( 3 \), RMSE values slightly increase as the preceding vehicle data are added. This is because the measurements from far preceding vehicles has less correlation with short future speed of the ego vehicle but high leverage in the regression; which could cause degraded prediction quality for the prediction near the current time. The additional preceding vehicle data improve prediction accuracy after the third step in the prediction horizon. The influences of prediction on the controller and vehicle fuel economy are further discussed in Section 3.4.3 and 3.4.4, respectively.

### 3.4.3 Control Results

The eco-ACC controller introduced in Section 3.3 has been implemented. The example results are shown in Figure 3.5. In these results, the V2V information from the five
preceding vehicles is used to predict the target vehicle’s future speed. The desired speed $v_{des}$ in (3.3) is set to 25 m/s which is the maximum speed in the UDDS. In Figure 3.5, the first two graphs show the longitudinal acceleration and speed of the target (red) and the ego vehicle (blue), respectively. The speed trace by the ego vehicle is smoothed effectively by using MPC with acceleration minimization. This speed smoothing effect is well described in Figure 3.6. The accelerations of the target and the ego vehicles are compared in the left subplot, and the corresponded speeds are shown in the right. In Figure 3.5, the third subplot presents the relative distance from the target vehicle and it is converted to the time headway in the bottom subplot. Overall, the ego vehicle can closely follow the target vehicle with the time headway of 2 s.

In Figure 3.7, the distributions of longitudinal acceleration in the UDDS drive cycle
and the ego vehicle driven by the eco-ACC are compared. The histograms show that the eco-ACC has smaller maximum and minimum accelerations than those of the UDDS. The standard deviation of acceleration ($\sigma$) by the eco-ACC is 22% less compared to that of the IDM for the UDDS. This effect is more significant when the ego vehicle follows an aggressive drive cycle such as the LA92. Fuel saving results corresponding to the smoothed cycles are described in the following section.

3.4.4 Fuel Economy Assessment

To evaluate vehicle fuel consumption, the vehicle energy consumption evaluation simulator developed by Argonne National Laboratory, Autonomie, is used [109]. The small SUV model with an ICE is used for energy evaluation. The configuration of a vehicle model considered for this analysis is summarized in Table 3.3.

Fuel economy of the original driving schedules are selected as the baselines. In Fig-
Table 3.3: Vehicle Model Configuration for Energy Consumption Evaluation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle total mass</td>
<td>kg</td>
<td>1745</td>
</tr>
<tr>
<td>Wheel radius</td>
<td>m</td>
<td>0.34</td>
</tr>
<tr>
<td>Frontal area</td>
<td>m²</td>
<td>2.841</td>
</tr>
<tr>
<td>Drag coefficient</td>
<td>-</td>
<td>0.356</td>
</tr>
<tr>
<td>Engine displacement</td>
<td>cc</td>
<td>2700</td>
</tr>
<tr>
<td>Number of cylinder</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>Engine max. torque</td>
<td>Nm</td>
<td>170</td>
</tr>
<tr>
<td>Engine max. power</td>
<td>kW</td>
<td>140</td>
</tr>
<tr>
<td>Final drive ratio</td>
<td>-</td>
<td>3.51</td>
</tr>
<tr>
<td>Transmission type</td>
<td>-</td>
<td>6-automatic</td>
</tr>
</tbody>
</table>

Table 3.3, vehicle fuel economy, miles per gallon (MPG), from the eco-ACC following the UDDS and the LA92 are compared with different number of V2V data from the preceding vehicles. The fuel economy of the baselines are presented in the blue and red dashed lines for the UDDS and the LA92, respectively. The solid lines with crosses present the fuel economy from the eco-ACC with the V2V-based speed predictor. We change the number of available V2V data from the preceding vehicles and assess fuel economy. As shown in Figure 3.8, up to five preceding vehicle data, positive correlations are observed between the number of the preceding vehicle data and MPG. Increased vehicle connectivity with more than five preceding vehicles does not provide significant additional improvement in the considered scenarios.

The relationship between the number of the preceding vehicle data and prediction quality is explained in Section 3.4.2; as the number of the preceding vehicle data increases up to five preceding vehicles, prediction accuracy can be improved. These observations imply that five preceding vehicles would be enough to maximize the performance of the eco-ACC as well as the prediction in our simulation setting. Further benefits are expected for longer prediction horizon with the increased number of preceding vehicle data. Moreover, we assume that all the vehicles drive with the same time headway which is 2 s. Thus, the maximum and the minimum numbers of preceding vehicle data that the eco-ACC system benefits from are varied by simulation set-ups.

The fuel economy improvements by the different prediction approaches are summarized in Table 3.4. Perfect prediction is the case that the future locations of the target vehicle over prediction horizons are exactly known. Thus, the perfect prediction shows the best fuel economy improvements that can be achieved by the eco-ACC system given in this chapter. The constant acceleration model is introduced in Section 3.4.2, which has the
Figure 3.8: Comparison of fuel economy from the UDDS and the LA92, IDM, and the eco-ACC while following each original cycle with different numbers of connected vehicles. Similar study with PHEV can be found in [4] (see Figure 8.5).

Table 3.4: Fuel Economy Comparison [MPG]

<table>
<thead>
<tr>
<th>Method</th>
<th>UDDS</th>
<th>LA92</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Drive Cycle</td>
<td>28.2</td>
<td>25.9</td>
</tr>
<tr>
<td>IDM-based Follower</td>
<td>28.5</td>
<td>26.0</td>
</tr>
<tr>
<td>Eco-ACC, Perfect Prediction</td>
<td>31.0</td>
<td>30.0</td>
</tr>
<tr>
<td>Eco-ACC, Constant Acceleration</td>
<td>27.5</td>
<td>-</td>
</tr>
<tr>
<td>Eco-ACC, V2V Prediction, 1 Preceding Vehicle</td>
<td>28.6</td>
<td>26.2</td>
</tr>
<tr>
<td>Eco-ACC, V2V Prediction, 5 Preceding Vehicles</td>
<td>30.2</td>
<td>29.3</td>
</tr>
</tbody>
</table>

worst prediction quality. As shown in Table 3.4, the eco-ACC with predictions from the constant acceleration model degrades fuel economy for the UDDS, and results in infeasibility for the LA92. By using the V2V-based speed predictor with five connected preceding vehicles, fuel economy is improved by 6 % (UDDS) and 13 % (LA92) compared to the IDM driver.

In order to consider a more realistic driving environment, the penetration rate of connected vehicles is regulated in simulation. In our simulation, the penetration rate of connected vehicles is varied as 20, 40, 60, 80, 100 % in the traffic of five preceding vehicles.

---

1The infeasibility issue is resolved by adding a slack variable in the hard constraint (3.6) and penalizing safe time headway tracking in the cost function. The formulation and results are presented in Chapter VI.
Figure 3.9: Average position prediction root-mean-squared errors (RMSE) and fuel economy improvement compared to the original UDDS and LA92 cycles.

For each penetration rate, different combinations of connected preceding vehicles are realized by permutation; for instance, there are five combinations of connected vehicles under 20% penetration rate with five vehicles. The simulation results from implementing all the scenarios are summarized in Figure 3.9. The left subplot shows the average RMSE of the target vehicle’s future position estimates computed by the predictor in Chapter II. The right subplot presents the ego vehicle’s average fuel economy relative to driving the original drive cycles. The figures clearly show that as the penetration rate of the connected vehicles increases, the prediction RMSE decreases, and consequently, the fuel economy of the ego vehicle is enhanced.

3.4.5 Comparison with Various Powertrain Types

This section validates our findings in the previous section by applying the eco-ACC to various vehicle powertrains with the same class: a mid-size ICEV, a BEV, and a power-split HEV model. Vehicle models are provided by Autonomie Express [112] developed by Argonne National Laboratory. Autonomie Express can evaluate vehicle energy consumption over given travel trajectories with high computation efficiency by parallel computing. The detailed information on vehicle configuration is summarized in Table 3.5. For the HEV model, the maximum and minimum SOC of the power-split HEV model are 90% and 40%, and the initial SOC is set to 50%.

Trip energy consumed by the eco-ACC is compared with the IDM-based follower in Figure 3.10 with the various powertrain types. Similar to Figure 3.8, the energy benefits rises as the number of connected vehicles increases and converges at around five regardless of powertrain types. However, if the number of connected preceding vehicles is limited to one, the energy consumption increases in the LA92 cycle for all the powertrain types. This
Table 3.5: Vehicle Model Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Unit</th>
<th>ICEV</th>
<th>EV</th>
<th>HEV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle total mass</td>
<td>kg</td>
<td>1730</td>
<td>1753.5</td>
<td>1669</td>
</tr>
<tr>
<td>Wheel radius</td>
<td>m</td>
<td>0.3014</td>
<td>0.3014</td>
<td>0.3014</td>
</tr>
<tr>
<td>Frontal area</td>
<td>m²</td>
<td>2.35</td>
<td>2.372</td>
<td>2.372</td>
</tr>
<tr>
<td>Drag coefficient</td>
<td>-</td>
<td>0.3</td>
<td>0.311</td>
<td>0.311</td>
</tr>
<tr>
<td>Engine displacement</td>
<td>cc</td>
<td>2655.6</td>
<td>-</td>
<td>1800</td>
</tr>
<tr>
<td>Number of cylinders</td>
<td>-</td>
<td>6</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>Engine max. torque</td>
<td>Nm</td>
<td>181.4</td>
<td>-</td>
<td>191.7</td>
</tr>
<tr>
<td>Engine max. power</td>
<td>kW</td>
<td>108</td>
<td>-</td>
<td>73</td>
</tr>
<tr>
<td>Motor max. torque</td>
<td>Nm</td>
<td>-</td>
<td>348.0</td>
<td>213.5</td>
</tr>
<tr>
<td>Motor max. power</td>
<td>kW</td>
<td>-</td>
<td>109.5</td>
<td>68.67</td>
</tr>
<tr>
<td>Number of cells in the battery</td>
<td>-</td>
<td>6</td>
<td>156</td>
<td>162</td>
</tr>
<tr>
<td>Energy of the battery in each cell</td>
<td>Wh</td>
<td>132</td>
<td>179.9</td>
<td>7.8</td>
</tr>
<tr>
<td>Battery nominal voltage</td>
<td>V</td>
<td>2</td>
<td>3.6</td>
<td>1.2</td>
</tr>
<tr>
<td>Final drive ratio</td>
<td>-</td>
<td>3.31</td>
<td>4.44</td>
<td>4.059</td>
</tr>
<tr>
<td>Transmission</td>
<td>-</td>
<td>5-automatic</td>
<td>2-manual</td>
<td>Planetary [111]</td>
</tr>
</tbody>
</table>

Figure 3.10: Energy consumption of the eco-ACC trajectories compared to the IDM over various powertrain types.

degradation in vehicle energy efficiency is caused due to the low prediction quality with the limited information for the speed prediction. On the other hand, when V2V information is sufficient, the eco-ACC improves the energy efficiency of the ICEV, BEV, and HEV. The maximum energy-saving from the IDM driving is 6.3-8.8% in the UDDS, and 8.9-14.2% in the LA92.

3.5 Summary

In this chapter, the efficacy of the short-horizon predictor developed in Chapter II is investigated by applying the predictor to the eco-ACC application and evaluating energy
benefits. Simulation results show that the eco-ACC systems using the proposed predictor can save energy consumption of the ego vehicle by 6% and 13% compared to the IDM follower when a sufficient amount of V2V information is given. On the contrary, in the face of inaccurate speed prediction of the target vehicle, the controller might perform worse than the IDM follower, and also, the worst prediction quality causes infeasibility in the optimizer. The next chapter extends our short-horizon predictor development by lengthening the horizon lengths up to 100 s. This mid-length horizon predictor is applied to different eco-driving control applications, which can expect higher energy saving by solving their optimal control problem for a longer prediction horizon.
CHAPTER IV

Forecasting Mid-Length Speed Trajectories of a Preceding Vehicle using V2V and V2I Information

4.1 Introduction

In Chapter II and III, the vehicle speed predictor using V2V information is developed and applied to the eco-ACC system, which requires short-term horizon for maximum energy saving. However, for some eco-driving control applications, a longer prediction horizon can further increase energy benefits. Therefore, this chapter proposes a novel vehicle speed predictor that can improve prediction accuracy over a short to mid-length horizon. This predictor uses a locally weighted polynomial regression (LWPR) algorithm, a data-driven prediction method that does not require expensive data collection for estimating the preceding vehicle’s future speed profiles. Moreover, this chapter proposes a prediction methodology using not only V2V information but also V2I information broadcast from traffic signs such as traffic lights. The predictor is applied to an eco-driving control system developed in [98], which optimizes the powertrain operations of BEVs.

The schematic overview of this method is shown in Figure 4.1. We call the prediction target the “target” vehicle (the red car in Figure 4.1), and the “ego” vehicle (the dark blue car in the figure) is the vehicle controlled by the eco-driving control algorithm. Other vehicles in front of the target vehicle are called “preceding vehicles” in this chapter.

The proposed method uses real-time vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) information as the prediction algorithm’s input. The algorithm is designed based on a locally weighted polynomial regression (LWPR). This method differs from the predictor developed in Chapter II, which uses regular weighted-least-square estimation and only V2V data for short-term speed forecasting. Using V2I information enables us to extend the prediction horizon length from a short to a medium range. The LWPR algorithm also allows us to predict a preceding vehicle’s speed trajectory with better accuracy at locally
The principle of the developed predictor is illustrated in Figure 4.2. First, the longitudinal speed and position of the target vehicle \( (v_{tgt}, s_{tgt}) \), and the preceding vehicles \( (v_{pv}, s_{pv}) \) are transmitted to the ego vehicle via V2V communication. Then, the estimated time gap between the target vehicle and each of the preceding vehicles is mapped to the time domain as the estimated time of arrival \( (ETA) \). Finally, the estimated speed at any one of the query points \( (t_q) \) is computed by interpolating the measurements based on the LWPR method. The rest of this section explains the details of the proposed predictor. First, the
speed prediction model is derived mathematically. Then, several design factors, including how to assign weights to training data and deal with V2I information, are discussed. The last part of this section explains our methodologies for exploiting V2I information by message type.

4.2.1 Prediction Algorithm

Speed prediction is implemented using the LWPR method. The main idea of LWPR is to assign higher weights to the training data points taken near a query point when training the polynomial coefficients [113]. Therefore, the prediction calculation should be repeated at every query point in a prediction horizon. The $n$th prediction in a prediction horizon can be calculated as follows:

$$\hat{v}(n) = x_q^{(n)} \beta^{(n)}$$ (4.1)

In the above equation, $\beta^{(n)}$ is the polynomial coefficient vector, and $x_q^{(n)}$ is the polynomial basis of a query point, $t_q^{(n)}$, defined as follows:

$$x_q^{(n)} = \left[ 1, t_q^{(n)}, \ldots, (t_q^{(n)})^m \right]$$ (4.2)

where $m$ indicates the order of the polynomial. In this chapter, quadratic polynomial fitting is used, that is $m = 2$. A query point, $t_q^{(n)}$, is one of the time steps in a prediction horizon. By solving a linear regression problem analytically, we can obtain the coefficient vector:

$$\beta^{(n)} = (X^T W^{(n)} X)^{-1} X^T W^{(n)} v.$$ (4.3)

In the solution, $X$ is a design matrix, $W^{(n)}$ is a weighting matrix for the $n$th query point, and $v$ is a training data vector, containing vehicles’ current speed measurements or zeros if the target vehicle is expected to stop due to traffic signs. The proposed methods of building the design matrix and the weighting matrix are elaborated upon in the rest of this section.

The interpolation fails when the time gap from the farthest preceding vehicles is shorter than the prediction horizon length (see after $\tilde{t}_{pv1}$ in Figure 2.1). In this case, the farthest preceding vehicle’s speed represent the speed of the rest of the prediction horizon.

4.2.2 Design Matrix

A design matrix, or a regressor matrix, is a matrix of values associated with the explanatory variables. A predicted speed trajectory is generated on a time domain, as illustrated in Figure 4.2. On the fitting domain, the origin is the current time. The training data are pro-
Figure 4.2: Illustration of the LWPR-based prediction method proposed in this chapter.

jected onto the time scale of the fitting domain. Here, the estimated time of arrival (ETA) at the preceding vehicles’ position or a traffic sign is introduced to compute time information for each measurement. The ETA calculation is an iterative program based on integrating the predicted speed calculated in (4.1). The ETA calculation process can be summarized as Algorithm 1.

Here, \( s_{pv,j} \) is the longitudinal position of the \( j \)th preceding vehicle “preceding vehicles” 1 to 4 in Figure 4.1), traffic lights or stops. \( s_{tgt} \) and \( v_{tgt} \) are the longitudinal position and speed of target vehicle (“target” in Figure 4.1), respectively. We assume that \( s_{pv} \) is obtained through V2V and V2I communication, and \( s_{tgt} \) is obtained by radar equipment on the “ego” vehicle. \( N \) is the number of training data (or fitting data points), \( N_{iter} \) is the number of iterations of the ETA calculation (can be defined by system designers) and \( N_p \) is the number of query points (or prediction steps) in a prediction horizon. \( T_s \) is the length of discretized time step in a predicted trajectory. Therefore, \( N_p \cdot T_s \) produces a prediction horizon length. \( \hat{v} \) is the prediction of the target vehicle’s speed. \( v_0 \) is set to 5 m/s, imposed to prevent irrationally large ETA values due to low \( v_{tgt} \).

The accuracy of the ETA is vital to accurately forecast the target vehicle’s speed profile, especially when traffic light timing is considered in the prediction. The target vehicle’s ETA at the traffic light determines whether it decelerates or passes through the light without stopping, which affects the ego vehicle’s driving strategy and, as a result, affects energy consumption as well. After ETA is computed, the design matrix is constructed as shown
Algorithm 1 ETA calculation

Initialize:  \( ETA_{(j,0)} \leftarrow \frac{(s_{pv,j} - s_{tgt})}{v_{max}(v_{tgt},v_0)}, \forall j \in \{1,...N\} \).

for \( 0 \leq k \leq N_{iter} - 1 \) do

Compute \( \hat{v}_k \) by LWPR with \( ETA_{(j,k)}, \forall j \in \{1,...N\} \).

Compute \( \hat{s}_k \) by integrating \( \hat{v}_k \).

for \( 1 \leq j \leq N \) do

if \( \hat{s}_k(Np) \geq s_{pv,j} \) then

Search \( t^*_j \) that minimizes: \( |s_{pv,j} - s_{tgt} - \frac{T_s}{2} \sum_{n=0}^{t^*_j} (\hat{v}(n) + \hat{v}(n+1))| \).

\( ETA_{(j,k+1)} \leftarrow t^*_j \)

else

\( ETA_{(j,k+1)} \leftarrow N_p \cdot T_s + \frac{(s_{pv,j} - \hat{s}_k(Np))}{v_{max}(\hat{v}(Np),v_0)} \)

end if

end for

end for

return \( ETA_j \leftarrow ETA_{(j,N_{iter})}, \forall j \in \{1,...N\} \).

below:

\[
X = \begin{bmatrix}
1 & ETA_1 & \cdots & ETA^m_1 \\
\vdots & \vdots & \ddots & \vdots \\
1 & ETA_N & \cdots & ETA^m_N
\end{bmatrix}
\] (4.4)

where \( m \) is the order of polynomial fitting, and \( N \) is the number of training data.

4.2.3 Weight Matrix

The LWPR algorithm indicates the importance of training data by assigning them different weights, which is generally defined as their distances from the query point. The distance between a query point and a training data point is determined by the difference between the predicted time (a query point) and the ETAs at the locations of V2V and V2I measurement sources.

One of the most common methods for determining value weight is using a Gaussian kernel. Thus, the weight given to the \( j^{th} \) training data for the \( n^{th} \) query point is determined by the squared distance between the training data point and a query point, projected in an
exponential space, as follows:

\[ w_j^{(n)} = \exp \left( -\frac{(t_q^{(n)} - ETA_j)^2}{2\tau^2} \right) \]  

(4.5)

where \( t_q^{(n)} \) is the \( n \)th query point, \( ETA_j \) is the ETA of the \( j \)th training data, and \( \tau \) is the kernel width. The kernel width is a hyperparameter that determines the width of the data window. A larger kernel width allows more training data to be utilized for predicting a query point. We chose the kernel width as 100, found by trial and error. A weighting matrix used for computing the \( n \)th query point in (4.2) is constructed as follows:

\[
W^{(n)} = \begin{bmatrix}
w_1^{(n)} & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & w_N^{(n)}
\end{bmatrix}.
\]  

(4.6)

4.2.4 Speed Forecasting with V2I Information

In this chapter, we consider that V2I information includes messages broadcast by roadside equipment such as traffic lights, stop signs, and speed limits. This section explains prediction policies for stop signs and traffic lights. Speed limit information is only used as the maximum constraint of the predicted speed profiles in this chapter. In this section, we notate the sources of the V2V/V2I measurements on the subscripts of ETA for clarification. For example, \( ETA_{PV} \), \( ETA_{stop} \), and \( ETA_{light} \) are the ETA at the preceding vehicle’s location, a stop sign, and a traffic light, respectively.

4.2.4.1 Stop Signs

A vehicle’s behavior when approaching a stop sign is to decelerate until the car stops. With this assumption, the stop sign is taken into account by adding training data as follows:

\[ v(ETA_{stop}) = v(ETA_{stop} + \Delta T_{stop}) = 0 \]  

(4.7)

where \( \Delta T_{stop} \) is the estimated stop duration at the stop sign. In this chapter, we assume \( \Delta T_{stop} = 1 \)s. The ETA at the stop sign is computed by the method introduced in the previous section. Instead of using the LWPR, the target vehicle’s speed at the stop sign is determined to be zero, as shown below:

\[ \dot{v}(ETA_{stop} + k) = 0, \text{ for } 0 \leq k \leq \Delta T_{stop}. \]  

(4.8)
Figure 4.3: The proposed predictor can distinguish six scenarios with V2I information. Three traffic light statuses are considered in this chapter. The six cases represent the combinations of the current status (on the left) of the next traffic light and its estimated upcoming status (on the right) at the arrival of the target vehicle.

To account for the delay at the stop sign, which is not covered by the ETA calculation, we add the estimated stop duration $\Delta T_{\text{stop}}$ to the ETAs at the preceding vehicles’ current locations as follows:

$$\text{ETA}_{\text{PV}} \leftarrow \text{ETA}_{\text{PV}} + \Delta T_{\text{stop}} \quad (4.9)$$

where $\Delta T_{\text{stop}}$ is the estimated stop duration of the target vehicle. Note that this ETA modification is made for the preceding vehicles which have already passed the stop sign.

### 4.2.4.2 Traffic Lights

In this chapter, three light statuses are considered: red, green, and yellow. To emulate realistic communication, we adopt the standards of DSRC. DSRC broadcasts traffic light information by compiling them into SPaT messages [8]. SPaT messages contain the following information:

1. Traffic light status (color) at the current time
2. Remaining time for the current light status

With all possible combinations of SPaT information, the speed predictor can expect a total of six scenarios, shown in Figure 4.3. We design prediction policies for each case. The predictor first determines which case the target vehicle will most likely be in by computing the ETA at the light position, $\text{ETA}_{\text{light}}$, and comparing it to the remaining time in the current light phase. Then, the prediction policy designated for that case is applied.
**Case 1: Red Light Stays Red**

In case 1, the light is currently red and stays red until the target vehicle arrives at the light. If the target vehicle is estimated to reach the light before the remaining time of the current red phase ends, that is, $\text{ETA}_\text{light} \leq \Delta T_{\text{red}}$, this case is activated. By using the method proposed in the section dedicated to stop signs, the ETA at preceding vehicles’ positions that are beyond the traffic light should be adjusted as follows:

$$\text{ETA}_\text{PV} \leftarrow \text{ETA}_\text{PV} + \Delta T_{\text{red}} + \Delta T_{\text{queue}}$$  \hspace{1cm} (4.10)

where $\Delta T_{\text{red}}$ is the remaining time in the current red phase, and $\Delta T_{\text{queue}}$ is the additional time delay due to the queue of preceding vehicles waiting at the red light. We assume this queue time to be 2 s multiplied by the number of preceding vehicles behind the traffic light.

First, measurements that represent the stop during the red-light phase are added to the training data:

$$v(\text{ETA}_\text{light}) = v(\text{ETA}_\text{light} + \Delta T_{\text{red}} + \Delta T_{\text{queue}}) = 0.$$  \hspace{1cm} (4.11)

The predictor produces zero speed for query points during the red-light phase:

$$\hat{v}(\text{ETA}_\text{light} + k) = 0, \text{ for } 0 \leq k \leq \Delta T_{\text{red}} + \Delta T_{\text{queue}}.$$  \hspace{1cm} (4.12)

**Case 2: Red Light Turns Green**

In the second case, the light is currently red and is estimated to turn green before the target vehicle arrives at the light’s location. Thus, the condition for case 2 is $\text{ETA}_\text{light} > \Delta T_{\text{red}}$. Since the green light lets traffic flow and does not explicitly dictate any changes to the target vehicle’s maneuvers, the predictor ignores the light and implements regression only with the speed measurements of the preceding vehicles.

**Case 3: Yellow Light Stays Yellow**

In case 3, the light status is currently yellow and is estimated to stay yellow until after the target vehicle passes through the light. The condition for case 3 is $\text{ETA}_\text{light} \leq \Delta T_{\text{yellow}}$. In practice, it is up to the driver to decide whether to pass through the light or brake for the coming red phase. We assume the target vehicle’s decision coincides with the estimation $\text{ETA}_\text{light} \leq \Delta T_{\text{yellow}}$, and it passes through the yellow light without slowing down. Hence, as in case 2, the light is ignored in the prediction.

**Case 4: Yellow Light Turns Red**

In case 4, the current yellow light is estimated to turn red when the target vehicle arrives at the traffic light location. This case is activated when it satisfies $\text{ETA}_\text{light} > \Delta T_{\text{yellow}}$. The policy in case 4 is as follows:
1. Ignore the measurements from preceding vehicles that have passed the traffic light.

2. Add training data: \( v(ETA_{light}) = 0. \)

3. Set the predicted speed to zero for the query points after the light is red: \( \hat{v}(ETA_{light} + k) = 0, \) for \( 0 \leq k \leq T_{horizon} - ETA_{light}. \)

**Case 5: Green Light Stays Green**

In case 5, the light status is currently green, and it stays green as the target vehicle arrives at the traffic light. The condition for this case is \( ETA_{light} \leq \Delta T_{green}. \) Since the green light does not impose any change on the target vehicle’s speed, no additional training data from V2I information are needed. The speed prediction is based only on V2V information about preceding vehicles’ speed.

**Case 6: Green Light Turns Yellow**

In the last case, the current green light status is estimated to change to yellow before the target vehicle arrives at the traffic light. This case is active at the condition \( ETA_{light} > \Delta T_{green}. \) Since a yellow light is treated the same as a green light according to our previous assumption, the predicted speed is computed as in case 5: Use preceding vehicles’ speed measurements as training data directly without any V2I-related training data.

4.3 Eco-Driving Control Strategy (PMP-Pwt)

The eco-driving control applied in this chapter is a simplified version of the controller introduced in Section 3.3 of [98], trimmed down to meet a stricter real-time requirement. The optimal control algorithm is deployed in a model predictive control framework, which follows a repetitive series of obtaining look-ahead information, optimizing the control sequence, and applying the first action.

The lower layer of the algorithm generates optimal state and control trajectories over the receding horizon for free-flow cruising (the absence of traffic lights and preceding vehicles) and follows the steps below:

1. Split the look-ahead horizon into constant grade and constant speed limit segments.

2. Adjust target steady-state speed, \( v^*_{ss}, \) based on a typical driver’s desired travel speed for each segment, according to grade and speed limit.

3. Calculate junction speed points between segments subject to the continuity conditions of optimal control theory (see Section IV.A in [114]).
4. Develop optimal solution trajectories of acceleration, deceleration, and constant speed on each segment, connecting the junction points and steady-state speed $v_{ss}^*$. 

5. Consolidate the solution trajectories of segments.

In order to consider constraints pertaining to the real-world driving environment, the algorithm is augmented with upper layers to override the above-derived target steady-state speed $v_{ss}^*$. When a vehicle approaches a traffic light, crossing time is optimized such that the vehicle passes the next green phase without stopping, resulting in a new estimate of target speed, $v_{ss,1}^*$. In the presence of a preceding vehicle, a target speed $v_{ss,2}^*$, as close as possible to the free-flow steady-state speed without running into a dangerous distance is estimated. The updated steady-state speed $v_{ss}^* = \min(v_{ss,1}^*, v_{ss,2}^*)$ is passed to the lower layer to generate an adjusted solution trajectory over the horizon.

### 4.4 Simulation Setup

The proposed prediction method and its efficacy in working with the eco-driving control are validated in simulation. The simulation is set up and run in RoadRunner, a simulation platform of multi-vehicle energy consumption and performance evaluation developed by Argonne National Laboratory [115]. This software automatically constructs a traffic environment by extracting information from real-world maps. Multiple CAVs connected to V2V and V2I communication channels can be simulated simultaneously in the selected closed-loop traffic scenarios. Energy consumption is evaluated by high-fidelity powertrain models which have been validated with chassis dynamometer test data [116, 117]. We use a BEV model in this chapter. Details on the configuration of the tested vehicle model can be found in [98].

#### 4.4.1 Traffic Scenarios

Two types of simulation designs are used for our analysis, as illustrated in Figure 4.4:

1. Type 1 uses map data provided by HERE, including route features such as road geometries, stop signs, traffic lights, and speed limits. The five front vehicles (see Figure 4.4) are driven by a human driver model [100], reacting to the road environment autonomously. Three routes are studied: one on the I-70 highway in Utah (“Highway”), another in an urban area in Chicago (“Urban”), and the third in a suburban area in Pittsburgh (“Suburban”).
2. Type 2 uses real driving data recorded by on-road experiments. The recorded drive cycle is followed by the first preceding vehicle (“Experiment”). The other four vehicles in front of the ego vehicle are controlled by the same human driver model used in Type 1. The real driving data are collected in a suburban area in Michigan and provided by Hyundai America Technical Center, Inc.

In both simulation designs, the last one in the vehicle string is the “ego” vehicle, which is an automated vehicle controlled by the optimal eco-driving strategy. The ego vehicle predicts the target vehicle’s future trajectories over prediction horizons and considers them as distance constraints when solving the eco-driving optimal control problem [98, 114]. The maximum V2V and V2I communication ranges are set to 250 meters and fixed over trips for simplification. We assume all the vehicles drive without lane changes throughout their trips. The drive cycles used for the simulations are plotted in Figure 4.5.

4.5 Simulation Results

This section presents results from the simulation implementation described in the previous section. The performance of the proposed prediction method is validated with the eco-driving controller described in Section 4.3. First, the prediction accuracy of the target vehicle’s speed is discussed, then the impact of the proposed prediction method on eco-driving control performances is shown. Finally, we discuss how the choice of parameters
affects the method’s performance.

4.5.1 Target Vehicle Speed Forecasting Results

The target vehicle’s future speed trajectories are estimated using the proposed predictor, and its accuracy is assessed. Two snapshots of speed prediction simulation are shown in Figure 4.6. The solid blue lines indicate the actual speed profiles of the target vehicle. The red lines are the predicted speed profiles at every time step, and the cyan lines represent the speed limits. In both subplots in Figure 4.6, the event of the target vehicle arriving at the stop sign is marked on the time domain with a red dotted line.

In Figure 4.6, the top subplot shows the prediction results using only V2V communication. In this case, the ego vehicle had no knowledge of the upcoming stop sign and just observed preceding vehicles’ driving. In contrast, in the bottom figure, the predictor used V2I information and anticipated that the target vehicle would decelerate to a full stop and wait for a certain time before leaving.

The accuracy of the proposed predictor is highly dependent on the number of training data allowed by the connectivity. Figure 4.7 clearly shows this phenomenon with snapshots of predictions with different horizon lengths. In both figures, the blue lines are the actual speed trajectories, the red lines are predicted speed profiles, and the black crosses are the speed measurements of preceding vehicles. The color of the circles presents the current and next light signal phases.

In Figure 4.7(a), the prediction horizon length is 10 s. The current light signal phase is green and changes to red at 19 s. The target vehicle’s ETA at the traffic light is 16
Figure 4.6: Example of speed prediction when the target vehicle approaches a stop sign.

s. In this example, the number of training data is sufficient to cover the short horizon. Figure 4.7(b) presents a prediction result over a 50 s horizon with the same number of training data. In this case, the number of the training data is insufficient to cover the whole horizon; therefore, the predictor failed to foresee the sharp deceleration and acceleration in the near future. After the last sampled measurement at 18 s, the predicted profile exhibited a constant speed in the absence of training data supporting the prediction.

Figure 4.8 displays the prediction accuracy at each horizon step. A root-mean-square error (RMSE) is selected as an accuracy metric. For all the driving conditions, the RMSEs increase as the prediction step goes farther, which is natural, considering the trait of time-series forecasting. It is noteworthy that the highway route results show a higher prediction accuracy compared to those from the urban and suburban routes. The main reason is that vehicles execute acceleration/deceleration maneuvers more often in urban and suburban roads than on highways in general; frequent variations in speed are more difficult to be predicted than a near-constant speed. The trace of the “Experiment” data backs to the suburban area, resulting in a similar accuracy with that of the suburban drive cycle of Type
Figure 4.7: Snapshots of speed prediction using both V2V and V2I information.

Figure 4.8: Speed prediction root-mean-squared errors (RMSE) from the proposed predictor over various driving cycles simulations.

4.5.2 Ego Vehicle Eco-Driving Control Results

In different driving conditions, the proposed prediction method is compared to a baseline to validate the proposed speed predictor’s impacts. The baseline speed predictor is based on an intelligent driver model (IDM), which describes the car-following behaviors of a human driver in a microscopic scale of traffic dynamics analysis [73]. More informa-
Figure 4.9: Energy consumption (top) and travel times (bottom) of the proposed method compared to the baseline.

The proposed prediction model and the IDM-based baseline run in separate simulation tests and feed the prediction of the target vehicle’s speed to the eco-driving control (see Section 4.3).

To assess the speed predictor’s performance in cooperation with the eco-driving optimal control, we investigate the resulting energy consumption and travel time, as shown in Figure 4.9. Three different prediction horizon lengths are tested: 10 s, 20 s, and 50 s. The results show that using the proposed speed predictor reduces energy consumption in most cycles, except for a negligible increase observed in the urban drive cycle with a 10 s horizon. The energy savings increase as the prediction horizon length becomes longer. As the predictor can look farther into the future, the eco-driving control is able to make better decisions to react to future constraints.

In most cases, the travel time increases, in general, due to the trade-off with energy savings. In some cases, however, travel time remains unchanged or is even reduced. For
Table 4.1: Summary of energy consumption and travel time with different prediction methods. The numbers in the parenthesis indicate relative values in percentage compared to the baseline.

<table>
<thead>
<tr>
<th>Type</th>
<th>Energy [kWh]</th>
<th>Travel Time [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDM-based predictor (baseline)</td>
<td>1.295</td>
<td>1051</td>
</tr>
<tr>
<td>Perfect preview</td>
<td>1.218 (−5.95 %)</td>
<td>1044 (−0.67 %)</td>
</tr>
<tr>
<td>Proposed predictor (fully connected)</td>
<td>1.219 (−5.87 %)</td>
<td>1037 (−1.33 %)</td>
</tr>
<tr>
<td>Proposed predictor (range = 250 m)</td>
<td>1.229 (−5.10 %)</td>
<td>1039 (−1.14 %)</td>
</tr>
</tbody>
</table>

Example, in the suburban cases, energy consumption decreases a little, while travel times remain the same. In addition, the simulation results of the urban and “Experiment” scenarios show energy savings, while at the same time, travel times decrease for the cases of 10 s and 20 s horizons. These results occur because the ego vehicle optimizes its maneuvers in a timely manner in reaction to the constraints imposed by the front vehicle and traffic regulating devices. The earlier reactions also enable the vehicle to avoid unnecessary speed changes, which leads to greater energy savings.

Table 4.1 compares the energy consumption and travel time of the ego vehicle using the baseline prediction, the “perfect preview,” the proposed prediction with fully connected V2I/V2V communication, and the proposed prediction with a limited 250 m communication range. Simulations are conducted with a 20 s prediction horizon and use the “Experiment” data set. For “perfect previews,” we assume the eco-driving control has the accurate a priori knowledge of the “target” vehicle’s movement. This setting is unrealistic but is intended to benchmark the impacts of prediction errors on optimization performances. The proposed predictor is analyzed for two cases: (1) fully connected V2V and V2I communications with no range limitation, and (2) the limited V2V and V2I communication range of 250 meters. The speed prediction RMSE of the proposed predictor with full communication is 4.15 m/s, while the prediction RMSE with the limited communication range is 4.84 m/s with a relative increase of 17%. The higher prediction accuracy in the fully connected case leads to extra energy saving with an even shorter travel time than the limited-range case, as shown in Table 4.1.
4.6 Summary

This chapter develops a novel vehicle speed predictor targeting mid-length horizons (<100 s) based on the LWPR algorithm. The predictor utilizes both V2V and V2I information for learning polynomial coefficients. The predicted trajectories are used to calculate position constraints on the PMP-Pwt controller and validated by SIL tests. Simulation results demonstrate that using the proposed mid-length horizon predictor in the PMP-Pwt controller reduces the energy consumption of the ego vehicle by up to 9.4% without a significant increase in travel time compared to the IDM-based speed predictor. In this chapter, the proposed predictor’s performance is evaluated with a limited number of connected vehicles, that is four. This number of connected vehicles may not be sufficient to address a long prediction horizon such as 100 s. In order to implement an extensive study on the potential energy benefits obtained by using the proposed predictor, in the next chapter, we implement eco-driving control simulations with extended V2V connectivity. Moreover, the performance reliability of the proposed predictor is assessed by implementing the predictor for a different eco-driving control formulation over various traffic conditions.
CHAPTER V

Extended Case Study on the Impacts of Vehicle Connectivity and Traffic Conditions for Eco-Driving Control and Prediction

5.1 Introduction

In the previous chapter, the vehicle speed predictor for mid-length prediction horizons is developed. In this chapter, more thorough investigation on the energy benefits from using the proposed predictor is conducted by implementing extensive scenario study. Prediction accuracy is affected by several factors, such as V2V connectivity range, which determines the number of connected vehicles, as well as by the heterogeneous car-following driving behaviors of connected vehicles. For example, in Chapter III, we see that the predictors using neighboring vehicles’ states as input features could be affected by the number of V2V-connected vehicles. In addition, low coherence between adjacent vehicles’ behaviors may influence prediction accuracy. To the best of our knowledge, no previous study systematically identifies such factors and investigate their impacts on both prediction accuracy and corresponding eco-driving behaviors.

On top of that, eco-driving control performance can also be affected by internal system parameters, such as prediction and control horizon lengths. The relationship between the horizon length and energy efficiency improved by eco-driving systems has been investigated by [104], [37] and [90], but the relationship is variable by different control formulations. This chapter identifies factors impacting the prediction accuracy and investigates the extent to which these impacts lead to energy savings of the eco-driving control system of our interest. Following hypotheses are formulated and confirmed through a systematic parameter study:

1. Heterogeneity in car-following maneuvers may affect the performance of predictors
Predefined drive cycles e.g. UDDS

Human-driven CVs

Automated target ego pv2

Predict . . .

Traffic Generation by Analytical Anticipative Drivability Model (A2ODM)

+ Time gap length
+ Heterogeneity

Real-time Prediction & Optimization

Eco-Driving Control

Target’s final position

Trajectory Forecasting

Automated

Predict

target

Optimal Commands

Position, Speed

Human-driven CVs

Position, Speed

V2V Connectivity

Figure 5.1: Overview of this chapter focusing on single-lane scenarios. The ego vehicle is the CAV controlled by the PMP-Spd controller introduced in Section 5.3; the target vehicle is the vehicle immediately in front of the ego vehicle and the prediction target of the ego vehicle, and the preceding vehicles are all the vehicles driving in front of the target vehicle.

using V2V information and consequently the energy saving from the eco-driving control.

2. A longer prediction horizon increases the vehicle energy efficiency achieved by our focused eco-driving control system.

3. Collecting V2V messages from more preceding vehicles enables the predictor to achieve higher accuracy over a longer prediction horizon, which improves the eco-driving control performance.

To prove the above hypotheses, the speed predictor developed in the previous chapter is integrated into a powertrain-agnostic eco-driving controller developed by Argonne National Laboratory [99] or the PMP-Spd controller introduced in Chapter 1.4.1 to validate the energy saving effect over various powertrain types. Two types of information are used as input to this predictor: (1) the preceding vehicles’ current speed and position delivered via the V2V basic safety messages, and (2) traffic signal location and timing obtained from the V2I SPaT messages. Since this study focuses on car-following scenarios, the V2I information is not considered. In addition, a sophisticated human driver model developed in
is used, which can produce reliable traffic scenarios with large number of vehicles, e.g. 100.

As Figure 5.1 shows, we generate numerous traffic conditions in which vehicles follow one another and the lead vehicle is driven by a pre-defined drive cycle, and we add heterogeneity by allowing each vehicle to maintain a different desired time gap. We then investigate prediction accuracy and the PMP-Spd controller’s driving behaviors varying the look-ahead horizon for various scenarios in which the number of connected vehicles is changed. Finally, our findings derived from the parameter study are validated through SIL testing.

This chapter is organized as follows: Section 5.2 presents the traffic generation methodology with a summary of the human-driven car-following model. Section 5.3 summarizes the eco-driving control algorithm adopted in this chapter. The eco-driving system is simulated with various combinations of the simulation parameters in Section 5.4, and this parameter study is extended in the SIL testing environment in Section 5.5. Finally, Section 7.5 presents our conclusions.

5.2 Heterogeneous Traffic Generation

Using A2ODM introduced in Section 5.2, we generate naturalistic single-lane car-following traffic scenarios where vehicles follow one another and the lead vehicle is driven by a pre-defined drive cycle. Assuming that vehicle time gaps are normally distributed, we use mean and standard deviation values to assign a time gap to each vehicle to produce different desired distance gaps. As an example, the trajectories of 100 vehicles are presented in Figure 5.2. In this example, the first preceding vehicle drives the UDDS cycle and the rest of the following vehicles are driven by A2ODM. The time gap between vehicles are generated by a normal distribution with a mean of 0.5 s and a standard deviation of 1 s. All A2ODM-driven vehicles maintain different distance gaps by following their assigned constant time gap parameters and arrive at the destination of the UDDS-cycle vehicle without any collisions. It can be seen that A2ODM can stabilize car-following behaviors, as vehicles in upstream traffic can reduce braking levels and avoid stops.

5.3 Eco-Driving Controller (PMP-Spd)

The PMP-Spd controller developed by Argonne National Laboratory [99] is used for our analyses in this chapter. In this chapter, we integrate the vehicle speed predictor into the PMP-Spd controller to calculate the car-following distance gap constraint based on
the predicted information. The PMP-Spd controller has a two-level approach to providing the reference states of the next step in a model predictive control fashion: (1) The free-flow speed planning level selects and uses one eco-mode (for example, the eco-approach mode), and (2) the car-following speed planning level maintains the desired distance gap between the ego vehicle and the preceding vehicle. The reference states of the next step are finalized by the ensemble module. Specifically, the car-following speed planning level uses the predicted final position and speed of the preceding vehicle at the end of the prediction horizon to optimize final position and speed for an eco-car-following problem. It then generates state-constrained trajectories by taking into account the predicted trajectory of the preceding vehicle. This analytical approach guarantees that the real-time computing capability of the speed-only eco-driving control algorithm does not suffer from the increase in the prediction horizon.

5.4 Parameter Study

This section presents a systematic parameter study that examines the impact of each parameter on prediction performance and eco-driving behaviors. We generate traffic using 100 vehicles on the UDDS cycle and the following parameters:

1. The mean ($\mu_\tau$) and standard deviation ($\sigma_\tau$) of a time gap parameter for the A2ODM
2. The length of prediction horizons (T)

3. The number of connected vehicles (N_{cv})

To measure the change in the resulting eco-driving behaviors, we use acceleration energy, which is a cost function for the PMP-Spd controller. Acceleration energy indicates the smoothness of trajectory: the smoother the trajectory, the lower the acceleration energy. It can be computed by the following:

\[
E_a = \frac{1}{2} \sum_{k=0}^{N_{trip}} a^2(k)
\]

where \(a\) is longitudinal acceleration (and deceleration) that the ego vehicle employs, and \(N_{trip}\) is the total number of time steps in a trip. In this chapter, four types of predictors for an ego vehicle are compared:

1. Accurate preview (“perfect”): the target vehicle’s accurate final position is given.

2. Constant speed (“CS”): the current speed of the target vehicle is propagated to compute the target vehicle’s final position.

3. Constant acceleration (“CA”): the current acceleration of the target vehicle is propagated to compute the target vehicle’s final position.

4. V2V predictor (“V2V”): the current status of all connected vehicles is used to calculate target vehicle’s future position based on the LWPR algorithm introduced in Chapter IV.

In this chapter, we assume that the vehicle wireless communication range is unlimited.

5.4.1 Mean and Standard Deviation of Time Gap

To analyze the impact of the A2ODM time gap parameters on the eco-driving control performance, the A2ODM traffic is implemented with three mean values of time gap—0.5, 1, and 1.5 s—given the characteristics of real-world driving according to [118]. In addition, the standard deviation is set to 0 s and 1 s for homogeneous and heterogeneous time gaps, respectively. Figure 5.3 shows the impact of the mean value of the time gap on speed trajectories in car-following scenarios. In this example, the prediction horizon length is set to 100 s. The results show that with a longer time gap, a vehicle maintains a larger desired gap from the preceding vehicle, which gives it sufficient distance to react to the preceding vehicle’s abrupt maneuvers (e.g., braking and stops). Consequently, having a longer time
Figure 5.3: Speed profiles of the lead vehicle (black solid line), 99 preceding vehicles (black dotted lines), and target vehicle (blue solid line) for the time gap mean values of 0.5 s (top) and 1.5 s (bottom).

Figure 5.4: Comparison of acceleration energy and corresponding final position accuracy resulted from different mean and standard deviation of the A2ODM time gap parameter.

gap results in smoother speed profiles for the preceding and target vehicles, generating milder traffic conditions.
As shown in Figure 5.4(a), acceleration energy generally decreases with the mean value of the time gap for all the ego vehicles, regardless of the prediction methods and the target vehicle (lines with the cross markers), except for the V2V ego vehicle with $N_{cv} = 100$. This V2V ego vehicle can achieve the minimum acceleration energy performance for all time gaps, meaning that neither its prediction accuracy nor its eco-driving behaviors are affected by the upcoming traffic conditions.

On the other hand, the V2V ego vehicle with $N_{cv} = 8$ lacks information, because the limited number of connected vehicles is not sufficient for interpolating the entire prediction horizon of 100 s. As explained in Chapter 4.2.1, when the V2V data is not sufficient to address the entire prediction horizon, the V2V predictor cannot implement interpolation. Where interpolation is not available, the farthest preceding vehicle’s speed is used as a constant model over the tail of the horizon. For this reason, the trend of its acceleration energy, depending on mean time gap, is consistent with that of the CS ego vehicle. The prediction accuracy is represented by the RMSE of target vehicle’s final position and compared in Figure 5.4(b). It is worth noting that a limited V2V ego vehicle still can reduce acceleration energy better than a CS ego vehicle due to its improved prediction accuracy. Its reduced scale increases with the mean time gap, as longer time gaps between eight connected vehicles enable elongation of the interpolation section using sparsely distributed V2V data over a prediction horizon.

From our analysis on the time gap parameter, we conclude that the standard deviation of the time gap parameter does not have significant impacts on the eco-driving control performance. On the contrary, the influence of the mean time gap on acceleration energy is weaker as the heterogeneity of the time gap parameter is added. This phenomenon happens because as time headway in traffic ranges from various values based on a normal distribution, the results from different time gap is mixed. For example, if a vehicle driving with a longer desired time gap is added in the traffic generated with a short mean time gap (e.g. 0.5 s) can smooth car-following trajectories, and as a result, enable ego vehicles to reduce acceleration energy.

5.4.2 Prediction Horizon Length

This section focuses on assessing the impact of various prediction horizon lengths—10, 20, 50, and 100 s—on eco-driving behaviors for homogeneous car-following scenarios with $\mu_\tau = 0.5$ s and $\sigma_\tau = 0$, where all vehicles are connected, i.e., $N_{cv} = 100$. Here, the lengths of the prediction horizon and control horizon are equivalent.

The acceleration energy results from using different prediction horizon lengths in the simulations are compared in Figure 5.5(a). The results show that all ego vehicles can
reduce acceleration energy compared to the target vehicle, regardless of the predictor type. Figure 5.6 shows speed and distance gap trajectories of the V2V ego vehicle with different prediction horizon lengths. The distance gap is more flexible when the prediction horizon length is 100 s. In contrast, in the case with a 10 s horizon, the ego vehicle closely follows the target vehicle. When a longer prediction horizon is applied, the controller can produce more satisfactory optimal solutions by looking further ahead. Hence, applying a longer prediction horizon results in a smoother speed trajectory.

5.4.3 Number of Connected Vehicles

This section analyzes the influence of the number of connected vehicles representing V2V connectivity—8, 20, 50, 80, and 100—on the performance of the V2V predictor with a 100-s prediction horizon length in the same homogeneous car-following scenarios described in the previous section. In Figure 5.7(a), the RMSE of the target vehicle’s final position estimates are plotted for varying numbers of connected vehicles. The results show that as the number of connected vehicles increases, the RMSE of the V2V predictor decreases, and when the number of the connected vehicles is eight, the V2V predictor’s RMSE is similar to the CS predictor’s. This result is due to the strategy the V2V predictor adopts: When the V2V information for interpolating the entire horizon is limited, the CS prediction is employed, using the current speed of the farthest connected preceding vehicle. The relationship between the V2V predictor’s performance and the number of connected preceding vehicles, seen in Figure 5.7(a), appears in the acceleration energy results as well.
(a) The V2V ego vehicle’s speed with different prediction horizon lengths.

(b) The distance gap trajectories from the target vehicle with different prediction horizon lengths.

Figure 5.6: Trajectories of the V2V ego vehicle with the various lengths of prediction horizons.

(a) Root-mean-squared errors (RMSE) of the target vehicle’s final position estimates from all types of the predictors.

(b) Acceleration energy from all types of the ego vehicles.

Figure 5.7: Comparison of prediction accuracy and acceleration energy using different numbers of connected vehicles.

as shown in Figure 5.7(b). The acceleration energy savings achieved by using the V2V predictor is significantly increased if more V2V messages from the preceding vehicles are available.
5.5 Validation based on Software-in-the-Loop Testing

5.5.1 Simulation Setting

We use RoadRunner [115], the CAV SIL testing software used in Chapter IV, to validate our findings from the parameter study by evaluating the energy consumption and performance of ego vehicles. Running RoadRunner with 100 vehicles in the scenarios of the parameter study demands a heavy computational load. To avoid this issue, we link the traffic generation part to the RoadRunner environment, in which only the ego and target vehicles are emulated. The trajectories that the target vehicle follows are provided by the traffic generation part. In addition, the V2V messages transmitted from the connected vehicles are generated by the traffic generation task and delivered to the ego vehicle driven in RoadRunner during simulations. We consider homogeneous car-following scenarios with a UDDS-driven lead vehicle and preceding vehicles, and employed perfect, V2V, and CS predictors for ego vehicles. In this chapter, the energy consumption of ego vehicles is evaluated using the same BEV model used in Chapter IV.

5.5.2 Results and Discussion

The SIL test results with various combinations of parameters are plotted in Figure 5.8. Battery energy consumption is reduced when a longer time gap is applied in the traffic generation (Figure 5.8(a)). This trend coincides with the trend in the acceleration energy presented in Figure 5.4(a). In addition, the target vehicle’s battery energy consumption decreases with the time gap due to the smoothed speed profile, which weakens the benefit of using V2V communication. Figure 5.8(b) and Figure 5.8(c) show battery energy consumption depending on the prediction horizon lengths and number of connected vehicles, respectively. These further battery energy savings are correlated to reduced acceleration.
energy, as shown in Figure 5.5(a) and Figure 5.7(b). The V2V predictor enables the ego vehicle to decrease its battery energy consumption and reach the minimum closer to the perfect predictor, as it elongates its prediction horizon while guaranteeing higher accuracy using richer V2V information. The 2% performance gap between the perfect and V2V predictors still exists and could be filled by future-intent sharing between connected vehicles. Note that when the number of connected vehicles is limited ($N_{cv} = 4$), the energy savings from using the V2V predictor is similar to that of using the CS predictor.

The impact of the parameters on prediction and control performance are displayed in Figure 5.9 and Figure 5.10, respectively. Three cases are studied: 1) short-term prediction with insufficient V2V information, 2) long-term prediction with insufficient V2V connectivity, and 3) long-term prediction with sufficient V2V information. In the first case, the ego vehicle’s speed profile is similar to the target vehicle’s trajectory (Figure 5.10(a)) due to the short-term prediction horizon (Figure 5.9(a)). The second case’s V2V prediction is similar to the CS prediction (Figure 5.9(b)); however, the long-term prediction horizon enables the ego vehicle to drive more smoothly than the short-term one (Figure 5.10(b)). In the last case, the prediction quality is significantly improved (Figure 5.9(c)) since the sufficient V2V information can address the long-term prediction horizon. The enhanced prediction accuracy also leads to the smoothest speed profile of the ego vehicle, as compared to the first and second cases (Figure 5.10(c)).
Figure 5.11: Energy consumption increase compared to the perfect preview case over various powertrain configurations. In the simulations, the time gap mean and standard deviation, horizon length, and the number of connected vehicles vary with the same parameter sets used in Section 5.4.

The performance of the different predictors is compared with various powertrain types. Similar to the analyses in Chapters 3.4.5, Autonomie Express [112] is used to evaluate vehicle energy consumption over the eco-driving control trajectories produced by the MATLAB-based simulation environment. The same ICEV, BEV, and power-split HEV models used in Chapters 3.4.5 are tested. Details on vehicle configuration can be found in Table 3.5. Simulations are implemented by using the various combinations of the parameters considered in this chapter (0.5 ≤ μτ ≤ 1.5, 10 ≤ T ≤ 100, and 8 ≤ Ncv ≤ 100) over various drive cycles including UDDS, US06, HWFET, WLTP, and LA92.

In Figure 5.11, the energy consumption increases compared to the perfect preview are compared over the ICEV, BEV, and power-split HEV powertrains and the different predictors. On each box, the central mark indicates the median of the results. The top and bottom edges of the boxes indicate the 75th and 25th percentiles, respectively. The whiskers extend to the most extreme data points not considered outliers. The outliers are shown as the plus sign markers. For all the powertrain types, the V2V predictor produces the least energy increases from the perfect preview, which shows the benchmark performance. These results prove that the use of V2V communication can significantly and robustly improve energy efficiency in various driving environment.

5.5.3 Comparison between Polynomial Regression and LWPR Predictors

This section compares the performance of the polynomial regression (PR)-based predictor developed in Chapter II, and the LWPR-based predictor developed in Chapter IV, or V2V predictor as named in this chapter. In this analysis, both predictors use V2V informa-
Figure 5.12: Comparison between the polynomial regression-based predictor and the LWPR-based predictor.

(a) Root-mean-squared errors of the target vehicle’s final position estimates.

(b) Prediction computation time

(c) Acceleration energy

(d) Electricity consumption in the BEV model
short computation time ($\leq 0.2$ ms) regardless of a horizon length.

5.6 Summary

This chapter investigates the energy-saving potential of the mid-length horizon predictor developed in Chapter IV through an extensive scenario study. The performance of the predictor and the PMP-Spd controller is assessed by varying (1) the mean and standard deviation of time gap parameters, (2) the number of connected vehicles, and (3) the prediction horizon length. Simulation results show that energy consumption can be decreased by 12% using the proposed predictor for forecasting 100 s horizon with 100 connected vehicles’ information in the A2ODM traffic generated with the UDDS cycle. This energy-saving performance is close to the benchmark using the accurate previews of the target vehicle’s future trajectories, with a performance gap of 2 %. Finally, we see the trade-off of the proposed predictor: it can produce accurate prediction results over a longer prediction horizon; however, the computational load increases linearly with the prediction horizon length. This issue is critical because it can be an obstacle to real-time implementation. On the other hand, the PR-based predictor can be helpful in real-world applications because it does not demand a high computational load regardless of prediction horizon lengths. However, the PR-based predictor’s accuracy is lower than the LWPR, especially for a longer horizon. The next chapter introduces one strategy to improve the PR-based predictor’s performance to take advantage of its high computational efficiency.
CHAPTER VI

Data-Driven Parameter Tuning Methodology for Vehicle Speed Forecasting based on Polynomial Regression

6.1 Introduction

While the complex data-driven methods can automatically learn their input weights through their backpropagation mechanism, no schemes are given for tuning input weights for the simple regression techniques such as polynomial regression developed in Chapter II. This chapter proposes a novel strategy for tuning input weights of the simple data-driven regression methods to enhance the eco-driving control performance. The overview of the chapter is illustrated in Figure 6.1. The proposed weighting method is applied to our polynomial regression-based predictor developed in Chapter II and tested with an eco-driving control application.

First, an exponentially-decaying function is developed as a input weighting function based on temporal correlations in real-world driving records. The decaying rates are defined as a “forgetting factor” for the past information and a “discount factor” for the future information. Those parameters are determined from two sets of real-world speed data: 1) to guide the forgetting factor, a set of driving data collected by the University of Michigan Transportation Research Institute as a part of the Integrated Vehicle-Based Safety Systems (RWD-UM) is used, and 2) to determine the discount factor, a set of data including three human-driven vehicles’ trajectories, closely following each other on the same lane during an entire trip in Ann Arbor (RWD-AA) is analyzed. The methodologies for these data analyses are illustrated in Figure 6.2.

To assess the benefits of the developed weighting method, a MPC-based eco-ACC problem is simulated, where various driving scenarios are considered. The predictor produces the immediate preceding vehicle’s speed trajectory, and the speed is integrated to obtain position, which is then fed into the eco-ACC problem. Two types of test data are used to
evaluate the performance of the new method compared to other rule-based or conventional approaches. Energy consumption is calculated for a battery electric powertrain by using the vehicle simulator Autonomie [119]. The results show that by applying the new weighting method, electricity consumption can be reduced by up to 4% compared to the regular least-squares estimation. The main contributions of this chapter are therefore as follows:

1. A input weighting strategy is developed for vehicle speed predictors that do not have dedicated optimization rules for the weights on input features.

2. To design a input weighting function, systematic data analyses are conducted to find temporal correlations in the human driving states from historical driving data.

3. The predictor developed in Chapter II is implemented with the new weighting method. Then, the predictor is simulated with an eco-ACC and compared with baseline predictors in terms of energy consumption and tracking capability.

4. The robustness of the predictor is demonstrated by testing it in a real-world traffic scenario and a simulated driving environment with various traffic conditions.

The rest of this chapter is organized as follows. Section 6.2 explains the proposed input weighting method. In Section 6.3, the eco-ACC problem implemented in this chapter is
Figure 6.2: Flowcharts of the processes to obtain the weighting parameters, the forgetting factor (top) and the discount factor (bottom), respectively. Detailed explanation of these processes are presented in Section 6.2.2 and 6.2.3.

described. Simulation settings and results are presented in Section 6.4. Finally, Section 7.5 summarizes the contributions of this chapter and introduces future work.

6.2 Tuning Input Weights using Real-World Driving Records

In this section, a novel methodology for determining weights on V2V and past measurements is proposed. The proposed method determines the input weights based on findings from data observations. Specifically, a weight on the $i^{th}$ measurement, $w_i$, is determined by the following equations:

$$ w_i = \begin{cases} 
\lambda (N_{\text{past}} - i + 1) T_s & \text{if the } i^{th} \text{ input is the past speed of the target vehicle}, \\
\gamma T_s^{<j>} & \text{if the } i^{th} \text{ input is the current speed of the preceding vehicle}. 
\end{cases} $$

(6.1)

Here, $\lambda$ is defined as the forgetting factor of the target vehicle’s past speed, $\gamma$ is the discount factor of the preceding vehicles’ current speed, and both parameters satisfy: $0 < \lambda \leq 1$ and $0 < \gamma \leq 1$. Hence, both weighting functions exponentially decay over time. This shape is designed based on our hypothesis about data: as the gaps of time or distance from the data sources become larger, the correlations between future speed and the fitting data would decay exponentially. The decaying rates are determined by two parameters: the
forgetting and the discount factors. The designs of the forgetting and discount factors are the key contributions of this chapter and each design process is introduced in the following section. In the following section, the real-world driving records used for this analysis are described.

### 6.2.1 Data

Two types of datasets are used in this chapter, and the detailed information on the datasets is given below:

**RWD-UM:** The RWD-UM dataset is collected and provided by the University of Michigan Transportation Research Institute (UMTRI) as a part of the Integrated Vehicle-Based Safety Systems (IVBSS) program [120]. In these datasets, human-driven vehicles’ driving states in various conditions including city and highway driving are recorded by their on-board sensors. The vehicle models used in the experiments were 2006 and 2007 Honda Accord EX four-door sedans. Preceding vehicles’ driving states are not available in these datasets. The example trajectories driven by a driver are plotted on the map in Figure 6.3. Longitudinal speed and acceleration information from 1668 trips driven by ten human drivers are utilized. These datasets are used for determining weights on the past measurements of the predicted vehicle in Section 6.2.2.

---

Figure 6.3: An example of routes traveled by one of the human drivers in the RWD-UM dataset.
Figure 6.4: The route driven by the three vehicles in the RWD-AA dataset. The map shows the Ann Arbor area in Michigan.

Figure 6.5: Speed profiles of the three vehicles in the RWD-AA data.

**RWD-AA**: The RWD-AA data is collected by Toyota Motor North America in Ann Arbor, Michigan. The data include three human-driven vehicles’ trajectories, closely following each other on the same lane during an entire trip. The route is plotted in Figure 6.4, and the speed profile of each vehicle is presented in Figure 6.5. The route includes both city and highway sections, and the total trip time is about 30 minutes. The vehicles used in the experiment are four-door sedans. The vehicles were equipped with DSRC transmitters and receivers. In the experiments, the vehicles were capable of communicating with each other. The speed and position trajectories transmitted in DSRC basic safety messages by the three vehicles are used. This dataset is used for tuning weights on the V2V mea-
surements. Details on this process are reported in Section 6.2.3. In addition, the data are utilized to reproduce the realistic traffic for our simulation introduced in Section 6.4. This platooning dataset is chosen for our analysis, because it has the sufficient length of trip for comparing energy consumption by human driving and eco-ACC.

It is noted that both RWD-UM and RWD-AA datasets were collected from the similar vehicle classes, which are four-door sedans. Thus, the influence of a vehicle class on driving styles is assumed to be negligible in our data analyses.

### 6.2.2 Information from Past Speed Measurements

For the past measurements, the temporal correlation of longitudinal acceleration is analyzed, because the predictor based on the polynomial regression estimates future speed based on the trend in the past speed measurements. The RWD-UM datasets are used to analyze the correlations of acceleration. To consider various driving styles, we use driving records collected from ten different human drivers. Each vehicle drove different routes, mainly in the North East and a few in the South East areas in the United States. The correlation is computed for high-speed and low-speed driving separately. To divide the high-speed and low-speed sections in trips, the speed criterion of 60 mile-per-hour (mph) is adopted from [121], which is one of the typical speed limits on highways in the United States.

The accelerations with different time gaps are presented by the scatter plots in Figure 6.6 and 6.7 for the low-speed and high-speed sections, respectively. In these results, the relationships between accelerations with different time gaps from 1 to 5 s are shown. Both figures show that the correlation between accelerations rapidly decreases as the time gap increases. The correlation coefficients ($\rho$) multiplied by the coefficients of determination ($R^2$) are plotted in Figure 6.8. The left subplot shows the product of the correlation coefficient and the coefficient of determination ($\rho \cdot R^2$) from the low-speed sections, and the right subplot presents the results from the high-speed parts. The lines with the dot markers are actual statistics from the ten drivers’ trips in the RWD-UM data, and the solid lines fit the statistics.

### 6.2.3 Look-ahead Information from V2V Measurements

In a typical traffic environment, as the locations of two vehicles become closer, their speed trajectories become more similar. Thus, for the speed measurements transmitted from the preceding vehicles, the correlations between two vehicles’ speeds are studied. Since our predictor maps the preceding vehicles’ speed onto the fitting domain based on ETA, we define “V2V errors,” which is a new concept introduced to define the temporal
Figure 6.6: Accelerations with different time gaps at low-speed sections in the real-world driving records traveled by the UMTRI driver 1.

Figure 6.7: Accelerations with different time gaps at high-speed sections in the real-world driving records traveled by the UMTRI driver 1.

Figure 6.8: Calculations between correlation and coefficient of determination of linear fit between accelerations with time differences. Ten human drivers’ driving records from the RWD-UM datasets are included in this analysis. The solid lines are the fitted curves of results from real data that are plotted in the lines with the dot markers.

correlation between two vehicle’s speed profiles with respect to ETA. The V2V error is defined as follows:

\[
\text{V2V Error} = v^FV(t + T_h) - v^LV(t)
\]  

(6.2)
Figure 6.9: Inverse standard deviations of the V2V errors with different time headway between preceding and following vehicles from the RWD-AA data. The data are used for designing the discount factor ($\gamma$).

Table 6.1: Summary of Prediction Parameter Tuning

<table>
<thead>
<tr>
<th>Type</th>
<th>Low speed ($&lt; 60$ mph)</th>
<th>High speed ($\geq 60$ mph)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forgetting factor ($\lambda$)</td>
<td>0.51</td>
<td>0.43</td>
</tr>
<tr>
<td>Discount factor ($\gamma$)</td>
<td>0.77</td>
<td>0.71</td>
</tr>
</tbody>
</table>

where $v_{FV}(t + T_h)$ is the speed of a following vehicle at time $t + T_h$ and $v_{LV}$ is the speed of a lead vehicle at time $t$. $T_h$ is the time headway from the lead vehicle to the following vehicle, which is used as ETA in our prediction process. The intuitive meaning of the V2V errors is the mismatch between the lead car’s current speed and the following car’s future speed when the following car arrives at the lead’s location, since time headway is used as the estimated time of arrival to the preceding car’s location in our predictor.

The V2V errors are produced from the RWD-AA data. The statistics of V2V errors are presented in Figure 6.9. The standard deviation of the V2V errors are calculated with the measurement samples within the time headway window of 0.5 s. In Figure 6.9, the blue asterisks show the inverse standard deviations of the V2V errors for every time headway window. The red curves fit the trend of the inverse standard deviations of V2V errors with different time headway.

The observation of the data in Figs. 6.8 and 6.9 points to an exponentially decaying function versus time as defined in (6.1). Hence, the forgetting and discount factors represent the decaying rates of the curves in Figure 6.8 and 6.9, and their values are summarized in Tab. 6.1.
6.3 Ecological Adaptive Cruise Control (QP-SC)

Since eco-ACC, typically formulated in a model predictive control scheme, relies on the future speed information, it is considered as a good example application to evaluate the performance of the proposed weighting method in speed forecasting. Thus, this section describes the formulation of an optimal control problem for the eco-ACC used in this chapter. The control formulation introduced in this chapter corresponds to the “QP-SC” type introduced in Chapter 1.4.1. The eco-ACC is realized by minimizing the control efforts, the surrogate optimization problem of energy minimization, by penalizing the square sum of longitudinal acceleration. This approach allows us to formulate the optimization problem as a quadratic programming. The optimal control problem is derived as follows:

\[
\begin{align}
\min_{u, \xi} & \quad J \\
\text{s.t.} & \quad s(k|t) \leq \hat{s}^{TV}(k|t) - v(k|t)T_h^* - L - d^* + \xi(t), \\
& \quad v_{\min} \leq v(k|t) \leq v_{\max}, \\
& \quad u_{\min} \leq u(k|t) \leq u_{\max}, \\
& \quad \xi(t) \geq 0, \\
& \quad s(k+1|t) = s(k|t) + v(k|t)T_s + 0.5u(k|t)T_s^2, \\
& \quad v(k+1|t) = v(k|t) + u(k|t)T_s,
\end{align}
\]

where the objective function of the eco-ACC is formulated as follows:

\[
J = \sum_{k=1}^{N_p} \phi_s(\hat{s}^{TV}(k|t) - s(k|t) - v(k|t)T_h^* - L - d^*)^2 \\
+ \sum_{k=1}^{N_p} \phi_v(v(k|t) - v^*(k|t))^2 \\
+ \sum_{k=0}^{N_p-1} \phi_u u(k|t)^2 + \phi_\xi \xi(t)^2.
\]

In (6.3) and (6.4), \(k|t\) represents that the variable is the \(k\)th-step prediction when the current time is \(t\). \(T_s\) is the sampling time and \(N_p\) is the number of steps in a prediction horizon. The variables \(s\) and \(v\) are the ego vehicle’s longitudinal position and speed, respectively. \(\hat{s}^{TV}\) is the target vehicle’s longitudinal position estimate predicted by the ego vehicle. \(d^*\) is a marginally safe distance between the lead and ego vehicles for safety, \(T_h^*\) is the desired time headway, and \(v^*\) is the desired speed, which is set to the speed limit. \(\xi(t)\) is a slack
variable imposed to soften the positional constraints to achieve recursive feasibility.

The objective function (6.4) is composed of four terms. The first term, multiplied by a penalty coefficient of $\phi_s$, ensures a safe distance gap from the target vehicle by following the desired time headway. The second term with a coefficient of $\phi_v$ recommends following the speed limit imposed on the road section. The third term, multiplied by a penalty coefficient of $\phi_u$, indicates that the acceleration should be minimized. The last term is a penalty on a slack variable for the safety.

The penalty coefficients, $\phi_s$, $\phi_u$, $\phi_v$, and $\phi_\xi$, are determined by the following logic. First, the penalty coefficient for the slack variable is determined as a relatively large value, $\phi_\xi = 100$, because the slack variable is introduced to address numerical feasibility issues from hard constraints; therefore, it is preferred to be minimized as much as is feasible due to safety concern. Then, the minimization of acceleration is scaled with that of the slack variable by considering the maximum magnitude of acceleration and the maximum of the slack variable: $\phi_u = \phi_\xi \cdot (d^*/u_{\text{max}})^2$. Here, the marginal distance gap $d^*$ is the desired maximum value of $\xi$. Similarly, the penalty coefficient on the desired speed tracking, $\phi_v$, is determined: $\phi_v(t) = \phi_u \cdot (u_{\text{max}}/v^*(t))^2$. $\phi_v$ is a time varying coefficient, since the speed limit changes on different road sections. Finally, the penalty coefficient on the desired headway tracking, $\phi_s$, is selected as $\phi_s \cdot (u_{\text{max}}/T^*_h v_{\text{max}})^2$. The target vehicle’s acceleration is estimated as follows:

$$\hat{a}^{TV}(k-1|t) = \frac{1}{T_s} (\hat{v}^{TV}(k|t) - \hat{v}^{TV}(k-1|t))$$

(6.5)

where $\hat{v}^{TV}$ is the future speed estimate of the target vehicle predicted by the ego vehicle.

<table>
<thead>
<tr>
<th>Type</th>
<th>Unit</th>
<th>Notation</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Car Length</td>
<td>m</td>
<td>$L$</td>
<td>4.5</td>
</tr>
<tr>
<td>Marginal distance gap</td>
<td>m</td>
<td>$d^*$</td>
<td>2</td>
</tr>
<tr>
<td>Desired headway</td>
<td>s</td>
<td>$T^*_h$</td>
<td>2</td>
</tr>
<tr>
<td>Max. acceleration</td>
<td>m/s$^2$</td>
<td>$u_{\text{max}}$</td>
<td>4</td>
</tr>
<tr>
<td>Min. acceleration</td>
<td>m/s$^2$</td>
<td>$u_{\text{min}}$</td>
<td>-4</td>
</tr>
<tr>
<td>Max. speed</td>
<td>m/s</td>
<td>$v_{\text{max}}$</td>
<td>40</td>
</tr>
<tr>
<td>Min. speed</td>
<td>m/s</td>
<td>$v_{\text{min}}$</td>
<td>0</td>
</tr>
</tbody>
</table>
Then, the target vehicle position is integrated as follows during one-step integration:

\[
\hat{s}^{TV}(k|t) = \hat{s}^{TV}(k-1|t) + T_s \hat{v}^{TV}(k-1|t) + 0.5T_s^2 \hat{a}^{TV}(k-1|t)
\]

The equality constraints (6.3f)-(6.3g) are derived from the vehicle dynamics with a point-mass system. More details on the MPC parameters are summarized in Tab. 6.2. The parameters are selected considering mid-size sedan vehicles. The marginal distance gap is determined to avoid collision, and the desired headway is determined considering safe time headway. The maximum and minimum acceleration and speed are determined considering vehicles’ physical limitations.

### 6.4 Simulation Study

To evaluate the effect of the proposed weighting method, it is applied to the polynomial regression-based vehicle speed predictor developed in Chapter II. Specifically, in the polynomial regression algorithm, the weight matrix \( W \) in (2.3) has its diagonal elements as (6.1) where the forgetting and discount factors are found in the previous section. The predictor produces the future speed profiles of the target vehicle to the eco-ACC system in the simulations. The eco-ACC system is tested in various driving environment by considering: a real-world driving scenario and artificially-generated traffic with various traffic conditions. This section provides detailed information including simulation set-up and baselines used for performance comparison and discussion of the results obtained in various driving situations.

#### 6.4.1 Simulation Set-up

Only single-lane traffic is considered, and following vehicles behind the ego vehicle are not taken into account. Two types of traffic simulation are implemented:

1. Real-world traffic: Three vehicles’ trajectories from the RWD-AA dataset are used.

2. Simulated traffic: The RDM introduced in Chapter 1.4.1 is adopted to generate the preceding and target vehicles’ trajectories.

The real-world traffic from the RWD-AA data provides a realistic traffic condition, in which vehicles on the same lane drive with speed variations. Moreover, the performance of the eco-driving can be directly compared with human driving in a car-following scenario. On
the other hand, the simulated traffic allows us to investigate the performance of the new prediction in various traffic environments by applying different drive cycles, different number of on-road vehicles, and different headway between vehicles. In this case, the V2V communication range is limited to 1 km considering typical range of C-V2X [122]. To consider various driving scenarios, five standard drive cycles are selected for analysis: the urban dynamometer driving schedule (UDDS), US06, the highway fuel economy test (HWFET), LA92, and the worldwide harmonized light vehicles test procedure (WLTP). Autonomie [119] is used to analyze the energy consumption of a vehicle. A midsize BEV model with 300-mile range and fixed gear transmission is chosen for the analyses. The details on the vehicle configuration are summarized in Table 6.3

Table 6.3: Vehicle Configuration Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Unit</th>
<th>EV-300mi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle total mass</td>
<td>kg</td>
<td>2619</td>
</tr>
<tr>
<td>Wheel radius</td>
<td>m</td>
<td>0.3014</td>
</tr>
<tr>
<td>Frontal area</td>
<td>m²</td>
<td>2.372</td>
</tr>
<tr>
<td>Drag coefficient</td>
<td>-</td>
<td>0.311</td>
</tr>
<tr>
<td>Motor max. torque</td>
<td>Nm</td>
<td>620</td>
</tr>
<tr>
<td>Motor max. power</td>
<td>kW</td>
<td>195</td>
</tr>
<tr>
<td>Number of cells in the battery</td>
<td>-</td>
<td>594</td>
</tr>
<tr>
<td>Energy of the battery in each cell</td>
<td>Wh</td>
<td>172</td>
</tr>
<tr>
<td>Battery nominal voltage</td>
<td>V</td>
<td>3.6</td>
</tr>
<tr>
<td>Final drive ratio</td>
<td>-</td>
<td>3.02</td>
</tr>
<tr>
<td>Transmission type</td>
<td>-</td>
<td>Fixed</td>
</tr>
</tbody>
</table>
6.4.2 Prediction Horizon Length Selection

To determine the effective length of the prediction horizon, the eco-ACC is simulated with different prediction horizon lengths. Particularly, perfect previews on the target vehicle’s speed over a prediction horizon are used in the eco-ACC. The eco-ACC performance is represented by the standard deviation of acceleration, since the objective function of the eco-ACC minimizes acceleration. In Figure 6.10, the eco-ACC performances are presented with different prediction lengths. The results show that as the length of horizon increases, the standard deviation of acceleration decreases and converges at around 20 s for most drive cycles. This observation is in agreement with the finding in [37], in which eco car-following is realized by acceleration minimization. Electricity consumption (EC) is also reduced as the prediction horizon length increases, but converges faster than the acceleration. Based on these results, we select 20 s for the prediction horizon length as a conservative choice. The same value is also used for the control horizon length.

6.4.3 Baselines

For a quantitative analysis of the performance of the new predictor, constant speed and acceleration models are implemented as baseline strategies. Moreover, the polynomial regression without input weighting is implemented to test the efficacy of the new input weighting method. The prediction methods compared are summarized as follows:

1. Constant speed (CS), which uses the current speed of the target vehicle over the prediction horizon.

2. Constant acceleration (CA), which propagates the current acceleration of the target vehicle to generate a speed trajectory over the prediction horizon.

3. Polynomial regression with the least-square estimation (LS).

4. Polynomial regression with the weighted-least-square estimation (WLS).

<table>
<thead>
<tr>
<th>Table 6.4: IDM Parameter Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDDS</td>
</tr>
<tr>
<td>( \bar{a} ) [m/s^2]</td>
</tr>
<tr>
<td>( \hat{v} ) [m/s]</td>
</tr>
</tbody>
</table>
Figure 6.11: Comparison between the trajectories driven by the real human driver in the Ann Arbor area (RWD-AA) and the eco-ACC with the predictor based on WLS.

Since the goal of the eco-ACC is to reduce energy consumption by automated driving, a human driver model is considered for performance comparison. The eco-ACC with the new predictor is compared with following trajectories of human drivers. For the real traffic simulation, the third vehicle of RWD-AA data is used as the baseline for the ego vehicle. For the simulated traffic, a baseline is generated by the IDM [73]. The IDM parameter values are presented in Tab. 6.4. Here, the parameter values are chosen to demonstrate general driving styles, i.e., not too aggressive or mild, by setting the maximum acceleration and speed to be similar to those of the original cycles. Among the IDM parameters, the length of a car is set to 4.5 m. The safe distance gaps $d_0$ and $d_1$ are set to 2 m and zero, respectively. A safe time headway chosen as 2 s, and the desired deceleration for passenger comfort, selected as 1.4 m/s$^2$. The acceleration exponent, $\delta$, is set to 4 [123].

6.4.4 Simulation Results with the Real-World Traffic Data

The three vehicles’ trajectories from the RWD-AA are used in this simulation. The first leading vehicle is a preceding vehicle as defined in Figure 2.1, which shares its driving states to the ego vehicle through V2V communication. The second vehicle is considered the target vehicle. The third vehicle is used as a human baseline for the eco-ACC. The trajectories of the human-driven third vehicle and the eco-ACC driven one are compared
Figure 6.12: Cumulative probability distributions of prediction errors from the entire prediction horizon (left) and the first prediction step (right).

Figure 6.13: Comparison of eco-ACC performance over the real traffic data (RWD-AA) with different prediction methods.

In Figure 6.11. In the subplots, the black lines represent the human baseline (the third vehicle’s trajectories), and the red lines indicate the eco-ACC trajectories. The eco-ACC’s speed smoothing effect by acceleration minimization is shown in the speed and acceleration trajectories given in the first and second subplots from the top. The third and the fourth subplots from the top present the tracking performance by showing the distance gap and the time headway between the target vehicle and the ego vehicle. Since the eco-ACC formulation penalizes tracking performance in the cost function, the headway is within a reasonable range around the desired headway of 2 s. The simulations are terminated based
on the predefined trip time. During the given time, the eco-ACC drives 7.9 m longer than the human driving. This demonstrates that the eco-ACC does not extend the trip time compared to the human driving.

The accuracy of the predictors is compared in Figure 6.12. The left subplot provides the cumulative probability distributions of prediction errors over an entire prediction horizon, while the right subplot displays those at the first prediction step. Since the sampling time of our simulation is 1 s, these errors result from 1-s ahead predictions. Although WLS and LS seem to have almost the same accuracy for the entire prediction horizon, WLS outperforms LS at the first-step accuracy. This observation shows that the new input weighting improves the prediction accuracy at the near-future steps while not degrading overall accuracy, unlike CA does. CA has the highest accuracy for the first step, since CA propagates the most current acceleration sample, which is highly correlated with the acceleration a second later as shown in Figure 6.6 and 6.7. However, this approach is not effective for addressing the rest of the horizon; CA results in the worst prediction performance compared to the other three methods.

In Figure 6.13, the simulation results of the eco-ACC performed with different predictors are compared. In addition, the results from using perfect previews of the target vehicle’s future speed are also compared as an ideal benchmark performance. These results are labeled as “Perfect” in the bar graphs. The left top subplot present the relative values of the standard deviation of longitudinal acceleration compared to the third human vehicle in RWD-AA data. Electricity consumption (EC) is compared in the right top subplot. The results show that the polynomial regression predictors outperform CS and CA in terms of acceleration minimization, and consequently electricity consumption saving.

The bottom subplots of Figure 6.13 present tracking capability: mean time headway (left) and the mean magnitude of slack variables (right), $\xi$, imposed on the position constraint in (6.3b), respectively. By penalizing the 2-s headway tracking in the objective function (6.4), the time headway from the eco-ACC does not exceed 5 % variation from the human following with almost 30 % electricity consumption reduction. Here, the average time headway of the human following vehicle is 1.99 s. Thus, the eco-ACC follows the target vehicle with the reasonable range of distance gaps. The magnitude of the slack variables indicates the safety performance by showing the level of soft constraint violation. CA and WLS outperform the other two predictors because CA and WLS have high accuracy for the first-step prediction, which is the most critical for safety. However, the performance gaps between WLS and LS are not significant for the other aspects overall. Also, the electricity consumption saving by both polynomial regression predictors is 10% less compared to the benchmark performance (“Perfect”). This is due to the limited availability of mea-
Figure 6.14: Comparison of root-mean-square errors (RMSE) of prediction accuracy by prediction steps. The shaded areas are results from the variation in traffic simulation parameters: the number of preceding vehicles from 1 to 10, and the time headway from 1 to 4 s. The lines with the diamond and triangle markers indicate the medians of simulation results with traffic variations. In RWD-AA, the number of preceding vehicles is 1 and the average time headway is 2 s.

measurements; the V2V messages from only one preceding vehicle are given in the RWD-AA data. Thus, to assess the performance more thoroughly, the new and the baseline predictors are evaluated under various simulated traffic conditions and the results are discussed in the next section.

6.4.5 Simulation Results under Various Simulated Traffic Conditions

To assess the accuracy of the predictors, RMSE at each prediction step are compared in Figure 6.14. In the subplots, the black lines with crosses are the results of the constant speed prediction, and the gray lines with circles are from the constant acceleration method. To evaluate the predictor’s performance in various traffic conditions, we vary the number of preceding vehicles from 1 to 10 and the time headway between the vehicles from 1 to 4 s. In Figure 6.14, the shaded areas present the range of RMSE with various traffic parameters when the least-square and the weighted-least-square estimation are used. The lines with diamond markers are the medians of RMSE from the polynomial regression using the least-square estimation. The lines with triangles correspond to the median of RMSE from polynomial regression with weighted least-square estimation.

For all standard drive cycles, WLS has the highest accuracy up to 15 s, while sacrificing a little accuracy at the end of the horizon. LS also outperforms CS and CA, except for the first few prediction steps. The worst accuracy of the polynomial regression predictors are almost the same. The worst accuracy corresponds to the case when only one preceding
Figure 6.15: Comparison of the eco-ACC performance with different speed prediction methods. The clouds of black dots show results from various traffic simulation parameters ($1 \leq N_{PV} \leq 10$ and $1 \leq T_h \leq 4$ s).

Figure 6.16: Relative values of electricity consumption by using the weighted-least-square method compared to the least-square method.

vehicle’s data is available and driving with 1-s time headway. In this case, most of the steps in a prediction horizon cannot be covered by V2V data points. According to our polynomial regression predictor’s logic, the rest of the prediction horizon steps is addressed by propagating the current speed of the target vehicle. Consequently, the worst accuracy of LS and WLS are almost the same with the accuracy of CS.

The performance of the eco-ACC using different predictors is compared in Figure 6.15. In the bar graphs, the green bars present simulation results using perfect previews on the target vehicle’s speed over a prediction horizon, which is considered as the ideal benchmark performance. The clouds of black dots on the bars of LS (blue) and WLS (red) plot the eco-ACC performance with various traffic parameters: the number of preceding vehicles from 1 to 10, and the time headway between every preceding car from 1 to 4 s.

The top left subplot presents the standard deviation of acceleration compared to that of
the IDM-driven trips. The results show that WLS achieves the largest improvement from the baseline except the perfect preview case. This improvement is reflected to electricity use as presented in the top right subplot. The simulation results show that WLS reduces electricity consumption the most among the predictors except the case using perfect previews, which is by up to 15% compared to the baseline.

One general concern for the eco-ACC is that it could sacrifice tracking performance. The bottom left subplot of Figure 6.15 provides the tracking performance in terms of mean time headway compared to the IDM baseline. In the graph, the time headway of the eco-ACC ranges from 2.3 to 2.7 s. Considering the recommended time headway for safety is generally considered as 2 s, the eco-ACC can maintain reasonable and safe time headway from the target vehicle.

The mean magnitudes of the slack variable, $\xi$, in the inequality constraint (6.3b) are compared in the bottom right subplot of Figure 6.15. Overall, CA produces the smallest slack variables except the perfect preview case, because CA has the least RMSE at the first prediction step as shown in Figure 6.14. Due to the same reason, WLS also results in small slack variables, especially compared to LS. Thus, WLS is the most effective prediction approach by minimizing electricity consumption most considerably among the other predictors while not significantly sacrificing tracking performance and maintaining the safety.

The electricity consumption from using LS and WLS is compared with various traffic conditions in Figure 6.16. The results show that using WLS can achieve up to 4.7% more energy saving than LS. The largest improvement is mostly made when the entire horizon is fully covered by the V2V data, where the number of vehicles is larger and the time headway is longer. Hence, obtaining sufficient number of measurements would be crucial to maximize the advantage of the new prediction method.
6.4.6 Comparison with Various Powertrains

In this section, the efficacy of the proposed weighting method is validated and compared with various powertrain types. Similar to the analysis in Chapter 3.4.5, Autonomie Express [112] is used to evaluate vehicle energy consumption with the resultant eco-ACC trajectories. The same vehicle models used in Chapter 3.4.5 is used: the mid-size ICEV, BEV, and power-split HEV. More information on vehicle configuration and parameters can be found in Table 3.5. In Figure 6.17, the energy consumption from the eco-ACC compared to the human driving over RWD-AA is plotted with different powertrain types. The uniform trends are shown in the results from all the powertrains, coinciding with the results in Figure 6.13.

In addition, the results under various simulated traffic conditions compared in Figure 6.18 with the same ICEV, BEV, and power-split HEV models. In each box plot, the red line indicates the mean of the data (grey circles), and the pink bands show the 95% confidence intervals for the mean. The purple bands present the standard deviations of the data. The results show that the WLS’s energy saving is higher than the LS in average, ranging from -0.5% to 4.6% regardless of powertrain types. Thus, the proposed weighting method shows robust and reliable performance over the various types of vehicle powertrain.
6.5 Summary

This chapter develops a novel input weight tuning strategy for vehicle speed predictors to improve energy saving in the eco-ACC system. The temporal correlations in human driving states are investigated by analyzing actual driving trajectories recorded from multiple drivers and DSRC basic safety messages. The forgetting factor and discount factor are defined to address the correlation trends and determine weights on the target vehicle’s past speed and the preceding vehicle’s current speed, respectively. The developed weighting method is applied to the short-horizon vehicle speed predictor developed in Chapter II, and validated in the real-world traffic scenarios. Moreover, the performance reliability is validated by simulating various driving environments. The efficacy of the predictor is tested by applying it to the eco-ACC application formulated as the QP-SC. The results reveal that using the proposed input weights reduces the energy consumption of a BEV by up to 4.7% compared to implementing the same predictor algorithm without the proposed input weights. This chapter studies an input layer design by tuning input weights based on the exponential kernel functions. The next chapter focuses on the output layer design of data-driven predictors by designing a loss function enhancing the optimization performance of the eco-ACC system.
CHAPTER VII

Loss Function Design of Data-Driven Vehicle Speed Predictors for Enhancing Energy Efficiency

7.1 Introduction

In the previous chapter, we observe that higher accuracy at prediction steps close to present leads to larger energy savings. From this observation, one hypothesis is deduced: uncertainties at certain prediction steps impact more on the energy efficiency of the ego vehicle. This chapter proves this hypothesis by investigating the influence of uncertainty at an individual prediction step. Moreover, we propose a novel predictor design strategy based on our findings that can improve the eco-ACC performance further by improving accuracy at the most influential prediction steps. To enhance the accuracy at certain prediction steps, we focus on designing the output layer of a data-driven predictor, to be specific, the loss function of data-driven predictors.

Data-driven predictors are trained by optimizing their objective functions defined at an output layer. This objective function is also termed as a loss function. Predictors producing regression outputs generally define their loss functions as averaged errors over a prediction horizon, such as a mean-squared error (MSE). Thus, it can be conjectured that using a loss function with weighed prediction errors may improve optimal control performance.

The flowchart of this work is illustrated in Figure 7.1. First, an eco-ACC system is implemented with randomly generated errors at only one of the prediction steps in a prediction horizon. Then, the influence of uncertainty at every prediction step on the eco-ACC is quantified by comparing energy consumption increment due to the uncertainty. Then, a new loss function is derived in the form of weighted-mean-squared error, where the weights are determined based on the quantified influence.

The efficacy of the proposed loss function is validated by applying it to three state-of-the-art data-driven predictors: polynomial regression, ANN, and LSTM. These example
Data selection for training and testing predictors by analyzing characteristics of real-world driving data. Predictors are developed to predict the future speed profiles of the immediately preceding vehicle. The training and test datasets are constructed by using standard drive cycles selected based on real-world driving data analysis. Each predictor is trained with the proposed loss function and two baseline loss functions, and the energy consumption is compared in the eco-ACC systems. Energy consumption is calculated by using a battery electric vehicle (BEV) model provided in Autonomie [119]. An intelligent driver model (IDM) [73] is employed to produce car-following behaviors of a human driver and analyze the energy saving of the eco-ACC compared to the human drivers. Simulation results show that using the proposed loss function saves more energy and performs more robustly toward the inexperienced trip than using the MSE loss function, most commonly used for data-driven regression in the state of the art. The standard drive cycles used in this chapter and their usages are summarized in Table 7.1.

The contributions of this chapter are summarized as following:

1. The influence of forecasting uncertainty on the energy-saving achieved by the eco-ACC is quantified.
2. The loss function of data-driven prediction techniques is designed to minimize the degradation in the eco-ACC performance caused by forecasting uncertainty.
3. Various data-driven predictors are trained with different loss functions and implemented in the eco-ACC system to validate the efficacy and robustness of the proposed loss function.

The rest of this chapter is organized as follows. First, given the eco-ACC system in Chapter VI, the impacts of forecasting uncertainty on energy consumption are explored,
Table 7.1: Usage of the drive cycles in this chapter.

<table>
<thead>
<tr>
<th>Usage</th>
<th>UDDS</th>
<th>US06</th>
<th>WLTP</th>
<th>LA92</th>
<th>SC03</th>
</tr>
</thead>
<tbody>
<tr>
<td>Influence Analysis</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Predictor Training (Noised)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Predictor Validation (Noised)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Predictor Test</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Eco-ACC Simulation</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

and the loss function is designed based on our finding in Section 7.2. Then, the three data-driven vehicle speed predictors are introduced in Section 7.3. The proposed loss function is then tested by simulation, with settings and results presented in Section 7.4. Finally, Section 7.5 summarizes the contributions of this chapter and suggests future work.

7.2 Designing the Loss Function of a Vehicle Speed Predictor to Improve Eco-ACC Energy Saving

A loss function is the optimization objective of data-driven predictors. In mathematical optimization, a loss function or cost function is a function that maps an event or values of one or more variables onto a real number, intuitively representing some “cost” associated with the event. An optimization problem seeks to minimize a loss function. In regression, a mean-squared error (MSE) is one of the most popular choices for a loss function, because MSE is globally continuous and differentiable, which are desirable features for most optimization algorithms such as gradient descent [124]. For this reason, we formulate a loss function based on a weighted-mean-squared error as follows:

$$L(\hat{v}(t)) = \sum_{k=1}^{N_p} w(k) (\hat{v}(k|t) - v(t + kT_s))^2$$

(7.1)

where $\hat{v}(t)$ is a vector of speed prediction over a prediction horizon generated at $t$, that is $\hat{v} = [\hat{v}(1|t), \ldots, \hat{v}(N_p|t)]^T$. $\hat{v}(k|t)$ indicates the speed prediction at the $k$th step computed at $t$, and $v(t + kT_s)$ is the ground-truth speed at $t + kT_s$. $w(k)$ represents a weight assigned to the $k$th-step prediction error. Therefore, the goal of this chapter is to find the weight $w(k)$ to reflect the importance of every prediction step to the eco-ACC system, and ultimately, enhance vehicle energy efficiency. As the first step, the influence of each prediction step is investigated to model the weight $w(k)$ in the following section.
7.2.1 Derivation of Artificial Forecasting Uncertainty

Time-series forecasting errors in target vehicle’s speed trajectories can be bounded by the physical limitations of vehicle speed and acceleration. The example uncertainty bounds of the target vehicle’s future speed trajectory are illustrated in Figure 7.2. In the figure, \( v(t) \) denotes the ground-truth speed of the target vehicle at the current time \( t \) and \( T_s \) is the length of the prediction step. \( a_{\text{max}} \) and \( b_{\text{max}} \) represent the maximum magnitudes of longitudinal acceleration and deceleration, respectively, imposed by the physical limitations of vehicles. \( v_{\text{max}} \) and \( v_{\text{min}} \) are the maximum and minimum vehicle speed, respectively. Based on this information, the potential future speed of the target vehicle can be ranged over the shaded area. From the bounds, the maximum magnitude of speed prediction errors at the \( k \)th prediction step is derived as follows:

\[
\max |\hat{v}(k|t) - v(t+kT_s)| = (a_{\text{max}} + b_{\text{max}})T_s k.
\]  

(7.2)

In this chapter, it is assumed that \( a_{\text{max}} \) and \( b_{\text{max}} \) are equivalent to 4 m/s\(^2\). This assumption makes the distributions of forecasting uncertainty symmetric with respect to zero. Hence, a truncated normal distribution with zero mean is used to produce random uncertainties, where the minimum and maximum values are 0 and 40 m/s, respectively. Based on the maximum error bounds in (7.2), the standard deviation of the \( k \)th prediction step is derived as follows:

\[
\sigma_v(k) = \frac{2a_{\text{max}} T_s k}{K_{\text{max}}}
\]  

(7.3)

where \( K_{\text{max}} \) is a scalar to represent the 0.01 and 99.99-percentiles of a normal distribution.
Figure 7.3: Electricity consumption from the eco-ACC trips with the artificial forecasting uncertainty at each prediction step are plotted compared to the eco-ACC with accurate previews. The unit of the y-axis is a percentage [%].

approximated as $K_{\text{max}} \approx 4$.

7.2.2 Eco-ACC Simulation with Forecasting Uncertainty

The forecasting uncertainty is randomly generated based on $\sigma_v(k)$ and added to one of the prediction steps during a eco-ACC simulation. The eco-ACC is formulated as QP-SC introduced in Chapter VI. Accurate speed information of the target vehicle is given for the rest of the prediction steps. The simulation is repeated fifty times for each prediction step to address the randomness of forecasting uncertainty fully throughout the prediction horizon.

In this chapter, the length of the control and prediction horizons is equivalent to 20s adopted from VI. Four standard drive cycles are used as the target vehicle’s speed profiles: the Urban Dynamometer Driving Schedule (UDDS), US06, LA92, and the Worldwide harmonized Light-duty vehicles Test Procedure (WLTP). The drive cycles are selected to consider various driving conditions, including highway (high speed with fewer stops), urban
(low speed with frequent stops), or a combination of both.

After the simulations, the trip energy is evaluated based on the speed profiles of the ego vehicle. Autonomie, a high-fidelity vehicle powertrain simulator developed by Argonne National Laboratory [109], is used. The midsize BEV model with a 300-mile range and fixed-gear transmission used in Chapter VI is selected for the analysis.

To evaluate the influence of the uncertainty, the eco-ACC system is also implemented with accurate information on the target vehicle’s speed trajectory over the same length of prediction horizons. Then, the electricity consumption from the forecasting uncertainty is compared with those from the accurate previews. In Figure 7.3, the box graphs present the electricity consumption resulting from the fifty eco-ACC simulations with the forecasting uncertainty over the UDDS drive cycle. The x-axis represents the prediction steps where the forecasting uncertainty is added. The y-axis displays the electricity consumption compared to the eco-ACC with the accurate target vehicle’s speed. Thus, the box graphs show the influence of the forecasting uncertainty at each prediction step on the energy-saving achieved by the eco-ACC.

An overall trend in Figure 7.3 shows that the influence of uncertainties on energy savings is considerably high in the prediction steps around 3-5s compared to other steps. This observation is counter-intuitive; since a model predictive controller applies only the first step in the optimal control command sequence, the uncertainty caused at the first prediction step is expected to be the most influential. One explanation is that larger uncertainties are added to the prediction steps at 3-5s than the earlier steps in our simulations, considering the characteristics of time-series forecasting.

Another noteworthy observation is that even though the prediction steps at the end of the prediction horizon (>10s) may possess the largest prediction errors, their influence is negligible compared to the earlier prediction steps (<10s). Our findings from these simulation results are utilized to determine the loss function weights \( w(k) \) in (7.1) as introduced in the next section.

### 7.2.3 Quantifying the Influence of Prediction Uncertainty on the Eco-ACC Energy Saving Performance

The loss function weight \( w(k) \) is determined based on the relative values of electricity consumption in Figure 7.3. First, to level the differences in the simulation results from using different drive cycles, the electricity consumption increments are scaled as following:

\[
\overline{EC}(k) = \frac{\Delta EC(k)}{\sum_{k=1}^{N_p} \Delta EC(k)}
\]  

(7.4)
where $\Delta EC(k)$ is the electricity consumption increment due to the forecasting uncertainty at the $k^{th}$ step. The scaled electricity consumption values, $\overline{EC}$, are plotted for the four drive cycles in Figure 7.4 (lines with dot markers).

The loss function weight, $w(k)$ in (7.1), is modeled as a continuous function to make the proposed loss function versatile to various time discretization of a prediction horizon. In addition, its value should be positive over all prediction steps. In order to satisfy these features, a probability density function (PDF) is utilized to model the loss function weight. We select concave PDFs including chi-squared, Rayleigh, Gamma, normal, noncentral F (NCF) and generalized extreme value (GEV) distributions. The results of fitting the $\overline{EC}$ data with the selected PDFs are presented in Figure 7.4.

The goodness-of-fit is assessed by evaluating the coefficient of determination ($R^2$). In Figure 7.5, $R^2$ and the number of parameters in a fitted function are compared for all the PDFs. The graph show that all the PDFs is able to fit the data with high $R^2 > 0.85$. The PDF of Rayleigh distribution has the least number of function parameters with that of chi-squared distribution, and it results in slightly higher $R^2$ than the chi-squared. To avoid overfitting with minimum number of parameters while gaining the high $R^2$, we select the PDF of Rayleigh distribution to model the loss function weights as follows:

$$w(kT_s) = \alpha \frac{kT_s}{\sigma^2} e^{(kT_s)^2/(2\sigma^2)}, \quad \text{for } 1 \leq k \leq N_p. \quad (7.5)$$
In the above equation, \( \alpha \) is a scalar coefficient that we impose and \( \sigma \) is the scale parameter of the Rayleigh distribution or the Rayleigh parameter. The parameters \( \alpha \) and \( \sigma \) are determined through an optimization problem of minimizing the MSE between the \( \overline{EC} \) data and the output from the model (7.5). This optimization problem is numerically solved by using the “fmincon” function in MATLAB. The resulting parameter values are: \( \alpha = 1.0814 \) and \( \sigma = 4.5485 \), with \( R^2 = 0.8874 \). The solid curve in Figure 7.4 shows the fitted function based on (7.5).

### 7.3 Example Vehicle Speed Predictors

To validate the efficacy of the loss function designed in the previous section, three data-driven predictors are considered as these are one of the most commonly used for time-series forecasting applications: polynomial regression, ANN, and LSTM. The predictors use the same information to predict the future speed trajectories of the target vehicle and produce the same types of output signals as described in Section 7.3.1. The architectures of the example predictors are explained in the rest of this section.

#### 7.3.1 Predictor Input Features and Outputs

To predict the target vehicle’s future speed trajectories, two types of on-road information are used as the input features of the predictors:

1. the past speed trajectory of the target vehicle,
2. the current speed of the preceding vehicles,

3. the current distance gaps between the target and the preceding vehicles.

The second and third types of information may be available via V2V basic safety messages \[?\]. It is assumed that all the vehicles transmit the basic safety messages, and the ego vehicle can successfully receive them with negligible delays. The first type of information can be obtained by either V2V communication or the sensors onboard the ego vehicle. In this chapter, we assume that the target vehicle’s current speed is always measurable by the ego vehicle’s radar. The length of past speed information is restricted to 10s adopted from the literature \[97\].

All the data types listed above can be directly used as input features for ANN and LSTM, because those techniques can infer relationships between input features and outputs by learning multiple layers. On the contrary, polynomial regression requires data processing, unless the relationship between inputs and outputs can be directly modeled as polynomial functions. In our polynomial regression predictor, the distance gaps from the preceding vehicles need to be converted to time information. More information on this data process and reasoning can be found in Chapter II. In summary, the distance gaps from the preceding vehicles are used for estimating the target vehicle’s time of arrival at their location.

The output vector of the predictors comprise the speed estimates of the target vehicle over a prediction horizon as follows:

\[
\hat{v}^{TV}(t) = [\hat{v}^{TV}(1|t), \hat{v}^{TV}(2|t), \cdots, \hat{v}^{TV}(N_p|t)]^T
\]  

(7.6)

where \(N_p\) is the number of prediction steps in a prediction horizon.

### 7.3.2 Polynomial Regression

The algorithm of the polynomial regression-based vehicle speed predictor is introduced in Chapter II. The most commonly used loss function for linear regression is a least-squared error. In this chapter, a weighted-least-squared error is considered as a loss function to apply the loss function weights designed in Section 7.2:

\[
L(\beta(t)) = ||W(t)(X^T\beta(t) - v(t))||_2^2
\]  

(7.7)

where \(W(t)\) is a weight matrix whose diagonal elements are the loss function weights. Details on the algorithm and the definitions of the variables can be found in Chapter II.
7.3.3 Artificial Neural Network

In time-series forecasting, ANN has been one of the well-known and powerful tools for estimating nonlinear trends [125]. Therefore, many previous studies have applied ANN for vehicle speed prediction [69, 96, 79]. It is therefore adopted as one of the representative state-of-the-art data-driven prediction approaches.

The ANN model is constructed with an input layer, an output layer, and hidden layers between the two layers, and the number of hidden layers is determined to retain high accuracy while minimizing overfitting risk. In Figure 7.6, the prediction RMSE from training and validation data are presented with the number of hidden layers from 1 to 10. In this analysis, the number of hidden units is 100 at each hidden layer and the loss function is MSE. The number of hidden layers is determined as three, because with this number both the validation and training RMSEs are reduced considerably, whereas more hidden layers cause overfitting, which can be inferred from the enlarged gap between the training and validation RMSEs. In conclusion, the ANN is constructed as a five-layer neural network with three hidden layers, and each layer is fully connected. The number of hidden units at each hidden layer is determined as 64, 32, and 24 by trial-and-error.

A rectified linear unit (ReLU) [126], one of the commonly used activation functions, is selected for the activation function of the hidden layers. The loss function is applied
Figure 7.7: Training and validation root-mean-squared errors (RMSE) with the various number of hidden units in the LSTM.

The input vector of ANN is composed of the past speed trajectory of the target vehicle and the current speed and relative location of the preceding vehicles:

\[
i_{ann}(t) = [v^{TV}(t - N_{past}T_s), \ldots, v^{TV}(t - T_s), v^{TV}(t), v^{<1>} (t), v^{<2>} (t), \ldots, v^{<N_{PV}>} (t), \Delta s^{<1>} (t), \Delta s^{<2>} (t), \ldots, \Delta s^{<N_{PV}>} (t)]^T
\]

where \(\Delta s^{<j>} (t)\) is the distance gap between the \(j^{th}\) preceding vehicle and the target vehicle at the current time \(t\).

7.3.4 Long Short-Term Memory Network

A recurrent neural network (RNN) is a type of deep learning network, commonly used to recognize characteristics in sequential data and predict the following scenarios. A LSTM network is a variant of RNNs, invented to deal with the vanishing gradient problem of
traditional RNNs and enable the retention of long-term dependencies in sequential data [128]. Thus, LSTM networks are helpful for forecasting time-series data, especially when the duration between information critical for prediction is obscure.

We develop a LSTM-based predictor with three layers: an input layer, an output layer, and an LSTM layer between them. All the layers are fully connected. The number of hidden units in the LSTM layer is determined by varying the number of the hidden units from 10 to 100 and analyzing the training and validation RMSEs as plotted in Figure 7.7. These results are produced by using only the US06 cycle to reduce the training time. Therefore, the RMSE shown in Figure 7.7 presents a pseudo-performance of the LSTM network trained by using the entire training dataset. Here, the number of hidden units is chosen to be 100, as the validation accuracy converges at around 100.

The state activation function of the LSTM layer is the hyperbolic tangent function, and the gate activation function is the sigmoid function. The LSTM network is trained by using the Adam optimizer. The loss function is defined at the output layer. Finally, the LSTM network is trained by the MATLAB built-in function “trainNetwork” in the Deep Learning Toolbox.

The input features of the LSTM network are constructed as follows:

$$i_{lstm}(t) = [v^{TV}(t), v^{<1>}(t), v^{<2>}(t), \ldots, v^{<N_{PV}>}(t), \Delta s^{<1>}(t), \Delta s^{<2>}(t), \ldots, \Delta s^{<N_{PV}>}(t)]^T.$$  \tag{7.9}$$

Unlike the input features of the polynomial regression and ANN, the input features of the LSTM predictor do not include the past speed measurements of the target vehicle because the LSTM network memorizes past input features in its cell states.

### 7.3.5 Training and Validation Data

In this chapter, datasets for training, validating, and testing the ANN and LSTM predictors are generated by simulating car-following traffic. The RDM introduced in Chapter 1.4.1 is used to generate the trajectories of the preceding and target vehicle with a time gap of 2s. The speed profiles are defined based on standard drive cycles selected by the following process.

First, real-world driving data of human drivers are used for extracting human-driven vehicles’ driving characteristics. The real-world driving dataset was collected by the University of Michigan Transportation Research Institute (UMTRI) as a part of the Integrated Vehicle-Based Safety Systems (IVBSS) program [120]. This dataset contains driving states collected from vehicles driven by ten human drivers in various conditions, including city
and highway driving, using their onboard sensors. The vehicle models used in the experiments were 2006 and 2007 Honda Accord EX four-door sedans. Preceding vehicles’ driving states are not available in these datasets.

As the representative characteristics of human driving, the mean of speed and acceleration magnitudes, $|a|$, are computed for each trip record. The results of this analysis are plotted in Figure 7.8 in crosses. Then, we select standard drive cycles that can address real-world driving characteristics well. The selected drive cycles are US06, UDDS, LA92, and WLTP. The mean of speed and acceleration magnitudes of these drive cycles are plotted also in Figure 7.8. In the plot, the widths and heights of the ellipses indicate the standard deviations of speed and acceleration magnitudes, respectively. The scatter plot shows that the selected standard drive cycles can cover the driving statistics of the real-world driving data. The selected standard drive cycles are used as training data instead of the real-world driving data to save training time by using a smaller training dataset. Lastly, the SC03 drive cycle allocated near the training driving data is selected as a test drive cycle, which validates the efficacy of our method in Section 7.4.

![Figure 7.8: Mean of speed and acceleration magnitudes of the real-world driving dataset and the standard drive cycles.](image-url)
To improve the robustness of the ANN and avoid overfitting, random noises are added to the datasets based on standard normal distribution [129]. The generated dataset is randomly split into 85% and 15% for training and validation, respectively. The example speed profiles included in the training dataset are plotted in Figure 7.9. The original drive cycle is US06 as shown in the solid line, and the random noises are added in the training data as presented in the dashed line.

7.4 Validation

7.4.1 Simulation Settings

To test the prediction and energy saving performance, simulations are implemented. In the simulations, the number of preceding vehicles are 10 and all the preceding and target vehicles’ trajectories are generated by the RDM. The ego vehicle receives V2V messages transmitted from the preceding and target vehicles with the maximum V2V communication range of 1 km [122]. Furthermore, noises are added to the input features to consider the realistic characteristics of measurements. The noises are generated based on a zero-mean normal distribution with the standard deviation of 0.03 m/s and 2.7 m for speed and position, respectively, considering the typical accuracy of standalone global navigation satellite system receivers for automobile applications [130, 131].

Five standard drive cycles are used to generate the preceding and target vehicles’ trajectories: US06, UDDS, LA92, WLTP, and SC03. Here, the former four standard drive
cycles are similar to the training datasets, because the training datasets are generated by adding random noises in the same standard drive cycles. The last cycle, SC03, is used to test if the proposed method could still be effective even for the test drive cycles not similar to the training datasets. This analysis could enable us to understand the efficacy of the proposed loss function for inexperienced trips. The preceding and target vehicles drive the same standard drive cycles with a 2-s time gap in the simulations to eliminate the effect of car-following dynamics and analyze the effect of the eco-ACC system only. In addition, a distance gap of 2 m is assumed between each vehicle at the beginning of the trip.

### 7.4.2 Tested Loss Functions

Two baseline loss functions are developed and compared with the loss function developed in Section 7.2:

- **MSE loss function**: A mean-squared-error (MSE), whose $w(k) = 1$ for all $k$ in (7.1).
- **Exponential loss function**: An exponentially weighted MSE whose $w_k = \gamma^k$ in (7.1). The value of $\gamma$ is $\gamma = 0.77$ for high speed (> 26 m/s), $\gamma = 0.71$ for low speed (< 26 m/s) sections, adopted from Chapter VI.
- **Rayleigh loss function**: The proposed weighted MSE, where its weights are determined based on the function (7.5).

The first two loss functions are used as the baselines. The MSE loss function is the most common form of loss function for regression in the state-of-the-art deep learning field. The exponential loss function is tested to check whether the Rayleigh loss function outperforms the loss function that simply weights near-future prediction steps greater than the far future. Because our eco-ACC system is implemented in a receding horizon manner, the weights exponentially decrease as the prediction step is far from the future. The differences in the
7.4.3 Prediction Results

This section summarizes the prediction results from the eco-ACC simulations. The vehicle speed predictors introduced in Section 7.3 are trained by the different types of loss functions explained in the previous section.

Prediction accuracy from the three predictors trained by the different loss functions are compared in Figure 7.11. Each subplot shows prediction mean absolute errors (MAE) at each prediction step produced by simulating (a) the polynomial regression, (b) the ANN, and (c) the LSTM network over the SC03 drive cycle. For all predictors, the predictions with the exponential and Rayleigh loss functions have significantly smaller errors at the earlier prediction steps (<10s) than the end of the horizon, whereas the MSE loss function results in more even accuracy over the prediction steps. These results are reasonable, because the exponential loss function and the Rayleigh loss function weigh errors at the near-current steps more significantly than the far-future steps. The differences between
Figure 7.13: Snapshots of prediction results over the US06 cycle using the ANN predictor trained by the MSE loss function and the Rayleigh loss function.

The exponential and the Rayleigh loss function are subtle, but overall, the Rayleigh loss function has smaller MAEs in the middle of the prediction horizon (3-10s). The same performance trends are observed with other drive cycles.

This difference in prediction accuracy is investigated further by comparing the cumulative probability distributions of absolute prediction errors in Figure 7.12. The prediction errors are produced by simulating the polynomial regression over the US06 drive cycle. The distributions show that the MSE loss function’s prediction accuracy is greater at 1s and 5s than the exponential and Rayleigh loss functions. This trend is opposite at 20s as presented in Figure 7.12(c). In addition, the Rayleigh loss function results in higher accuracy than the exponential loss function at 5s.

In Figure 7.13, prediction trajectories are plotted as snapshots to compare actual prediction results from the different loss functions. In this figure, the dashed lines are the true speed profile of the target vehicle, and the solid line segments are the predicted trajectories of 20s produced by the ANN at every sampling time (1s). In the results with the Rayleigh loss function (b), the predictions at the end of the prediction horizon are more diverted from the true trajectory compared to when using the MSE loss function. On the contrary, the beginning of the prediction horizon is more accurate than the MSE loss function as Figure 7.12(a) shows. The same trends are found in the simulation results with the other predictors. Finally, these differences in prediction performance lead to different eco-ACC results. These control results are described in the following section.
7.4.4 Control Results

The longitudinal speed and acceleration trajectories of the target vehicle and the following ego vehicle controlled by the eco-ACC are plotted in Figure 7.14. The penalty coefficients in the eco-ACC cost function are kept the same in all the simulations. Here, the eco-ACC results are produced using the polynomial regression trained by the MSE loss function (dashed lines) and the Rayleigh loss function (dotted dashed lines). The speed profiles from using the MSE and the Rayleigh loss functions are smoother than the target vehicle’s speed, since the cost function of our eco-ACC minimizes acceleration. This effect is well seen in the acceleration trajectories. In addition, it is clearly shown that the Rayleigh loss function smooths the speed and acceleration profiles significantly more than the MSE loss function.

The acceleration minimization results are compared with the different loss functions in Figure 7.15. Each bar graph compares the standard deviation of the ego vehicle’s longitudinal acceleration with the MSE loss function. The bars represent the results with the

Figure 7.14: Trajectories of the target vehicle and the eco-ACC using the MSE and the Rayleigh loss functions.
Figure 7.15: Standard deviations of longitudinal acceleration from using the exponential and Rayleigh loss functions compared to the MSE loss function. The polynomial regression (left), ANN (middle), and the LSTM (right) predictors are tested.

The polynomial regression and the Rayleigh loss function. The Rayleigh loss function results in the smallest standard deviations of acceleration for all drive cycles compared to the other two loss functions. Using the Rayleigh loss function reduces the standard deviations of acceleration by 5% on average, and 11.6% in maximum compared to the MSE loss function, and by 2.7% in average, and 5.1% in maximum compared to the exponential loss function.

The reduced acceleration demands lead to energy benefits in the eco-ACC systems. Here, the energy consumption of the ego vehicle is evaluated by using the same BEV model used in Section 7.2. The impacts of the loss functions on electricity consumption (EC) are compared in Figure 7.16. Each box graph compares electricity consumed by the ego vehicle to the human driver model. In order to generate human car-following trajectories, the IDM is used in this analysis. The IDM parameters are adopted from Chapter VI. In the results, the Rayleigh loss function saves electricity the most compared to the baseline exponential loss function, except the case with polynomial regression over the US06 drive cycle. In this case, the exponential loss function outperforms the Rayleigh loss function; however, the difference is minimal. Using the MSE loss function, the eco-ACC saves electricity consumption by an average of 10%. However, applying the Rayleigh loss function increases the average saving to 12%. This improvement is more significant than the exponential loss function with 11% saving on average.

The same conclusion is valid with the inexperienced trip; the SC03 cycle is not used to generate the training dataset. Similar to Figure 7.15, the standard deviations of the ego vehicle’s longitudinal acceleration from using the exponential and Rayleigh loss functions are compared with that of the MSE loss function in Figure 7.17(a). The benefit of the Rayleigh loss function still holds with the inexperienced trip for all the predictors, and the numbers are consistent with the experienced trips shown in Figure 7.15. In addition, the Rayleigh loss function results in a more robust performance than the exponential loss function in that the Rayleigh loss function outperforms the MSE loss function in all cases,
Figure 7.16: Electricity consumption from using the exponential and Rayleigh loss functions compared to the MSE loss function. The polynomial regression (left), ANN (middle), and LSTM (right) predictors are tested with three different loss functions.

Figure 7.17: Comparison of acceleration and electricity consumption with the inexperienced trip (SC03).

whereas the exponential loss function occasionally results in larger acceleration than the MSE loss function with the LSTM.

Finally, the electricity consumption is compared in Figure 7.17(b). Similar to the results in Figure 7.16, the bar graphs present the electricity consumption compared to that of IDM. The IDM parameters for the SC03 cycle are adopted from Chapter VI, except the maximum acceleration and speed as 2.3 m/s$^2$ and 24.5 m/s, respectively. For all predictors, the Rayleigh loss function saves trip energy the most, from 1.5 to 2% more than the MSE loss function.

These results are noteworthy in that the eco-ACC systems can achieve extra energy saving by simply adjusting the weights of a loss function, which does not demand hardware modification or add complexity to the control algorithms.
7.4.5 Tests with Various Powertrain Configuration

In this section, the proposed loss function design strategy is applied to different powertrain types and assesses its efficacy is analyzed. The proposed weight function (7.5) is tuned based on the influence of the forecasting uncertainty on the electricity consumption in the BEV. Therefore, in order to apply the proposed weight function to different powertrain types, we need to confirm that the influence of the uncertainty on the energy saving is shown the same with different powertrain types.

In order to confirm this, the influence of forecasting uncertainty on acceleration is assessed because our eco-ACC minimizes longitudinal acceleration to reduce energy consumption. In Figure 7.18, the increases of acceleration in the eco-ACC trips after adding the forecasting uncertainty are presented in a box plot for every prediction step. The y-axis shows the increase of the standard deviation of acceleration compared to the eco-ACC results with perfect previews ($\Delta \sigma_u$). The simulation results show that the trends of acceleration in Figure 7.18 are highly similar to the one of electricity consumption in Figure 7.3.

![Figure 7.18](image)

Figure 7.18: Increase of the standard deviations of acceleration demanded by the eco-ACC with the artificial forecasting uncertainty compared to the eco-ACC with accurate previews.
Similar to the analysis in Section 7.2.3, the increase of the standard deviation of acceleration is scaled to obtain:

$$\sigma_u(k) = \frac{\Delta \sigma_u(k)}{\sum_{k=1}^{N_p} \Delta \sigma_u(k)}$$

(7.10)

where $\Delta \sigma_u(k)$ is the increment of $\sigma_u(k)$ due to the forecasting uncertainty at the $k^{th}$ step. Then, the goodness-of-fit of the weight function in (7.5) is tested over the $\sigma_u$ data obtained from the UDDS, US06, WLTP, and LA92 cycles. The coefficient of determination ($R^2$) is selected as the goodness-of-fit metric, which results in $R^2 = 0.8622$. Considering the goodness-of-fit test results with electricity consumption in Figure 7.5, this $R^2$ value is sufficiently high; therefore, the developed weight function is sufficiently accurate to address the influence of the uncertainty on the acceleration as well as the electricity consumption.

Finally, the travel trajectories driven by the eco-ACC with the different predictors trained with the MSE, exponential, and Rayleigh loss function are fed into the vehicle models to evaluate energy consumption. As we analyze in Chapters 3.4.5 and 6.4.6, the energy consumption of the vehicles is evaluated by Autonomie Express [112]. The mid-size class ICEV, BEV, and power-split HEV models are tested for analysis. The information on vehicle configurations and model parameters is presented in Table 3.5.

In Figure 7.19, the energy consumption from the eco-ACC compared to the IDM is plotted for different powertrain types. The gray circles represent the results from using the three predictors (PR, ANN, and LSTM) over the five drive cycles (UDDS, US06, LA92, WLTP, and SC03). The red lines in the middle of boxes indicate the mean of the results, and the pink bands represent the 95% confidence interval of the means. The purple bands show the standard deviations of the data. The results show that the use of the Rayleigh loss function can contribute to reduce energy consumption significantly compared to the MSE.
and exponential loss functions for all the powertrain types. From these results, it can be concluded that the proposed loss function is beneficial regardless of powertrain types and can deliver reliable performance.

7.5 Summary

This chapter proposes a novel strategy for designing the loss function of data-driven vehicle speed predictors to maximize energy saving in the eco-ACC system. The loss function is proposed as a weighted-mean-squared error to address the different influences of the prediction steps on the optimization results in the eco-ACC. To investigate the influence, artificial forecasting uncertainty is generated considering vehicle speed time-series forecasting characteristics. Next, the uncertainty is added to each prediction step in multiple eco-ACC simulations, and the electricity consumption demanded by the eco-ACC is compared. The weights on the loss function are modeled by the PDF of Rayleigh distribution, which can describe the trend of the influence produced in the simulation results. The Rayleigh loss function is validated in the eco-ACC system by comparing the control results with the different formulations of loss functions. The loss functions are used for training various data-driven speed predictors including polynomial regression, an ANN, and an LSTM network. The datasets for training and testing the predictors are constructed based on the five standard drive cycles. The selected cycles can cover the statistics of speed and acceleration of human drivers’ trips collected by experiments. Simulations results show that using the Rayleigh loss function, the magnitude of acceleration demanded by the eco-ACC is reduced by an average of 5%, which leads to additional energy saving in the BEV by 3% more than the MSE loss function commonly used in state of the art. This result is remarkable, because it shows that the eco-ACC system with the data-driven predictor can increase energy saving by 3% further by simply modifying a loss function.
CHAPTER VIII

Conclusions and Future Work

8.1 Results and Conclusions

With the recent advances in vehicle communication and automation technologies, eco-driving control has been one of the promising technologies to enhance the energy efficiency of vehicles. Forecasting a preceding vehicle’s trajectory plays a crucial role in energy-optimal speed planning in realistic traffic environments. The majority of prior studies focus on the accuracy of trajectory prediction by developing complex learning-based prediction algorithms. This dissertation focuses on developing energy-optimal forecasting strategies for eco-driving control systems. The organization of this dissertation and key takeaways from each topic are summarized in Figure 1.4.

First, in Chapter II and IV, the novel data-driven speed predictors are developed, targeting short to mid-length prediction horizons (10-100s). The developed predictors do not demand expensive data cost and time-consuming training processes by using polynomial regression and locally weighted polynomial regression. Although the predictors use simple linear-regression-based algorithms, they produce competitive prediction accuracy by leveraging V2V and V2I information. Moreover, the proposed predictor designs have high availability even when the vehicle has lost V2V signals.

The predicted trajectories are used to calculate position constraints for various eco-driving control formulations based on optimal control theory. The performance of the predictors and controllers is assessed in various traffic and network conditions in the simulations. In addition, the robustness of the eco-ACC system is also tested across an ICE, a BEV, and a HEV model. Based on our findings, the eco-driving control systems can significantly benefit from vehicle connectivity, for example, 6% with the short horizon and 12% with the mid-length horizon in the UDDS traffic with sufficient vehicle communication information.
This dissertation also proposes a methodology for tuning input weights for the predictor with a single layer, such as the polynomial regression. First, human driving records are collected in the real world and analyzed to identify the temporal correlations in driving maneuvers. In addition, V2V messages are collected by the experiment with DSRC equipment and analyzed to compare the correlations between the preceding vehicle's V2V information and the following vehicle's future trajectories. Then, the correlations found in two types of datasets are modeled based on exponential functions. The forgetting and discount factors are defined to address the correlation models and determine input weights accordingly. This approach enables the predictor to prioritize information obtained from different timing and sources, and therefore, improve prediction quality. Finally, the developed weighting method is applied to the eco-ACC system and implemented for validation in the real-world traffic scenario. Simulations realize various driving conditions to confirm the performance reliability of the proposed method. Simulation results show that the proposed weighting method improves energy consumption by average 1% more than applying no input weighting over various powertrain types.

Lastly, the novel loss function of the data-driven vehicle speed predictor is derived in this dissertation, optimally designed for the eco-ACC system. The proposed loss function is formulated as a weighted-mean-squared error. The weights are determined to address the different influence of prediction steps on the optimization results in the eco-ACC. The influence of uncertainty at different prediction steps is quantified by implementing the eco-ACC system with uncertainty at a single prediction step. The weights on the loss function are modeled based on the quantified influence. The proposed loss function is used for training various data-driven predictors based on: polynomial regression, ANN, and an LSTM network. Then, the predictors are integrated into the eco-ACC system to validate the efficacy of the loss function. Simulation results show that energy saving from using the proposed loss function in the eco-ACC is added by 3% to the MSE loss function with the identical eco-ACC system. This result is remarkable: the eco-ACC system with all data-driven predictor tested can increase energy saving further by simply modifying a loss function, which does not require additional costs or hardware modifications.

8.2 Future Work

This dissertation proposes strategies to predict a preceding vehicle's future trajectory, optimally designed for eco-driving control systems. The work presented in this dissertation can be extended by considering further validation and applications. Moreover, several limitations and assumptions mentioned throughout the dissertation are summarized in this
section followed by some future research directions.

- **Experimental validation:**

  The next step would be validation of the results on a vehicle. The prediction and control algorithms developed here are easily implementable on a vehicle electronic control unit (ECU). However, real-time computation has to be addressed, especially for locally weighted polynomial regression, which demands longer computation time with longer prediction horizons.

- **Realistic driving environment:**

  Our work can be extended by considering a more realistic driving environment. For example, if road grade is addressed in our eco-ACC problems, speed forecasting errors could add uncertainty to road load preview by misleading the feasible set of the ego vehicle. It will be interesting to analyze how this uncertainty impacts the eco-driving control system and vehicle energy efficiency.

  Other potential uncertainties critical to our eco-ACC systems include V2V signal delay and measurement errors. In our work, we assume the current states of the target and preceding vehicles are accurate and consider them as ground-truth. However, in the real-world, V2V message packets are delivered with delay, and their information possesses errors due to sensor noise and bias. The realistic communication and sensor specification in predictor and controller development can be considered in future work to improve the robustness of the eco-ACC systems. For instance, if the current speed of the target vehicle has uncertainty due to signal delay or sensor noise, the uncertainty bounds derived in Figure 7.2 would be expanded as a trapezoidal shape instead of a triangle.

  In addition, fixed IDM parameter values are considered in our simulations. However, the driving styles of IDM are dependent on the IDM parameter values such as maximum acceleration. Therefore, in order to consider the various driving styles of human drivers, the simulations need to be repeated with various IDM parameter values so that the sensitivity of the prediction and control performance can be assessed.

  Furthermore, in Chapter IV, an assumption is applied for the simulated vehicles: a uniform decision is made by all the vehicles under a yellow light. However, this assumption is not realistic in real-world traffic, especially with human drivers. Therefore, understanding different driving styles and corresponding behaviors under yellow lights would be interesting. Moreover, the predictor can be modified to reflect the identified driving styles.
• Extending applications:

The reproduction of Chapter VII with various eco-driving applications would be interesting. For instance, the influence of forecasting uncertainty can be investigated for different types of control formulations and tuned the loss function weights accordingly. The efficacy of the proposed method to powertrain-specific eco-driving controllers for various powertrain types would be worth studying.

In addition, this dissertation only considers deterministic prediction approaches. It would be interesting to consider how to adjust the proposed prediction strategies to stochastic prediction approaches such as hidden Markov models.

• Comparison with intention sharing:

Another direction for future research would be comparing the proposed trajectory forecasting methods with vehicles’ intention sharing via V2V communication. Different types of intention sharing can be considered depending on vehicle automation levels: (1) sharing the subsequent powertrain operations or navigation information for non-automated vehicles, and (2) sharing a planned trajectory that an automated vehicle is tracking. Energy-saving from using either service can be compared by applying them to the same eco-driving control systems.

• Risk-sensitive control designs:

Finally, developing a risk-sensitive controller would be valuable for road safety and energy benefits, considering potential high forecasting uncertainty. Conventional robust MPC may degrade energy efficiency by imposing excessively conservative constraints. On the contrary, risk-sensitive MPC can mitigate the conservative constraints by assessing risk metrics during implementation, and consequently, result in greater energy efficiency than the robust MPC [132]. Moreover, probabilistic prediction approaches can be considered in future work in order to obtain a prediction error covariance in real-time and feed it to the risk-sensitive controller.
BIBLIOGRAPHY


