



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사 학위논문

Robust Hierarchical Motion
Planning and Control
for Automated Bus

자율주행 버스를 위한
강건한 계층적 거동 계획 및 제어

2022년 8월

서울대학교 대학원

기계항공공학부

조 아 라

자율주행 버스를 위한 강건한 계층적 거동 계획 및 제어

Robust Hierarchical Motion Planning and Control
for Automated Bus

지도교수 이 경 수

이 논문을 공학박사 학위논문으로 제출함

2022년 4월

서울대학교 대학원

기계항공공학부

조 아 라

조아라의 공학박사 학위논문을 인준함

2022년 6월

위원장 : 송 한 호 (인)

부위원장 : 이 경 수 (인)

위 원 : 김 아 영 (인)

위 원 : 김 시 호 (인)

위 원 : 오 광 석 (인)

Abstract

Robust Hierarchical Motion Planning and Control for Automated Bus

Ara Jo

Department of Mechanical and Aerospace Engineering

The Graduate School

Seoul National University

This dissertation presents novel, robust hierarchical motion planning and motion control algorithm for an automated bus in urban bus-only lanes. The proposed motion planning and control algorithm aims to achieve three goals: (1) enhance ride comfort and safety considering a limited field of view (FOV) and detection range, (2) cope with localization uncertainty, and (3) secure robustness against environmental disturbances and model uncertainties.

Although the FOV and detection range of an automated bus are limited due to on-board sensors' performance and complex road structures, ride comfort and safety for the automated bus should be secured in terms of public transport requirements. Ensuring ride comfort and safety with limited FOV and detection range was accomplished using the proposed longitudinal motion control algorithm. The algorithm was constructed as a hierarchical framework with two components: (1) long time-horizon longitudinal motion planning and (2) short time-horizon longitudinal motion planning. The long time-horizon planning was designed for optimizing the reference motion with limited FOV and detection range. The optimization problem was formulated using simple kinematics and dynamic programming (DP) to cover the limited FOV and detection range. The short time-horizon motion planning was constructed for ride comfort and safety by replicating

human driving patterns. The risk index-based weights were decided from the fuzzy inference system considering newly suggested rules and parameters, which were inferred from the analysis of human driving data in urban bus-only lanes. The optimization problem was formulated using vehicle dynamics and chance-constrained, Fuzzy-tuned model predictive control (MPC). The MPC was designed to minimize control effort and jerk while driving an automated bus and avoiding collisions within the speed limits. For improving safety, the uncertainties from localization and perception modules were additionally considered by designing the chance constraint for travel distance. Consequently, the longitudinal motion planning algorithm could secure ride comfort and safety with limited FOV and detection range. In addition, it was necessary to investigate the suggested approach in comparison to a deep reinforcement learning (DRL)-based algorithm in terms of the recent research trends related to this topic. After the DRL-based longitudinal motion planning algorithm was hierarchically constructed, the performance was compared to that of the algorithm using DP. Since the comparison attempted to review the suggested approach using DP, simulations under representative test conditions were planned to investigate advantages and disadvantages in terms of characteristics of the solution and computational cost.

Urban driving environments include underpasses and overpasses to separate bus-only lanes from roadways and high buildings along the lanes. They cause shading in the global positioning system (GPS) satellite signals for the automated bus. The effect of a GPS-denied environment on the lateral clearance in a lane is critical when a bus is driving along a narrow road. To prevent the potential dangers in such a GPS-denied environment, it was necessary to suggest an approach that should cope with the localization uncertainty. The localization uncertainty was treated in both the longitudinal and lateral motion planning algorithms since the uncertainty could cause heading, lateral position, and longitudinal position errors with respect to the reference path. In detail, the longitudinal position error was considered as the chance constraint in the longitudinal motion planning algorithm. The effect of the heading error was addressed by implementing the disturbance estimation. The lateral position error was compensated for by updating the reference path based on the safe drivable

corridor to avoid collisions with static obstacles. Offset-free MPC was used to determine the desired yaw rate considering the desired path from the drivable corridor. After the bias in heading error was estimated by implementing the disturbance estimation using a moving horizon estimation (MHE), the corrected heading error was used as the current state of the offset-free MPC, and the desired yaw rate was feasibly acquired. As a result, the localization uncertainty could be coped with in the longitudinal and lateral motion planning algorithms.

Motion control is affected by environmental disturbances (due to changes in road slope at an underpass and overpass) and model uncertainties (due to changes in mass caused by onboarding and stopover of passengers). What is worse, feedback information is limited since the number of sensors equipped in a full-sized bus is fewer than that in a private car. The model-reference adaptive sliding mode control (ASMC) scheme with radial basis function neural network (RBFNN)-based uncertainty bound estimator was implemented for the longitudinal motion control. The sliding mode control (SMC) scheme was applied to the lateral motion control. Consequently, robustness against environmental disturbances and model uncertainties with limited feedback information could be achieved using both the proposed longitudinal and lateral motion control algorithms.

After the feasibility of the suggested approaches was evaluated based on computer simulations, the applicability was confirmed from actual vehicle tests on the test track and bus rapid transit (BRT) course. Finally, the proposed algorithms were confirmed as applicable for fully autonomous driving of the full-sized bus in urban bus-only lanes.

Keywords : Automated bus, Dynamic programming, Model predictive control, Deep reinforcement learning, Moving horizon estimation, Offset-free model predictive control, Adaptive sliding mode control

Student Number : 2018-37061

Table of Contents

Chapter 1. Introduction.....	1
1.1. Research Background and Motivation.....	1
1.2. Previous Researches	5
1.2.1. Research on Motion Planning for Ride Comfort and Safety with Limited FOV and Detection Range	5
1.2.2. Research on Motion Planning for Coping with Localization Uncertainty	7
1.2.3. Research on Motion Control Algorithm considering Environmental Disturbance and Model Uncertainty	10
1.3. Dissertation Objectives	12
1.4. Dissertation Outlines	15
Chapter 2. Overall Architecture	18
2.1. Software Configuration	18
2.2. Hardware Configuration and Test Vehicle	21
2.3. Vehicle Test Environments	25
2.4. Definition of ODD for Research Objective	27
Chapter 3. Longitudinal Motion Planning	29
3.1. Problem Definition	29
3.2. Outline of Longitudinal Motion Planning	32
3.3. Analysis of Human Driving Data.....	35
3.3.1. Effects of Limited FOV and Detection Range	35
3.3.2. Human Driver's Characteristics against Ride Comfort and Perceived Safety	39
3.4. Long Time-horizon Motion Planning	40
3.4.1. Definition of System Dynamics Model.....	40
3.4.2. Long Time-horizon Motion Planning using DP.....	42

3.5. Short Time-horizon Motion Planning.....	4 8
3.5.1. Definition of System Dynamics Model.....	4 8
3.5.2. Short Time-horizon Motion Planning using chance-constrained, Fuzzy-tuned MPC	4 9
3.6. Development of Longitudinal Motion Planning using DRL for Comparative Study	6 8
3.6.1. Long Time-horizon Motion Planning	6 9
3.6.2. Short Time-horizon Motion Planning	8 0
Chapter 4. Lateral Motion Planning.....	8 2
4.1. Problem Definition	8 2
4.2. Outline of Lateral Motion Planning.....	8 4
4.3. Examination of Localization Uncertainty in Urban Environments	8 6
4.4. Disturbance Estimation.....	9 3
4.4.1. Definition of System Dynamics Model.....	9 3
4.4.2. Disturbance Model	9 5
4.4.3. Disturbance Estimation using MHE.....	9 9
4.5. Drivable Corridor Determination.....	1 0 4
4.6. Lateral Motion Planning using Offset-free MPC.....	1 0 6
Chapter 5. Longitudinal and Lateral Motion Control	1 0 8
5.1. Problem Definition	1 0 8
5.2. Outlines of Longitudinal and Lateral Motion Control.....	1 1 0
5.3. Longitudinal Motion Control.....	1 1 2
5.3.1. Definition of System Dynamics Model.....	1 1 2
5.3.2. Longitudinal Motion Control using MR-ASMC.....	1 1 6
5.4. Lateral Motion Control	1 2 7
Chapter 6. Performance Evaluation	1 2 9
6.1. Performance Evaluation of Longitudinal Motion Planning.....	1 2 9

6.1.1. Computer Simulation Results.....	1 2 9
6.1.2. Vehicle Test Results.....	1 4 4
6.2. Performance Evaluation of Lateral Motion Planning.....	1 4 9
6.2.1. Vehicle Test Results in FMTC Track	1 4 9
6.2.2. Vehicle Test Results in Urban Bus-only Lanes.....	1 5 3
6.3. Performance Evaluation of Longitudinal Motion Control.....	1 5 7
6.3.1. Computer Simulation Results.....	1 5 7
6.3.2. Vehicle Test Results in FMTC Track	1 6 7
6.3.3. Vehicle Test Results in Urban Bus-only Lanes.....	1 7 2
6.4. Performance Evaluation of Lateral Motion Control.....	1 7 5
6.4.1. Vehicle Test Results in FMTC Track	1 7 5
Chapter 7. Conclusion & Future Works.....	1 7 8
7.1. Conclusion.....	1 7 8
7.1.1. Contributions for Ride Comfort and Safety with Limited FOV and Detection Range.....	1 7 8
7.1.2. Contributions for Coping with Localization Uncertainty.....	1 7 9
7.1.3. Contributions for Robustness against Environmental Disturbance and Model Uncertainty	1 8 0
7.2. Future Works.....	1 8 1
Bibliography	1 8 3

List of Figures

Fig. 1.1. Challenges in autonomous driving for the urban bus-only lanes in Sejong City, South Korea.	4
Fig. 2.1. The overall architecture of an autonomous driving algorithm and research scope.....	1 9
Fig. 2.2. The software architecture of the proposed algorithm.	2 0
Fig. 2.3. The hardware configuration of the test vehicle: full-sized, electric-driven bus (Hyundai Elec-city).	2 3
Fig. 2.4. Vehicle test track in FMTC, Siheung City, South Korea.	2 5
Fig. 2.5. Vehicle test course in BRT lanes, Sejong City, South Korea.	2 6
Fig. 3.1. Overview of the proposed longitudinal motion planning algorithm.	3 3
Fig. 3.2. Outline of the proposed longitudinal motion planning algorithm.....	3 4
Fig. 3.3. Box plot of the detection ranges of the automated bus acquired from analysis of driving data acquired in BRT course.....	3 6
Fig. 3.4. Vehicle test results of longitudinal motion planning against a stationary target vehicle	3 8
Fig. 3.5. Distribution of longitudinal speed, acceleration, and TTC^{-1} of human driving data acquired in BRT course.....	4 0
Fig. 3.6. Simulation result of long time-horizon motion planning using DP.....	4 5
Fig. 3.7. Backward reachability plot and trajectory of the simulation result using DP.	4 7
Fig. 3.8. Fuzzy rules and distribution of the weights for reference tracking cost of the short time-horizon longitudinal motion planning.....	5 3
Fig. 3.9. Chance-constraint for uncertainty of position from perception and localization modules.....	5 8
Fig. 3.10. Clearance and longitudinal speed of target vehicle acquired in vehicle test	

through both perception module and V2V communication.	6 2
Fig. 3.11. Distribution of clearance and speed errors acquired in vehicle tests. .	6 3
Fig. 3.12. Procedure for calculation of curvature-based speed profile.....	6 5
Fig. 3.13. Example of curvature-based speed profile.....	6 7
Fig. 3.14. The architecture of the actor and critic network for DDPG.....	7 8
Fig. 3.15. The structure of the actor and critic networks for the proposed algorithm.	7 9
Fig. 4.1. Definition of path tracking error about lane centerline.....	8 3
Fig. 4.2. Outline of the proposed lateral motion planning algorithm.....	8 5
Fig. 4.3. Lane width of BRT course and width of the test vehicle.	8 7
Fig. 4.4. GPS status and relative errors between the measured value of RTK GPS and estimated value of localization module.....	8 9
Fig. 4.5. Estimated deviation of longitudinal position, lateral position, and heading angle from localization module.....	9 0
Fig. 4.6. Overview of the proposed offset-free MPC for lateral motion planning. .	9
7	
Fig. 4.7. Concept of a moving horizon estimation.	1 0 3
Fig. 4.8. The determination of the desired path considering drivable corridor	1 0 5
Fig. 5.1. Steering system of the test vehicle.....	1 0 9
Fig. 5.2. Outline of the proposed longitudinal motion control algorithm.	1 1 1
Fig. 5.3. Outline of the proposed lateral motion control algorithm.....	1 1 2
Fig. 5.4. External forces acting on a vehicle with respect to the longitudinal direction.	1 1 3
Fig. 5.5. Zero-throttle acceleration of a vehicle according to vehicle speed...	1 1 6
Fig. 5.6. The structure of the adopted RBFNN.	1 2 1
Fig. 5.7. Switching condition of throttle and brake modes.	1 2 5
Fig. 6.1. Simulation results when implementing DP-based algorithm: a stationary target detected at the initial clearance of 40 m.	1 3 1

Fig. 6.2. Simulation results when implementing DP-based algorithm: a slowly moving target with vehicle speed of 20 kph detected at the initial clearance of 40 m.....	1 3 3
Fig. 6.3. Simulation results when implementing DP-based algorithm: a moving target with vehicle speed of 40 kph which cuts-in at the initial clearance of 10 m (ego vehicle speed: 40 kph).....	1 3 5
Fig. 6.4. Simulation results when using DRL-based algorithm: a stationary target detected at the initial clearance of 40 m.	1 3 7
Fig. 6.5. Simulation results when implementing DRL-based algorithm: a slowly moving target with vehicle speed of 20 kph detected at the initial clearance of 40 m.....	1 3 9
Fig. 6.6. Simulation results when using DRL-based algorithm: a moving target with vehicle speed of 40 kph which cuts-in at the initial clearance of 15 m (ego vehicle speed: 40 kph).	1 4 0
Fig. 6.7. Simulation results when implementing proposed algorithm: Monte-Carlo simulation for different FOV ranges from 30 m to 100 m.	1 4 2
Fig. 6.8. Vehicle test results when implementing the conventional algorithm for a stationary target detected at the initial clearance of 40 m.	1 4 6
Fig. 6.9. Vehicle test results when implementing the proposed algorithm for a stationary target detected at the initial clearance of 40 m.	1 4 7
Fig. 6.10. Vehicle test results of the conventional algorithm for a slowly moving target (20 kph) detected at the initial clearance of 30 m.	1 4 8
Fig. 6.11. Vehicle test results when implementing the proposed and reference algorithms with an additional heading bias of -1 deg.	1 5 0
Fig. 6.12. Vehicle test results when implementing the proposed and reference algorithms with an additional heading bias of -2 deg.	1 5 1
Fig. 6.13. Vehicle test results when implementing the proposed and reference algorithms with an additional heading bias of ± 1 deg.....	1 5 2
Fig. 6.14. Vehicle test results when implementing the proposed and reference	

algorithms in BRT course.....	1 5 4
Fig. 6.15. Vehicle test results when implementing the proposed and reference algorithms in the underpasses of BRT course.	1 5 6
Fig. 6.16. Outline of the computer simulation for the proposed longitudinal motion control algorithm.....	1 5 8
Fig. 6.17. Control performance of the proposed and reference longitudinal motion control algorithms on flat ground.....	1 6 4
Fig. 6.18. Reference speed tracking delays on flat ground.	1 6 5
Fig. 6.19. Results of sensitivity analysis for (a) change in road slope (b) change in mass (c) change in equivalent control input when using the proposed and reference longitudinal motion control algorithms.....	1 6 7
Fig. 6.20. Driving route for testing the proposed longitudinal motion control algorithm in FMTC track.....	1 6 8
Fig. 6.21. Vehicle test results of the proposed and reference longitudinal motion control algorithms on the test track.	1 7 1
Fig. 6.22. Vehicle test results of the proposed longitudinal motion control algorithm on the urban bus-only lanes.....	1 7 4
Fig. 6.23. Vehicle test results when using the proposed lateral motion control algorithm in the FMTC track	1 7 6
Fig. 6.24. Vehicle test results when implementing the conventional lateral motion control algorithm in the FMTC track	1 7 7

List of Tables

Table 2.1. Specification of the test vehicle.....	2 4
Table 2.2. Definition of ODD for the research objective.....	2 8
Table 3.1. Objectives of proposed overall motion planning algorithm.	3 1
Table 3.2. Parameters of objective function for long time-horizon optimization....	4 4
Table 3.3. Grids of state and control variables for discrete DP of long time-horizon motion planning.	4 4 4 4
Table 3.4. Parameters for state and control input constraints for short time-horizon longitudinal motion planning	5 2
Table 3.5. Pros and cons of DP and DRL for long time-horizon motion planning.	6 8
Table 3.6. Structure of algorithms using DP and DRL for comparative study. ...	6 9
Table 3.7. Parameters of reward function for DRL.	7 2
Table 3.8. Pseudo code of DDPG algorithm for long time-horizon motion planning.	7 7 7 7
Table 3.9. Training parameters for the proposed agent of the DDPG method. ...	7 9
Table 3.10. Parameters of the simulation environment and bound of state and action space.....	7 9 7 9
Table 6.1. Summary of calculation time of proposed and conventional algorithms	1 4 4
Table 6.2. Design parameters of the proposed longitudinal motion control algorithm.	1 6 0 1 6 0
Table 6.3. Summary of the tracking performance from computer simulations when using the proposed longitudinal motion planning algorithm.....	1 6 1 1 6 1

Chapter 1. Introduction

1.1. Research Background and Motivation

In autonomous driving, automated public buses have been actively researched because the efficiency of transportation systems can be improved, and traffic accidents can be prevented

[1, 2, 3, 4]. Recently, studies are focusing on satisfying the requests of passengers on the automated bus: Ride comfort and safety, which are key performance in public transport [5, 6]. To meet the demands, it is necessary to develop motion planning and motion control algorithms for an automated bus. The applicability of constructed approaches has to be confirmed from simulations and vehicle tests. When an existing logic was tested for evaluating the performance in terms of ride comfort and safety, it was possible for me to recognize the potential weaknesses in motion planning and motion control as below.

First, an automated bus can be faced with dangerous situations because of the effective field of view (FOV) and detection range of the perception module. The FOV of the bus is limited according to on-board sensors' performance and complex road structures. The detection ranges of existing off-the-shelf sensors are shorter than those of human drivers. What is worse, complex road structures result in occluded regions. The potential risk of the limited FOV and detection range can be understood from Fig. 1.1. The bus rapid transit (BRT) lanes include inclined roads with buses that are in operation at short intervals as shown in Fig. 1.1(a), and complex road structures including underpasses and overpasses, as shown in Fig. 1.1(b). They make the FOV and detection range of the automated bus limited. To secure ride comfort

and safety, it was necessary to develop the longitudinal motion planning algorithm for the automated bus with the limited FOV and detection range.

Second, a global positioning system (GPS) denied area exists since urban driving environments include underpasses and overpasses to separate bus-only lanes from roadways and high buildings along the lanes. The problem related to the GPS-denied environment can be understood from Fig. 1.1(a), where the automated bus passes through an underpass along the lanes surrounded by high buildings. They make the shading of GPS satellite signals. The performance of the localization module depends on the quality of GPS signals. The effect of the localization performance on the lateral clearance in a lane is quite significant when the automated bus is driven along a narrow road, as shown in Fig. 1.1(c). Thus, potential danger can be recognized for the bus compared to a private car whose width is smaller than that of the bus. Furthermore, it is a high risk for an automated bus to be interfered with by a median strip, sidewalls, and barriers along the lane in the GPS-denied environment. To overcome the potential dangers, it was necessary to suggest an approach that could cope with the localization uncertainty for the lateral motion planning algorithm.

Third, motion control is affected by environmental disturbance and model uncertainty. What is worse, feedback information is limited since the number of sensors equipped in a full-sized bus is fewer than that in a private car. The environmental disturbance and model uncertainty can be understood from Fig. 1.1(a) and Fig. 1.1(b): road slope on an overpass, curves due to the bus stop, and change in the number of passengers at a bus stop. In addition, limited feedback information due to the number of equipped sensors (for example, longitudinal accelerometer) on

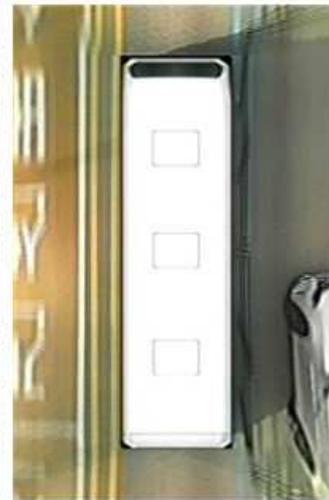
a commercial vehicle and public bus make motion control not robust. To accomplish the robustness in the motion control for an automated bus, it was necessary to construct a control algorithm against environmental disturbance and model uncertainty.



(a) Limited FOV and GPS-denied environments due to road structures



(b) Environmental disturbance due to road slope for an overpass



(c) Narrow road comparing to a bus captured from Around-view Monitor (AVM)

Fig. 1.1. Challenges in autonomous driving for the urban bus-only lanes in Sejong City, South Korea.

1.2. Previous Researches

Since the essential enhancement of autonomous driving for an automated bus was recognized, the state of the arts was reviewed to investigate contributions. From Sections 1.2.1 to 1.2.3, literature reviews are summarized according to the recognized weaknesses: (1) ride comfort and safety with limited FOV and detection range, (2) localization uncertainty, and (3) environmental disturbance and model uncertainty.

1.2.1. Research on Motion Planning for Ride Comfort and Safety with Limited FOV and Detection Range

Cooperative intelligent transport systems (C-ITS), including vehicle to vehicle (V2V) and vehicle to infrastructure (V2I) communications, were developed to support autonomous driving of a vehicle with limited FOV and detection range. V2V and V2I communications were used to improve the accuracy of targets' state estimation [7], the safety in signalized intersections [8], and the performance of the cooperative adaptive cruise control (ACC) with nearby automated vehicles [9]. A V2I-based advisory algorithm was constructed to prevent collisions between vehicles and pedestrians [10]. However, there are several restrictions: First, the approaches suggested in the referred studies were only simulated or partially implemented on vehicle tests. Second, if the number of unconnected vehicles increased, the effectiveness of the information from infrastructure was insufficient to overcome problems due to the limited FOV and detection range.

The virtual target concept was also proposed to overcome the limited FOV and

detection range without support from infrastructures. Representatively, the virtual target vehicle was implemented for proactive motion planning under safe lane changing and overtaking situations [11] and in urban intersections [12]. However, it is difficult for the virtual target concept to be implemented in urban bus-only lanes since there are complex road structures, including underpasses, overpasses, and bus stops. Therefore, in the presence of the limited FOV and detection range, an optimal motion planning algorithm is required.

In general, motion planning algorithms have multiple objectives: Vehicle stability, traffic rules, traffic flow efficiency, ride comfort, and safety [13]. Model predicted control (MPC) has been applied to motion planning algorithms for solving the multi-objective optimization problem in real-time [11, 12, 14]. The algorithms in the previous researches [11, 12, 14] utilized a short time-horizon, between 2 and 4 sec, for real-time optimization considering computational cost. The algorithm using only MPC had to be designed as a performance-centered approach due to the restriction of the short time-horizon. Thus, it could not secure ride comfort since drastic control efforts were ordered for maximizing the tracking performance, unlike human drivers.

Several researchers proposed deep reinforcement learning (DRL) based planning algorithms to mimic human drivers' characteristics. A conditional deep Q-network for end-to-end path planning was proposed and tested in the CARLA simulator [15]. Human-like longitudinal motion planning models were proposed using DRL based on actual human drivers' data [16, 17]. However, the human drivers' data may not be optimum in terms of traffic flow efficiency and control effort minimization [18]. In addition, human driving data is not directly applicable to

autonomous driving algorithms because the FOV and detection range of autonomous vehicles are different from those of human drivers.

Long time-horizon optimization approaches have been taken into account for the motion planning algorithm using dynamic programming (DP) and DRL. Eco-optimal strategy at a signalized intersection was proposed using DP and MPC [19]. DP was adopted to find an energy-efficient speed profile. MPC was utilized to compute the desired control input to follow the speed profile while securing safe inter-vehicle distance. A comparative study of DRL and MPC for the car-following problem was carried out [20]. A widespread time-delay model and constant time gap (CTG) policy were adopted for the system dynamics and policy of the algorithms, respectively. The algorithm utilizing DRL was compared to those using short time-horizon MPC and long time-horizon MPC. Although an optimal solution could be calculated by the long time-horizon MPC, the real-time application was restricted due to its long calculation time. From the simulation studies, DRL showed better performance than that of short time-horizon MPC but slightly inferior performance than that of long time-horizon MPC. However, it was hard for the state and control input constraints to be implemented using only DRL. In addition, the researchers did not consider the uncertainty in the observed targets' states.

1.2.2. Research on Motion Planning for Coping with Localization Uncertainty

In autonomous driving, the term, which is written as uncertainty, includes environmental disturbance, incompleteness of dynamics model, and disturbances in

localization and perception modules. The uncertainties, which are related to environmental disturbance and dynamics model, will be addressed in the section for motion control algorithms. Here, the uncertainties in the localization and perception modules are coped with for motion planning algorithms.

The proactive approaches, including stochastic MPC and robust tube MPC, have been suggested for considering uncertainties in the perception and dynamics model. The stochastic MPC was implemented for the lane changing and merging situations by considering the uncertainties in the ego vehicle's states and perception results [21]. The robust tube MPC was established using a deep learning method [22]. After the deep learning-based approaches were proposed considering learning tubes that contain the possible trajectories, the learned tube was used for nonlinear MPC. Both approaches have a common limitation: An infeasible solution can be obtained for a lateral motion planning problem that an automated bus is driven along a narrow road.

In the proposed research, the motion planning algorithm was interworked with the localization module, which was developed using a state-of-the-art map-based localization algorithm. A localization algorithm estimates the position and orientation of an ego vehicle with respect to a global reference frame [23]. The map-based localization algorithm utilized the information from GPS, inertial measurement unit (IMU), and map-matching results. Road markings detected by a front camera and around-view monitor (AVM) were compared and matched to the surface markers in a high definition (HD) map. Then, the result was used to calibrate the position and orientation that were initially estimated from the data measured by GPS and IMU.

Changes in GPS status and the accuracy of map-matching results affect the performance of the map-based localization approach. As mentioned, the accuracy of GPS could be dropped because of the shading of GPS satellite signals in underpasses and near high buildings. The map-matching results could deteriorate due to insufficiently detected road markings and inaccurate position of the markers defined in HD map.

Most of the motion planning approaches utilized the information in HD map and used the information by transforming the values from the global reference frame into the ego vehicle's body-fixed reference frame. The approaches assumed that the exact position and orientation of the ego vehicle were obtainable, and localization uncertainty was negligible. Yoo et al. proposed a virtual target-based proactive longitudinal motion planning for urban intersections [12]. To consider the limited FOV and detection range of onboard sensors, the virtual targets were constructed at intersections using road information acquired from HD map. Xu et al. presented a motion planning and control system with experimental validation [24]. The system consisted of routing, behavior planning, trajectory generation, and trajectory tracking. HD map information, including road type, lane features, connections, and waypoints, was utilized for each sub-system. Jeong proposed a motion planning approach for urban environments [25]. The drivable corridor was determined by a probabilistic static obstacle map, predicted trajectory of moving obstacles, and the road boundaries from HD map.

However, in reality, since the localization uncertainty affects the accuracy of reference values in HD map transformed into the body-fixed frame, the effect of the uncertainty on a motion planning algorithm cannot be negligible. The motion

planning approaches, which attempted to consider the localization uncertainty, have been studied in two ways: applying the variance of the localization uncertainty, and regarding the uncertainty as a disturbance and estimating it. Artunedo et al. proposed the motion planning approach based on a probabilistic occupancy grid map to utilize the localization uncertainty [26]. The uncertainty was described as variances of heading angle, longitudinal position, and lateral position errors. The occupancy grid map was calculated by fusing the information of road from HD map and propagating the localization uncertainty. Then, the optimal trajectory was chosen from the drivable corridor within the grid map. Ge et al. developed a motion control algorithm using an integrated vehicle dynamics model [27]. The mismatches in the integrated vehicle dynamics model were modeled as constant disturbances and estimated by Kalman filter. Then, the motion control algorithm was developed using an offset-free MPC scheme for disturbance rejection.

However, the described approaches were inapplicable for an automated bus due to the following limitations; The algorithms were designed for small objects which were not significantly affected by the bias in the localization uncertainty. Since the motion planning for an automated bus is restricted to driving on a narrow path, the planned motion can be impracticable unless the bias is coped with. In addition, most of the algorithms assumed that the uncertainty was a constant value. As the localization uncertainty can be varied with spatial movement and changes in the referred sensor status, the uncertainty should be treated as a variable.

1.2.3. Research on Motion Control Algorithm considering Environmental Disturbance and Model Uncertainty

Many algorithms for longitudinal controllers have been proposed. The controllers were developed by estimating the disturbances using the dynamics model and feedback information. The self-tuning controller was proposed based on the vehicle powertrain model [28]. The adaptive control algorithm was proposed using a tire-road friction coefficient estimator [29]. The artificial neural network sliding mode control (ANN-SMC) was proposed using the feedback information of throttle and brake torques [30]. The summarized approaches have a common premise: An accurate model has to be taken into account for vehicle dynamics, and sufficient feedback information is transmitted. In actual driving conditions, there are environmental disturbances due to changes in road slopes. In addition, the dynamics model is significantly affected by model uncertainty related to changes in mass when onboarding or stopover of passengers in addition to changes in model parameters of tire-road friction and actuator control gains. The feedback information can be limited due to the number of sensors equipped in a test vehicle. Therefore, it is difficult to meet the premise which has been applied to conventional control algorithms.

Some researchers have proposed model-free control approaches to overcome the model uncertainty. The intelligent PID controllers, that utilized the estimation of nonlinear system and PID control together, were proposed [31, 32, 33]. The adaptive controllers were proposed using a time-varying parameter adaptation of the vehicle dynamics model [34, 35]. These studies assumed that the effect of unknown disturbances on the performance of the controller was insignificant. The model-reference adaptive controller, that followed the reference target speed and updated the disturbance parameter of the system with the adaptation law, was proposed [36]. This scheme can adjust the controller's parameters against the change in system dynamics.

Instead, the adaptive control depends on the explicit structure of the dynamics model. However, the performance of the controller is significantly affected by the initial value settings of the adaptation parameters. The sliding mode control (SMC) technique is widely used in tracking control, especially for compensating for the model uncertainty. The technique ensures robustness to parameter variations and environmental disturbances in a practical application [37]. However, this controller needs the upper bound of model uncertainty and has chattering. Adaptive sliding mode control (ASMC) has the advantage that the scheme can achieve the outstanding tracking performance of adaptive control schemes with robustness against model uncertainty. However, the upper bound of model uncertainty should be set, in advance.

A robust ASMC strategy using upper bound estimation has the advantage that the logic can reduce the effect of environmental disturbance and model uncertainty on the performance of the controller. Fuzzy logic and neural networks were implemented to design the switching control gain and uncertainty bound estimation [38, 39]. The novel ASMC strategies using a radial basis function neural network (RBFNN) have been presented [40, 41, 42]. The RBFNN showed good estimation performance and robustness against the sensor noise with small computational cost [43]. The adaptation law for the network was designed based on Lyapunov stability theory to secure the stability and convergence of the system.

1.3. Dissertation Objectives

This dissertation aimed to suggest the motion planning and control algorithm for an automated bus, to enhance ride comfort and safety with limited FOV and detection range, cope with localization uncertainty, and secure robustness against

environmental disturbance and model uncertainty. The suggested approaches for overcoming the weaknesses were implemented in longitudinal motion planning, lateral motion planning, and longitudinal and lateral motion control, respectively. The constructed autonomous driving algorithm was divided into several subcomponents not to be restricted by the calculation time in vehicle tests. The detailed objectives of this research are described below.

First, the longitudinal motion planning algorithm was suggested to obtain the optimized plan using smaller acceleration and jerk (change of acceleration) against an unexpected target, which was related to the limited FOV and detection range. After the detection range in the automated bus was investigated, DP and chance-constrained MPC were implemented to acquire the optimum longitudinal motion planning within the detection range. Especially, weights in the objective function of the MPC were tuned using a risk-index based fuzzy inference system, which was constructed from the analysis of human driving data for a full-sized bus. Finally, longitudinal motion planning, which could secure ride comfort and safety, was constructed through the optimized solution from the hierarchical framework using DP and chance-constrained MPC. In addition, it was necessary to investigate the suggested approach in comparison to DRL based algorithm developed with the same design purpose in terms of the recent research trends related to this topic. After the DRL-based longitudinal motion planning algorithm was hierarchically constructed, the performance was compared to that of the algorithm using DP. Since the comparison was attempted to review the suggested approach using DP, simulations under representative test conditions were planned for investigating advantages and disadvantages in terms of characteristics of the solution and calculation time.

Second, the lateral motion planning algorithm was constructed to cope with the effect of localization uncertainty on the heading error and lateral position error of the ego vehicle. To reduce the effect of heading error on the path tracking performance, the disturbance estimation was suggested. To avoid collisions with static obstacles, the lateral position error was compensated for by updating the reference path based on the safe drivable corridor. The localization uncertainty observed in longitudinal position error affects longitudinal motion planning. Although the error was addressed with the localization uncertainty, which was coped with for the lateral motion planning, it was taken into account in the longitudinal motion planning as the chance constraint. Finally, the result of the lateral motion planning was feasibly acquired in spite of the effect of localization uncertainty on path tracking performance.

Third, longitudinal and lateral motion control schemes were implemented considering environmental disturbance, including changes in road slope, and model uncertainty, including changes in mass due to onboarding or stopover of passengers. In addition, since advanced driver assistance systems (ADAS) were not equipped on a full-sized bus, the feedback information acquired from the chassis controller area network (CAN) was limited, unlike a private car. In the test vehicle, a longitudinal accelerometer was not equipped but wheel speed sensors were installed. Thus, the motion control algorithm was developed considering the feedback information from the wheel speed sensors only. Finally, motion control was robustly carried out under environmental disturbance and model uncertainty with limited feedback information.

1.4. Dissertation Outlines

This dissertation is organized in the following manner. In Chapter 2, the overall architecture is briefly introduced. The outline of software architecture, including localization, perception, motion planning, and motion control, is described by summarizing key elements. Hardware configurations and details of the test vehicle are listed with values that are referred to from specifications. In addition, the vehicle test track and course are schematically introduced. The track is used for assessing the feasibility of the proposed approach. The course in urban bus-only lanes is used to confirm the applicability of the approach under actual driving conditions. After that, the operational design domain (ODD) is defined with the dissertation objective. Chapter 3 presents the suggested longitudinal motion planning algorithm. The algorithm utilizes a hierarchical framework; DP and chance-constrained, fuzzy-tuned MPC are proposed for ensuring ride comfort and safety with limited FOV and detection range. The long time-horizon optimal planning is carried out considering the limited FOV and detection range using DP. The detection range of the test vehicle is acquired from the analysis of the driving data along the BRT course. The optimization problem for the planning adopts a long time-horizon and simple error dynamics model of a car-following problem to cover the FOV and detection range. The short time-horizon optimal planning is conducted using chance-constrained, fuzzy-tuned MPC. The state constraints of the MPC are designed to secure safety to collision, and non-violating speed limits, and to consider the uncertainties in localization and perception modules to the longitudinal direction. In addition, a fuzzy inference system and reasoning rules are constructed for replicating human drivers' decision-making in a bus. To investigate the performance of the proposed algorithm

in terms of the optimization performance and computational cost, a DRL-based algorithm is additionally developed. Then, the advantages and disadvantages of the two algorithms are summarized.

Chapter 4 shows the suggested lateral motion planning algorithm, which copes with localization uncertainty. The offset-free MPC using disturbance estimation is proposed to consider the localization uncertainty. The effects of localization uncertainty on heading error, lateral position error, and longitudinal position error are respectively investigated. To attenuate the heading error, the disturbance estimation is carried out using moving horizon estimation (MHE). To consider the lateral position error, the drivable corridor is determined considering collisions with static obstacles. The offset-free MPC implements the estimated states from MHE and desired path from drivable corridor determination. The desired yaw rate is calculated from the offset-free MPC.

Chapter 5 explains the suggested longitudinal and lateral motion control algorithms to secure robustness against environmental disturbance and model uncertainty. The longitudinal motion control algorithm is developed using the model-reference adaptive sliding mode control (MR-ASMC). The sliding mode controller updates the upper bound of uncertainty from RBFNN and secures robustness without the pre-defined bound. The lateral motion planning algorithm is constructed using SMC.

Chapter 6 presents the computer simulation results for evaluating the feasibility of the proposed algorithm and the vehicle test results for confirming the applicability of the suggested approach both in the test track and the urban bus-only lanes. To assess the feasibility of the suggested longitudinal motion planning algorithm,

computer simulations for a stationary target, slowly moving target, and cut-in and urgently decelerating target are conducted, respectively. In addition, Monte-Carlo simulation is conducted for targets detected from different ranges to improve the credibility of the suggested approach. Finally, it is possible to confirm the applicability of the algorithm for a stationary target and a slowly moving target through vehicle tests. For the suggested lateral motion planning, the proposed algorithm is investigated from vehicle tests in FMTC track and BRT course. From the feasibility that a heading bias is intentionally added to a signal acquired from the localization module, the effectiveness of the proposed approach is recognized. Finally, the applicability is confirmed from the vehicle tests along the urban bus-only lanes. For the suggested longitudinal motion control, computer simulations for the assumed environmental disturbance and model uncertainty are firstly carried out. Sensitivity analysis is additionally conducted to investigate the effectiveness of the controller. The applicability of the longitudinal and lateral motion control algorithms is confirmed through vehicle tests.

Chapter 7 shows the conclusion by summarizing the contributions of this dissertation and future works which are recognized from the simulations and vehicle tests.

Chapter 2. Overall Architecture

2.1. Software Configuration

The overall architecture of an autonomous driving algorithm and the research scope of this dissertation are plotted in Fig. 2.1. An autonomous driving algorithm is generally composed of localization, perception, motion planning, and motion control [11, 12, 21]. As highlighted in Fig. 2.1. (blue solid-line box), the research scope of this dissertation is the motion planning and motion control algorithms. In this research, the motion planning and control algorithms were divided into longitudinal and lateral planning/control to improve the computational efficiency of implementing the algorithms in an actual vehicle. The longitudinal and lateral motion planning algorithms determined desired longitudinal acceleration and desired yaw rate. The longitudinal motion control algorithm manipulated throttle and brake actuators to follow the desired longitudinal acceleration. The lateral motion control algorithm manipulated the steering angle to follow the desired yaw rate.

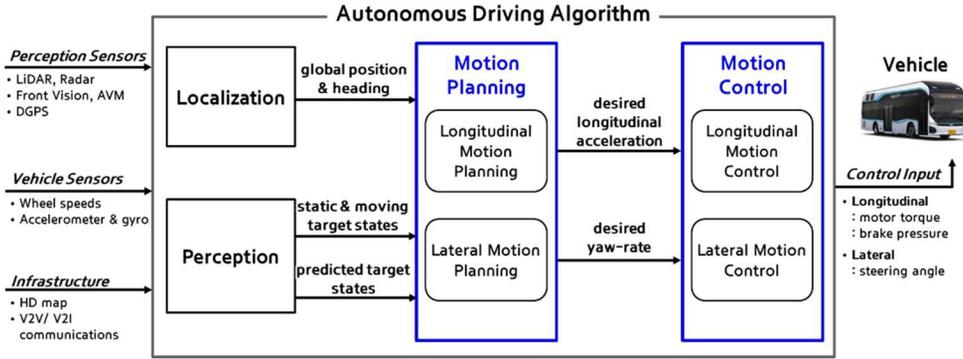


Fig. 2.1. The overall architecture of an autonomous driving algorithm and research scope.

Fig. 2.2 shows the software architecture of the proposed motion planning and motion control algorithms. As shown in Fig. 2.2, the longitudinal motion planning algorithm consists of three subsections: the risky target decision, the long time-horizon motion planning using DP, and the short time-horizon motion planning using chance-constrained, Fuzzy-tuned MPC. The lateral motion planning algorithm is composed of three subsections: the drivable corridor determination, the disturbance estimation using moving horizon estimation (MHE), and the motion planning using offset-free MPC. The risky target decision was conducted by selecting the closest preceding target among the stationary and moving targets within the region of interest (ROI). After the selected target's states were predicted, the drivable corridor was determined considering the predicted trajectories of the targets and the ego vehicle, and the road boundaries acquired from HD map. The ego vehicle's longitudinal and lateral state constraints were calculated using the drivable corridor. The longitudinal motion planning algorithm consisted of a long time-horizon using DP and a short time-horizon using chance-constrained, Fuzzy-tuned MPC to determine the desired longitudinal acceleration. The lateral motion planning

algorithm was constructed using offset-free MPC with disturbance estimation using MHE to determine the desired yaw rate. The motion control algorithms manipulated motor torque, brake pressure, and steering angle to track the desired longitudinal acceleration and yaw rate with accuracy.

As shown in Fig. 2.2, the motion planning algorithm is interfaced with the localization and perception modules. The global position and heading of the ego vehicle were obtained by the localization module. The relative position and heading, and speed of the surrounding targets were obtained by the perception module. All modules utilized the information from on-board sensors. Representatively, the longitudinal and lateral states of the ego vehicle were estimated through an ego vehicle filter using the information from data acquired by chassis controller area network (CAN).

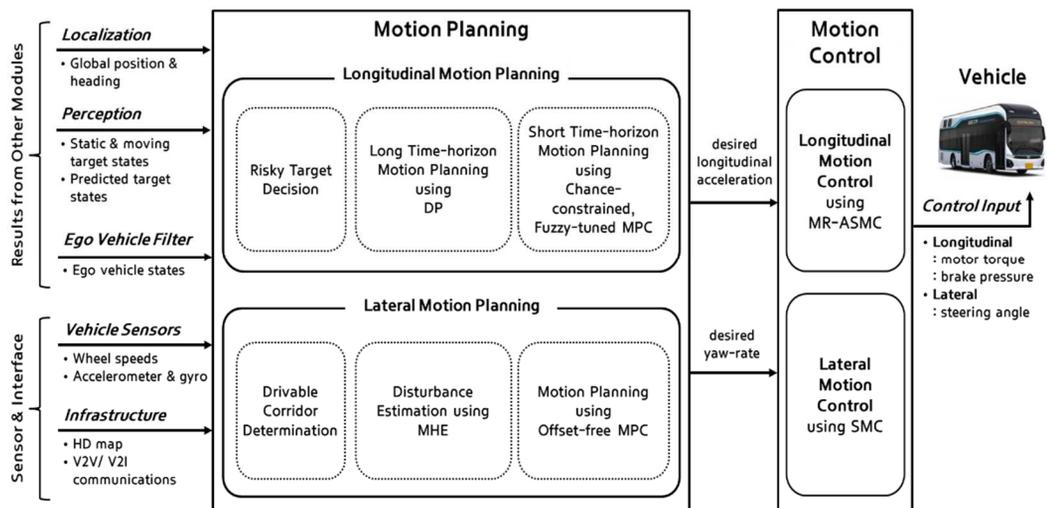


Fig. 2.2. The software architecture of the proposed algorithm.

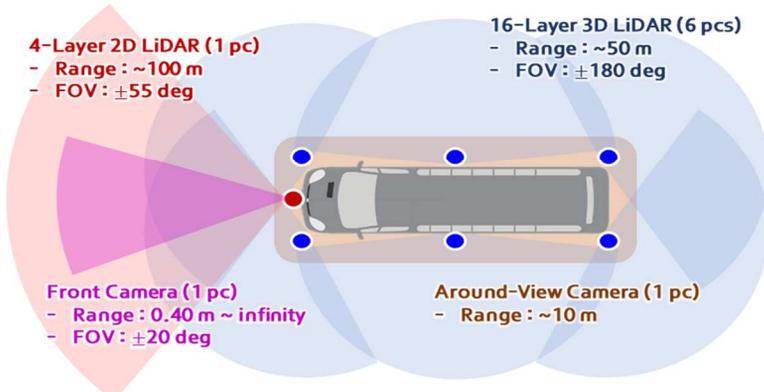
2.2. Hardware Configuration and Test Vehicle

Fig. 2.3 shows the on-board sensors, processors, C-ITS, and control interfaces of the test vehicle. The test vehicle, a Hyundai Elec-city, is a 47-seater electric-driven full-sized public bus, as shown in Table 2.1. The measured data from on-board sensors were used by the localization and perception algorithms. A high-precision GPS sensor (OXTS RT3000) was installed to measure the global position, orientation, speed, and acceleration of the vehicle. The measured data were utilized by the localization algorithm. The data were also collected to evaluate the performance of the algorithms. The longitudinal and lateral states of motion were estimated through an ego vehicle filter considering measured values of wheel speeds, longitudinal and lateral acceleration, and yaw rate from chassis CAN messages. As there was no longitudinal acceleration sensor on-board, the ego vehicle filter estimated the acceleration using the four wheels' speed and yaw rate. On-board unit (OBU) for the C-ITS interface was connected to the industrial PC for the motion planning algorithm. The rapid control prototyping (RCP) unit computed the actuator commands employing the longitudinal and lateral motion control algorithms to follow the desired acceleration and yaw rate. The electronic control units (ECUs) were linked with the RCP. The ECU was customized for the control interface using the motor torque, brake pressure, and steering wheel angle.

The middle part of Fig. 2.3 shows the FOV and detection range of the perception sensors. Although the maximum detection range of the 4-Layer 2D LiDAR (Ibeo LUX 2010) is up to 200 m from its specification, the effective detection range was approximately 100 m. In the case of the 16-Layer 3D LiDAR (Velodyne VLP-16), the maximum detection range from the specification is up to 100 m. Nevertheless,

the effective detection was limited to approximately 40 m. Since point clouds are required above a certain level to detect and track an object, the effective detection ranges have to be shorter than those described in the specifications. As shown in Fig. 2.3, LiDAR sensors are parallelly mounted along the ground at the bumper's height. Installing sensors on the roof of the automated bus was difficult because of the interference with the vehicle body. Therefore, it is understandable that the effective detection range of the automated bus is shorter than that of a private car using identical sensors.

The localization, perception, and motion planning algorithms were computed using an industrial PC. The iteration frequency and sampling time of the motion planning algorithm were 25 Hz and 40 ms, respectively. The motion control algorithm was computed using the RCP (dSPACE MicroAutobox-II), and its iteration frequency and sampling time were 100 Hz and 10 ms, respectively. Due to the difference between iteration frequencies of the upper-level and lower-level algorithms, the desired acceleration and yaw rate from the motion planning algorithms were updated to the motion control algorithms at four steps intervals. However, since the reference inputs were filtered and delivered to the motion control algorithm, the control performance was insignificantly affected. The RCP calculated the control commands and sent them to the ECUs for the control interface via the chassis CAN bus. The sampling time of the ECUs was 10 ms.



Industrial PC CPU : Intel Xeon 3,6 GHz RAM : DDR4 32 GB GPU : Geforce GTX3070		16-layer 3D LiDAR²⁾ Range : up to 100 m FOV : ± 180 deg horizontal ± 10 deg vertical	
Rapid Control Prototyping (RCP) Unit CPU : IBM PPC 750GL, 900 MHz / Memory : 16 MB CAN ¹⁾ Interface		4-Layer 2D LiDAR²⁾ Range : up to 200 m FOV : ± 55 deg horizontal, ± 3.2 deg vertical	
On Board Unit (OBU) WAVE/LTE hybrid Ethernet Interface Antenna : WAVE (4 pcs), LTE (2 pcs), GPS (1 pcs)		GPS³⁾ Dual Antenna Accuracy : 0,1 deg (heading) / 1 cm (RTK ⁴⁾) (position)	
Throttle Motor Torque to Engine Control Unit (ECU)		Front Camera FOV : ± 20 deg Imager Type : CMOS Focus: 0,4 m ~ inf	
Brake Brake Pressure to Electric Stability Control (ESC) System		Around-View Camera For Fisheye Cameras FOV : 360 deg Width : ~6 m Height : ~10 m	
Steering Electro-Hydraulic Steering System		<div style="border: 1px dashed black; padding: 5px;"> : Processor : V2X Interface : Control Interface : Onboard Sensor </div>	
1) CAN : controller area network 2) LiDAR : light detection and ranging 3) GPS : global positioning system 4) RTK : real-time kinematic			

Fig. 2.3. The hardware configuration of the test vehicle: full-sized, electric-driven bus (Hyundai Elec-city).

Table 2.1. Specification of the test vehicle.

Vehicle	Full-sized bus (Hyundai Elec-city)
Seater [pcs]	47
Gross mass [kg]	15,405
Curb mass [kg]	12,285
Mass ratio [gross/curb, %]	125
Length / width / height [mm]	10,995 / 2,490 / 3,420
Wheelbase [mm]	5,400
Powertrain	Electric motor
Power steering	Electro-hydraulic power steering system
Control interface	Throttle: motor torque input to ECU Brake: brake pressure input to ESC system Steering: steering wheel angle to electro-hydraulic steering system

2.3. Vehicle Test Environments

The vehicle tests were conducted on a vehicle test track and course. Fig. 2.4 illustrates the vehicle test track in the future mobility technology center (FMTC) at Seoul National University (SNU), Siheung City, South Korea. The vehicle tests for comparing the performance of the proposed algorithm with that of the reference algorithm were carried out in the FMTC test track. In the test track, the experimental variables, including surrounding obstacles, could be controlled. For confirming the applicability of the overall algorithm, fully autonomous driving was conducted in the BRT course, Sejong City, South Korea, which is urban bus-only lanes. The experiment aimed to demonstrate the performance of the proposed algorithm under the actual driving conditions, including interactions with other vehicles, environmental disturbances, and localization uncertainties. As illustrated in Fig. 2.5, the course is about 5.8 km long and includes three underpasses, one overpass, and six bus stops.

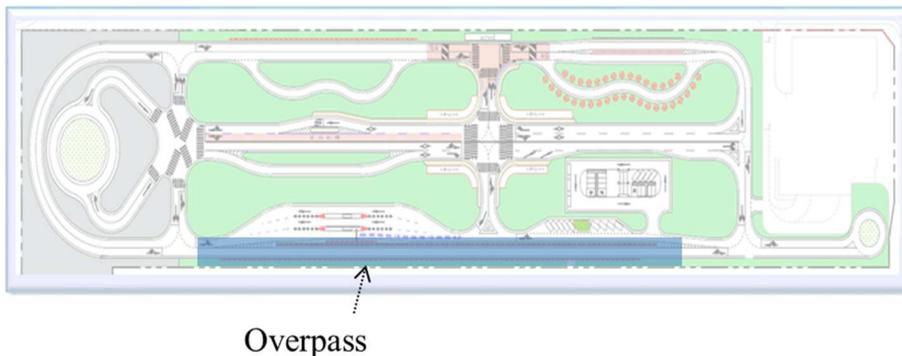


Fig. 2.4. Vehicle test track in FMTC, Siheung City, South Korea.



Fig. 2.5. Vehicle test course in BRT lanes, Sejong City, South Korea.

2.4. Definition of ODD for Research Objective

The definition of ODD is the operating environment within an automated driving system (ADS) that can perform the dynamic driving task (DDT) safely. For the proposed research objectives, the ODD of the research was defined by referring to the study [44] where details of the same vehicle test course in Sejong City were summarized.

As summarized in Table 2.2, the ODD of this search is largely divided into operational constraints, objects, and environmental conditions. The operational constraints included the operational speed range from 0 to 50 kph. The maximum speed of the range is the same as the speed limit of the test course. The detected objects for the research were composed of signage and roadway users. The status of the traffic light signals, which is the signage, was used for driving at the signalized intersection. Roadway users around the ego vehicle included surrounding vehicles and vulnerable road users. It was assumed that these users could be recognized as stationary and moving obstacles in the perception module.

In the case of environmental conditions, the weather-induced roadway conditions were restricted to normal driving conditions. For normal driving conditions on urban roads, the assumptions of small slip-rate and small slip angle of a tire can be regarded as valid. These assumptions were adopted utilizing dynamics models of the lateral motion planning and longitudinal motion control algorithms.

Table 2.2. Definition of ODD for the research objective.

Operational constraints		
Operation speed limit	ODD	Limit / Boundaries
Range of operation speed	<ul style="list-style-type: none"> • 0 ~ 50 kph 	<ul style="list-style-type: none"> • Exceeding 50 kph
Objects		
Signage	ODD	Limit / Boundaries
Traffic light signals	<ul style="list-style-type: none"> • Availability of V2I communication • Up-to-date HD map 	<ul style="list-style-type: none"> • Non-availability of V2I communication • Out-of-date HD map
Roadway users	ODD	Limit / Boundaries
Vehicles	<ul style="list-style-type: none"> • Up-to-date HD map • Detectability of static and dynamic objects • Availability of identifying lanes 	<ul style="list-style-type: none"> • Out-of-date HD map
Venerable road users (Pedestrians, Cyclists, and Motorcyclists)		
Environmental conditions		
Weather	ODD	Limit / Boundaries
Rain	<ul style="list-style-type: none"> • Availability of securing vision and LiDAR measurement • Availability of identifying lanes and objects 	<ul style="list-style-type: none"> • Non-availability of securing vision and LiDAR measurement • Non-availability of identifying lanes and objects
Snow	<ul style="list-style-type: none"> • Small snow with less impact on vehicle stability • Availability of securing vision and LiDAR measurement • Availability of identifying lanes and objects 	<ul style="list-style-type: none"> • Snow with a significant impact on vehicle stability. • Non-availability of securing vision and LiDAR measurement • Non-availability of identifying lanes and objects
Fog	<ul style="list-style-type: none"> • Availability of identifying lanes and objects 	<ul style="list-style-type: none"> • Non-availability of identifying lanes and objects
Wind	<ul style="list-style-type: none"> • Wind with less impact on the stability of the vehicle 	<ul style="list-style-type: none"> • Wind with a significant impact on the stability of the vehicle
Weather-induced roadway conditions	ODD	Limit / Boundaries
Road flooding	<ul style="list-style-type: none"> • Availability of identifying lanes and objects • Road condition with less impact on the stability of the vehicle 	<ul style="list-style-type: none"> • Non-availability of identifying lanes and objects • Road conditions with a significant impact on the stability of the vehicle
Road snowing		
Road freezing		
Illumination	ODD	Limit / Boundaries
Daytime / Dawn / Sunset / Night	<ul style="list-style-type: none"> • Availability of identifying lanes and objects 	<ul style="list-style-type: none"> • Non-availability of identifying lanes and objects
Streetlight / headlight		

Chapter 3. Longitudinal Motion Planning

3.1. Problem Definition

The objectives of a motion planning algorithm can be classified into traffic flow efficiency, ride comfort, safety, and traffic rules [13]. The objectives of the proposed motion planning algorithm are summarized as shown in Table 3.1. The traffic flow efficiency included maintaining proper inter-vehicle distance and relative speed to the preceding vehicle and not exceeding the speed limit. The ride comfort was divided into the smoothness of longitudinal motion and curvature smoothness of lateral motion. The safety could be divided into the perceived safety sensed by passengers and the driving safety against collisions with obstacles and violations of stop lines. The perceived safety is generally expressed as time gap (TG) and time to collision (TTC) which are representative parameters of a car-following problem. The driving safety was considered by securing the robustness of the algorithm and considering state constraints. The traffic rule indicates the lane geometry, speed limit, and stop sign.

The proposed longitudinal motion planning algorithm covered the features depicted in Table 3.1, except for the yaw rate for curve comfort and the lane geometry included in the traffic rule. The lateral acceleration and jerk, which are sorted in the curve comfort, were taken into account as the speed limit of the longitudinal motion planning. After calculating the curvature for a given target trajectory, the curve-based speed was obtained by considering the limits of lateral acceleration and jerk. Then, the smallest value between the curve-based speed and the regulatory speed limit of the road was adopted as the speed limit in the longitudinal motion planning. The

lateral motion planning algorithm considered the lane geometry. The algorithm calculated desired motion to drive the automated bus along the centerline while maintaining the vehicle within the lane. The longitudinal motion planning algorithm considered the traffic rules including the speed limit and stop sign. The stop signs for the automated bus are the stop-and-go at bus stops and signalized intersections. The automated bus should stop precisely at designated locations of bus stops for safe boarding and unloading of passengers. Also, the automated bus must not violate the stop lines at intersections.

Table 3.1. Objectives of proposed overall motion planning algorithm.

Category	Sub-category	Feature		Adopted part for motion planning
Traffic flow efficiency		Deviation from reference position	$p_h - p_{h.des}$	Longitudinal motion planning (long time-horizon)
		Deviation from desired speed	$v_{x.h} - v_{x.p}$	
Ride comfort	Smoothness	Longitudinal acceleration	$a_{x.h}$	Longitudinal motion planning (long time-horizon)
		Longitudinal jerk	$\dot{a}_{x.h}$	
	Curve comfort	Lateral acceleration	a_y	Longitudinal motion planning (determination of curvature-based speed) & Lateral motion planning
		Lateral jerk	\dot{a}_y	
		Yaw rate	γ	Lateral motion planning
Perceived safety		Deviation from desired TG	$t_{TG} - t_{TG.des}$	Longitudinal motion planning (short time-horizon)
		Deviation from desired TTC	$t_{TTC} - t_{TTC.des}$	
Traffic rule	Lane geometry	Deviation from centerline position (laterally)	$e_y - e_{y.des}$	Lateral motion planning
		Deviation from centerline orientation	$e_\psi - e_{\psi.des}$	
	Speed limit	Deviation from speed limit	$v_{x.h} - v_{x.limit}$	Longitudinal motion planning (short time-horizon)
	Stop sign	Deviation from stopping position	$p_h \leq p_{stop}$	Longitudinal motion planning (short time-horizon)

3.2. Outline of Longitudinal Motion Planning

In this dissertation, the longitudinal motion planning algorithm, which was hierarchically constructed, was suggested to solve the optimization problem with a long time-horizon and short time-horizon. As shown in Fig. 3.1, when the automated bus recognizes a stationary target, the optimum plan is that the ego vehicle stops behind the target by keeping a distance more than the minimum clearance. The desirable state (in other words, the terminal reference state) is significantly different according to the current state of the ego vehicle. If the optimum solution is acquired by using the long time-horizon, the optimum reference motion trajectory can be reachable to the terminal state. The reference motion can be obtained by implementing the long time-horizon motion planning, which can cover the detection range. After the solution for the long time-horizon was acquired, the vehicle model, ride comfort, risk, states, and input constraints were additionally considered to obtain the solution for the short time-horizon. When the desired acceleration was determined by solving both the long time-horizon and short time-horizon, the acquired longitudinal motion planning could be similar to that of human drivers.

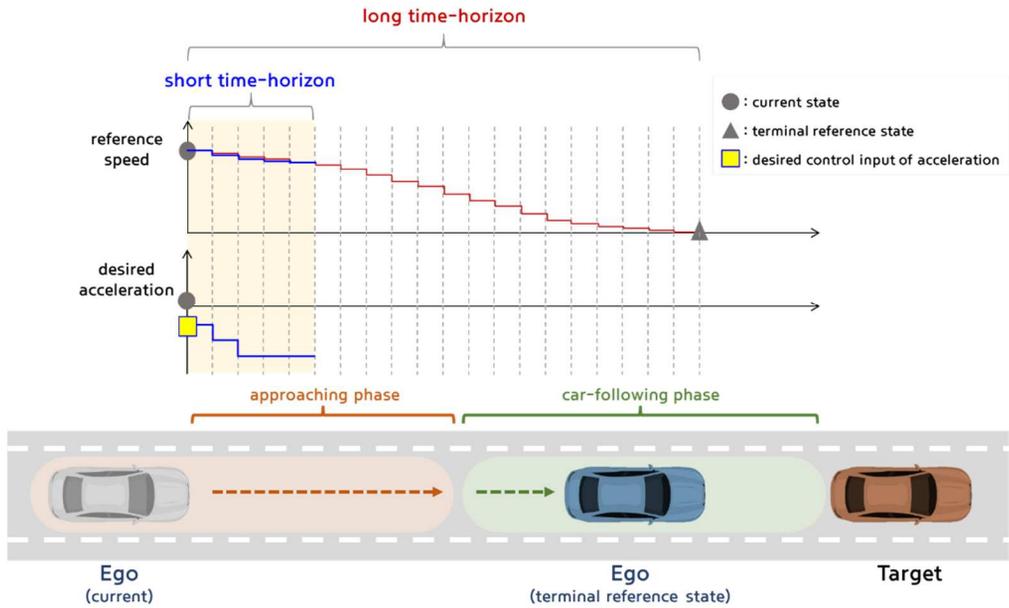


Fig. 3.1. Overview of the proposed longitudinal motion planning algorithm.

By simplifying the long time-horizon motion planning problem, the optimal reference motion was obtained. Through two assumptions, which were the CTG policy and error dynamics model, the problem became the Markov decision process (MDP). Consequently, the optimal solution could be obtained by implementing the long time-horizon optimization approaches, which are DP or DRL. However, it is hard to consider all the vehicle dynamics or constraints, since the approach cannot be implemented on an actual vehicle due to the increase in the calculation time. Thus, the reference motion was obtained using a simple model.

Short-horizon motion planning algorithm was constructed by considering the dynamics model, state, and control input constraint. Here, consideration of human driving characteristics was attempted when the optimization problem was defined to get the optimal control input acquired from the longitudinal motion planning (namely, desired acceleration). In general, human drivers show gradual patterns with

sophisticated control even against a dangerous target. Ride comfort against a non-hazardous target is prioritized by human drivers. In addition, human drivers show behavioral characteristics that they use insignificant control input at high speed and large control input at low speed. Since the described patterns could be formulated using the risk-based indices (TG and TTC), the indices were implemented to determine the weight of the optimization problem.

The proposed longitudinal motion planning module consists of three subsections: risky target selection, long time-horizon motion planning using DP, and short time-horizon motion planning using MPC (Fig. 3.2). Among the surrounding targets recognized from perception modules and the virtual target converted from the stop-lines of intersections or bus-stops within ROI, the riskiest target was selected. Long time-horizon motion planning algorithm found the optimal reference motion, which was defined as a sequence of travel distance and speed, considering the risky target's state. Short time-horizon motion planning algorithm determined desired acceleration using linear MPC. The weight of the optimization function was determined according to the ego vehicle's speed and the risk index (TG and TTC).

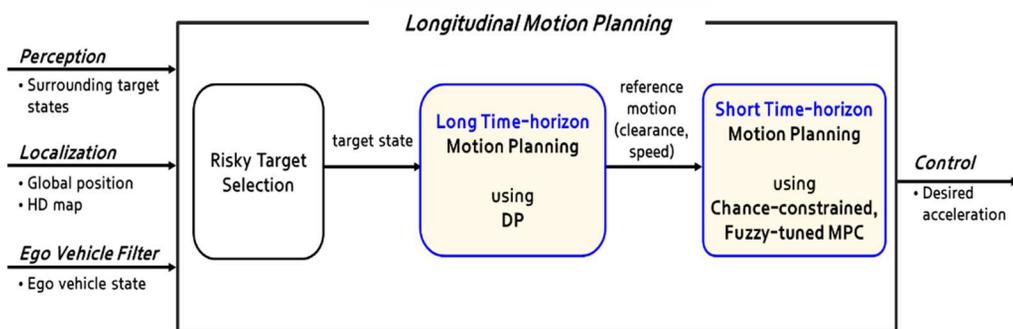


Fig. 3.2. Outline of the proposed longitudinal motion planning algorithm.

Since only a car-following situation existed in the urban bus-only lanes (two-

way, two-lane), three safety indices from a human driver's car-following model were investigated: clearance, TG, and TTC. The clearance is the relative distance between the ego vehicle and the target. TG is the time interval between the two vehicles, calculated as the clearance divided by the ego vehicle's speed. TTC is the remaining time of the collision. It is obtained by dividing the clearance by the relative speed. From these indices, records of human drivers' behavior could be investigated. As TTC becomes small, the collision risk becomes large [45]. Therefore, the reciprocal number of TTC, denoted as TTC^{-1} , is defined as a risk of collision.

3.3. Analysis of Human Driving Data

3.3.1. Effects of Limited FOV and Detection Range

The detection ranges of an autonomous vehicle using close-to-market sensors are shorter than those of human drivers. The effective detection range could be obtained by analyzing human driving data in the BRT course, Sejong city. The statistical properties are marked in Fig. 3.3. From the human driving data, the recognition of a preceding target, which was front-in, was available in the range from 32.2 m to 43.3 m. The perception module could not recognize the tracking target, which was front out, in the range from 39.5 m to 47.1 m. The effective detection range was approximately a quarter of the human driver's visible range of about 230 m [46]. In a private car using the identical sensors tested on flat ground, the effective detection ranges for the front-in target and front-out target were from 71.3 m to 86.1 m, and from 77.7 m to 95.7 m, respectively [47]. There was a big gap between the range of the automated bus and that of the private car. There are two main reasons: First,

several LiDAR points have interfered with the bodywork of the automated bus. The full-sized bus is 11 m long, which is about twice as long as that of the private car. It was difficult for rear LiDAR sensors to recognize the front region. Second, the visibility of sensors was restricted by the curvature of the BRT course. That was in line with the observations that the effective detection range was significantly affected by road curvature or slope [48]. Consequently, the limited FOV and detection range have to be considered in the longitudinal motion planning of an automated bus against targets.

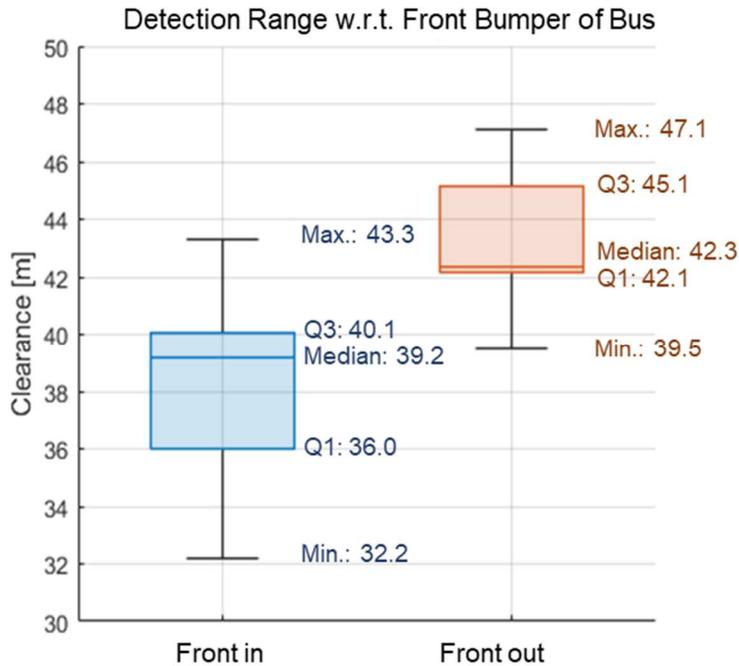


Fig. 3.3. Box plot of the detection ranges of the automated bus acquired from analysis of driving data acquired in BRT course

A human driver can take into account ride comfort and safety for driving with preceding vehicles since the human driver with sufficient visibility makes a longer

plan in both spatial and temporal domains. MPC algorithms based on CTG have been suggested for replicating human driving patterns and implemented in simulations and vehicle tests for longitudinal motion planning problems [7, 11, 49, 50, 51]. The model was constructed with the assumption that the relative distance between the ego vehicle and the target vehicle was proportional to the speeds of the vehicles. The proportional gain of the speed was defined as TG. The TG was assumed as a constant value according to a driving pattern of a human driver. The optimal control input was obtained by solving the optimization problem in MPC. In the objective function, terms for tracking performance and control effort minimization were included.

A distinct difference between the driving patterns of a human driver and those of an autonomous driving algorithm was observed in the preliminary vehicle test. Fig. 3.4 shows vehicle test results of a conventional longitudinal motion planning algorithm against a stationary target vehicle that is suddenly recognized at a specific location. (Here, an assumption was necessary for recognition of the human driver. The effective detection range of a human driver is longer than that of sensors; The human driver got noticed the target at a specific location where the target could be detected by the sensors.) As shown in Fig. 3.4(a), when an existing longitudinal motion planning algorithm is applied to the automated bus with limited FOV and detection range, the large control input is requested against a target that is suddenly recognized on sensors. When implementing the longitudinal motion planning algorithm, the automated bus decelerated largely as soon as the virtual target was recognized. Once the bus speed was reduced due to the significant control input, deceleration decreased as the bus got closer to the target. On the contrary, the human driver stepped on the brake with small deceleration as soon as the virtual target was

found. As the bus approached the virtual target (or assumed location as a target), the control input increased. As shown in Fig. 3.4(a) and Fig. 3.4(b), both the maximum peaks of deceleration and rate of deceleration are larger than those of human drivers when implementing the existing longitudinal motion planning algorithm. The larger peaks make passengers uncomfortable in terms of ride comfort and safety. From the preliminary tests, it was possible to recognize the necessity of an improved longitudinal motion planning algorithm that could mimic human driving patterns.

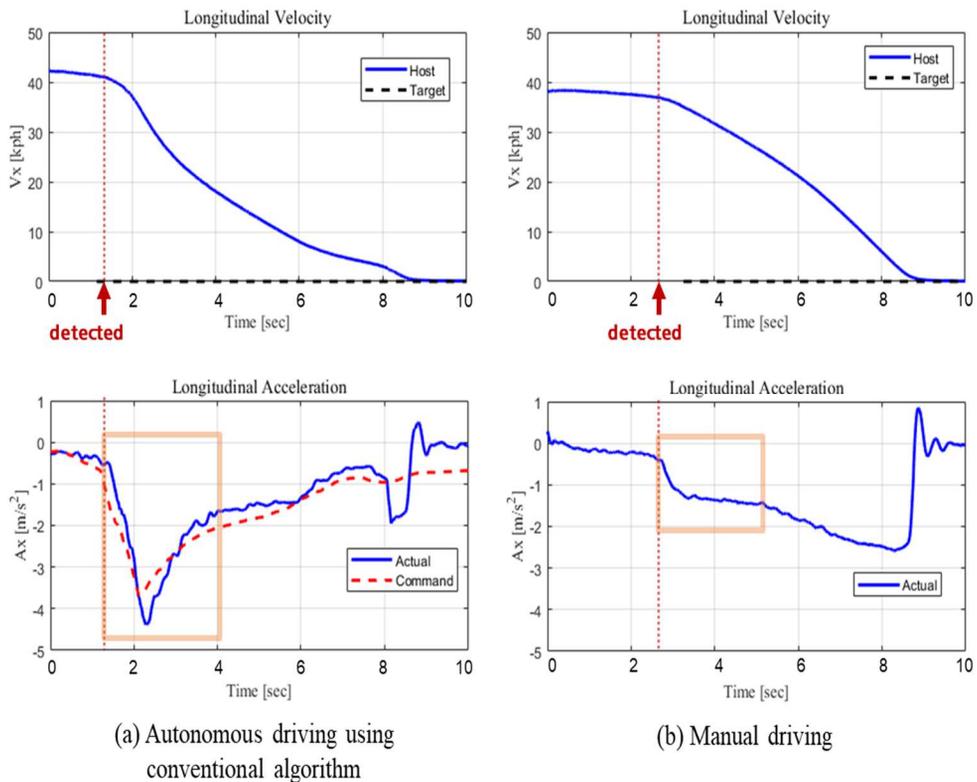
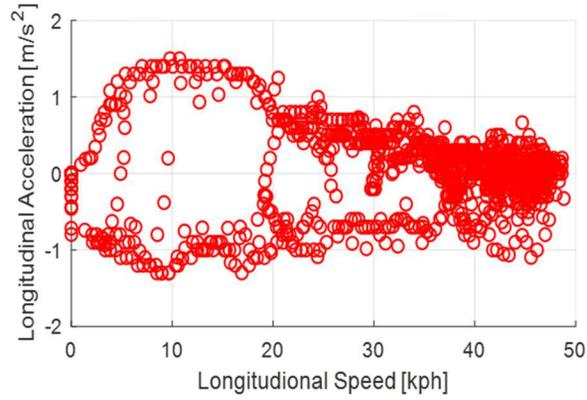


Fig. 3.4. Vehicle test results of longitudinal motion planning against a stationary target vehicle

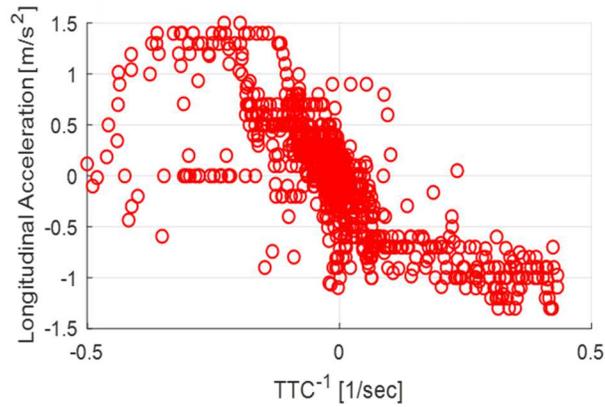
3.3.2. Human Driver's Characteristics against Ride Comfort and Perceived Safety

Ride comfort and safety are essential requirements of passengers in public transport [5]. Ride comfort is affected by vehicle motions: speed, acceleration, and jerk. Passengers feel safe when the automated bus responds quickly to the risk of collision and secures the minimum clearance and operates below the speed limits.

The experienced drivers' driving data was collected from the multiple connected vehicles and V2V interface in the BRT course. Two features were recognized from the collected data-driven analysis: the cautiousness of bus drivers; and correlations between control effort and risk index and between control effort and ego vehicle's speed. The bus drivers began to decelerate when the preceding vehicles were driven at TTC from 5 sec to 10 sec. The bus drivers' average TG was 3.9 sec, and the minimum clearance was 2.6 m. In contrast, it was reported that private car drivers' average TG was 1.4 sec and the minimum clearance was 2.0 m [45]. The cautiousness of bus drivers is in line with the observation in the previous research [52] that bus drivers tend to drive cautiously considering standing passengers and passengers who are boarding or alighting from a bus. As shown in Fig. 3.5(b), The bus drivers decelerate using small effort at low TTC^{-1} and large control effort at high TTC^{-1} . On the other hand, as shown in Fig. 3.5(a), large control effort is used at low speed, and small control effort is used at high speed. The observations are consistent with those of the previous research [45].



(a) Longitudinal Speed vs Acceleration



(b) TTC^{-1} vs Longitudinal Acceleration

Fig. 3.5. Distribution of longitudinal speed, acceleration, and TTC^{-1} of human driving data acquired in BRT course.

3.4. Long Time-horizon Motion Planning

3.4.1. Definition of System Dynamics Model

DP has been selected as a long time-horizon optimization scheme. Since the DP approach could be suffered from the curse of dimensionality issue, the state for the long time-horizon motion planning should be selected considering the number of state variables and state transitions that could be handled by the DP approach. Therefore, the formulation of the state equation adopted the kinematics-based error dynamics model of car-following problem, which is defined as follows:

$$\begin{aligned}
\Delta p_x(t) &= p_{x.ref}(t) - p_{x.h}(t) \\
&= \left\{ p_{x.p}(t) - \left(c_0 + \tau_g v_{x.p}(t) \right) \right\} - p_{x.h}(t) \\
\Delta v_x(t) &= v_{x.ref}(t) - v_{x.h}(t) = \min(v_{x.p}(t) - v_{x.limit}) - v_{x.h}(t) \\
\Delta a_x(t) &= a_{x.ref}(t) - a_{x.h}(t) = -a_{x.h}(t)
\end{aligned} \tag{3.1}$$

where, $\Delta p_x(t)$, $\Delta v_x(t)$, and $\Delta a_x(t)$ are the errors of travel distance, longitudinal speed, and longitudinal acceleration, respectively. $p_{x.ref}(t)$ is the reference travel distance, which is defined in terms of the clearance of target vehicle, denoted as $p_{x.p}(t)$, TG, denoted as τ_g , and minimum clearance, denoted as c_0 . $p_{x.h}(t)$ is the travel distance of ego vehicle. The reference travel distance ($p_{x.ref}(t)$) is determined by the desirable clearance, which is proportional to the nominal TG of bus drivers. The minimum clearance (c_0) is determined by considering the bus drivers' driving patterns. $v_{x.ref}(t)$ is the reference speed, which is the smallest value between the target vehicle's speed, denoted as $v_{x.p}(t)$, and the speed limit, denoted as $v_{x.limit}$. $a_{x.h}(t)$ is the longitudinal acceleration of ego vehicle. Here, the subscripts p and h represent the preceding (risky target) vehicle and host (ego) vehicle, respectively.

By using the Euler discretization approach, the equation of the continuous

system (3.1) is discretized and represented as a state-space equation as follows:

$$\begin{aligned}
\mathbf{x}_{\text{long}}(k+1|t) &= \mathbf{A}_{\text{long,D}}\mathbf{x}_{\text{long}}(k|t) + \mathbf{B}_{\text{long,D}}u_{\text{long}}(k|t) \\
\text{s.t. } \mathbf{A}_{\text{long,D}} &= \begin{bmatrix} 1 & \Delta t & -\Delta t^2/2 \\ 0 & 1 & -\Delta t \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{B}_{\text{long,D}} = \begin{bmatrix} -1/6 \cdot \Delta t^3 \\ -1/2 \cdot \Delta t^2 \\ -\Delta t \end{bmatrix} \\
\mathbf{x}_{\text{long}}(k|t) &= \begin{bmatrix} \Delta p_x(k|t) \\ \Delta v_x(k|t) \\ \Delta a_x(k|t) \end{bmatrix}, u_{\text{long}}(k|t) = \dot{a}_{x,h}(k|t)
\end{aligned} \tag{3.2}$$

where Δt is the time step. $\dot{a}_{x,h}$ is the jerk. For simplification of the problem, the speed of the preceding vehicle was assumed to be constant ($v_{x,p}(k|t) = v_{x,p}(0|t)$) within the time-horizon. The control input, denoted as $u_{\text{long}}(k|t)$, is defined as the jerk, denoted as $\dot{a}_{x,h}(k|t)$, because the state-space equation is constructed without considering the vehicle dynamics to reduce the number of states.

3.4.2. Long Time-horizon Motion Planning using DP

The long time-horizon motion planning algorithm optimized the reference motion profile considering the traffic flow efficiency and the ride comfort (as smoothness), as summarized in Table 3.1. The proposed optimization problem for the long time-horizon motion planning is constructed as follows:

$$\begin{aligned}
& \min_{u_{\text{long}} \in U} J(u_{\text{long}}, \mathbf{x}_{\text{long}}) \\
& = G_{N_p}(\mathbf{x}_{\text{long}}(N_p|t)) + \sum_{k=0}^{N_p-1} L(\mathbf{x}_{\text{long}}(k|t), u_{\text{long}}(k|t)) \quad (3.3) \\
& \text{s.t. } \mathbf{x}_{\text{long}}(k+1|t) = \mathbf{A}_{\text{long,D}}\mathbf{x}_{\text{long}}(k|t) + \mathbf{B}_{\text{long,D}}u_{\text{long}}(k|t) \\
& \quad , k \in \{0, \dots, N_p - 1\}
\end{aligned}$$

$$G_{N_p}(\mathbf{x}_{\text{long}}(N_p|t)) = \|\mathbf{x}_{\text{long}}(N_p|t)\|_{\mathbf{P}}^1 \quad (3.4)$$

$$L(\mathbf{x}_{\text{long}}(k|t), u_{\text{long}}(k|t)) = \|\mathbf{x}_{\text{long}}(k|t)\|_{\mathbf{Q}}^1 + \|u_{\text{long}}(k|t)\|_{\mathbf{R}}^1 \quad (3.5)$$

where G_{N_p} is the terminal cost, and L is the stage cost from $k-1$ step to k step. N_p is the maximum prediction step, which is determined by dividing the prediction time horizon into sampling time step. \mathbf{Q} and \mathbf{R} are the weight matrices of the state and control input for intermediate and terminal steps. \mathbf{P} is the weight matrix for the terminal state. The parameters of the objective function for the long time-horizon optimization are defined in Table 3.2.

As defined in Eq.(3.3) and Eq.(3.5), the objective function, denoted as J , includes the costs for minimizing reference tracking error, acceleration, and jerk to consider the traffic flow efficiency and ride comfort as smoothness. In addition, the terminal state constraint is considered for the convergence of the optimization problem as shown in Eq.(3.4). The state constraint for the terminal condition was represented as a soft constraint, and it was included in the objective function. By using the terminal state constraint, the calculated reference motion could be optimized from the initial to the terminal state.

Table 3.2. Parameters of objective function for long time-horizon optimization.

Parameter	Variable
Time step, Δt [sec]	0.1
Maximum prediction step, N_p [-]	150
Weight for state, Q	[0, 0, 5]
Weight for control input, R	[25]
Weight for terminal state, P	[5000/50, 5000/14, 5000/5]

DP is an optimization scheme that solves an optimization problem by using a multi-stage in which the final result is the consequence of a number of separate decisions. The solution of the optimization problem was obtained recursively on the basis of Bellman's principle of optimality. To solve the problem using DP, the grids for state and control variables are defined in Table 3.3.

Table 3.3. Grids of state and control variables for discrete DP of long time-horizon motion planning.

	Variable	Grid [Min.:Step size:Max.]
State	Error of travel distance, Δp_x [m]	[-10.0:0.05:50.0]
	Error of speed, Δv_x [m/s]	[-14.0:0.1:14.0]
	Error of acceleration, Δa_x [m/s ²]	[-5.0:0.2:1.0]
Control input	Jerk, $\dot{a}_{x,h}$ [m/s ³]	[-10.0:2.0:10.0]

The example of the optimization result is depicted in Fig. 3.6. The simulation

was conducted that the ego vehicle with a speed of 10 m/sec detected a stationary target at a relative distance of 50 m. As shown in Fig. 3.6(a), and Fig. 3.6(b), the solution converges to the terminal reference state, which is the origin. As shown in Fig. 3.6(c), the acceleration profile of the algorithm is gradual without a peak. This tendency is similar to that of human drivers as seen in Fig. 3.4.

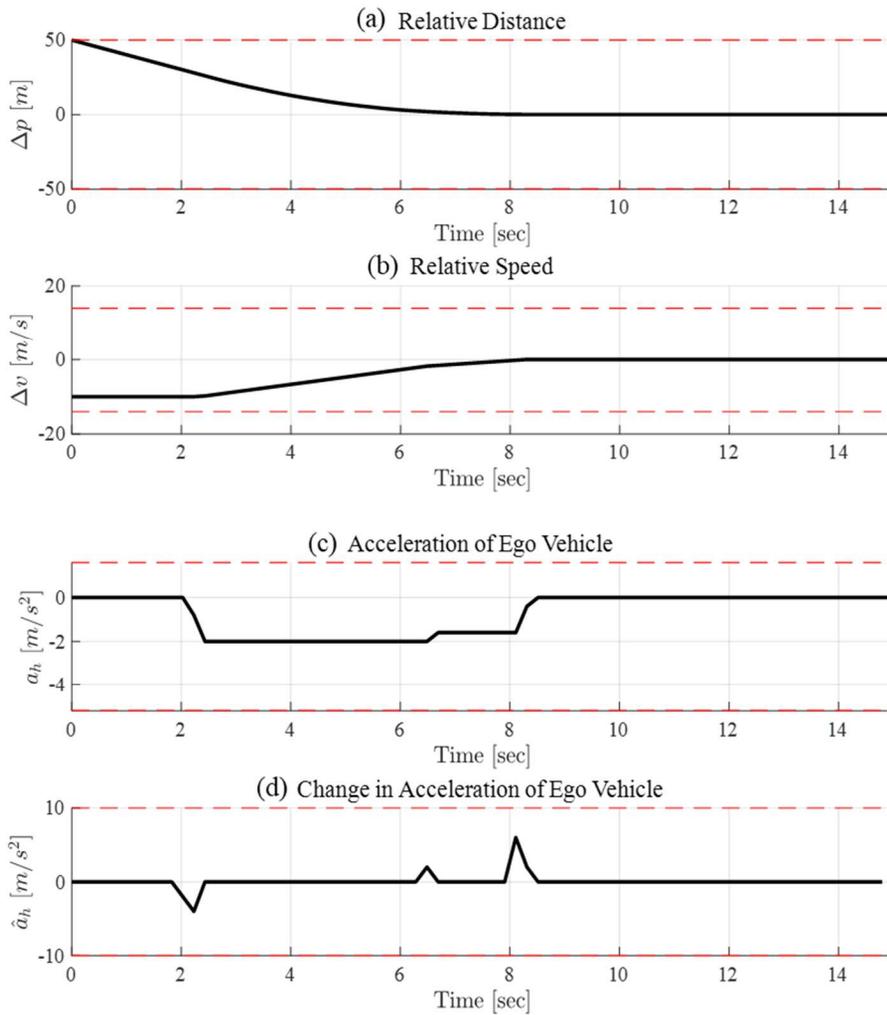
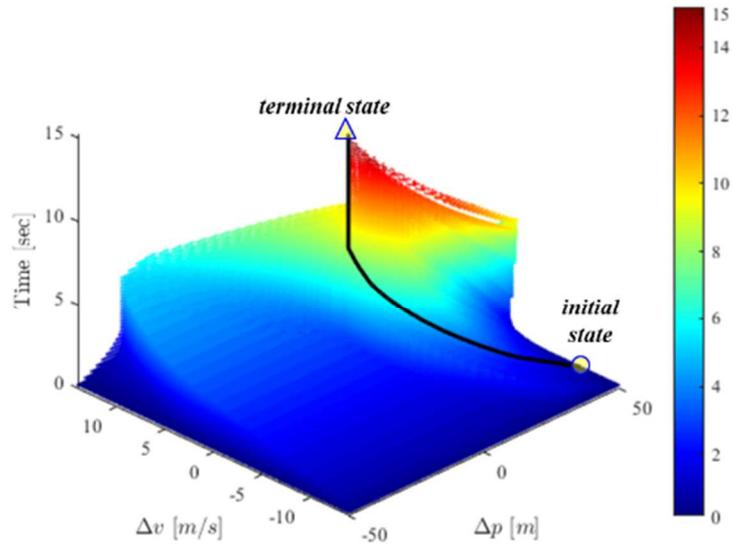


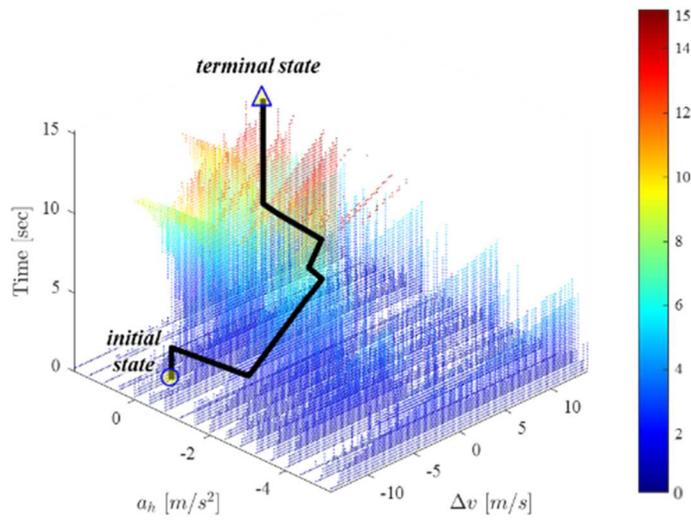
Fig. 3.6. Simulation result of long time-horizon motion planning using DP.

The reachability was investigated to analyze the feasibility of the proposed

algorithm. Backward reachability analysis was performed using the DP result. Fig. 3.7 shows the result of the backward reachability analysis and trajectory of the simulation result that is also shown in Fig. 3.6. From the result depicted in Fig. 3.7, it can be seen that the initial states within the detection range are reachable to the terminal state within the assumed time-horizon of 15 sec. Therefore, it was confirmed that the time horizon when using DP was appropriate.



(a) Backward Reachability w.r.t. Δp and Δv



(b) Backward Reachability w.r.t. Δv and a_h

Fig. 3.7. Backward reachability plot and trajectory of the simulation result using DP.

3.5. Short Time-horizon Motion Planning

3.5.1. Definition of System Dynamics Model

The ego vehicle's dynamics model was constructed using a kinematics model; the actual acceleration of the vehicle is assumed to track the desired acceleration with the first-order delay, represented as a time constant (τ). The continuous-time dynamics model is written in Eq.(3.6).

$$\begin{aligned} \dot{\mathbf{x}}_{lon}(t) &= \mathbf{A}_{lon.C}\mathbf{x}_{lon}(t) + \mathbf{B}_{lon.C}u_{lon}(t) \\ \text{s.t. } \mathbf{A}_{lon.C} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau_d} \end{bmatrix}, \mathbf{B}_{lon.C} = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{\tau_d} \end{bmatrix} \\ \mathbf{x}_{lon}(t) &= \begin{bmatrix} p_x(t) \\ v_x(t) \\ a_x(t) \end{bmatrix}, u_{lon}(t) = a_{x.des}(t) \end{aligned} \quad (3.6)$$

where, $\mathbf{x}_{lon}(t)$ is the state vector. The state consists of travel distance, denoted as $p_x(t)$, longitudinal speed, denoted as $v_x(t)$, and longitudinal acceleration, denoted as $a_x(t)$. $u_{lon}(t)$ is the control input of desired acceleration, denoted as $a_{x.des}(t)$. Using a zero-order hold (ZOH) discretization approach, the state equation (Eq.(3.6)) is discretized with a fixed sampling time, denoted as Δt , which is as follows:

$$\begin{aligned}
\mathbf{x}_{lon}(k+1|t) &= \mathbf{A}_{lon.D}\mathbf{x}_{lon}(k|t) + \mathbf{B}_D u_{lon}(k|t) \\
\text{s.t. } \mathbf{A}_{lon.D} &= e^{\mathbf{A}_{lon.C}\Delta t}, \mathbf{B}_{lon.D} = \int_0^{\Delta t} e^{\mathbf{A}_{lon.C}\tau} d\tau \cdot \mathbf{B}_{lon.C} \\
\mathbf{x}_{lon}(k|t) &= \begin{bmatrix} p_x(k|t) \\ v_x(k|t) \\ a_x(k|t) \end{bmatrix}, u_{lon}(k|t) = a_{x.des}(k|t)
\end{aligned} \tag{3.7}$$

3.5.2. Short Time-horizon Motion Planning using chance-constrained, Fuzzy-tuned MPC

3.5.2.1. Chance-constrained MPC for Short Time-horizon Motion Planning

The short time-horizon motion planning utilized chance-constrained MPC. As shown in Table 3.1, the optimization problem is formulated to follow the reference motion acquired from the long time-horizon motion planning while keeping the traffic rules. The speed limit was treated as a state constraint of the longitudinal speed. The stop signs at signalized intersections and bus stops were included in the state constraint of the travel distance.

The chance-constrained MPC for the short time-horizon motion planning is set up as shown in Eq.(3.8), including a chance constraint as shown in Eq.(3.8e). The objective function is formulated as defined in Eq.(3.8a), which consists of reference tracking, control input, change in control input (jerk), and slack variables for travel distance and speed. Among them, the amount of control input (desired acceleration) ($u_{lon}(k|t)$) and jerk ($\Delta u_{lon}(k|t)$) were added to the objective function in order to attenuate the chattering that may exist in the reference motion. To consider the

convergence of the hard constraints for states, slack variables were implemented for constraint relaxation. Here, the slack variables acted as penalties when violating the clearance and speed limit. After obtaining the optimal control input profile that satisfies the constraints, the first control input, denoted as $u_{lon(0|t)}$, was used as the desired acceleration of the longitudinal motion control module.

$$\begin{aligned}
\min_{u_{lon} \in U} J(u_{lon}, \mathbf{x}_{lon}) &= \sum_{k=1}^{N_p} \|\mathbf{x}_{lon}(k|t) - \mathbf{x}_{lon.ref}(k|t)\|_{Q_{ref}(k)}^2 \\
&+ \sum_{k=0}^{N_p-1} \|u_{lon}(k|t)\|_{R_u(k)}^2 + \sum_{k=0}^{N_p-2} \|\Delta u_{lon}(k|t)\|_{R_{\Delta u}(k)}^2 \\
&+ \sum_{k=1}^{N_p} \|q_p(k|t)\|_{Q_p}^2 + \sum_{k=1}^{N_p} \|q_v(k|t)\|_{Q_v}^2
\end{aligned} \tag{3.8a}$$

s.t.

$$\mathbf{x}_{lon}(k+1|t) = \mathbf{A}_{lonD} \mathbf{x}_{lon}(k|t) + \mathbf{B}_{lonD} u_{lon}(k|t) \quad , k \in \{0, \dots, N_p - 1\} \tag{3.8b}$$

$$\begin{aligned}
a_{x.min} &\leq u_{lon}(k|t) \leq a_{x.max} & , k \in \{0, \dots, N_p - 1\} \\
|\Delta u_{lon}(k)| &\leq S_u & , k \in \{0, \dots, N_p - 2\} \\
|u_{lon}(0|t) - u_{prev}| &\leq S_{u.prev}
\end{aligned} \tag{3.8c}$$

$$\begin{aligned}
-\varepsilon_{position} &\leq p_x(k|t) & , k \in \{1, \dots, N_p\} \\
-\varepsilon_{speed} &\leq v_x(k|t) & , k \in \{1, \dots, N_p\}
\end{aligned} \tag{3.8d}$$

$$\Pr(p_x(k|t) \leq p_{max}(k|t) + q_p) \leq 1 - \varepsilon \quad , k \in \{1, \dots, N_p\} \tag{3.8e}$$

$$v_x(k|t) \leq v_{max}(k|t) + q_v \quad , k \in \{1, \dots, N_p\} \tag{3.8f}$$

where $\mathbf{x}_{lon}(k|t)$ denotes the state vector, which is consisted of of travel distance ($p_x(t)$), longitudinal speed ($v_x(t)$), and longitudinal acceleration ($a_x(t)$).

$\mathbf{x}_{lon(k).ref}$ is the vector of reference motions, which is travel distance ($p_{x.ref}(t)$) and speed ($v_{x.ref}(t)$). $\mathbf{Q}_{ref(k)}$ is the diagonal weight matrix for reference motion. $R_{u(k)}$ and $R_{\Delta u(k)}$ are the weights for control input and jerk, respectively. q_p and q_v are the slack variables for the travel distance and speed. Q_p and Q_v are the weights for the penalties. N_p is the maximum prediction step. The prediction horizon is 2 sec; 20 steps with a sampling time of 0.1 sec.

The following constraints were implemented for the MPC formulation: equality constraint for vehicle dynamics model as in Eq.(3.8b), inequality constraints of control input and change in control input as in Eq.(3.8c), inequality constraints of travel distance and speed as in Eq.(3.8d), Eq.(3.8e), and Eq.(3.8f). The parameters for the state and control input constraints are summarized in Table 3.4. The equality constraint in Eq.(3.8b) is the dynamics model as defined in Eq.(3.7). The lower bounds for the travel distance ($\varepsilon_{position}$) and speed (ε_{speed}) in Eq.(3.8d) were designed to obtain the future states with non-negative signs since the backward motion of the ego vehicle was disregarded. The upper bounds (p_{max}, v_{max}) in Eq.(3.8e) and Eq.(3.8f) were implemented for safe limits; Eq.(3.8e) is the probabilistic constraint (herein, chance-constraint) for describing the upper bound of travel distance with a probability of $(1-\varepsilon)$. Here, $\varepsilon \in (0,1)$ is the violation parameter. The detailed estimation of uncertainty distribution and reformulation of chance constraint are described in Section 3.5.2.4. $v_{max}(k|t)$ is a speed limit; The speed limit was determined according to road curvatures and regulatory speed limit. Since the speed limit was usually defined in spatial domain, it was transformed into time domain by using the predicted travel distance of the ego vehicle.

Table 3.4. Parameters for state and control input constraints for short time-horizon longitudinal motion planning

Parameters	Variable
Upper bound of control input, $a_{x.max}$ [m/s ²]	1.0
Lower bound of control input, $a_{x.min}$ [m/s ²]	-5.0
Bound of change in control input, S_u [m/s ³]	5
Bound of change in control input compared to previous control input, $S_{u.prev}$ [m/s ³]	5
Chance constraint parameter, ε [-]	0.1
Minimum clearance, c_0 [m]	3.0

3.5.2.2. Risk Index-based Fuzzy Inference System for Weight

Tuning

After investigating the driving data as summarized in Section 3.3.2, a promising way to consider the driving patterns in motion planning was devised. The combination of TTC^{-1} and ego vehicle's speed was taken into account in the suggested approach. The driving conditions in terms of safety were linguistically classified into three functions according to TTC^{-1} : "mild" under 0.26, "moderate" up to 0.45, and "dangerous" over 0.45. Likewise, the ego vehicle's speeds were sorted into three groups: "low" under 20 km/h, "medium" from 20 km/h to 40 km/h, and "high" over 40 km/h. They are values for membership functions of the fuzzy inference system.

For replicating the human driving patterns, a fuzzy inference system was

devised to obtain the weights of the multi-objective function. The fuzzy inference system of Takagi-Sugeno type was utilized with the triangular membership functions for fuzzy input and output. The fuzzy inference system adopted the risk index (TTC^{-1}) and ego vehicle's speed as converted into the linguistic variables, which are fuzzy inputs. The weights of the objective function were defined as output variables. They were converted into fuzzy output variables named as "small," "medium," and "large." The fuzzy rules were devised from the data analysis; the safety-related weights were proportional to TTC^{-1} . The weights for ride comfort were inversely proportional to TTC^{-1} . The weights for reference tracking performance were inversely proportional to the ego vehicle's speed and proportional to TTC^{-1} . From the fuzzy rules and inputs, the distribution of the weights is obtained as shown in Fig. 3.8.

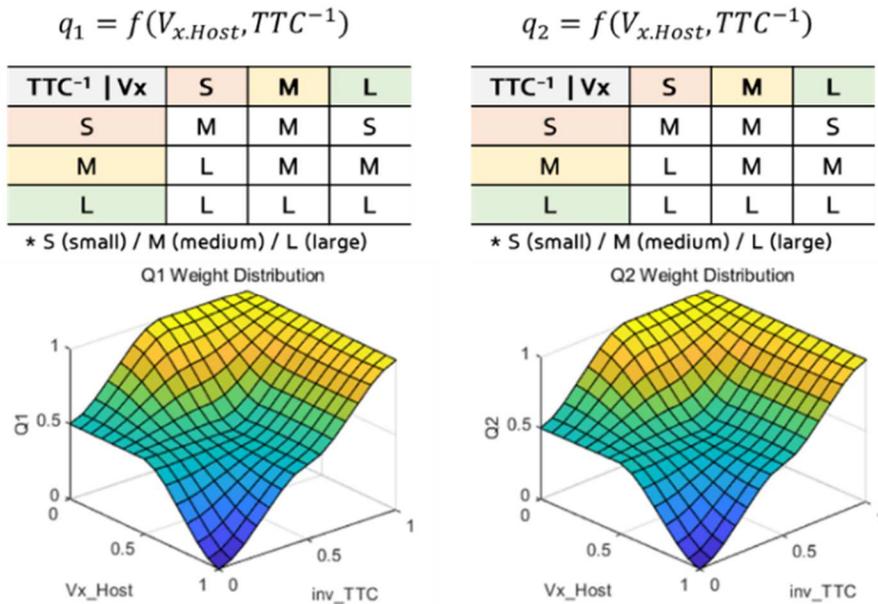


Fig. 3.8. Fuzzy rules and distribution of the weights for reference tracking cost of the short time-horizon longitudinal motion planning.

3.5.2.3. Stability Analysis of MPC under Change in System

Dynamics

Only a simple, first-order time-delay model was adopted as the system dynamics of short time-horizon motion planning. This assumption is a general approach when formulating a longitudinal motion planning algorithm. However, the system's characteristics may differ from the assumed time-delay model due to the changes in the vehicle's model parameter. Therefore, stability analysis was conducted for the MPC of the short time-horizon motion planning. In the analysis, the actual system's dynamics was assumed to be modified from those in the MPC formulation.

To analyze the stability of the system, the system dynamics in Eq.(3.6) was reformulated in terms of the error dynamics of the car-following problem, which is defined as follows:

$$\begin{aligned}
 \dot{\mathbf{x}}_{lon,e}(t) &= \mathbf{A}_{C,e}\mathbf{x}_{lon,e}(t) + \mathbf{B}_{C,e}u_{lon}(t) + \mathbf{w}(t) \\
 \text{s.t. } \mathbf{A}_{C,e} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau_d} \end{bmatrix}, \mathbf{B}_{C,e} = \begin{bmatrix} 0 \\ 0 \\ -\frac{1}{\tau_d} \end{bmatrix}, \mathbf{w}(t) = \begin{bmatrix} -\tau_g a_{x,p} \\ a_{x,p} \\ 0 \end{bmatrix} \\
 \mathbf{x}_{lon,e}(t) &= \begin{bmatrix} \Delta p_x(t) \\ \Delta v_x(t) \\ \Delta a_x(t) \end{bmatrix} = \begin{bmatrix} \left\{ p_{x,p}(t) - (c_0 + \tau_g v_{x,p}(t)) \right\} - p_{x,h}(t) \\ v_{x,p}(t) - v_{x,h}(t) \\ 0 - a_{x,h}(t) \end{bmatrix} \\
 u_{lon}(t) &= a_{x,des}(t)
 \end{aligned} \tag{3.9}$$

where $\mathbf{x}_{lon,e}(t)$ is the state vector. The state consists of the errors of travel distance, denoted as $\Delta p_x(t)$, longitudinal speed, denoted as $\Delta v_x(t)$, and longitudinal acceleration, denoted as $\Delta a_x(t)$. In Eq.(3.9), the error dynamics adopted the CTG

policy.

Eq.(3.9) is simplified by assuming the preceding vehicle's acceleration as zero and discretized using ZOH discretization approach as follows:

$$\mathbf{x}_{lon,e}(k+1|t) = \mathbf{A}_{D,e}\mathbf{x}_{lon,e}(k|t) + \mathbf{B}_{D,e}u_{lon}(k|t) \quad (3.10)$$

For the simplicity of the analysis, the finite time-horizon, state constraints, and control input constraints were not considered. Then, the MPC problem could be transformed into a discrete-time linear quadratic regulator (LQR) problem. The control gain of the discrete-time LQR was obtained by solving the discrete algebraic Riccati equation (DARE) as follows:

$$\begin{aligned} \mathbf{P}_\infty &= \mathbf{Q} + \mathbf{A}_{D,e}^T \mathbf{P}_\infty \mathbf{A}_{D,e} - \mathbf{A}_{D,e}^T \mathbf{P}_\infty \mathbf{B}_{D,e} (\mathbf{B}_{D,e}^T \mathbf{P}_\infty \mathbf{B}_{D,e} + \mathbf{R})^{-1} \mathbf{B}_{D,e}^T \mathbf{P}_\infty \mathbf{A}_{D,e} \\ \mathbf{K}_\infty &= (\mathbf{B}_{D,e}^T \mathbf{P}_\infty \mathbf{B}_{D,e} + \mathbf{R})^{-1} \mathbf{B}_{D,e}^T \mathbf{P}_\infty \mathbf{A}_{D,e} \end{aligned} \quad (3.11)$$

where \mathbf{P}_∞ is the positive definite, unique solution of the DARE.

Using the feedback control gain, denoted as \mathbf{K}_∞ , obtained from Eq.(3.11), the closed-loop system dynamics is rewritten as follows:

$$\begin{aligned} \mathbf{x}_{lon,e}(k+1|t) &= \mathbf{A}_{D,e}\mathbf{x}_{lon,e}(k|t) + \mathbf{B}_{D,e}u_{lon}(k|t) \\ &= (\mathbf{A}_{D,e} - \mathbf{B}_{D,e}\mathbf{K}_\infty)\mathbf{x}_{lon,e}(k|t) \end{aligned} \quad (3.12)$$

As shown in Eq.(3.12), the feedback control gain (\mathbf{K}_∞) affects the closed-loop

system's stability. The control gain could be calculated using the system dynamics and weights of the problem. For the analysis, the weights of the chance-constrained MPC were utilized; The weights, denoted as \mathbf{Q} and \mathbf{R} , were given by $\text{diag.}(40.0, 20.0, 0.0)$ and 40.0 . Then, the control gain (\mathbf{K}_∞) and the eigenvalues of the system were calculated as follows: $\mathbf{K}_\infty = [-0.9343 \quad -1.3146 \quad -2.1821]^T$ $\text{eig}(\mathbf{A}_{D,e} - \mathbf{B}_{D,e}\mathbf{K}_\infty) = 0.9373 \pm 0.0670i, 0.8944$. Since all eigenvalues are inside of the unit circle, the original closed-loop system is asymptotically stable.

Suppose that the system's time delay is modified to τ^* . Then, let us check that the closed-loop system is still asymptotically stable when implementing the control gain (\mathbf{K}_∞) which is calculated using the initial time delay, τ . Suppose that the delay of the actual system is 5.1 times greater than that of the initial assumption. Then, the eigenvalues are calculated as follows: $\text{eig}(\mathbf{A}_{D,e}^* - \mathbf{B}_{D,e}^*\mathbf{K}_\infty) = 0.9978 \pm 0.0651i, 0.9571$. This system is still stable because all eigenvalues are inside the unit circle. If the delay of the actual system becomes 5.2 times greater than that of the initial assumption, the eigenvalues are calculated as follows: $\text{eig}(\mathbf{A}_{D,e}^* - \mathbf{B}_{D,e}^*\mathbf{K}_\infty) = 0.9982 \pm 0.0646i, 0.9574$. Since the eigenvalues are located outside the unit circle, the system becomes unstable.

The analysis shows that the system's stability is lost if the system responds more slowly than that with the initially assumed delay. The dynamic characteristics of the longitudinal motion control and the test vehicle had been analyzed using system identification. Then, the most significant time delay of the system was chosen as the delay of the MPC parameter. Therefore, it can be confirmed that the stability of the proposed chance-constrained MPC for short time-horizon motion planning is guaranteed as long as the performance of the longitudinal motion controller is

preserved.

3.5.2.4. Design of Chance Constraints

The chance constraint in Eq.(3.8e) was adopted as the inequality constraint of travel distance for coping with the uncertainty from perception and localization modules. A chance constraint is a method for solving optimization problems under uncertainty. Among the inequality constraints in Eq.(3.8), the upper bound for travel distance is affected by the target's state. The targets considered in the longitudinal motion planning are composed of the targets from the perception module and targets which are virtually generated using stop-lines in HD map, as shown in Fig. 3.9. In the case of the virtual target, its state was affected by the uncertainty in ego vehicle's global position and heading from the localization module. As the position of the stop-lines is defined in the global reference frame, it is transformed into the ego vehicle's local coordinates. Then a stationary, virtual target is generated on the location. Therefore, the error of the virtual target's position comes from the localization module.

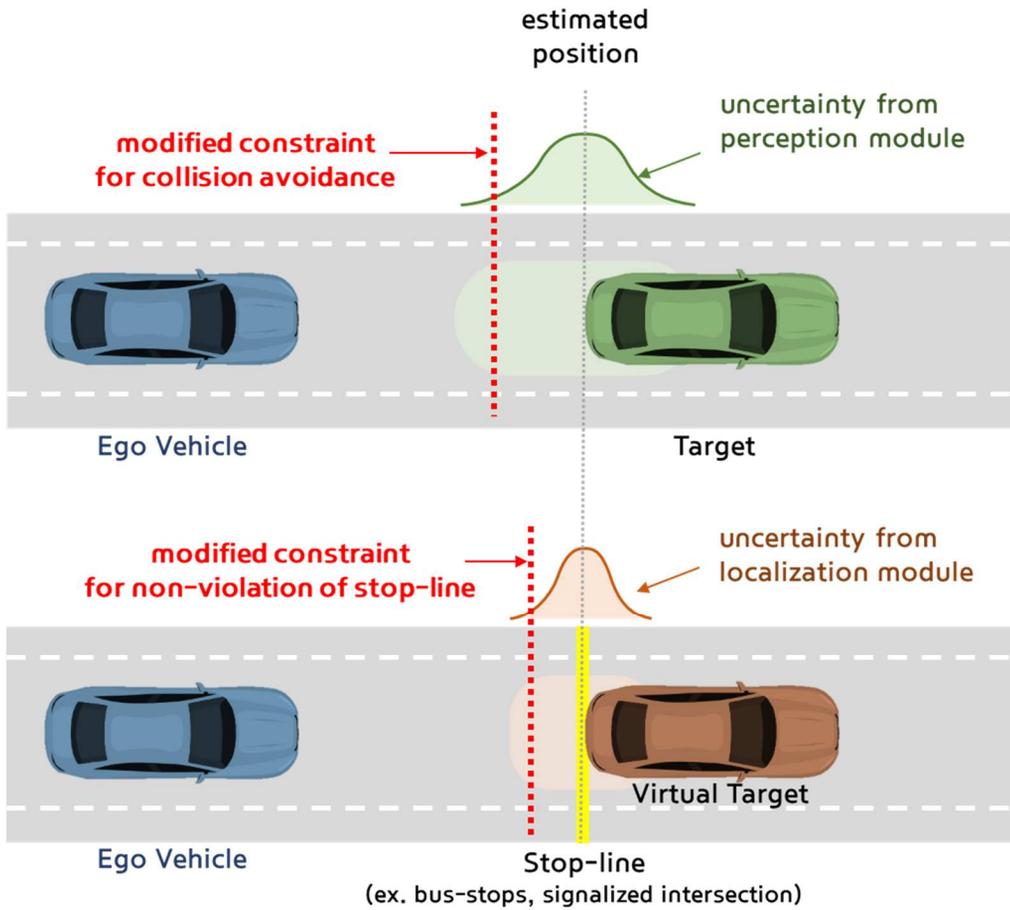


Fig. 3.9. Chance-constraint for uncertainty of position from perception and localization modules.

As the chance constraint is a stochastic form, it is hard to be implemented in the optimization problem. Therefore, the chance constraints are required to be reformulated into deterministic constraints. The reformulation method aims to determine a tightening parameter appropriately considering the uncertainty of the system. The tightening parameter, denoted as $\gamma(k|t)$, is a margin to provide a space to account for the uncertainty in the constraint.

Two main approaches are widely implemented [53]; analytic reformulation and

scenario-based reformulation methods. The analytic reformulation method assumes that the uncertainty can be defined using the uniform or Gaussian distribution. If the distribution of the uncertainty is unbiased and is normally distributed, then the chance constraint in Eq.(3.8e) can be analytically reformulated, which is described as follows:

$$\begin{aligned}
p_x(k|t) &\leq p_{max}(k|t) - \gamma(k|t), k \in \{1, \dots, N_p\} \\
\text{s. t. } \gamma(k|t) &= \sqrt{2\Sigma_k^e} \text{erf}^{-1}(1 - 2\varepsilon)
\end{aligned} \tag{3.13}$$

where, Σ_k^e is the covariance of the error (here, uncertainty) at kth step.

On the contrary, the scenario-based reformulation method does not restrict the distribution of uncertainty. Instead, it finds the tightening parameter with high confidence by taking a large number of samples. The scenario-based reformulation result is as follows [54].

$$\begin{aligned}
p_x(k|t) &\leq p_{max}(k|t) - \gamma_i(k|t), k \in \{1, \dots, N_p\}, i \in \{1, \dots, M\} \\
\text{s. t. } M &\geq \frac{2}{\varepsilon} \left(\ln \frac{1}{\beta} + n \right), \beta \in (0,1), u(k|t) \in R^n
\end{aligned} \tag{3.14}$$

where M denotes the number of samples that are independent and identically distributed, β means the confidence parameter.

To calculate the tightening parameter, denoted as $\gamma(k|t)$, by choosing the appropriate reformulation method, it is necessary to properly estimate the

distribution of uncertainty in advance. The uncertainties from the localization and perception modules were estimated using the vehicle test results. The tightening parameters for each target were calculated.

The uncertainty of the virtual target's state in terms of the travel distance was assumed to be same with that of the ego vehicle's longitudinal position with respect to local coordinates. As investigated in Section 0, the covariance of the longitudinal position was estimated using the Kalman filter of the localization module. Therefore, the covariance could be obtained in real-time. According to Fig. 4.5(a), it can be seen that the covariance depends on the sensor's state and remains almost constant when the sensor's state does not change. Therefore, it was assumed that the error covariance keeps its level of noise during the prediction horizon, which is described as follows.

$$\Sigma_k^e = \Sigma_{localization}^e(1|t), k \in \{1, \dots, N_p\} \quad (3.15)$$

In real vehicle applications, the covariance of the longitudinal position was obtained from the localization module in real-time. Intuitively, to check the level of uncertainty in terms of the tightening parameter, the vehicle test results in Fig. 4.5(a) were investigated. According to Fig. 4.5(a), the nominal values of covariance regarding the RTK GPS's status (RTK integer, single, and others) are observed as 0.0123, 0.8122, and 3.3062 m². The tightening parameters of these nominal values are calculated as 0.1418, 1.1550, and 2.3100 m, respectively. In the calculation, as the timestep adopted in the localization module was 0.04 sec, these values of covariance from the localization module were converted using a timestep of 0.1 sec,

which was the timestep of the motion planning module.

In the case of the perception module, the uncertainty in the target's state was represented by the errors in the target's clearance and speed. To analyze the distribution of the errors, the clearance and speed of the targets acquired from the perception module were compared to those acquired through V2V communication. Fig. 3.10 shows the clearance and speed of the target vehicle. As shown in Fig. 3.10(a) and Fig. 3.10(b), the black solid line and red dotted line indicate the target's state from the perception module and V2V communication, respectively. The collected data through V2V communication could be regarded as a ground truth value since the data was transmitted from the target vehicle to the ego vehicle with a low transmission delay. Fig. 3.11 shows the distribution of the errors for the estimated clearance and target's speed. As shown in Fig. 3.11(a), the error values can be regarded as normally distributed. By using the collected data, the covariance matrix was obtained assuming a bivariate normal distribution, as summarized in Eq.(3.16). Using the covariance matrix, the approximated probabilistic distribution function is plotted in Fig. 3.11(b).

$$\mathbf{w}(k|t) \sim N(0, \boldsymbol{\Sigma}_{\text{perception}}^e)$$

$$\boldsymbol{\Sigma}_{\text{perception}}^e = \begin{bmatrix} 0.2356 & 0.0600 & 0 \\ 0.0600 & 0.0570 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (3.16)$$

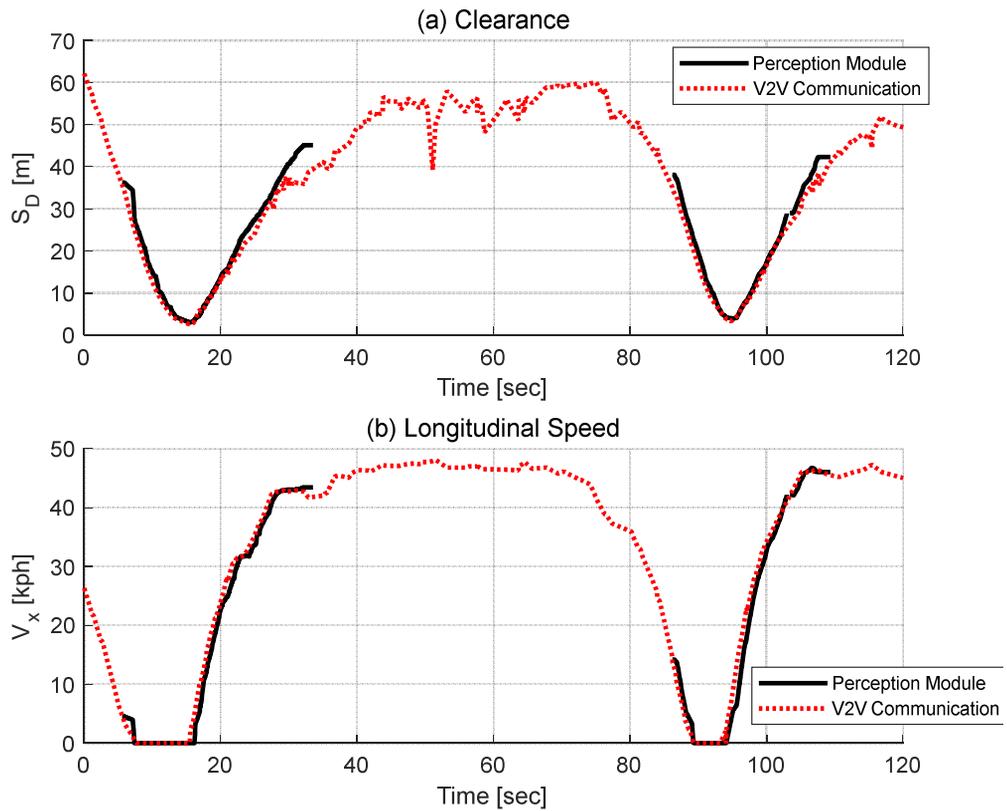


Fig. 3.10. Clearance and longitudinal speed of target vehicle acquired in vehicle test through both perception module and V2V communication.

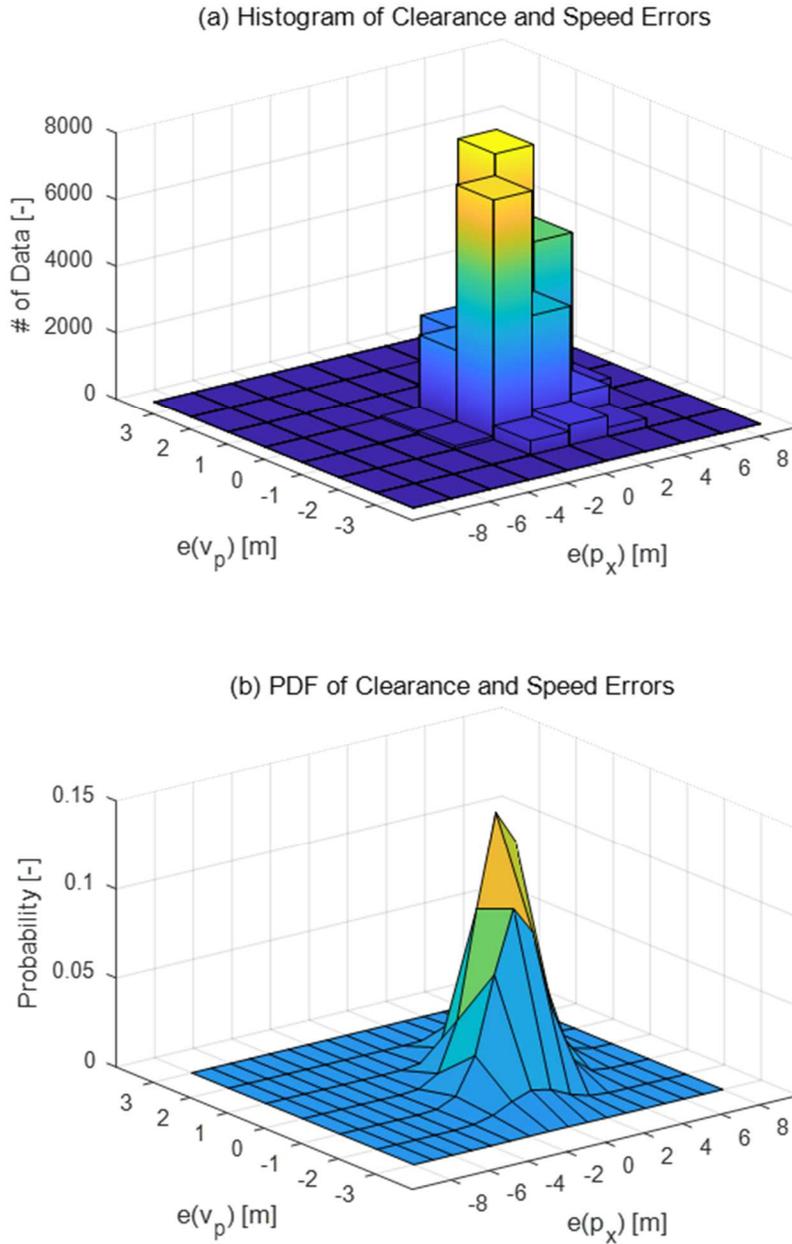


Fig. 3.11. Distribution of clearance and speed errors acquired in vehicle tests.

Before reformulating the probabilistic constraint in Eq.(3.8a) into the deterministic constraint, an error covariance matrix was calculated. Unlike the

localization module that provided the estimated covariance in real-time, the perception module did not provide nor estimate the covariance of the target's state. Therefore, referring to the system dynamics in Eq.(3.12) and the error covariance in Eq.(3.16) calculated with the aforementioned collected data, the error covariance was propagated over the prediction horizon as follows:

$$\begin{aligned} \Sigma_{k+1}^e &= (\mathbf{A}_{D,e} - \mathbf{B}_{D,e} \mathbf{K}_\infty) \Sigma_k^e (\mathbf{A}_{D,e} - \mathbf{B}_{D,e} \mathbf{K}_\infty)^T + \Sigma_w^e \\ &, k \in \{1, \dots, N_p - 1\} \end{aligned} \quad (3.17)$$

The tightening parameter over the prediction horizon could be obtained using the parameters adopted in Section 3.5.2.2, and the covariance matrix written in Eq.(3.16). Representatively, the tightening parameters for the first timestep, denoted as $\gamma(1|t)$, and that for the last timestep, denoted as $\gamma(20|t)$, were 0.6221 and 3.0058 m. With the result discussed for the localization module, it can be seen that the tightening parameters at the first timestep and that at the last timestep can be conservatively set as 2.3100 m and 3.0058 m, respectively.

3.5.2.5. Curvature-based Reference Speed Calculation

As depicted in Table 3.1, the optimization problem of longitudinal motion planning includes not violating the speed limit. The speed limit was used for constructing MPC for short time-horizon motion planning as a state constraint. The speed limit was determined from the regulatory speed limit and curvature-based speed from the lateral acceleration limit. The regulatory speed limit was calculated by the

information from HD map. The curvature-based speed was calculated from the limit of lateral acceleration and the curvature of the desired trajectory [55], which is defined as written in Eq.(3.18).

$$v_{x.curvature} = \sqrt{a_{y.limit}/\kappa} \quad (3.18)$$

where κ is the curvature of the road.

Fig. 3.12 shows the procedure for calculating the curvature-based speed profile. The curvature of the road could be calculated using the change in yaw rate of the given waypoints acquired from HD map. The waypoints may be given which makes it difficult for the automated vehicle to drive along. Also, the curvature calculated using the waypoints could contain noise. Therefore, the calculated curvature was smoothed using moving window filter and low-pass filter. Curvature-based speed was calculated using the smoothed curvature and lateral acceleration limit. After that, the change of the curve-based speed profile was corrected again using the longitudinal acceleration limit.



Fig. 3.12. Procedure for calculation of curvature-based speed profile.

Fig. 3.13 shows an example of the proposed curve-based speed calculation. The example was obtained by calculating the speed profile of the FMTC track, as depicted in Fig. 3.13(a). Fig. 3.13(b) shows the curvature calculated using the

waypoints of the track. Since the curvature could contain noise, a low pass filter was additionally applied. Fig. 3.13(c) shows the curvature-based speed profile. In Fig. 3.13(c), the blue solid line is the speed calculated from the curvature by applying the moving average and low pass filters. The red solid line indicates the speed by considering the longitudinal acceleration limit. From the result of Fig. 3.13, it can be seen that the curvature-based speed profile can be calculated considering the limit of lateral acceleration while attenuating the jerk.

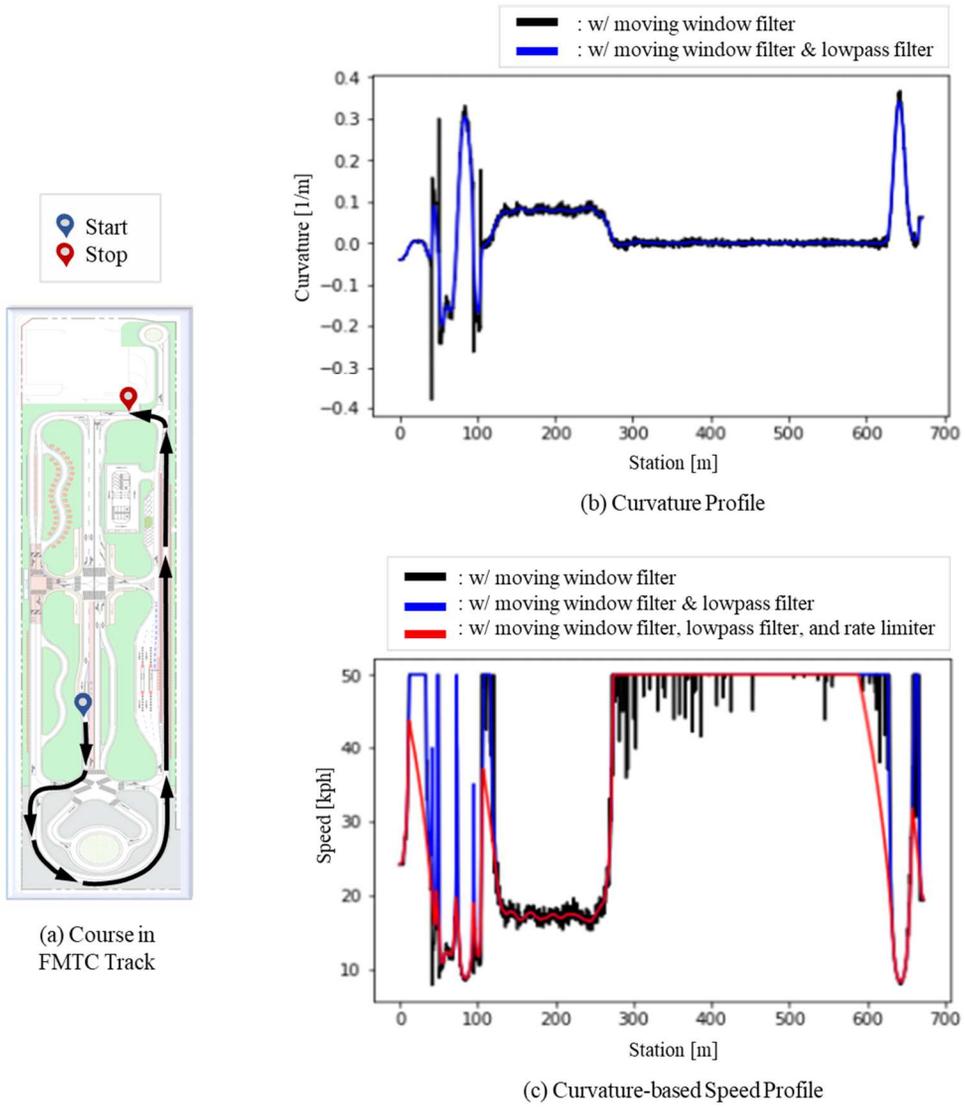


Fig. 3.13. Example of curvature-based speed profile.

3.6. Development of Longitudinal Motion Planning using DRL for Comparative Study

The pros and cons of DP and DRL algorithms when implementing for the long time-horizon motion planning are summarized in Table 3.5.

In the proposed longitudinal motion planning, DP was utilized as a tool for long time-horizon optimization. DP algorithm has a curse of dimensionality issue according to the increase in the dimension of a state; the computational cost is increased according to the number of states and time-horizon. On the contrary, the DRL algorithm is known as computationally efficient since it can handle a large number of states. Therefore, the proposed hierarchical longitudinal motion planning algorithm was compared with the DRL-based algorithm.

Table 3.5. Pros and cons of DP and DRL for long time-horizon motion planning.

	DP	DRL
Pros	<ul style="list-style-type: none"> Exact optimal solution Low pre-calculation cost, compared to DRL 	<ul style="list-style-type: none"> Determination of control input in continuous domain Easy adoption of policies
Cons	<ul style="list-style-type: none"> Curse of dimensionality Numerical error due to discretization 	<ul style="list-style-type: none"> Degradation of performance using state inputs that are outside the range of training data

Table 3.6 shows the structure of the algorithms developed using DP and DRL for comparative study. The newly designed longitudinal motion planning algorithm using DRL was developed with the same design purpose. Previous researches

utilized only DRL [16, 17]. However, it was hard for the algorithms to guarantee safety, which is one of the main objectives of this research. Therefore, as previously proposed, DRL and chance-constrained MPC were implemented as a hierarchical framework.

Table 3.6. Structure of algorithms using DP and DRL for comparative study.

	DP	DRL
Long time-horizon	<ul style="list-style-type: none"> Modeling w/o vehicle dynamics Dynamic programming 	<ul style="list-style-type: none"> Modeling w/o vehicle dynamics Deep reinforcement learning (using DDPG) Using rewards tuned by safety indices
Short time-horizon	<ul style="list-style-type: none"> Modeling w/ vehicle dynamics Chance-constrained MPC Objective function's weights tuned by Fuzzy inference system 	<ul style="list-style-type: none"> Modeling w/ vehicle dynamics Chance-constrained MPC

3.6.1. Long Time-horizon Motion Planning

3.6.1.1. Definition of System Dynamics Model

The long time-horizon motion planning was designed to obtain the optimized reference motion profile. The long time-horizon motion planning utilized the state vector with the following states: clearance, relative speed, ego vehicle's speed, and acceleration. Unlike DP, the DRL algorithm is less affected by the dimension of the state. Therefore, the system dynamics was modified by adding the state of the ego

vehicle. The state equation utilized the kinematic relationship of the ego vehicle's state, clearance, and relative speed, which is defined as follows:

$$\begin{aligned} \dot{\mathbf{x}}_{long}(t) &= \mathbf{A}_{long.C} \mathbf{x}_{long}(t) + \mathbf{B}_{long.C} u_{long}(t) \\ \text{s.t. } \mathbf{A}_{long.C} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \mathbf{B}_{long.C} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \\ \mathbf{x}_{long}(t) &= \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} p_{x.p}(t) - p_{x.h}(t) \\ v_{x.p}(t) - v_{x.h}(t) \\ v_{x.h}(t) \\ a_{x.h}(t) \end{bmatrix}, u_{long}(t) = \dot{a}_{x.h}(t) \end{aligned} \quad (3.19)$$

where the state vector is composed of clearance ($x_1(t) = p_{x.p}(t) - p_{x.h}(t)$), relative speed ($x_2(t) = v_{x.p}(t) - v_{x.h}(t)$), ego vehicle's speed ($x_3(t) = v_{x.h}(t)$), and ego vehicle's acceleration ($x_4(t) = a_{x.h}(t)$). The control input ($u_{x.long}(t)$) is defined by the ego vehicle's jerk, denoted as $\dot{a}_{x.h}(t)$. The state-space equation was formulated only using the kinematic relationship. For simplifying the problem, the speed of the preceding vehicle was assumed to be constant ($a_{x.p}(t) = 0$).

By using the Euler discretization approach, the continuous system in Eq.(3.19) is discretized as follows:

$$\mathbf{x}_{long}(k+1|t) = \mathbf{A}_{long,D}\mathbf{x}_{long}(k|t) + \mathbf{B}_{long,D}u_{long}(k|t)$$

$$\text{s.t. } \mathbf{A}_{long,D} = \begin{bmatrix} 1 & \Delta t & 0 & -\Delta t^2/2 \\ 0 & 1 & 0 & -\Delta t \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{B}_{long,D} = \begin{bmatrix} -\Delta t^3/6 \\ -\Delta t^2/2 \\ -\Delta t^2/2 \\ \Delta t \end{bmatrix} \quad (3.20)$$

where Δt is the time step.

3.6.1.2. Long Time-horizon Motion Planning using DRL

As summarized in Table 3.1, the optimization problem of the long time-horizon motion planning is aimed to optimize the traffic flow efficiency, ride comfort in terms of smoothness, and perceived safety. In addition, the planned motion was also required to avoid collision with the target vehicle. The reward function of the long time-horizon motion planning algorithm using DRL is formulated as shown in Eq.(3.21).

$$\begin{aligned} r(k|t) = & q_p \cdot (p_{x,des}(k|t) - p_{x,h}(k|t))^2 \\ & + q_v \cdot (v_{x,des}(k|t) - v_{x,h}(k|t))^2 \\ & + r_{a_x} \cdot (a_{x,h}(k|t))^2 + r_{\Delta a_x} \cdot (\dot{a}_{x,h}(k|t))^2 \\ & + q_{p_{min}} \cdot B (p_{x,p}(k|t) - p_{x,h}(k|t)) \end{aligned} \quad (3.21)$$

The definition and adopted parameters of the reward function for the proposed DRL algorithm are summarized in Table 3.7. After investigating the human driver's characteristics as described in Section 3.3.2, the range of weight for TG was

determined by referring to the bounds of the parameters.

Table 3.7. Parameters of reward function for DRL.

Parameter	Value
TG, τ_g	1.6 sec
Allowed minimum clearance, c_0	3.0 m
Threshold for barrier function of collision avoidance, ε	10^{-3}
Weight of collision avoidance, $q_{p_{min}}$	30
Weight of reference position, q_p	0.29
Weight of desired speed, q_v	0.54
Weight of longitudinal acceleration, r_{a0}	2.4
Weight of longitudinal jerk, $r_{\Delta a0}$	0.6
Weight of longitudinal speed, w_v [min, max, min. weight, max. weight]	[10 kph, 50 kph, 1.0, 5.0]
Weight of TG, w_{TG} [min, max, min. weight, max. weight]	[1.0 sec, 1.6 sec, 0.5, 1.0]
Weight of TTC, w_{TTC} [min, max, min. weight, max. weight]	[2.2 sec, 5.0 sec, 0.5, 1.0]

The first and second terms in Eq.(3.21) indicate the traffic flow efficiency, which is composed of the deviation from desired clearance and deviation from desired speed. The desired clearance and speed are defined as follows:

$$\begin{aligned}
p_{des}(k|t) &= c_0 + \tau_g v_{x.h}(k|t) \\
v_{des}(k|t) &= \min(v_{x.p}(k|t), v_{x.limit})
\end{aligned} \tag{3.22}$$

The third and fourth terms in Eq.(3.21) make the reference motion smooth for improving ride comfort considering longitudinal acceleration and jerk. The weights are defined as follows:

$$\begin{aligned}
r_{a_x}(k|t) &= r_{a0} \cdot w_v \cdot w_{TG} \cdot w_{TTC} \\
r_{\Delta a_x}(k|t) &= r_{\Delta a0} \cdot w_v \cdot w_{TG} \cdot w_{TTC} \\
w_v &= w_{v.min} + \Delta w_v \cdot \text{sigmoid} \left(10 \cdot \frac{v_{x.h} - v_{x.min}}{v_{x.max} - v_{x.min}} - 5 \right) \\
w_{TG} &= w_{TG.min} + \Delta w_{TG} \cdot \text{sigmoid} \left(10 \cdot \frac{t_{TG} - t_{TG.min}}{t_{TG.max} - t_{TG.min}} - 5 \right) \\
w_{TTC} &= w_{TTC.min} + \Delta w_{TTC} \\
&\quad \cdot \text{sigmoid} \left(10 \cdot \frac{t_{TTC} - t_{TTC.min}}{t_{TTC.max} - t_{TTC.min}} - 5 \right)
\end{aligned} \tag{3.23}$$

where w_v is the weight related to ego vehicle's speed, w_{TG} is the weight of TG, and w_{TTC} is the weight of TTC.

To mimic the human driving patterns, the weights in Eq.(3.23) adopted the perceived safety-related parameters, which are TG, TTC, and ego vehicle's speed. The minimum and maximum thresholds of the parameters were referred to from the data analysis. The policy of adjusting the weights was designed by referring to the tendency of human driving patterns; The weights related to perceived safety are

inversely proportional to TG and TTC. The weights for ride comfort are proportional to TTC and ego vehicle's speed. Here, if the weights for ride comfort are adjusted, the relative weights for collision avoidance and traffic flow efficiency are also tuned. Therefore, for convenience of calculation, the weights for ride comfort were only parameterized. Since the weights contained parameters, including clearance, relative speed, and ego vehicle's speed, the reward function became a nonlinear function. To make this reward function differentiable, the weight values were defined using the sigmoid function as described in Eq.(3.23).

The weights devised based on the human driving pattern affect the optimization performance and resultant reference motion. The effect can be recognized in terms of traffic flow efficiency, ride comfort, and perceived safety. First, the weight of the ego vehicle's speed in the ride comfort term can improve traffic flow efficiency at low speed. The terms of traffic flow efficiency contain a reference value that is proportional to the speed. In the conventional objective function, the terms of traffic flow efficiency are formulated as square terms. Suppose that the reference motion is obtained by using the objective function. If the speed of the ego vehicle is low, this reference motion value is relatively small. As a result, the reference motion is calculated so that traffic flow efficiency is less considered. However, in the proposed algorithm, the weight of the speed acts as normalizing the weights for traffic flow efficiency by the ego vehicle's speed. As a result, the quadratic cost function is modified as an absolute function. Consequently, the traffic flow efficiency at low speed can be improved.

Second, in terms of ride comfort, it can be expected that the ride comfort for a low-speed or stationary vehicle is improved in the approaching phase. Assume that

the ego vehicle is traveling at high speed. When recognizing a preceding vehicle traveling at a low speed, traffic flow efficiency is prioritized using the conventional quadratic cost function. As a result, the sharp profile of deceleration has to be calculated. As previously analyzed vehicle test results of the conventional algorithm, the deceleration profile differed from human behavior. In the same situation, the ride comfort prioritized-motion will be calculated by utilizing the proposed algorithm.

Finally, perceived safety can be improved by adopting the weights of TG and TTC. From the perspective of ride comfort, a human driver is careful of driving under conditions of small TG and TTC. This change in priority can improve the performance in response to the sudden acceleration/deceleration against the preceding vehicle in the car-following phase.

For the collision avoidance strategy of DRL, the penalty of violating the minimum clearance was included in the reward function. The penalty function, $B(x)$, is defined by referring to the logarithmic barrier function of convex optimization as follows:

$$B(x) = \log\left(\frac{\max(x, \varepsilon \cdot p_{min})}{p_{min}}\right) \cdot (x < p_{x.min}) \quad (3.24)$$

The logarithmic barrier function is differentiable and tends to negative infinity as the value goes to zero. To prevent the penalty value from becoming too small, a lower bound of the clearance input was set as $\varepsilon \cdot p_{x.min}$.

The structure of the DRL model was designed for optimizing the long time-horizon motion planning. The reinforcement learning (RL) agent interacted with an

environment and optimized its action from the obtained observation and reward. The action was defined by the jerk of the ego vehicle. The observation was defined by the clearance, relative speed, ego vehicle's speed, and acceleration, as described in Eq.(3.19). The ego and target vehicles were simulated using the simplified kinematics model in the simulation environment. The observation obtained in the simulation environment could be regarded as a ground truth value without uncertainty. The uncertainty from the real environments was covered in the short time-horizon motion planning.

RL algorithms are divided into value-based and policy-based methods. The value-based method measures the quality of the state or state-action pair. The action is chosen to maximize the expected future value from the given state. The policy-based method tries to directly optimize the policy by updating its parameters with the observed gradient of reward. A typical policy-based method is the deep deterministic policy gradient (DDPG). DDPG method is a model-free, off-policy method that utilizes an actor-critic agent. As the action of the DDPG agent is defined as a stochastic function, the method can be widely adopted for the control problems of a continuous system.

The pseudo-code of the DDPG algorithm with the defined simulation environment is described in Table 3.8, and the architecture of the actor and critic networks is described in Fig. 3.14 [56]. The algorithm starts by randomly initializing the actor, critic, and target networks of actor and critic. The agent cleans up the replay buffer data if necessary. At each episode, the initial states of the ego and target vehicles are randomly initialized for exploration. At each time step, the action is chosen according to the exploration policy. From the action, the simulation

environment updates its states. After that, the observation of the next states and the reward corresponding to the action are obtained. The pair of current states, action, and next states are stored in the replay buffer. The critic is trained with pre-defined mini-batches sampled from the replay buffer. Then, the actor is updated by using the policy gradient. Finally, the target critic and actor networks are slowly updated to track the critic and actor networks.

Table 3.8. Pseudo code of DDPG algorithm for long time-horizon motion planning.

Randomly initialize critic $Q(s, a \theta^Q)$ and actor $\mu(s \theta^\mu)$ networks with weights θ^Q and θ^μ .
Initialize target networks $Q'(s, a \theta^{Q'})$ and $\mu'(s \theta^{\mu'})$ networks with weights $\theta^{Q'} \leftarrow \theta^Q$ and $\theta^{\mu'} \leftarrow \theta^\mu$.
Setup replay buffer with a size of D.
for episode 1 to M, do
Begin with a random initial state of ego and target vehicles, process noise N for action exploration
Observe initial states of ACC: clearance, relative speed, and speed and acceleration of host vehicle
for t = 1 to T, do
Calculate reward r_t
Choose control input of change in acceleration (jerk) $a_t = \dot{a}_h = \mu(s_t, \theta^\mu) + N_t$
Implement the control input and transfer it to the new state, s_{t+1} , based on a kinematics-based ACC model
Save transition (s_t, a_t, r_t, s_{t+1}) into the replay buffer D
Sample random minibatch of N transitions (s_t, a_t, r_t, s_{t+1}) from D
Set $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1} \theta^{\mu'}) \theta^{Q'})$
Update critic through minimizing loss: $L = \frac{1}{N} \sum_i (y_i - Q(s_i, a_i \theta^Q))^2$
Update actor policy using sampled policy gradient: $\nabla_{\theta^\mu} J \approx \frac{1}{N} \sum_i \nabla_a Q(s, a \theta^Q) _{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu} \mu(s \theta^\mu) _{s_i}$
Update target networks: $\theta^{Q'} = \tau \theta^Q + (1 - \tau) \theta^{\mu'}$, $\theta^{\mu'} = \tau \theta^\mu + (1 - \tau) \theta^\mu$
end for
end for

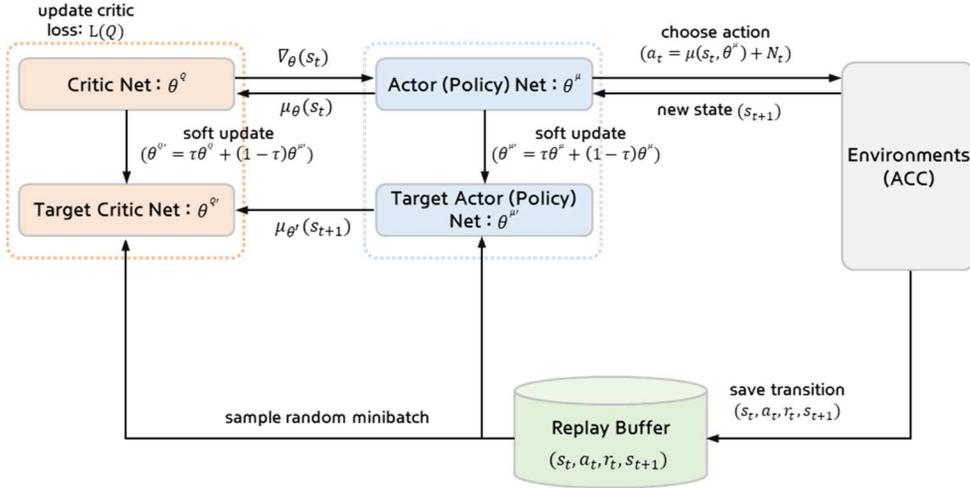


Fig. 3.14. The architecture of the actor and critic network for DDPG.

The actor and critic networks were designed by a neural network as shown in Fig. 3.15. The input of the actor network was the state at each time step. The jerk, which was defined as the control input, was applied to the ego vehicle at the next time step. The input of the critic network was the state and action pair, and its output was the expected cumulated reward. The actor network consisted of an input layer, output layer, and four hidden layers with 100 neurons. For the hidden layers, the rectified linear unit (ReLU) activation function was adopted. For the output layer, the hyperbolic tangent (a.k.a. tanh) activation function was used to map the real-valued data in the range of the jerk. Since the desired control input of acceleration in the longitudinal motion planning was designed to be bounded in the range from -5 m/s^2 to 1 m/s^2 , the jerk was also bounded in the range from -60 m/s^3 to 60 m/s^3 . The critic network consisted of an input layer for state, input layer for action, output layer, and three hidden layers. For the hidden layers, ReLU activation function was adopted. The structures of the networks were selected for acquiring the best performance. The training parameters of the DDPG method and environmental settings of the state and

action are listed in Table 3.9.

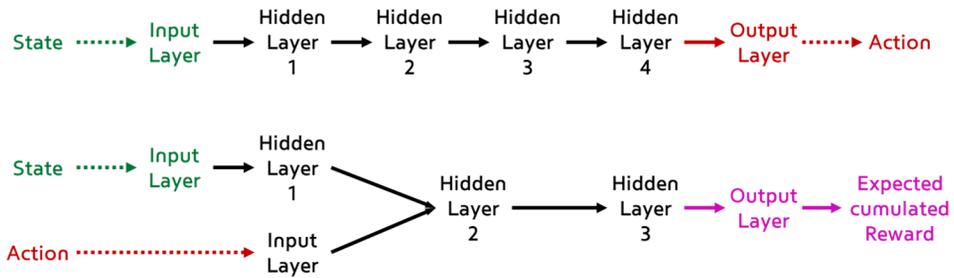


Fig. 3.15. The structure of the actor and critic networks for the proposed algorithm.

Table 3.9. Training parameters for the proposed agent of the DDPG method.

Parameter	Value
Target network update coefficient	0.001
Reward discount factor	0.999
Actor learning rate	0.00001
Critic learning rate	0.0001
Experience replay memory size	1000,000
Minibatch size	1024
Actor Gaussian noise mean	0
Actor Gaussian noise variance	0.05

Table 3.10. Parameters of the simulation environment and bound of state and

action space.

Parameter	Value
Episode steps	200
Simulation timestep	0.1 sec
Allowed maximum control input, u_{max}	60 m/s ³
Allowed minimum control input, u_{min}	-60 m/s ³
Allowed minimum clearance, c_{min}	0.0 m
maximum clearance, c_{max}	100.0 m
Allowed maximum speed of host vehicle, $v_{h.max}$	50 kph
Allowed maximum speed of target vehicle, $v_{p.max}$	50 kph

3.6.2. Short Time-horizon Motion Planning

The following two features are the differences between the longitudinal motion planning algorithms using DP and DRL. First, which part of the motion planning algorithm includes the weight tuning law. In the case of the algorithm using DP, the weight tuning law was implemented in the short time-horizon motion planning. However, the algorithm using DRL was designed so that the long time-horizon motion planning was in charge of the weight tuning law. The short time-horizon motion planning only played the role of a safety filter; Therefore, the motion planning implemented chance-constrained MPC, which adopted constant weight for the objective function.

Second, the way of utilizing the reference motion to the short time-horizon

motion planning is different. In the case of the algorithm using DP, different reference motions were used according to each driving mode: approaching, transient, and car-following mode. The main reason for adopting the driving modes was the number of states used in the DP algorithm. Due to the curse of dimensionality, the DP algorithm adopted three states: errors of desired clearance, speed, and acceleration. On the other hand, DRL adopted four states. Ego vehicle's speed was added to the state. In addition, the DRL algorithm implemented a more complex reward function, including indices of TG, TTC, and ego vehicle's speed. As the reference motion could be represented using more states, the modes of the reference motion, which were approaching, transient, and car-following modes, could be integrated into one algorithm. This integrated algorithm allowed for straightforward and optimized algorithm to be implemented.

Chapter 4. Lateral Motion Planning

4.1. Problem Definition

Overall motion planning algorithm for the automated bus was designed to achieve multiple objectives: traffic flow efficiency, ride comfort, perceived safety, and traffic rules, as shown in Table 3.1. Among them, the lateral motion planning algorithm was designed to consider ride comfort considering curves while keeping the traffic rule regarding lane geometry. Since these objectives are related to the geometrical information from HD map, the performance is significantly affected by the localization uncertainty. For example, the traffic rule related to lane geometry makes an autonomous vehicle follow the desired path while keeping a minimal distance to lane boundaries. Meanwhile, the lateral motion planning algorithm also tries to minimize the lateral acceleration, jerk, and yaw rate for ensuring ride comfort. To achieve the goals of the lateral motion planning algorithm, the algorithm determines the desired motion in the following way [21, 57]:

- 1) Determine the desired path considering the drivable corridor and the reference path which is initially guessed.
- 2) Calculate the desired motion to follow the path while ensuring ride comfort.

Fig. 4.1 shows the definition of reference path and path tracking error from the lane centerline. The reference path was assumed to be the centerline of the road, or calculated by referring to the left and right lanes and road boundaries recognized from sensors and given by the information from HD map. The drivable corridor was calculated by considering the present and expected locations of the static obstacles and moving targets within the ROI. From the desired path by considering the

acquired drivable corridor and reference path, the desired motion, which is the desired yaw rate, was obtained from the current states of the ego vehicle, and heading angle and lateral position errors from the desired path. If there are no obstacles, the path is regarded as the desired path. The nearest point (G_S) in the reference path from the center of mass (G) of the vehicle is calculated. The lateral distance between the nearest point (G_S) and center of mass (G) is defined as the lateral distance error, denoted as e_y . The angular difference between the tangential line at the point and the heading angle of the ego vehicle is defined as the heading error, denoted as e_ψ .

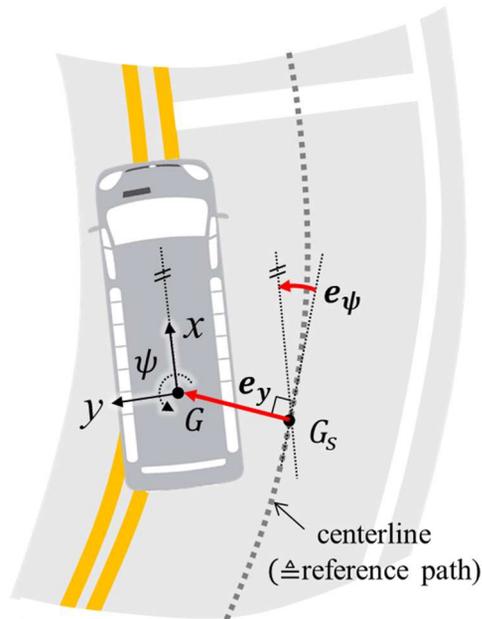


Fig. 4.1. Definition of path tracking error about lane centerline.

4.2. Outline of Lateral Motion Planning

This section presents the lateral motion planning algorithm for coping with localization uncertainty for an automated bus. Since the test vehicle was driven along the narrow road, infeasible motion planning could be acquired when implementing the existing approach of robust MPC using uncertainty propagation. Consequently, localization uncertainty-aware lateral motion planning logic is necessary. Here, the concepts of offset-free MPC, which were suggested in the previous researches [62, 63], was implemented and designed by compensating for the bias of the heading angle through a disturbance estimation. Since the lateral position error was less significant than the heading error, the bias of the heading angle was selected as the disturbance. However, the interference with the obstacles on the side was important for motion planning. Therefore, desired path planning was implemented through the drivable corridor as suggested in the previous researches [57, 21]. Longitudinal position error was considered in the chance constraints, which was already suggested in the longitudinal motion planning algorithm. After covariance of estimated longitudinal position in localization module was considered in chance constraint, the motion planning for stop lines was taken into account concerning the expected stopping position defined in HD map.

The proposed lateral motion planning module consists of three subsections: drivable corridor determination, disturbance estimation using MHE, and desired motion optimization using offset-free MPC, as shown in Fig. 4.2. Determination of drivable corridor was conducted for the desired path. The drivable corridor was calculated using lane boundary acquired from HD map and target states acquired from the perception module. Disturbance estimation was carried out using MHE.

MHE is the optimization-based estimation technique using the measured data acquired in a moving time-horizon. The system model for the MHE was constructed by augmenting the vehicle's lateral dynamics model and disturbance model. The disturbance model adopted the bias of the heading error, denoted as δe_ψ , as the disturbance value. The estimated values of the state (with the bias (δe_ψ) removed) were used for the present state of the ego vehicle in the desired motion optimization. The desired motion optimization logic using offset-free MPC determined the optimal desired motion, which is the yaw rate. The desired yaw rate was determined considering estimated states, desired path, state constraints, and control input constraints.

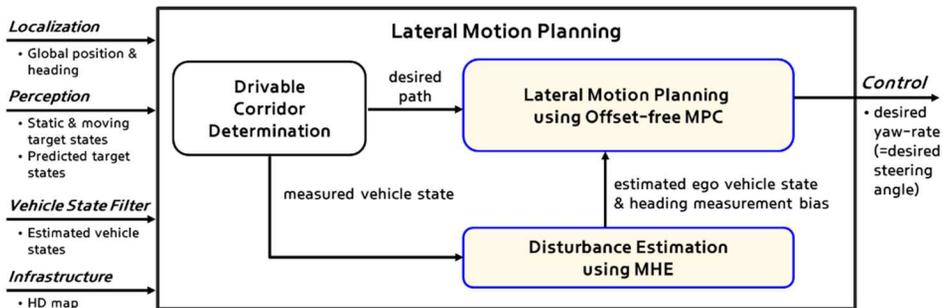


Fig. 4.2. Outline of the proposed lateral motion planning algorithm.

4.3. Examination of Localization Uncertainty in Urban Environments

Autonomous driving algorithms utilize the information acquired from HD map. The detection range of existing off-the-shelf sensors is shorter than that of human drivers. In addition, HD map includes additional information which cannot be recognized using onboard sensors. Therefore, the availability of the HD map increases for the advanced autonomous driving algorithm. Especially, most of the motion planning approaches extensively utilize the reference values in HD map. For example, the centerline, left lane, and right lane are utilized for lateral motion planning. Stop lines at bus stops and intersections are used for longitudinal motion planning. The reference values are transformed from the global reference frame into the body-fixed reference frame, referring to the position and orientation of the ego vehicle through the localization module.

Most of the motion planning approaches assumed that the exact position and orientation of the ego vehicle were obtainable, and localization uncertainty was negligible [11, 12, 24]. However, in reality, the localization uncertainty affects the accuracy of reference values acquired from HD map. Therefore, it is necessary to investigate the characteristics of localization uncertainty and the effects of the uncertainty on motion planning to develop a safe lateral motion planning algorithm.

To investigate the characteristics of localization uncertainty in an urban environment, the localization results and measured sensor data were collected at the vehicle test in BRT course, Sejong City. The test vehicle was driven manually. Fig. 4.3 shows the lane width of the BRT course. As shown in Fig. 4.3(a), the test vehicle is a full-sized public bus. The gap between the lanes and the sidewalls of the vehicle

was small; As shown in Fig. 4.3(b), the allowable distances between the lane and the test vehicle are 0.2 m on average. One of the primary measurements for the localization module is the lanes captured from AVM. Thus, the bus driver drove the vehicle carefully to maintain the vehicle within the lane during the test.

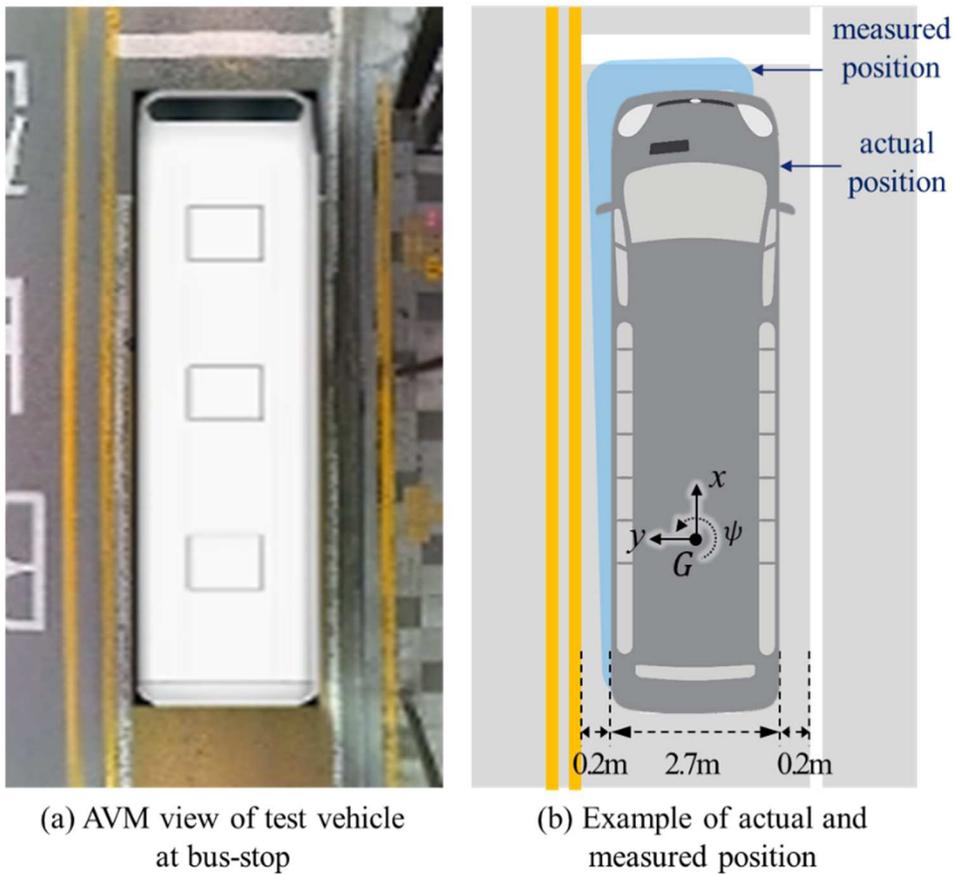


Fig. 4.3. Lane width of BRT course and width of the test vehicle.

Since this course contained GPS-denied environments, the measured data from RTK GPS could not be assumed to be the ground truth. Accordingly, the relative errors between the estimated value from the localization module and the measured value from RTK GPS were compared using the data acquired in the test. In addition,

the tendency of the deviation estimated by the localization module was compared with the results of referred researches.

Fig. 4.4 shows the GPS status and the relative errors between the measured value of RTK GPS and the estimated value from the localization module. As shown in Fig. 2.5, the test course includes three underpasses. The GPS status changed near the underpasses. As shown in Fig. 4.4(a), the GPS status deteriorates at stations of 780 m, 1600 m, and 3700 m, near the entrance of the underpasses. On the other hand, the recoveries of GPS status were observed at slightly different locations for all three driving tests. Fig. 4.4(b), Fig. 4.4(c), and Fig. 4.4(d) show the relative errors in a longitudinal position, lateral position, and heading angle. The red dotted lines in the figures indicate the validation gate of the first driving test. The localization module utilized the collected data as a measurement when the value was within the validation gate. Here, the validation gate was set to be the three-sigma region around the predicted observation. As shown in Fig. 4.4(b), Fig. 4.4(c), and Fig. 4.4(d), the increases in relative errors near the underpasses were similar, but the magnitude and sign of the errors are inconsistent. Among them, the errors of longitudinal and lateral positions are deviated from the bound, as shown in Fig. 4.4(b) and Fig. 4.4(c). On the contrary, the error of heading angle is within the bound most of the time, as shown in Fig. 4.4(d). As a result, the data out of bounds among the measured longitudinal and lateral positions from RTK GPS were not used as a measurement. However, most of the heading angle data were adopted for measurement.

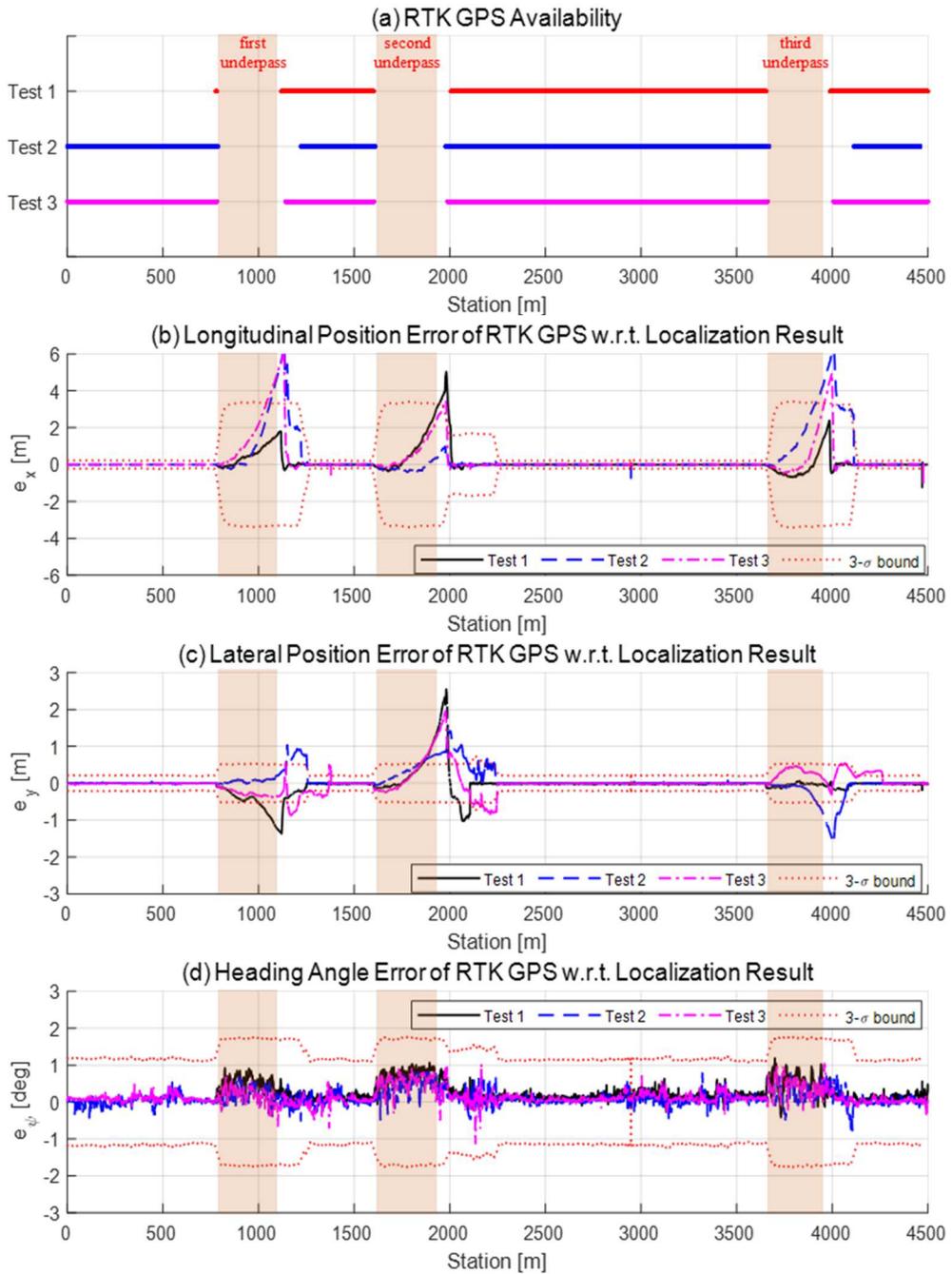


Fig. 4.4. GPS status and relative errors between the measured value of RTK GPS and estimated value of localization module.

Fig. 4.5 shows the deviation estimated by the localization module at the vehicle test. As depicted in Fig. 4.5(a) and Fig. 4.5(b), the maximum deviations of the longitudinal and lateral position errors are observed as 1.15 m at station of 1,822 m and 0.24 m at station of 2,138 m, respectively. As shown in Fig. 4.5(c), the maximum deviation of the heading angle is 0.59 deg at station of 1916 m. Similar to the specified values in Fig. 4.5, the maximum errors of lateral position and heading angle were reported as 0.13 m and 1.05 deg in the referred research [58].

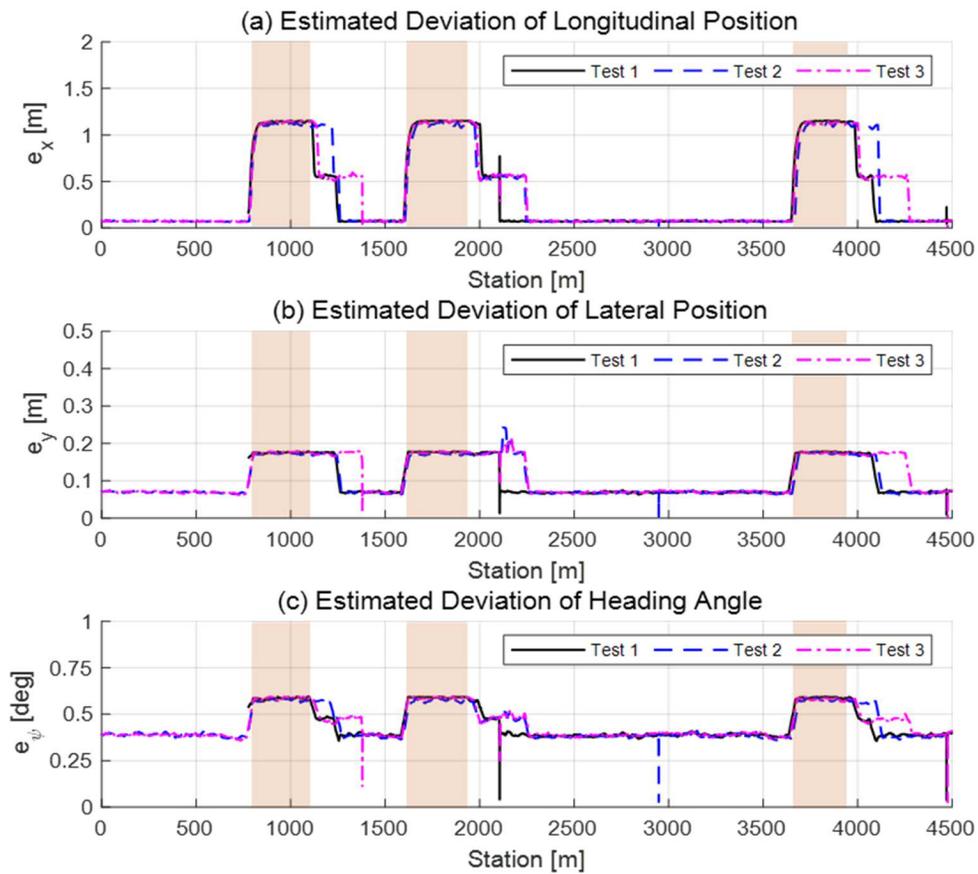


Fig. 4.5. Estimated deviation of longitudinal position, lateral position, and heading angle from localization module.

From the test results in Fig. 4.4(b), Fig. 4.4(c), Fig. 4.5(a), and Fig. 4.5(b), it can be seen that the lateral position error is smaller than the longitudinal position error. The tendency of the lateral position error is in line with the observation in the previous researches [59, 58]; The lateral position was credible since measurements for the lateral position including left/right lanes, walls of surrounding buildings, and walls of underpasses were frequently provided using the map matching-based localization methods. On the contrary, the longitudinal position was unreliable in GPS-denied environments, because the primary measurement for the longitudinal position, which was the RTK GPS values within the validation gate, was not frequently updated.

From the test results in Fig. 4.4(d) and Fig. 4.5(c), it can be seen that the bias of the heading angle is significantly observed at specific locations and under specific sensor status. In autonomous driving, the bias of heading angle could cause the lateral position error. Ge et al. pointed out that the disturbance of the heading angle bias of 1 deg could cause the lateral position error up to 0.35 m [27]. As shown in Fig. 4.5(c), the maximum standard deviation of the heading angle was 0.59 deg. The lateral position error caused by the heading angle bias can affect the performance of the localization module significantly. As shown in Fig. 4.2 (b), the gap between the lane and the sidewall of the vehicle is small; Suppose that there is a bias in the heading angle. The lateral position error becomes biased due to the heading angle error. Then, it may be difficult to recognize the lanes from AVM. Consequently, the heading uncertainty can result in danger during the autonomous driving of the bus.

From the vehicle test results, the high urgency that the motion planning algorithm should compensate for the heading angle bias was recognized. After that,

the motion planning algorithm should consider the longitudinal and lateral position errors in real-time.

4.4. Disturbance Estimation

4.4.1. Definition of System Dynamics Model

The vehicle lateral dynamic model was designed by combining a bicycle model and error dynamics equations along with a central path of the lane [60]. The bicycle model is applicable not only for low-speed driving but also for driving with wheel slip [60]. Newton-Euler equations of motion of a vehicle are decomposed into the body-fixed coordinate yields:

$$\begin{aligned} ma_y &= F_{yf} + F_{yr} \\ I_z \dot{\gamma} &= l_f F_{yf} - l_r F_{yr} \end{aligned} \quad (4.1)$$

where m is the mass of the vehicle, I_z is the mass moment of inertia, a_y is the lateral acceleration of the vehicle with respect to the inertial reference frame decomposed into a body-fixed coordinate, F_{yf} and F_{yr} are the lateral tire forces from front and rear tires, and l_f and l_r are the longitudinal distance from the center of mass to front and rear tires. The lateral acceleration of the vehicle (a_y) with respect to the inertial reference frame decomposed into body-fixed coordinate can be expressed in terms of the lateral acceleration and yaw rate with respect to the body-fixed reference frame under small angle assumption, as follows:

$$a_y = \dot{V}_y + V_x \dot{\psi} = V_x (\dot{\beta} + \dot{\gamma}) \quad (4.2)$$

The lateral tire forces from the front and rear tires can be approximated as a linear relationship with the angle of the velocity vector in mild driving conditions,

as follows:

$$\begin{aligned} F_{yf} &= 2C_f(\delta - \theta_{Vf}) \\ F_{yr} &= 2C_r(-\theta_{Vr}) \end{aligned} \quad (4.3)$$

where C_f and C_r are the cornering stiffness of the front and rear tires. δ is the front wheel steering angle. θ_{Vf} and θ_{Vr} are the body side slip angles at front and rear tires.

Using kinematic relationships and small-angle assumptions, the relationships between the body side slip angles (at the center of mass (β)) and the front (θ_{Vf}) and rear tires (θ_{Vr}) and vehicle state variables are simplified as follows:

$$\begin{aligned} \beta &= \tan^{-1}(V_y/V_x) \simeq V_y/V_x = \dot{e}_y/V_x - e_\psi \\ \theta_{Vf} &= \tan^{-1}\{(V_y + l_f\dot{\psi})/V_x\} \simeq (V_y - l_r\dot{\psi})/V_x = \beta + l_f\gamma/V_x \\ \theta_{Vr} &= \tan^{-1}\{(V_y - l_r\dot{\psi})/V_x\} \simeq (V_y - l_r\dot{\psi})/V_x = \beta - l_r\gamma/V_x \end{aligned} \quad (4.4)$$

By substituting Eq.(4.2), Eq.(4.3), and Eq.(4.4) into Eq.(4.1), the lateral vehicle dynamics model is simplified as follows:

$$\begin{aligned} \dot{\beta} &= \left(-\frac{2C_f + 2C_r}{mV_x}\right)\beta + \left(-1 + \frac{-2C_f l_f + 2C_r l_r}{mV_x^2}\right)\gamma + \frac{2C_f}{mV_x}\delta \\ \dot{\gamma} &= \left(\frac{-2C_f l_f + 2C_r l_r}{I_z}\right)\beta + \left(-\frac{2C_f l_f^2 + 2C_r l_r^2}{I_z V_x}\right)\gamma + \frac{2C_f l_f}{I_z}\delta \end{aligned} \quad (4.5)$$

As shown in Fig. 4.1, the path tracking problem is defined using terms of the

lateral distance error (e_y) and orientation error (e_ψ) with respect to the reference path. With an assumption that the radius (R_{path}) of the reference path is large, the error dynamics equation can be written as follows:

$$\begin{aligned}
e_y &= y - y_{des} \\
\dot{e}_y &= \dot{y} + V_x(\psi - \psi_{des}) = V_y + V_x e_\psi = V_x \beta + V_x e_\psi \\
e_\psi &= \psi - \psi_{des} \\
\dot{e}_\psi &= \dot{\psi} - \dot{\psi}_{des} = \gamma - V_x/R_{path} = \gamma - V_x \cdot \rho_{path}
\end{aligned} \tag{4.6}$$

Consequently, from Eq.(4.5) and Eq.(4.6), the lateral vehicle dynamics equation is written as follows:

$$\begin{aligned}
\dot{\beta} &= \left(-\frac{2C_f + 2C_r}{mV_x} \right) \beta + \left(-1 + \frac{-2C_f \ell_f + 2C_r \ell_r}{mV_x^2} \right) \gamma + \frac{2C_f}{mV_x} \delta \\
\dot{\gamma} &= \left(\frac{-2C_f \ell_f + 2C_r \ell_r}{I_z} \right) \beta + \left(-\frac{2C_f \ell_f^2 + 2C_r \ell_r^2}{I_z V_x} \right) \gamma + \frac{2C_f \ell_f}{I_z} \delta \\
\dot{e}_\psi &= \gamma - V_x \cdot \rho_{path} \\
\dot{e}_y &= V_x \beta + V_x e_\psi
\end{aligned} \tag{4.7}$$

4.4.2. Disturbance Model

The disturbance estimation model was constructed to estimate the lateral states and the disturbance of an ego vehicle: the slip angle, yaw rate, heading angle error, lateral distance error, and bias in heading angle error. The proposed concept of the algorithm

utilizing the offset-free MPC and disturbance estimation is shown in Fig. 4.6. As shown in Fig. 4.6(a), the positive heading angle error can be misestimated although there are no errors in both lateral position and heading angle. To reduce the misestimated error, the control input is calculated to steer the vehicle toward a clockwise direction, as shown in Fig. 4.6(b). As the control input is determined to minimize both the lateral position error and heading angle error, the control input results in the negative lateral position error. Because the signs of the two errors are different, lateral position and heading errors are not converged into zero but into values with bias. Fig. 4.6(c) shows the concept of the proposed algorithm using offset-free MPC. First of all, initially estimated heading error, denoted as $e_{\psi.measured}$, is assumed as a value bias. Then, the error is updated using the disturbance estimation. When performing optimization of MPC, the estimated heading angle without bias is taken into account as a state of the vehicle. If the error can be estimated with accuracy, the control input acquired by MPC becomes zero. Finally, the errors of lateral position and heading angle become zero by attenuating the estimated bias.

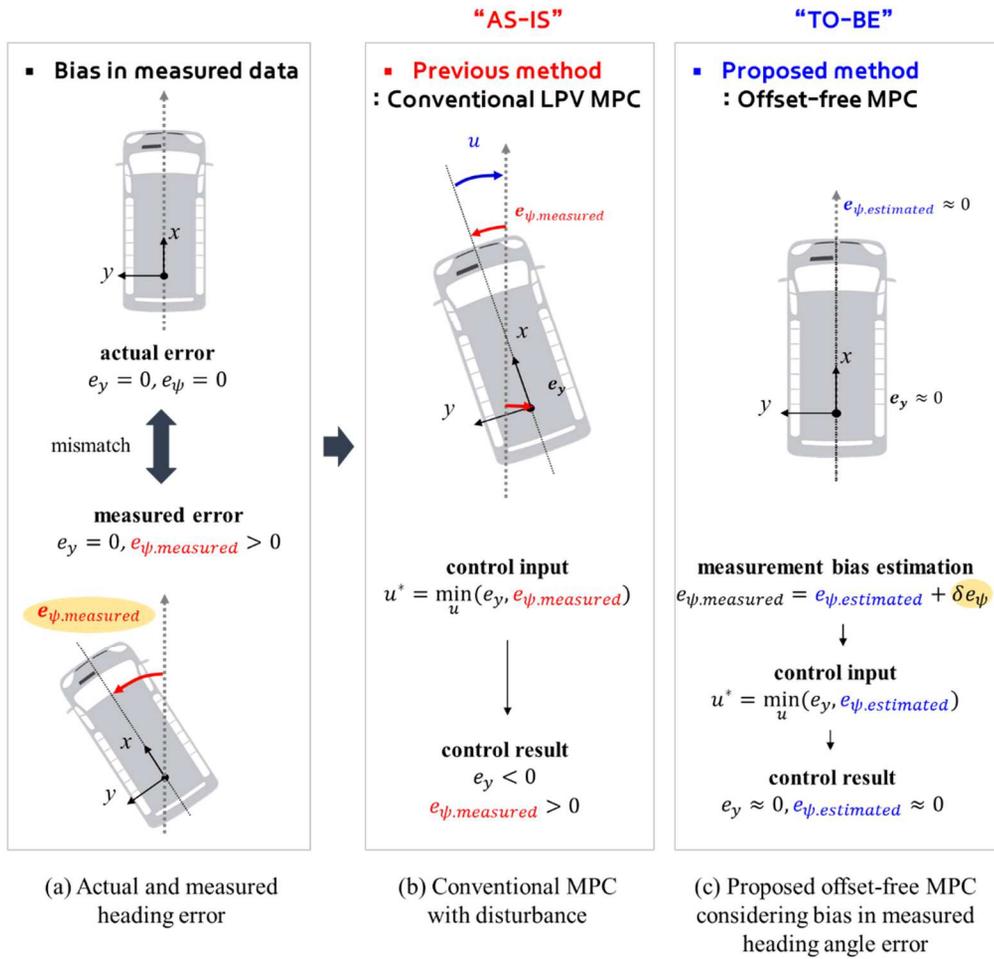


Fig. 4.6. Overview of the proposed offset-free MPC for lateral motion planning.

The state equation of the lateral vehicle dynamics can be represented as follows:

$$\begin{aligned}
\dot{\mathbf{x}}_{lat}(t) &= \mathbf{A}_{lat.c}(t)\mathbf{x}_{lat}(t) + \mathbf{B}_{lat.c}(t)u_{lat}(t) + \mathbf{F}_{lat.c}(t)\rho_{path}(t) \\
\text{s.t. } \mathbf{A}_{lat.c}(t) &= \begin{bmatrix} -\frac{2C_f+2C_r}{mV_x(t)} & -1 + \frac{-2C_f\ell_f+2C_r\ell_r}{mV_x^2(t)} & 0 & 0 \\ -\frac{2C_f\ell_f+2C_r\ell_r}{I_z} & -\frac{2C_f\ell_f^2+2C_r\ell_r^2}{I_zV_x(t)} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ V_x(t) & 0 & V_x(t) & 0 \end{bmatrix}, \\
\mathbf{B}_{lat}(t) &= \begin{bmatrix} \frac{2C_f}{mV_x(t)} \\ \frac{2C_f\ell_f}{I_z} \\ 0 \\ 0 \end{bmatrix}, \mathbf{F}_{lat}(t) = \begin{bmatrix} 0 \\ 0 \\ -V_x(t) \\ 0 \end{bmatrix} \\
\mathbf{x}_{lat}(t) &= [\beta(t) \ \gamma(t) \ e_\psi(t) \ e_y(t)]^T \\
u_{lat}(t) &= \delta(t)
\end{aligned} \tag{4.8}$$

where, $\mathbf{x}_{lat}(t)$ is the state vector. The state is composed of side slip angle, denoted as β , yaw rate, denoted as γ , orientation error, denoted as e_ψ , and lateral distance error, denoted as e_y . $u_{lat}(t)$ is the control input, which is the front wheel steering angle. $\rho_{path}(t)$ is the curvature of the reference road.

Since the state equation of the continuous system in Eq.(4.8) is a function of longitudinal speed (V_x), the system is originally a time-varying system. By using ZOH discretization approach, the state equation of the continuous system is discretized with a fixed sampling time (Δt) as follows:

$$\begin{aligned}
\mathbf{x}_{lat}(k+1|t) &= \mathbf{A}_{lat}(k|t)\mathbf{x}_{lat}(k|t) + \mathbf{B}_{lat}(k|t)u_{lat}(k|t) \\
&\quad + \mathbf{F}_{lat}(k|t)\rho_{path}(k|t) \\
\text{s.t. } \mathbf{A}_{lat}(k|t) &= e^{\mathbf{A}_{lat}(t)\cdot\Delta t} \\
\mathbf{B}_{lat}(k|t) &= \int_{\tau=0}^{\Delta t} e^{\mathbf{A}_{lat}(t)\cdot\tau}\mathbf{B}_{lat}(t)d\tau = \mathbf{A}_{lat}(t)^{-1}(e^{\mathbf{A}_{lat}(t)\cdot\Delta t} - I)\mathbf{B}_{lat}(t) \\
\mathbf{F}_{lat}(k|t) &= \mathbf{A}_{lat}(t)^{-1}(e^{\mathbf{A}_{lat}(t)\cdot\Delta t} - I)\mathbf{F}_{lat}(t)
\end{aligned} \tag{4.9}$$

4.4.3. Disturbance Estimation using MHE

4.4.3.1. Augmented System Equation

The state-space and measurement model of the discrete-time, linear time-varying state-space equation is as follows:

$$\begin{aligned}
\mathbf{x}_{lat}(k+1|t) &= \mathbf{A}_{lat}(k|t)\mathbf{x}_{lat}(k|t) + \mathbf{B}_{lat}(k|t)u_{lat}(k|t) \\
&\quad + \mathbf{F}_{lat}(k|t)\rho_{path}(k|t) \\
\mathbf{y}_{lat}(k|t) &= \mathbf{C}_{lat}\mathbf{x}_{lat}(k|t)
\end{aligned} \tag{4.10}$$

where the measurement matrix (\mathbf{C}_{lat}) is given by $\mathbf{C}_{lat}(k|t) = [0_{3\times 1} \quad \mathbf{I}_{3\times 3}]$.

Several disturbance models have been presented. In this research, the disturbance state ($\mathbf{d}_{lat}(k|t)$) is assumed as a random walk process [61], which is as follows:

$$\mathbf{d}_{lat}(k+1|t) = \mathbf{d}_{lat}(k|t) \text{ s.t. } \mathbf{d}_{lat}(k|t) \in R^{n_d}, n_d = 3 \quad (4.11)$$

As the number of states (n) and number of measurements (p) are given 4 and 3, respectively, the dimension of the disturbance (n_d) is assumed as 3, which is the same as the number of measurements.

Using Eq.(4.10) and Eq.(4.11), the augmented system model can be represented as follows:

$$\begin{aligned} \begin{bmatrix} \mathbf{x}_{lat}(k+1|t) \\ \mathbf{d}_{lat}(k+1|t) \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_{lat}(k|t) & \mathbf{B}_{lat.d}(k|t) \\ \mathbf{0}_{4 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{lat}(k|t) \\ \mathbf{d}_{lat}(k|t) \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{B}_{lat}(k|t) \\ \mathbf{0}_{3 \times 1} \end{bmatrix} u_{lat}(k|t) + \begin{bmatrix} \mathbf{F}_{lat}(k|t) \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \rho_{path}(k|t) \quad (4.12) \\ \mathbf{y}_{lat}(k|t) &= [\mathbf{C}_{lat}(k|t) \quad \mathbf{C}_{lat.d}(k|t)] \begin{bmatrix} \mathbf{x}_{lat}(k|t) \\ \mathbf{d}_{lat}(k|t) \end{bmatrix} \end{aligned}$$

In order to design the disturbance estimator, the matrices $\mathbf{B}_{lat.d}(k|t)$ and $\mathbf{C}_{lat.d}(k|t)$ should be chosen as that the augmented system is observable. The augmented system is observable if and only if the un-augmented system is observable and the following condition holds [62, 63].

$$rank \begin{bmatrix} \mathbf{A}_{lat} - \mathbf{I} & -\mathbf{B}_{lat.d} \\ \mathbf{C}_{lat} & \mathbf{C}_{lat.d} \end{bmatrix} = n + n_d \quad (4.13)$$

As the original system was observable, the rank of the left column of the matrix as shown in Eq.(4.13) is the number of states, denoted as n . The disturbance model matrices should be chosen as that the right column of the matrix in Eq.(4.13) is an

orthonormal basis of the null space of the left column of the matrix. Therefore, the disturbance variable was selected considering the observability of the augmented system. The disturbance variables, state, and measurement matrices are as follows:

$$\begin{aligned}
\mathbf{d}_{lat}(k|t) &= [\delta e_\psi(k|t) \ \delta u_{lat}(k|t) \ \delta \rho_{path}(k|t)]^T \\
\mathbf{B}_{lat.d}(k|t) &= [0_{4 \times 1} \ \mathbf{B}_{lat}(k|t) \ \mathbf{F}_{lat}(k|t)] \\
\mathbf{C}_{lat.d}(k|t) &= \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}
\end{aligned} \tag{4.14}$$

where, $\delta e_\psi(k|t)$ is the measurement bias in orientation error, $\delta u_{lat}(k|t)$ is the control input bias, and $\delta \rho_{path}(k|t)$ is the curvature bias. The number of control inputs in the system written in Eq.(4.10) is one. Therefore, the number of references to be tracked, which is the state for the zero offset, becomes one. In this problem, the manipulated variable is selected as the heading error (e_ψ). The other disturbances are chosen for two controlled variables, u_{lat} and ρ_{path} . Using the matrices in Eq.(4.14), the right column of the matrix in Eq.(4.13) is linearly independent. Therefore, the augmented system is observable.

The augmented system in Eq.(4.12) is simplified as follows:

$$\begin{aligned}
\mathbf{x}_e(k+1|t) &= \mathbf{A}_e(k|t)\mathbf{x}_e(k|t) + \mathbf{B}_e(k|t)u_{lat}(k|t) + \mathbf{F}_e(k|t)\rho_{path}(k|t) \\
&= f\left(\mathbf{x}_e(k|t), u_{lat}(k|t), \rho_{path}(k|t)\right) \\
\mathbf{y}_{lat}(k|t) &= \mathbf{C}_e(k|t)\mathbf{x}_e(k|t)
\end{aligned} \tag{4.15}$$

$$\begin{aligned}
\text{s.t. } \mathbf{x}_e(k|t) &= [\mathbf{x}_{lat}(k|t) \quad \mathbf{d}_{lat}(k|t)]^T \\
\mathbf{y}_{lat}(k|t) &= [\gamma(k|t) \quad e_\psi(k|t) + \delta e_\psi(k|t) \quad e_y(k|t)]^T
\end{aligned}$$

4.4.3.2. Design of State and Disturbance Estimator

MHE is an optimization-based state estimation technique [64, 65]. As shown in Fig. 4.7, the estimator considers only the most recent N measurements and finds only the most recent N values of the state trajectory. The states estimated by MHE are all the states in the estimation window. After finding the series of states using the given measurements and state-space model, the last state is inferred from the estimation of the current state. This estimation technique can be regarded as the dual problem of MPC. MHE requires a significant computational load to solve the optimization problem recursively for the updated moving window. Nevertheless, MHE showed a higher convergence rate, estimation accuracy, and robustness to large initial deviation in comparison to Kalman Filter-type estimators for vehicle state estimation problems [66, 67]. Moreover, the inequality constraints can be added to the estimated state, which is beneficial for the problem of estimating bounded bias in measurement.

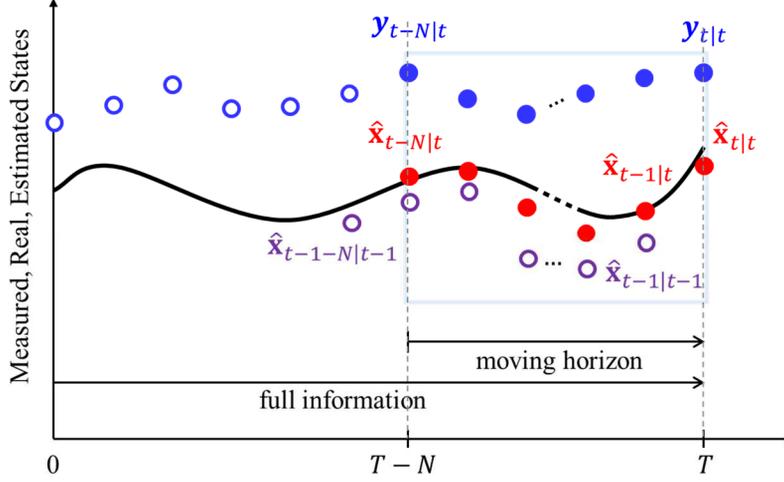


Fig. 4.7. Concept of a moving horizon estimation.

The optimization problem for the state and disturbance estimation is formulated as follows:

$$\begin{aligned}
 \min_{\mathbf{x}_{e,N}(t)} J &= \sum_{k=t-N}^t \|\mathbf{y}_{lat}(k|t) - \mathbf{C}_e(k|t)\mathbf{x}_e(k|t)\|_{\mathbf{V}_{lat}^{-1}}^2 \\
 &+ \sum_{k=t-N}^{t-1} \|\mathbf{x}_e(k+1|t) - f(\mathbf{x}_e(k|t), u_{lat}(k|t), \rho_{path}(k|t))\|_{\mathbf{W}_{lat}^{-1}}^2 \\
 \text{s.t. } &\delta e_{\psi.min} \leq \delta e_{\psi}(k|t) \leq \delta e_{\psi.max} \\
 &\delta u_{lat.min} \leq \delta u_{lat}(k|t) \leq \delta u_{lat.max} \\
 &\delta \rho_{path.min} \leq \delta \rho_{path}(k|t) \leq \delta \rho_{path.max}
 \end{aligned} \tag{4.16}$$

where $\mathbf{x}_{e,N}(t)$ is the set of states to be estimated; $\mathbf{x}_{e,N}(t) = \{\mathbf{x}_e(t-N|t), \dots, \mathbf{x}_e(t|t)\}$. \mathbf{P}_{lat} is the arrival cost at time $t-N$. \mathbf{V}_{lat}^{-1} and \mathbf{W}_{lat}^{-1} are the weighting matrices of measurements and process models, which are inversely proportional to the noise covariance. N is the maximum prediction step, which is

determined by dividing the prediction horizon length into sampling time.

4.5. Drivable Corridor Determination

A drivable corridor indicates the range that the ego vehicle is able to go through within the ROI. The drivable corridor is the envelop that the vehicle can pass the way. The envelop consists of the left/right envelops. The envelop is initially assumed that the left/right lanes, which is obtained by HD map. Within the drivable corridor, the desired path is obtained to avoid collision with obstacles.

Here, the algorithm suggested in the previous research was adopted to determine the drivable corridor [57]. Since the logic was applied to autonomous driving for a private car, the length of the bus should be additionally considered. The test vehicle (full-sized bus) is 11 meters long, about twice as long as a private car. The concept to determine the drivable corridor is shown in Fig. 4.8. The centerline and left/right lanes stored in HD map were utilized to determine the drivable corridor. In addition, static obstacles within the ROI of the ego vehicle were taken into account. The static obstacles were obtained in the way of an occupancy grid map which is named a static obstacle map (STOM) [47].

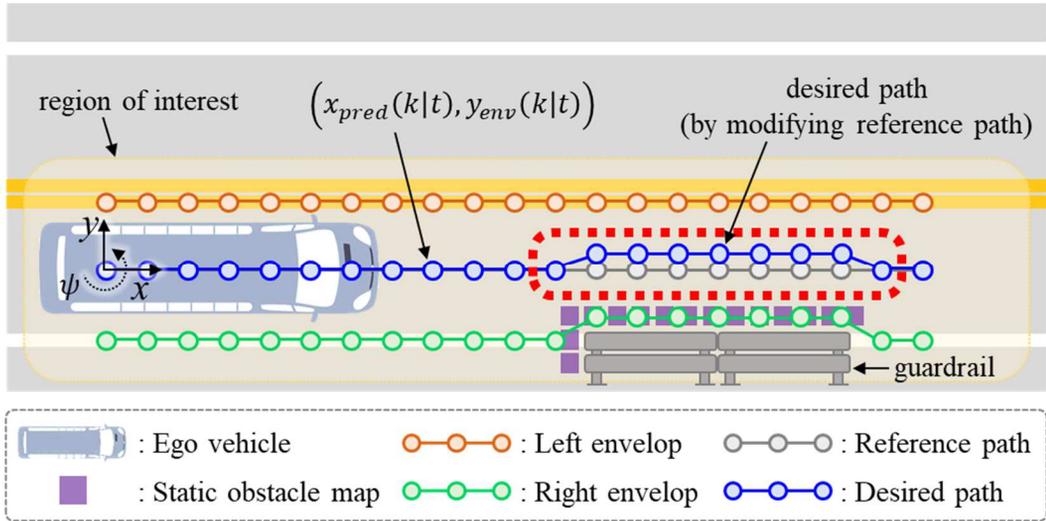


Fig. 4.8. The determination of the desired path considering drivable corridor

The way how to calculate the drivable corridor was a sequential process. First of all, the reference path and left/right boundaries were calculated using the centerline and left/right lanes stored in HD map. The value, denoted as x in envelop, indicates the expected path for the center of mass of the vehicle. The value, denoted as y in envelop, indicates the interpolated values from the centerline and left/right lanes in HD map.

After it was checked whether the static obstacles within the ROI of the ego vehicle invade the left/right envelopes, the envelopes were modified. The ROI was applied to the reduction of computational load by excluding the perception data, which was not related to the driving of the ego vehicle in the motion planning. As shown in Fig. 4.8, it is observed that the guardrail invades the right lane. As soon as STOM for the guardrail was calculated, the right envelop was corrected by referring to the STOM. After the driving path was acquired for the center of mass of the ego vehicle, the calculation was repeatedly conducted for the front bumper point. As a

result, the smallest envelop was regarded as the final envelop by combining two envelops. The initial desired path was assumed as the reference path. If the distance between the left/right envelop and reference path, considering the half of the ego vehicle's width, was smaller than the minimum clearance, the envelop was shifted to secure the minimum clearance.

4.6. Lateral Motion Planning using Offset-free MPC

Offset-free MPC was suggested to reject the bias when the bias exists in MPC formulation due to external disturbance. The conventional MPC cannot reject the bias, since the approach is based on state feedback. To overcome the limitation, the estimated disturbance was considered as the state in MPC formulation. Or, MPC was formulated using a state model, eliminating the disturbance [68, 69, 70].

The optimization problem for the state and disturbance estimation is formulated as follows:

$$\begin{aligned}
\min_{\mathbf{u}_{lat.N}(t)} J = & \|\mathbf{x}_{lat}(k|t) - \mathbf{x}_{lat.ref}(k|t)\|_{\mathbf{P}_{lat}}^2 \\
& + \sum_{k=0}^{N_p-1} \left\{ \|\mathbf{x}_{lat}(k|t) - \mathbf{x}_{lat.ref}(k|t)\|_{\mathbf{Q}_{lat}}^2 \right. \\
& \left. + \|u_{lat}(k|t)\|_{\mathbf{R}_{lat}}^2 \right\} \\
\text{s.t.} & \tag{4.17}
\end{aligned}$$

$$\begin{aligned}
\mathbf{x}_{lat}(k+1|t) = & \mathbf{A}_{lat}(k|t)\mathbf{x}_{lat}(k|t) + \mathbf{B}_{lat}(k|t)u_{lat}(k|t) \\
& + \mathbf{F}_{lat}(k|t)\rho_{path}(k|t)
\end{aligned}$$

$$\delta_{FSA.des.min} \leq u_{lat}(k|t) \leq \delta_{FSA.des.max}$$

$$|u_{lat}(k+1|t) - u_{lat}(k|t)| \leq dt \cdot \dot{\delta}_{FSA.max}$$

where $\mathbf{u}_{lat.N}(k|t)$ is the set of control input to be optimized ($\mathbf{u}_{lat.N}(k|t) = \{u_{lat}(0|t), \dots, u_{lat}(N_p - 1|t)\}$). \mathbf{P}_{lat} is the terminal cost at time k. \mathbf{Q}_{lat} and \mathbf{R}_{lat} are the weighting matrices of reference tracking and control input. N_p is the maximum prediction step, which is determined by dividing the prediction horizon length into sampling time. The constraints for the control inputs are the bounds of magnitude and jerk. These constraints were defined referring to the control interface specifications. In this study, $\delta_{FSA.des.min}$ of -900 deg, $\delta_{FSA.des.max}$ of 900 deg, and $\dot{\delta}_{FSA.max}$ of 360 deg/sec were used. Here, the current state of heading error ($e_\psi(0|t)$) is adopted by the estimated heading error from the disturbance estimation. Since the bias in heading error is eliminated, the MPC formulation becomes heading offset-free.

Chapter 5. Longitudinal and Lateral Motion Control

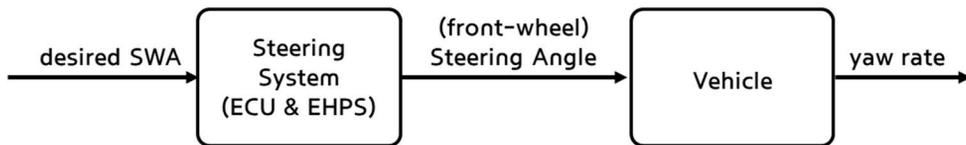
5.1. Problem Definition

The motion control algorithm of an automated bus aimed to develop a robust controller that can be applied to various driving conditions with limited feedback information. To improve the computational efficiency, the motion control module was developed separately from the motion planning module, referring to the previous researches [11, 21, 57]. After the motion planning algorithm determines the desired motion, the motion control algorithm tracks the desired motion, which is described as the desired longitudinal acceleration and desired yaw rate.

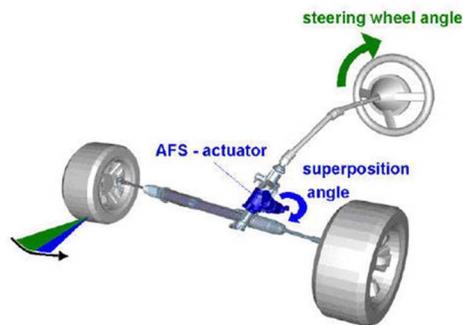
The longitudinal motion control algorithm was designed to be robust against uncertain disturbances using only limited feedback information. Performance of the longitudinal motion control algorithm is affected by environmental disturbance due to changes in mass (onboarding/stopover of passengers) and changes in slope. Consequently, a robust control algorithm is necessary against the disturbance. Besides, in the case of commercial vehicles and public buses, wheel speed sensors are installed. But, the longitudinal accelerometer is usually not installed. Therefore, the algorithm could utilize limited feedback information; wheel speed sensors.

The lateral motion control algorithm was required considering the dynamic characteristics of actuators equipped on a full-sized public bus. The steering system of the test vehicle is constructed as shown in Fig. 5.1. The control interface was developed as the type of desired steering wheel angle (SWA). As shown in Fig. 5.1(a),

when the desired SWA is the input into the steering system, the ECU of the steering system controls SWA through the feedback control. Once SWA changes, the front wheel's steering angle is changed by the actuator of the steering system, as shown in Fig. 5.1(b) [71]. As a result, the yaw rate of the vehicle changes due to the effect of the moment acting on the vehicle. The response related to yaw rate is slow on a commercial vehicle and full-sized bus when the steering wheel angle changes. On the other hand, a private car shows quick response without delay. Consequently, the dynamic characteristics of the steering system on a bus have to be taken into account.



(a) Interface of test vehicle's steering system



(b) Example of steering system

Fig. 5.1. Steering system of the test vehicle.

5.2. Outlines of Longitudinal and Lateral Motion Control

This section describes the architecture of the proposed longitudinal motion control algorithm for an autonomous vehicle. As summarized in Fig. 5.2, the goal of the longitudinal motion planning algorithm is to calculate the desired reference motion. The longitudinal motion control algorithm calculates the throttle and brake control inputs for tracking the reference motion.

The longitudinal motion planning algorithm and longitudinal motion control algorithm are regarded as upper-level and lower-level controllers, respectively. The upper-level controller adopted an MPC scheme assuming a first-order delay model. The delay includes the effects of the vehicle dynamics and the response of control actuators. Using the MPC scheme, the optimal control input was calculated as the desired acceleration considering the delay. The acceleration was converted into the desired speed in the reference speed model of the lower-level controller.

For the lower-level controller, a model-reference ASMC algorithm was proposed. Using the reference speed model, the desired acceleration, which was acquired from the upper-level controller, was converted into the reference speed. Then, the reference speed model calculated the desired speed with the reference and current speed of the vehicle. A first-order response function was adopted as the reference model, which worked as a low-pass filter. The ASMC scheme was utilized to achieve high performance in tracking with robustness against uncertainties. The adaptation parameters were updated from parameter adaptation law, while the RBFNN estimated the upper bound of the unstructured uncertainties for the SMC logic. The equivalent control input was also updated with the adaptation parameters.

Furthermore, this input was used for the switching algorithm according to the control mode. After the throttle and brake control commands were calculated from the switching algorithm, they were applied to inputs for the actuators of the ego vehicle.

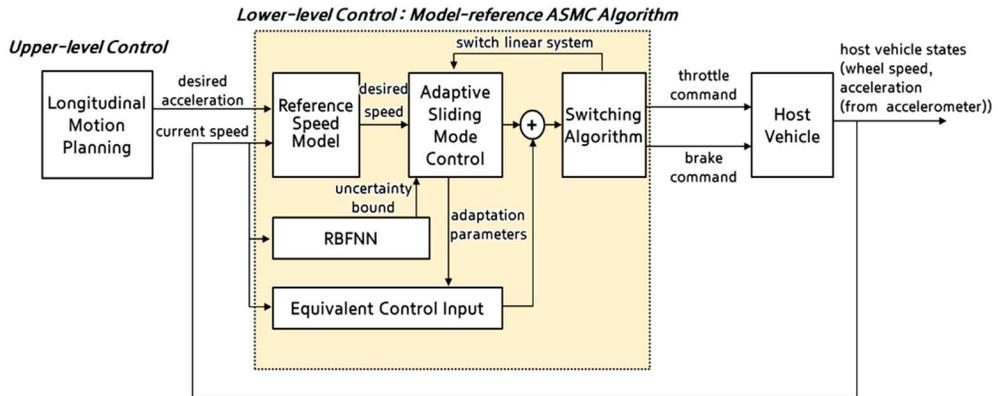


Fig. 5.2. Outline of the proposed longitudinal motion control algorithm.

The model with an assumption that there was no delay between front steering angle and yaw rate was used for the lateral motion planning algorithm. Consequently, the desired yaw rate could be obtained from the desired front steering angle acquired from the lateral motion planning result. Feedback information, which was measured SWA and actual yaw rate, was provided from chassis CAN. Consequently, the lateral motion control algorithm is constructed as shown in Fig. 5.3.

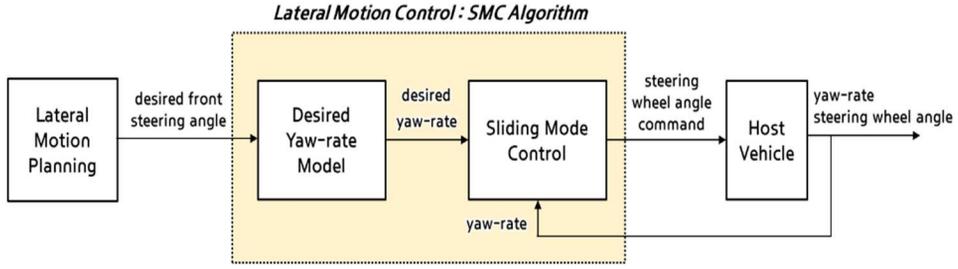


Fig. 5.3. Outline of the proposed lateral motion control algorithm.

5.3. Longitudinal Motion Control

The following section and Section 6.3 are the summaries of the published research paper [72].

5.3.1. Definition of System Dynamics Model

A longitudinal vehicle dynamics model considering the powertrain and brake system interface and throttle and brake control switching is formulated in this section. The equation of motion of a vehicle moving on an inclined load along with the vehicle's longitudinal axis yields [60]:

$$m\dot{v}_x = F_T - F_R \quad (5.1)$$

where, F_T is the sum of longitudinal traction force, and F_R is the resistance force.

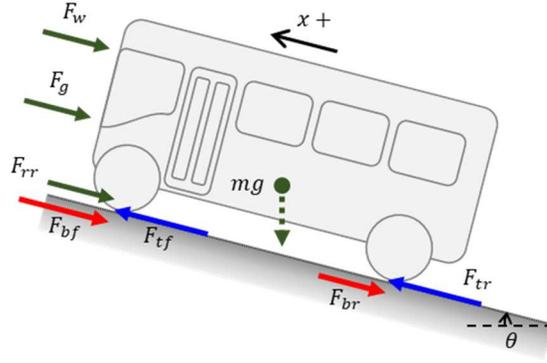


Fig. 5.4. External forces acting on a vehicle with respect to the longitudinal direction.

The longitudinal traction force is composed of driving force from the front and rear tires, denoted as F_{tf} and F_{tr} , and brake force from the front and rear tires, denoted as F_{bf} and F_{br} , as shown in Fig. 5.4. The driving force is generated by an electric motor or engine, and the brake force is driven by the brake system, respectively. These forces can be approximated as a linear relationship with the throttle and brake torques by assuming no tire-slip condition, as follows [60]:

$$\begin{aligned}
 F_T &= F_{tf} + F_{tr} + F_{bf} + F_{br} \\
 &= \frac{1}{r_{eff}} T_e - \frac{1}{r_{eff}} T_b
 \end{aligned} \tag{5.2}$$

where r_{eff} is the effective tire radius. T_e is the drive torque and T_b is the brake torque.

The longitudinal resistance force, F_R , is composed of gravitational force, F_g , rolling resistance force, F_{rr} , and aerodynamic force, F_w , as follows:

$$\begin{aligned}
F_R &= F_g + F_{rr} + F_w \\
&= mg\sin\theta + mg(C_{r0} + C_{rv}v_x)\cos\theta + \frac{1}{2}\rho A_{fr}S_x(v_x + v_{air})^2
\end{aligned} \tag{5.3}$$

where m is the mass of the vehicle, g is the gravitational acceleration, θ is the road slope, C_{r0} and C_{rv} are the rolling resistance coefficients, ρ is the density of air, A_{fr} is the frontal area, S_x is the aerodynamic coefficient, and v_{air} is the relative speed of the wind.

The characteristics of the powertrain and brake system depend on the actuator and control interface. The tire dynamics modeling is considered for the load transfer from the powertrain and brake system to a vehicle. Wheel slip is typically small for normal driving conditions. Therefore, the tire dynamics equation can be approximated as a linear relationship with the longitudinal speed and control input. The control gain was approximated by system identification results using the relation between the control input and the output of the measured speed. The delay of the control actuators may be considered together with the control gain for constructing the control interface model [35]. However, since the delay was already considered by the motion planning algorithm using the MPC scheme, it was neglected in the modeling. Then, the control gain was represented as a function of the speed and control input and normalized by mass, which yields:

$$\frac{1}{r_{eff}}T_e - \frac{1}{r_{eff}}T_b = mk^i(v_x, u) \cdot u \tag{5.4}$$

where k^i is the normalized control gain, and superscript i represents control mode

(throttle or brake). u is the control input. Besides, the resistance force can be approximated as follows:

$$\begin{aligned}
 F_R &= m(\bar{F}_R + f_{Rv}v_x + f_{R0}) \\
 \text{s. t. } \bar{F}_R &= g(C_{r0} + C_{rv}v_x) + \frac{1}{2m}\rho A_{fr}S_x v_x^2 \\
 &= \bar{F}_R(v_x)
 \end{aligned} \tag{5.5}$$

where the $\bar{\cdot}$ in Eq.(5.5) represents the nominal value, and f_{Rv} and f_{R0} are time-varying parameters due to the disturbances and uncertainties normalized by mass.

Consequently, by substituting Eq.(5.4) and Eq.(5.5) into Eq.(5.1), the longitudinal vehicle dynamics model is simplified as follows

$$\dot{v}_x = k^i(v_x, u)u - (f_{Rv}v_x + f_{R0} + \bar{F}_R(v_x)) \tag{5.6}$$

As stated in Eq.(5.5), the resistance force exerted on a vehicle is irrelevant to the throttle or brake control modes; it varies with the vehicle's speed. Due to this resistance force, a natural longitudinal acceleration, so-called zero-throttle acceleration, occurs even if there is no control input [73, 74]. The zero-throttle acceleration curve obtained from vehicle tests is plotted in Fig. 5.5. The vehicle tests were conducted using the test vehicle. As illustrated in Fig. 5.5, based on the acceleration line, the throttle control mode can be used for larger desired acceleration, and the brake control mode can be used for smaller desired acceleration. Here, h is a switching margin to prevent frequent and unnecessary mode changes.

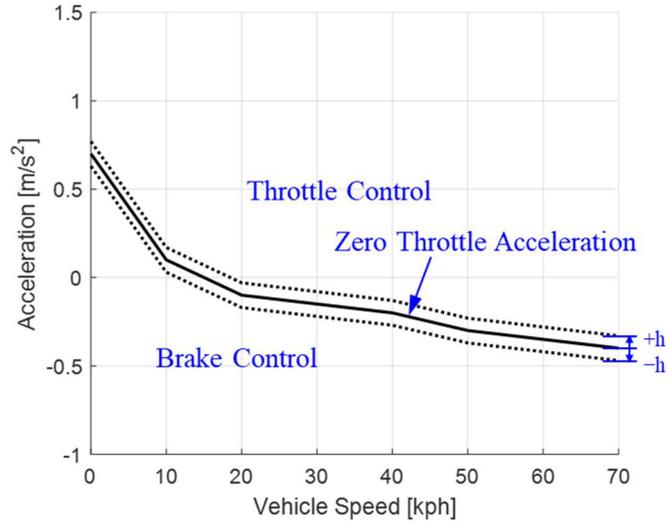


Fig. 5.5. Zero-throttle acceleration of a vehicle according to vehicle speed.

The zero-throttle acceleration for each speed can be measured under the normal road condition of flat ground in practical applications. Suppose that the effect of the zero-throttle acceleration is not considered in the controller. Then, the performance to keep the constant longitudinal speed can deteriorate, and the controller's convergence speed becomes slow. Therefore, the zero-throttle acceleration should be considered independently of the control mode to represent the actual dynamics.

5.3.2. Longitudinal Motion Control using MR-ASMC

In the longitudinal motion planning algorithm, the vehicle's longitudinal behavior was modeled using the term of longitudinal acceleration. The vehicle's acceleration with time delay was calculated through the estimator. If reference motion for the lower-level controller is expressed using the longitudinal acceleration, the control error has to contain the estimation delay. Due to the included delay, the performance of the

lower-level controller could be degraded. Therefore, the desired acceleration from the upper-level controller is converted into a reference speed as follows:

$$v_{ref}(t) = v_x + k_v a_{des}(t) \quad (5.7)$$

where v_{ref} is the reference speed, and k_v is the constant gain of the desired acceleration.

In the model-reference ASMC algorithm, a first-order response function was adopted as a reference model to cope with the measurement noise, and this model worked as a low-pass filter. Also, the error with constant gain, denoted as λ , was fed back to the reference model to achieve fast convergence [75]. The tracking error and the sliding surface can be formulated as:

$$\begin{aligned} e(t) &= v_x - v_{ref} \\ e_m(t) &= v_x - v_m \equiv s(t) \\ \dot{v}_m &= -a_m(v_m - v_{ref}) + \lambda e_m \end{aligned} \quad (5.8)$$

where $e(t)$ is the tracking error. v_m is the modified desired speed, which is calculated by the reference model, $e_m(t)$ is the modified tracking error, and a_m is the gain of the reference model. Here, a_m and λ are the design parameters.

Differentiating the sliding surface $s(t)$ with respect to time is written as follows:

$$\begin{aligned}
\dot{s}(t) &= \dot{v}_x - \dot{v}_m = \dot{e}_m(t) \\
&= k^i u - f_{Rv} v_x - f_{R0} - \bar{F}_R + a_m (v_m - v_{ref}) - \lambda e_m \\
&= k^i u - a_m v_{ref} - (f_{Rv} - a_m) v_x - f_{R0} - a_m (v_x - v_m) - \lambda e_m - \bar{F}_R \quad (5.9) \\
&= k^i \left\{ u - \frac{1}{k^i} [a_m v_{ref} + (f_{Rv} - a_m) v_x + f_{R0}] \right\} - (a_m + \lambda) e_m - \bar{F}_R
\end{aligned}$$

The equation can be simplified by introducing equivalent system parameters as follows:

$$\dot{s}(t) = k^i (u + \eta_1 v_{ref} + \eta_2 v_x + \eta_3) - (a_m + \lambda) s - \bar{F}_R \quad (5.10)$$

where η_1, η_2, η_3 are the system parameters, defined as follows:

$$\eta_1 = -\frac{a_m}{k^i}, \quad \eta_2 = -\frac{f_{Rv} - a_m}{k^i}, \quad \eta_3 = -\frac{f_{R0}}{k^i} \quad (5.11)$$

where the $\hat{\cdot}$ and $\Delta \cdot$ in Eq.(5.11) represent the estimated and uncertainty values, respectively.

The Lyapunov function V_1 for equivalent control input is as follows:

$$V_1(t) = \frac{1}{2} s(t)^2 \quad (5.12)$$

Differentiating V_1 with respect to time is written as follows:

$$\begin{aligned}\dot{V}_1(t) &= s(t) \cdot \dot{s}(t) \\ &= s\{k^i(u + \hat{\eta}_1 v_{ref} + \hat{\eta}_2 v_x + \hat{\eta}_3) - (a_m + \lambda)s - \bar{F}_R\} = 0\end{aligned}\quad (5.13)$$

The control objective of the ASMC is to keep the sliding surface as zero by choosing the equivalent control input as below:

$$u_{eq}(t) = -(\hat{\eta}_1 v_{ref} + \hat{\eta}_2 v_x + \hat{\eta}_3) + \frac{\bar{F}_R}{k^i} \quad (5.14)$$

The finite-time convergence can be guaranteed by choosing the switching control input as follows:

$$\begin{aligned}u_N(t) &= -\frac{K}{k^i} \cdot \text{sgn}(s) \\ \text{s. t. } \frac{K}{k^i} &> |\Delta_{un}| + \varepsilon_1\end{aligned}\quad (5.15)$$

where K is the upper bound of uncertainty. Δ_{un} is the lumped bounded uncertainty, and ε_1 is the positive constant for finite-time convergence. Then, the total control input becomes as given in Eq.(5.16).

$$u = u_{eq} + u_N \quad (5.16)$$

The control input in Eq.(5.16) is made to satisfy the reaching condition.

$$\begin{aligned}
\dot{V}_1(t) &= s(t) \cdot \dot{s}(t) \\
&= s\{k^i(u + \eta_1 v_{ref} + \eta_2 v_x + \eta_3) - (a_m + \lambda)s - \bar{F}_R\} \\
&= s\{k^i(u_N + \tilde{\eta}_1 v_{ref} + \tilde{\eta}_2 v_x + \tilde{\eta}_3) - (a_m + \lambda)s\} \\
&\leq s\left\{k^i\left(-\frac{K}{k^i} \cdot \text{sgn}(s) + \Delta_{un}\right)\right\} - (a_m + \lambda)s^2 \\
&\leq -k^i \varepsilon_1 |s| - (a_m + \lambda)s^2 \leq 0
\end{aligned} \tag{5.17}$$

where $\tilde{\eta}_i = \Delta \eta_i = \eta_i(t) - \hat{\eta}_i(t)$, $i = 1, 2, 3$. Since the derivative of the Lyapunov function in Eq.(5.17) is proved to be negative, the proposed ASMC is asymptotically stable [76].

RBFNN has been included in the ASMC by following the Lyapunov framework to design a robust ASMC scheme. It is essential to know the upper bound of the uncertainty, denoted as K , in advance. If the assumed bound is larger than the actual uncertainty, an excessive control input for the system can cause chattering. An adaptive neural network can approximate arbitrary nonlinear functions. Therefore, the RBFNN has been used to estimate the upper bound of system uncertainties online. This approach improves control performance against the system's unstructured uncertainty. Moreover, the network can reduce the chattering of sliding mode effectively [41, 42]. The bound may be designed using fuzzy logic. However, since the speed tracking controller should maintain robustness under various driving conditions, the estimation method of the upper bound without prior knowledge of the bound is desirable.

As illustrated in Fig. 5.6, the structure of the RBFNN is composed of a three-layer feed-forward network: an input layer, a hidden layer composed of radial basis function (RBF), and an output layer [77].

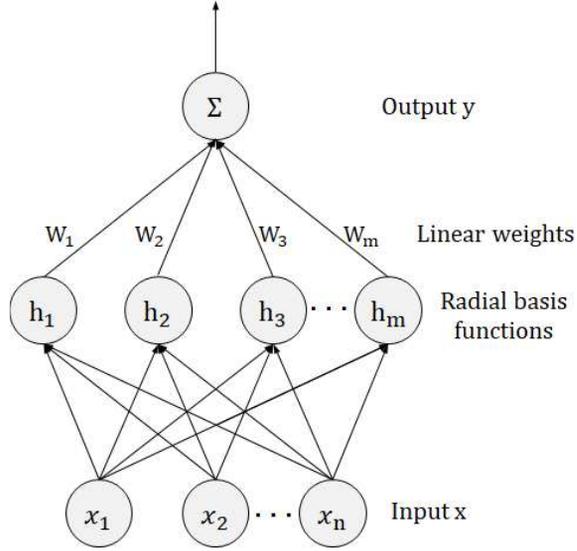


Fig. 5.6. The structure of the adopted RBFNN.

Suppose that $K(t)$ is the time-varying upper bound of the uncertainty. Then, the approximated bound $\hat{K}(t)$ can be defined as follows:

$$\begin{aligned}
 \hat{K}(t) &= \hat{W}^T h \\
 K(t) \cdot \text{sgn}(s) - \hat{K}(t) &= W^{*T} h + \varepsilon(t) - \hat{W}^T h \\
 &= \tilde{W}^T h + \varepsilon(t) \\
 \text{s. t. } |\varepsilon(t)| &\leq \varepsilon_2
 \end{aligned} \tag{5.18}$$

where W^* and \hat{W} are the optimal and estimated weight vectors, $\tilde{W} = \Delta W = W^* - \hat{W}$, and $\varepsilon(t)$ is the approximation error. The error $\varepsilon(t)$ is assumed to be bounded, and ε_2 denotes the maximum approximate error. $h = [h_1, h_2, \dots, h_m]^T$ is a known smooth vector function. The basis function $h_i(s)$ is chosen as the standard Gaussian function with the form [77] written as follows:

$$h_i(s) = \exp\left[-\frac{(s - \mu_i)^2}{\zeta^2}\right] \quad (5.19)$$

where μ_i and ζ are the center and the width of the basis function, respectively.

The Lyapunov function V_2 for parameter adaptation law is as follows:

$$V_2(t) = \frac{1}{2}s^2 + \frac{k^i}{2}(\gamma_1^{-1} \Delta \eta_1^2 + \gamma_2^{-1} \Delta \eta_2^2 + \gamma_3^{-1} \Delta \eta_3^2) + \frac{1}{2}\gamma_4^{-1} \Delta W^T \Delta W \quad (5.20)$$

Differentiating V_2 with respect to time, and substituting the time derivative of the Lyapunov function V_1 in Eq.(5.20) into Eq.(5.17), the derivative of the Lyapunov function V_2 is derived as follows:

$$\begin{aligned} \dot{V}_2(t) &= s\dot{s} + k^i(\gamma_1^{-1}\tilde{\eta}_1\dot{\tilde{\eta}}_1 + \gamma_2^{-1}\tilde{\eta}_2\dot{\tilde{\eta}}_2 + \gamma_3^{-1}\tilde{\eta}_3\dot{\tilde{\eta}}_3) + \gamma_4^{-1}\tilde{W}^T\dot{\tilde{W}} \\ &= sk^i(u_N + \tilde{\eta}_1v_{ref} + \tilde{\eta}_2v_x + \tilde{\eta}_3) - (a_m + \lambda)s^2 \\ &\quad - k^i(\gamma_1^{-1}\tilde{\eta}_1\dot{\tilde{\eta}}_1 + \gamma_2^{-1}\tilde{\eta}_2\dot{\tilde{\eta}}_2 + \gamma_3^{-1}\tilde{\eta}_3\dot{\tilde{\eta}}_3) - \gamma_4^{-1}\tilde{W}^T\dot{\tilde{W}} \\ &= sk^i u_N - (a_m + \lambda)s^2 \\ &\quad + k^i\{\tilde{\eta}_1(sv_{ref} - \gamma_1^{-1}\dot{\tilde{\eta}}_1) + \tilde{\eta}_2(sv_x - \gamma_2^{-1}\dot{\tilde{\eta}}_2) \\ &\quad + \tilde{\eta}_3(s - \gamma_3^{-1}\dot{\tilde{\eta}}_3)\} - \gamma_4^{-1}\tilde{W}^T\dot{\tilde{W}} \end{aligned} \quad (5.21)$$

The adaptation laws for the system parameters are designed as follows:

$$\begin{aligned} \dot{\tilde{\eta}}_1 &= \gamma_1 v_{ref} s \\ \dot{\tilde{\eta}}_2 &= \gamma_2 v_x s \\ \dot{\tilde{\eta}}_3 &= \gamma_3 s \end{aligned} \quad (5.22)$$

Substituting Eq.(5.22) into Eq.(5.21), the derivative of the Lyapunov function

V_2 becomes as follows:

$$\begin{aligned}
\dot{V}_2(t) &= sk^i \left(-\frac{\hat{K}}{k^i} + \Delta_{un} \right) - \gamma_4^{-1} \tilde{W}^T \dot{\hat{W}} - (a_m + \lambda) s^2 \\
&= sk^i \Delta_{un} - s\hat{K} - \gamma_4^{-1} \tilde{W}^T \dot{\hat{W}} - (a_m + \lambda) s^2 \\
&= \left(sk^i \Delta_{un} - sK \cdot \text{sgn}(s) \right) + \left(sK \cdot \text{sgn}(s) - s\hat{K} \right) - \gamma_4^{-1} \tilde{W}^T \dot{\hat{W}} - (a_m + \lambda) s^2 \\
&= s \left\{ k^i \left(-\frac{K}{k^i} \cdot \text{sgn}(s) + \Delta_{un} \right) \right\} + s \left(\tilde{W}^T h + \varepsilon(t) \right) - \gamma_4^{-1} \tilde{W}^T \dot{\hat{W}} - (a_m + \lambda) s^2 \quad (5.23) \\
&\leq -k^i \varepsilon_1 |s| + s \cdot \varepsilon_2 + \tilde{W}^T \left(hs - \gamma_4^{-1} \dot{\hat{W}} \right) - (a_m + \lambda) s^2 \\
&\leq -|s| (k^i \varepsilon_1 - \varepsilon_2) + \tilde{W}^T \left(hs - \gamma_4^{-1} \dot{\hat{W}} \right) - (a_m + \lambda) s^2
\end{aligned}$$

The adaptation law for adjusting the weights of RBFNN is designed as follows:

$$\dot{\hat{W}} = \gamma_4 hs \quad (5.24)$$

By substituting Eq.(5.24) into Eq.(5.23), and assuming that ε_2 is sufficiently small and satisfying $\varepsilon_2 < k^i \varepsilon_1$, the inequality can be obtained as follows:

$$\begin{aligned}
\dot{V}_2 &\leq -|s| (k^i \varepsilon_1 - \varepsilon_2) - (a_m + \lambda) s^2 \\
&\leq -(a_m + \lambda) s^2 \leq 0 \quad (5.25)
\end{aligned}$$

Since the derivative of the Lyapunov function in Eq.(5.25) is proved to be negative, the proposed RBFNN-based model-reference ASMC is asymptotically stable [76].

To attenuate the chattering phenomenon, the switching control input in Eq.(5.15)

is modified as follows:

$$\begin{aligned}
 u_N(t) &= -\frac{K}{k^i} \cdot \text{sat}\left(\frac{s}{\Phi}\right) \\
 \text{sat}(x) &= \begin{cases} 1 & \text{if } x > 1 \\ -1 & \text{if } x < -1 \\ x & \text{else} \end{cases} \quad (5.26)
 \end{aligned}$$

It is essential to design the switching algorithm of throttle and brake modes. An unreasonable switching algorithm will degrade the control performance and cause chattering problems.

The total control input Eq.(5.16) can be rewritten by substituting Eq.(5.11) into Eq.(5.16) as follows:

$$\begin{aligned}
 u &= u_{eq} + u_N \\
 &= -\left(\hat{\eta}_1(t)v_{ref}(t) + \hat{\eta}_2(t)v_x(t) + \hat{\eta}_3(t)\right) + \frac{\bar{F}_R(v_x)}{k^i(t)} - \frac{\hat{K}(t)}{k^i(t)} \cdot \text{sat}\left(\frac{s}{\Phi}\right) \\
 &= \frac{1}{\hat{k}^i(t)} \left[-a_m\{v_x(t) - v_{ref}(t)\} + \hat{f}_{Rv}(t)v_x(t) + \hat{f}_{R0}(t) + \bar{F}_R(v_x) - \hat{K}(t) \right. \\
 &\quad \left. \cdot \text{sat}\left(\frac{s}{\Phi}\right) \right] \quad (5.27)
 \end{aligned}$$

The zero-throttle acceleration illustrated in Fig. 5.5 can be reformulated as a function of the normalized control input and speed (Fig. 5.7). If the ego vehicle's speed is the same as the reference and reference model's speed, the total control input in Eq.(5.27) becomes the value to compensate for the zero-throttle acceleration. In Fig. 5.7, the value on the equivalent control input line is where the sliding surface is zero. The error is negative in the shaded part and positive in the other part. The

equivalent control input initially has a nominal value when there are no disturbances and uncertainties. While the parameter adaptation is performed, the equivalent control input is corrected by the adaptation result as follows:

$$u_{eq}(t|s=0) = \frac{1}{\hat{k}^i(t)} \{ \hat{f}_{Rv}(t)v_x(t) + \hat{f}_{R0}(t) + \bar{F}_R(v_x) \} \quad (5.28)$$

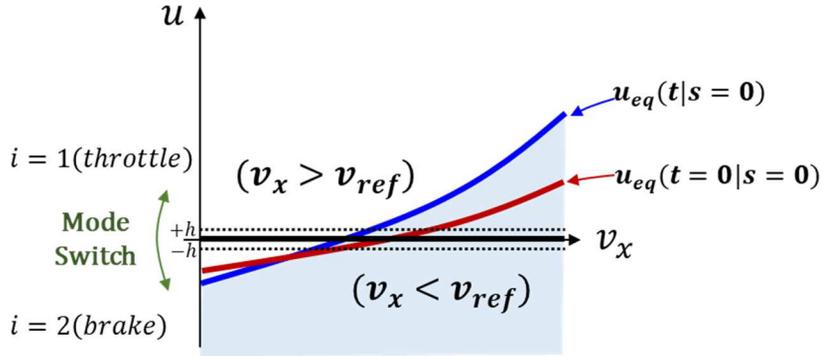


Fig. 5.7. Switching condition of throttle and brake modes.

The system's mode is switched if the sign of the total control input changes. If the total input changes from positive to negative, the control mode becomes brake mode. The control inputs are determined in the following way:

$$\begin{aligned} u_{th} &= u, & u_{br} &= 0 & \text{if } u > h \\ u_{th} &= 0, & u_{br} &= u & \text{if } u < -h \\ u_{th} &= 0, & u_{br} &= 0 & \text{elsewhere} \end{aligned} \quad (5.29)$$

where h is the switching margin to prevent frequent and unnecessary mode changes.

When the control mode is switched, the adaptation parameters of the

environmental disturbances and uncertainties shall not change. Thus, in the design of the switching algorithm, the adaptation parameters, denoted as \hat{f}_{Rv} and \hat{f}_{R0} , remain. Only the parameter of the vehicle control gain, denoted as \hat{k}^i , is adjusted according to the mode in the following way:

$$\hat{\eta}_i(t) = \begin{cases} \hat{\eta}_i(t) & \text{if } mode_{k-1} = mode_k \\ \hat{\eta}_i(t) \frac{k^1(t)}{k^2(t)} & \text{if } mode_{k-1} = 2, mode_k = 1 \\ \hat{\eta}_i(t) \frac{k^2(t)}{k^1(t)} & \text{if } mode_{k-1} = 1, mode_k = 2 \end{cases} \quad (5.30)$$

5.4. Lateral Motion Control

The actual yaw rate of the vehicle can be assumed to track the desired yaw rate with delay. The system dynamics can be derived using a simple second-order model with un-modeled system error as follows:

$$a\ddot{\gamma}_{act} + b\dot{\gamma}_{act} + \gamma_{act} = \gamma_{des} + \Delta_{un} \quad (5.31)$$

The tracking error and the sliding surface can be formulated as follows:

$$\begin{aligned} e &= \gamma_{act} - \gamma_{ref} \\ s &= e + \lambda \dot{e} \end{aligned} \quad (5.32)$$

The Lyapunov function V for equivalent control input is written as follows:

$$V = \frac{1}{2}s^2 \quad (5.33)$$

Differentiating V with respect to time is written as follows:

$$\dot{V} = s \cdot \dot{s} = s \left\{ (\dot{\gamma}_{act} - \dot{\gamma}_{ref}) + \lambda \left(\frac{\gamma_{des} - \gamma_{act} - b\dot{\gamma}_{act}}{a} - \ddot{\gamma}_{act} \right) \right\} = 0 \quad (5.34)$$

The equivalent control input can be obtained using the derivative of the sliding mode as below:

$$u_{eq} = \gamma_{act} + b\dot{\gamma}_{act} + a\ddot{\gamma}_{act} - \frac{a}{\lambda}(\dot{\gamma}_{act} - \dot{\gamma}_{ref}) = \gamma_{des} - \frac{a}{\lambda}(\dot{\gamma}_{act} - \dot{\gamma}_{ref}) \quad (5.35)$$

In order to compensate for the lumped uncertainty, switching control input is introduced as follows:

$$\begin{aligned} u_N &= -K \cdot \text{sgn}(s) \\ \text{s.t. } K &> |\Delta_{un}| \end{aligned} \quad (5.36)$$

The total control input, $u = u_{eq} + u_N$, is made to satisfy the following reaching condition.

$$\begin{aligned} \dot{V} &= s \cdot \dot{s} \\ &= s \left\{ (\dot{\gamma}_{act} - \dot{\gamma}_{ref}) + \lambda \left(\frac{u - \gamma_{act} - b\dot{\gamma}_{act} + \Delta_{un}}{a} - \ddot{\gamma}_{act} \right) \right\} \\ &= s \frac{\lambda}{a} (-K \text{sgn}(s) + \Delta_{un}) \leq 0 \end{aligned} \quad (5.37)$$

Since the derivative of the Lyapunov function in Eq.(5.37) is proved to be negative, the proposed SMC is asymptotically stable. To attenuate the chattering phenomenon, the switching control input in Eq.(5.36) is modified as follows:

$$\begin{aligned} u_N(t) &= -K \cdot \text{sat} \left(\frac{s}{\Phi} \right) \\ \text{where, } \text{sat}(x) &= \begin{cases} 1 & \text{if } x > 1 \\ -1 & \text{if } x < -1 \\ x & \text{else} \end{cases} \end{aligned} \quad (5.38)$$

where Φ is the thickness of the boundary layer. The saturation function was adopted to attenuate the chattering.

Chapter 6. Performance Evaluation

6.1. Performance Evaluation of Longitudinal Motion

Planning

6.1.1. Computer Simulation Results

The simulation studies for the longitudinal motion planning algorithm aimed to examine the feasibility of the proposed algorithm. First, as a representative scenario, simulation was performed against stationary and slowly moving targets and urgently decelerating target. These scenarios are to confirm the improvement of ride comfort and perceived safety. Second, a Monte-Carlo simulation was performed for each FOV range for a preceding vehicle (risky target) driving at a slower speed than an ego vehicle. This simulation was conducted to analyze the performance improvement for preceding vehicles under conditions with limited FOV and detection range. Third, simulations of the representative scenarios when using the DRL-based longitudinal motion planning algorithm were performed. The performance was compared to that of the algorithm using DP. The advantages and disadvantages in terms of characteristics of the solution and calculation time were investigated.

6.1.1.1. Simulation Results when using Proposed Algorithm

Fig. 6.1 shows the simulation results against a stationary target when implementing the proposed algorithm. The target is detected at the initial clearance of 40 m, as

shown in Fig. 6.1(d). The ego vehicle detected a stationary target at the ego vehicle's speed of 40 kph, as shown in Fig. 6.1(a). Fig. 6.1(b) and Fig. 6.1(c) show the longitudinal acceleration and desired acceleration profiles. The deceleration profiles were gradual without a peak, and the minimum deceleration was decreased. The maximum deceleration when using the proposed algorithm was 2.3 m/s^2 . Compared to the conventional algorithm, it was reduced by 107 %. In addition, the jerk was significantly decreased, as depicted in Fig. 6.1(f). The tendency of the deceleration profile is similar to that of the human driver; the deceleration starts softly and gradually increases as it approaches the preceding vehicle.

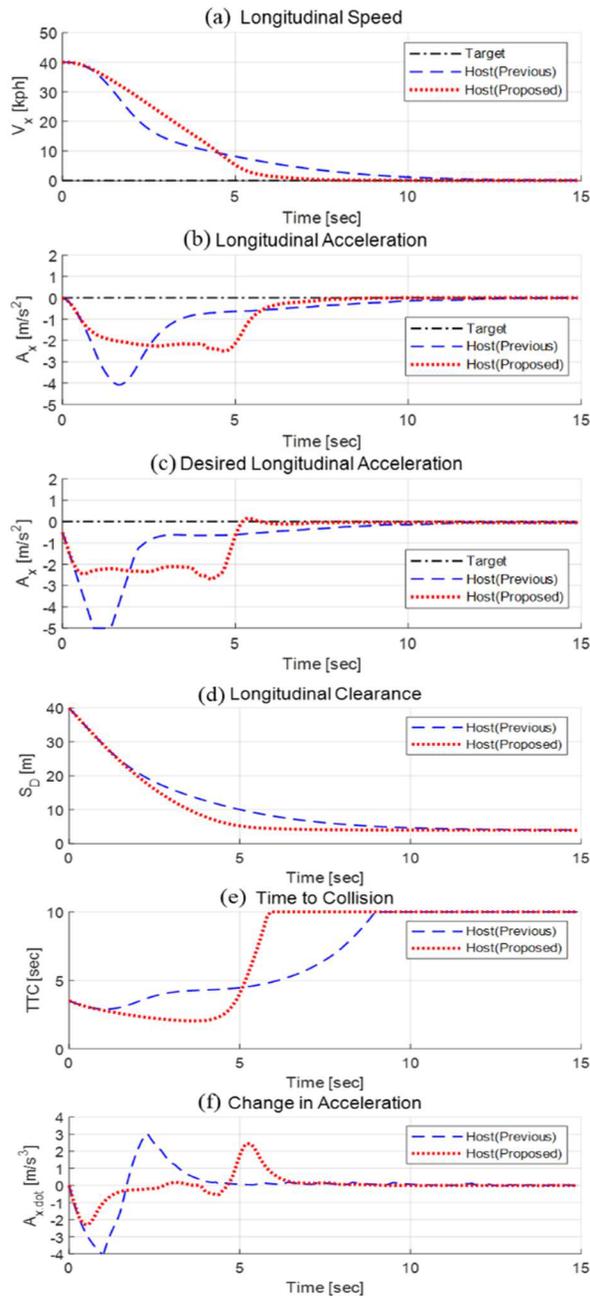


Fig. 6.1. Simulation results when implementing DP-based algorithm: a stationary target detected at the initial clearance of 40 m.

Fig. 6.2 shows the simulation results when using the proposed algorithm against a slowly moving target. As shown in Fig. 6.2(a) and Fig. 6.2(b), the ego vehicle starts to decelerate gradually at first. Then, the deceleration increases as it approaches the preceding vehicle. The preceding vehicle can be recognized at 40 m, which is the effective detection range assumption. The jerk that occurs at the time of recognition is -0.8 m/s^3 , which is reduced by 69 % compared with that of the conventional algorithm. It can be seen that the increase in deceleration as the ego vehicle approaches the preceding vehicle is the result of the fuzzy tuned MPC. As shown in Fig. 6.2(e), the TTC at the time of recognition is not large; a small deceleration is calculated at the beginning. As it approaches, a large deceleration is generated by the decreased TTC.

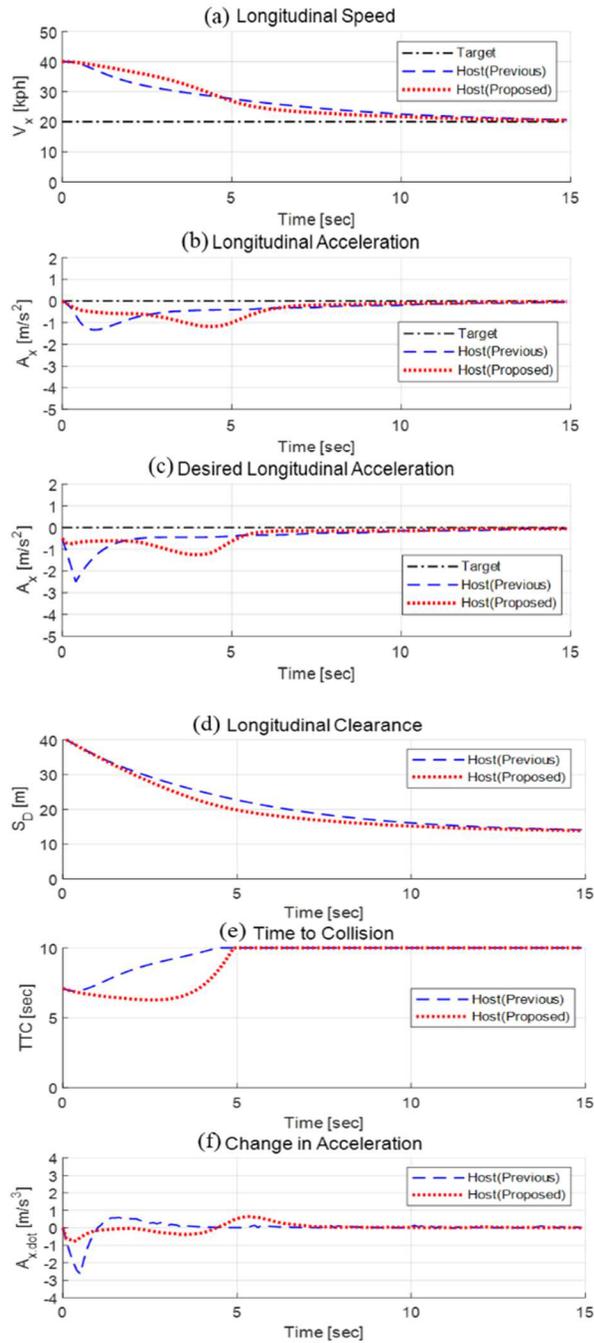


Fig. 6.2. Simulation results when implementing DP-based algorithm: a slowly moving target with vehicle speed of 20 kph detected at the initial clearance of 40 m.

Fig. 6.3 shows the simulation results of the longitudinal motion planning against the target, which cuts in severely at the clearance of 10 m with the same speed as the ego vehicle and stops urgently. As shown in Fig. 6.3(a), when the preceding vehicle cuts in, the speed of the ego vehicle is assumed to be the same as that of the preceding vehicle. In this case, TTC^{-1} becomes zero. As long as the minimum clearance is secured, it is not a dangerous situation as long as the minimum clearance is increased. As the fuzzy-tuned MPC is applied, the deceleration is reduced to 50 % compared to the conventional algorithm, as shown in Fig. 6.3(b). Fig. 6.3(a) shows that the preceding vehicle decelerates rapidly. Since it is a dangerous situation, the proposed approach is operated conservatively than the conventional algorithm, as shown in Fig. 6.3(b).

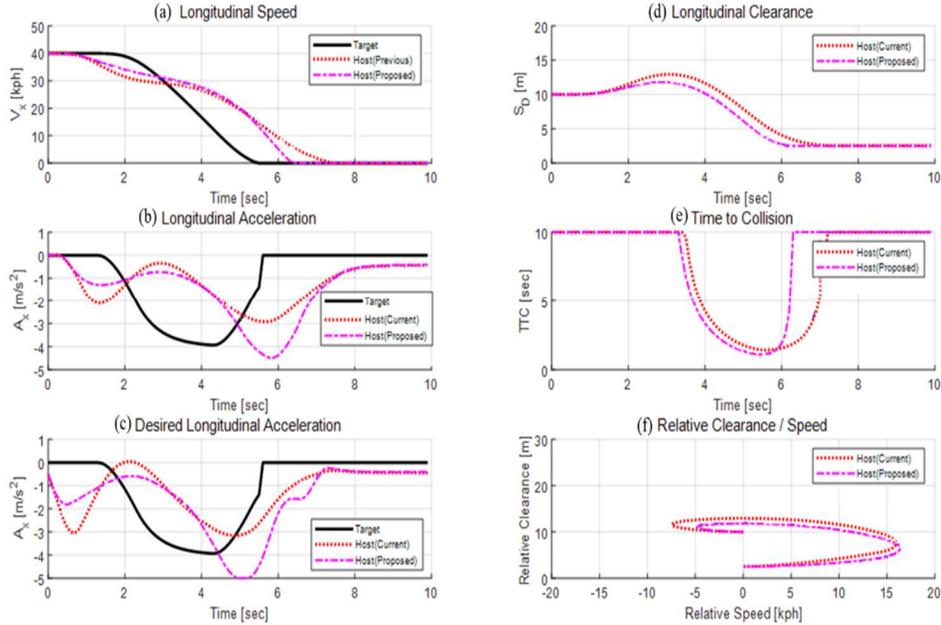


Fig. 6.3. Simulation results when implementing DP-based algorithm: a moving target with vehicle speed of 40 kph which cuts-in at the initial clearance of 10 m (ego vehicle speed: 40 kph).

6.1.1.2. Simulation Results when using DRL-based Algorithm

Fig. 6.4 shows the simulation results when using the DRL-based longitudinal motion planning algorithm. In the simulation, the weights of the MPC when using the conventional algorithm were set as the nominal weights of the DRL. It can be seen from Fig. 6.4(b) that the deceleration profile is gradual without a peak, and the minimum deceleration is smaller than that of the conventional algorithm. The maximum decelerations when using the proposed and previous algorithms were 2.5 m/s^2 and 2.8 m/s^2 . Compared to the conventional algorithm, it was reduced by 10.7%. In addition, the jerk was significantly decreased, as depicted in Fig. 6.4(c). The tendency of the deceleration profile is similar to that of the human driver and those

when using the proposed algorithm. The deceleration starts softly and gradually increases as it approaches the preceding vehicle. It can be seen that the planned motion from the proposed algorithm is more optimal than that from the conventional algorithm. The proposed algorithm's planned motion by the DRL is the result of long-time-horizon optimization. This time-horizon covers the range in which the vehicle reaches the target position. As a result, the planned motion is closer to the exact solution.

As shown in Fig. 6.4(d), the times for arriving at the target position using the proposed and conventional algorithms are 7 sec and 9 sec, respectively. Fig. 6.4(e) and Fig. 6.4(f) show the TG and TTC of the ego vehicle. The TG and the time when the ego vehicle reaches the target position are related to traffic flow efficiency. The difference in the traffic flow efficiency is related to the weights of the objective function in DRL and MPC formulations. In the proposed algorithm, the weights of acceleration and jerk are designed to be proportional to the ego vehicle's speed by imitating the human driver's characteristics. The modified weights results have the effect of dividing the weight for traffic flow efficiency by speed. As a result, the traffic flow efficiency at a low speed is improved.

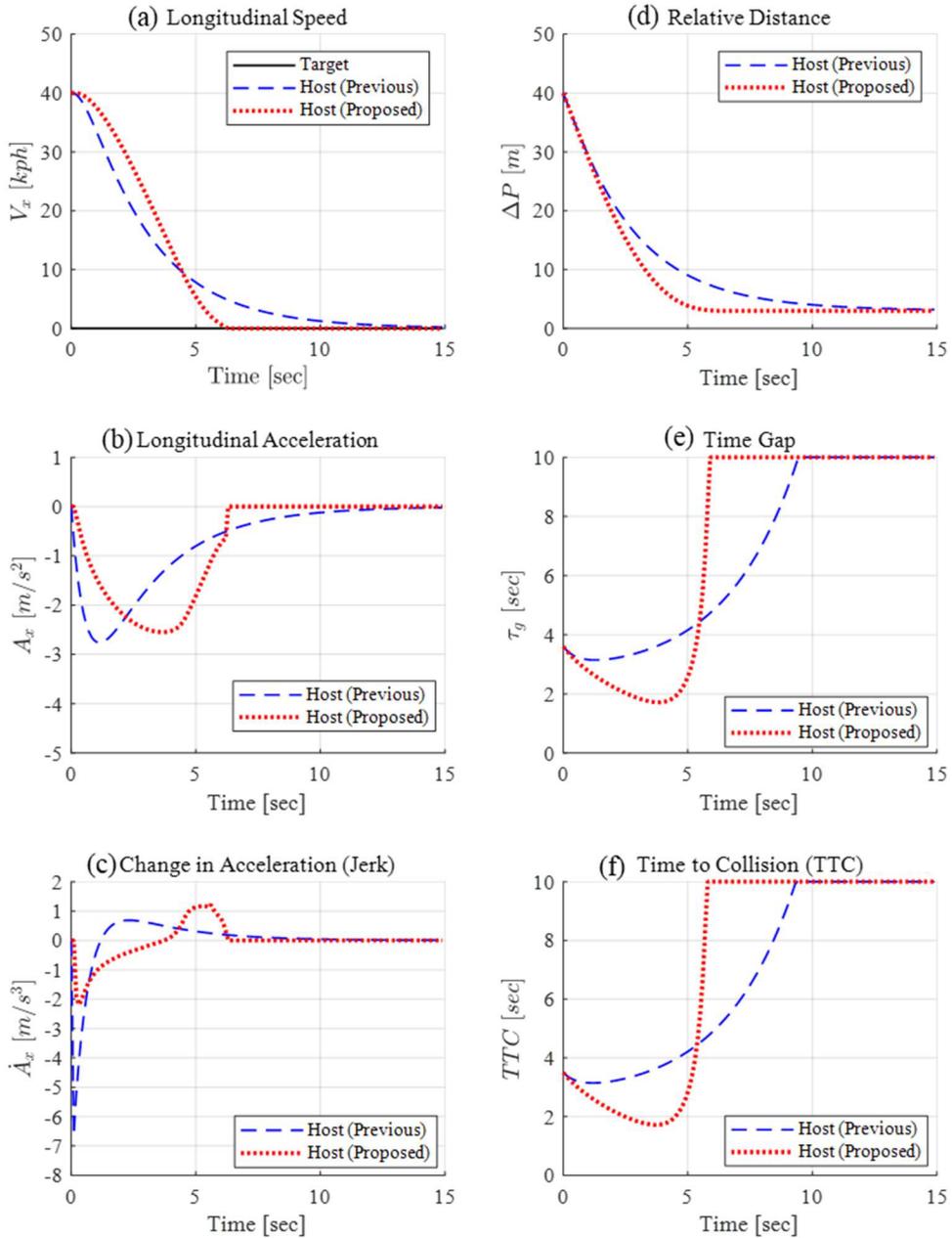


Fig. 6.4. Simulation results when using DRL-based algorithm: a stationary target detected at the initial clearance of 40 m.

It can be observed from Fig. 6.1 and Fig. 6.4 that both algorithms can reduce the maximum deceleration and jerk compared to the conventional algorithm.

Moreover, the traffic flow efficiency could be improved compared to the conventional algorithm. Although the deceleration and jerk profiles differ because of the differences in the weights implemented in the DP and DRL algorithms, it can be confirmed that both algorithms can improve ride comfort and traffic flow efficiency.

Fig. 6.5 shows the simulation results when using the DRL-based algorithm against a slowly moving target. As shown in Fig. 6.5(c), the jerk that occurs at the time of recognition from the proposed algorithm is -0.7 m/s^3 , which is reduced by 72 % compared with that of the conventional algorithm. It can be seen that the increase in deceleration as the ego vehicle approaches the preceding vehicle is the result of the risk index-based weights in reward function of DRL. As shown in Fig. 6.5(f), the TTC at the time of recognition is not large; a small deceleration is calculated at the beginning. As it approaches, a large deceleration is generated by the decreased TTC.

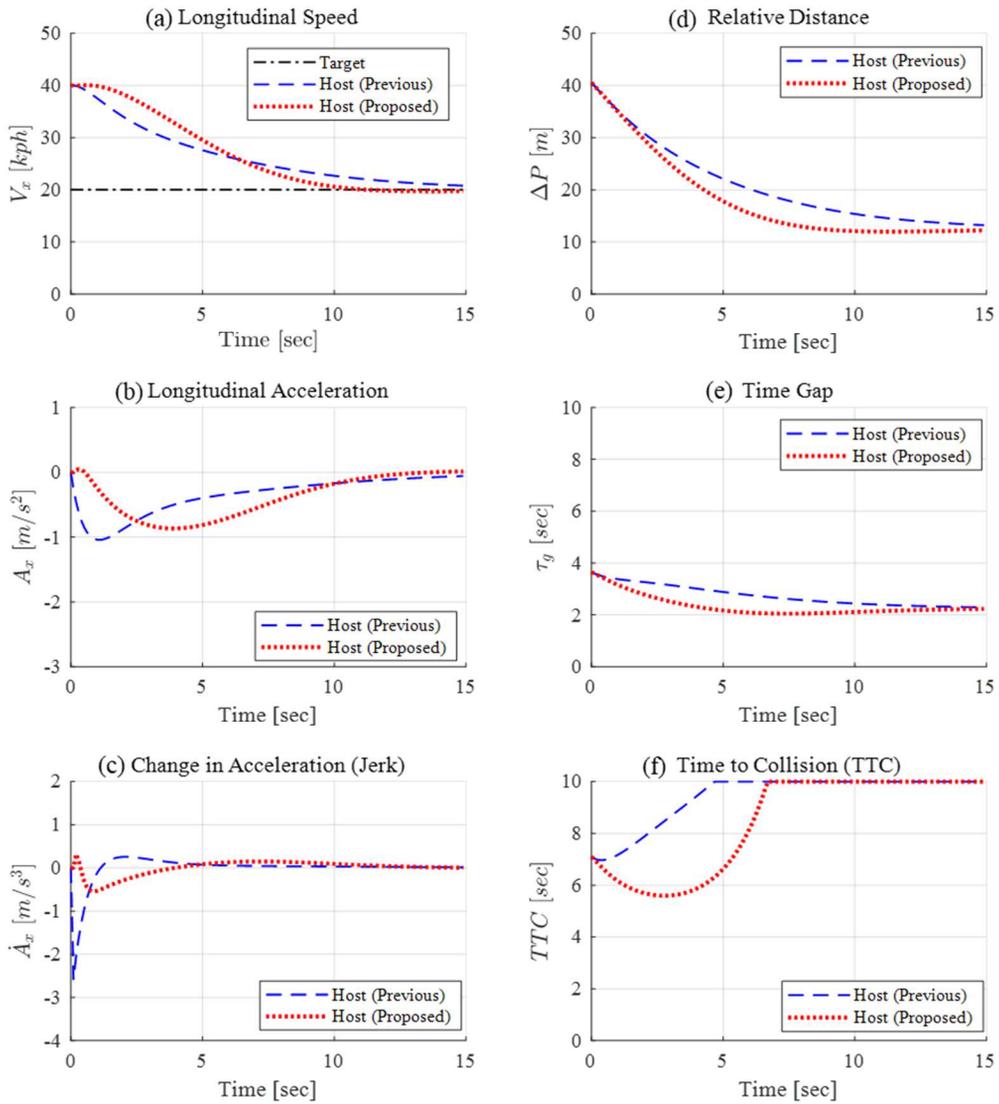


Fig. 6.5. Simulation results when implementing DRL-based algorithm: a slowly moving target with vehicle speed of 20 kph detected at the initial clearance of 40 m.

Fig. 6.6 shows the simulation results when implementing the DRL-based algorithm against the target, which cuts in severely at the clearance of 15 m with the

same speed as the ego vehicle and stops urgently. As shown in Fig. 6.6(a), when the preceding vehicle cuts in, the speed of the ego vehicle is assumed to be the same as that of the preceding vehicle. In this case, TTC^{-1} becomes zero. As long as the minimum clearance is secured, it is not a dangerous situation as long as the minimum clearance is increased. As the risk index-based tuning of weights is applied, the deceleration at cut-in is reduced to 20 % compared to the conventional algorithm, as shown in Fig. 6.6(b). The preceding vehicle decelerates rapidly from 3 sec to 7 sec, as shown in Fig. 6.6(b). Similar to the DP-based algorithm, the DRL-based approach was operated more conservatively than the conventional algorithm.

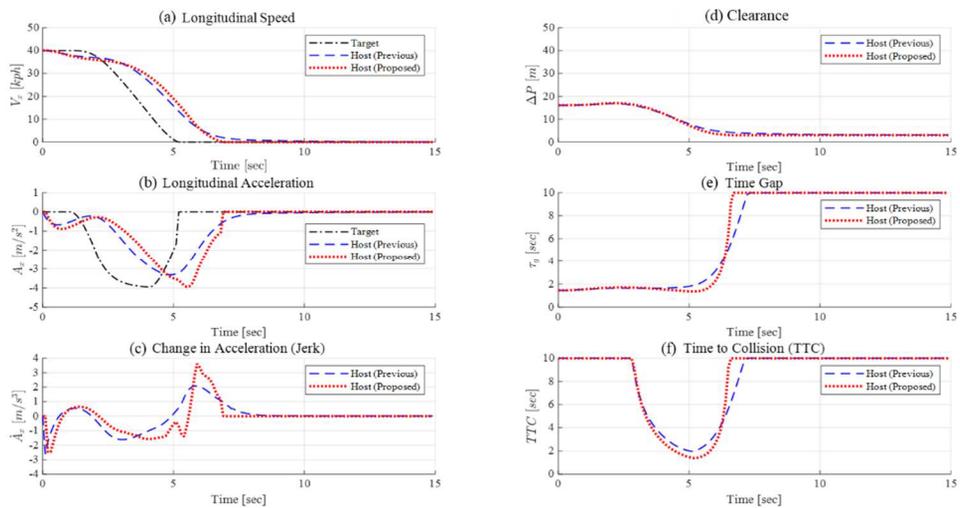


Fig. 6.6. Simulation results when using DRL-based algorithm: a moving target with vehicle speed of 40 kph which cuts-in at the initial clearance of 15 m (ego vehicle speed: 40 kph).

It can be observed from Figs.6.1 to 6.6 that both algorithms can reduce the maximum deceleration and jerk compared to the conventional algorithm. Moreover,

the traffic flow efficiency could be improved compared to the conventional algorithm. Since the weights for the DP and DRL algorithms were slightly different, the deceleration and jerk profiles were also different. Nevertheless, it can be seen that both algorithms improve ride comfort and traffic flow efficiency compared to the conventional algorithm.

In Fig. 6.6, the DRL-based algorithm computed a somewhat irregular profile when calculating the control input. Particularly, as shown in Fig. 6.6(c), the chattering was observed at 6 sec. It seems that the control input is not fully optimized because of the limitation of the learning-based algorithm.

Monte-Carlo simulation was conducted for targets detected from different ranges to improve credibility. The simulation was performed assuming that the ego vehicle's speed was 50 kph. The target vehicle speed was randomly selected by 100 samples using a uniform distribution. The detection range is set as 30 m to 100m, as shown in Fig. 6.7. As shown in Fig. 6.7(a), the deceleration is decreased significantly for FOV ranges from 40 m to 70 m. As shown in Fig. 6.7(b), the jerk is also decreased for FOV ranges from 30 m to 80 m. In the section with a clearance of 30 m, the effect of the improvement of ride comfort in terms of acceleration when implementing the proposed algorithm is insignificant. It can be understood that as the target is detected at a short clearance, the short time-horizon planning using chance-constrained MPC is carried out as a conservative behavior for safety from a collision. Consequently, it is confirmed that the ride comfort and traffic flow efficiency while securing driving safety can be improved by implementing the proposed algorithm.

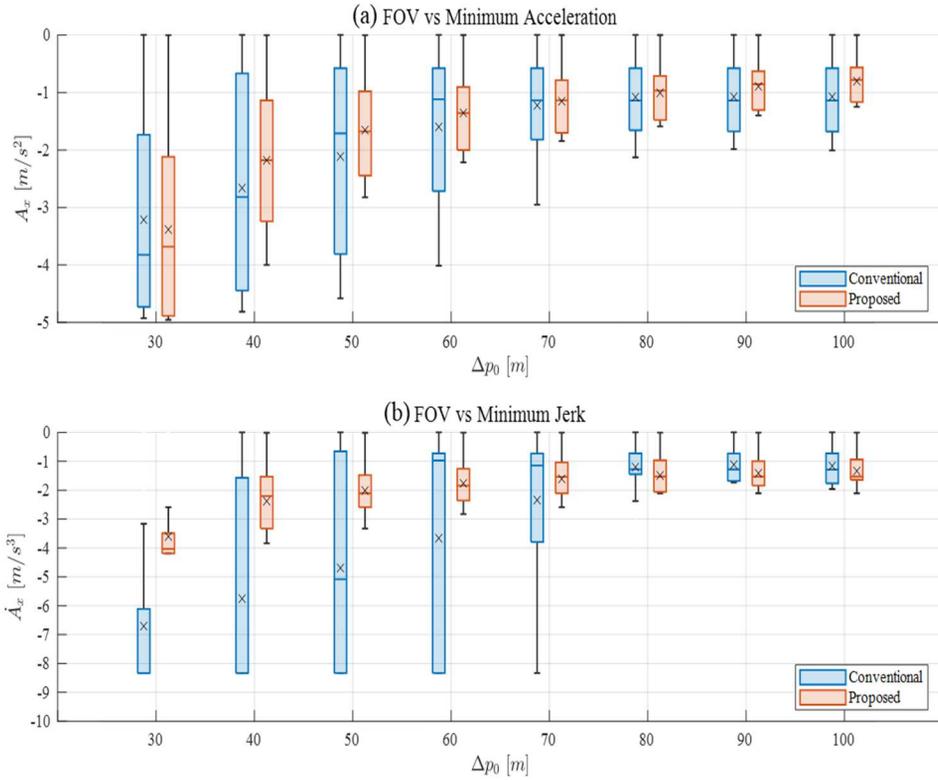


Fig. 6.7. Simulation results when implementing proposed algorithm: Monte-Carlo simulation for different FOV ranges from 30 m to 100 m.

The computational efficiency was analyzed to confirm whether the proposed algorithm is applicable to the actual automated vehicle. The calculation time is only the time for calculating the long time-horizon motion planning algorithm. The calculation time for pre-processing, post-processing, and other interworking logics required to implement the motion planning algorithms was excluded. In the simulation for calculating the calculation time, all algorithms used the same scenario. The scenario for a stationary vehicle was adopted. The algorithms were implemented in C++ and python and calculated on a PC equipped with an i7-9700K 3.6 GHz CPU. The analysis compared the results from the conventional algorithm using only MPC,

the proposed algorithm using DP and MPC, the algorithms for comparative study (DRL and MPC, DRL with 20 steps horizon, DRL with 1-step horizon), and interior-point optimization (IPO). The IPO refers to an MPC that uses the same objective function of DRL, and the same dynamics and constraints applied to chance-constrained MPC. The IPO was formulated for a 20-second time-horizon.

Table 6.1 shows the calculation time for each algorithm. As shown in Table 6.1, the mean and maximum computational times when implementing the proposed algorithm (DL and MPC) are 4.2014 msec and 9.2064 msec, respectively. Among the six algorithms excluding IPO, the DRL-based algorithm had the longest calculation time. When using only the DRL, the reference motion for the current state was calculated once. However, when the algorithm was applied to the MPC, reference motion had to be calculated for the prediction horizon of the MPC. Thus, at every calculation, the DRL was calculated 20 times, which is the prediction time step of MPC. As a result, the computation time of the DRL-based algorithm increased compared to other algorithms. In an actual test vehicle, the calculation cycle of the perception and localization modules is 40 msec. Therefore, it can be confirmed that the six algorithms excluding IPO are applicable to the actual vehicle.

Table 6.1. Summary of calculation time of proposed and conventional algorithms

Algorithm	Calculation Time [msec]		
	Min.	Max.	Mean
Proposed (DP only)	3.6351	9.2064	4.2014
Proposed (DP+MPC)	4.7574	10.4151	5.4487
MPC	0.9901	2.3889	1.1313
DRL+MPC	5.1784	17.6079	5.9058
DRL Only, 20 steps	3.3683	16.2694	3.8862
DRL Only, 1 step	3.1768	16.0691	3.7578
IPO	388.5	8,631	746.7

6.1.2. Vehicle Test Results

Vehicle tests were conducted to confirm the applicability from actual vehicle tests. The vehicle test results when implementing the conventional algorithm and proposed algorithm for a stationary target are presented in Fig. 6.8 and Fig. 6.9. In order to limit the surrounding obstacles, the test was conducted in FMTC. Fig. 6.8(a) shows that the speed was 38 kph when the ego vehicle recognized the other vehicle. At this time, the control input when using conventional algorithm applied to the ego vehicle is -2.5 m/s^2 , as shown in Fig. 6.8(b). What is worse, the vehicle started to decelerate quickly and used large jerk input as the target vehicle was recognized. The ego vehicle is stopped at 11.3 sec, as shown in Fig. 6.8(c). It takes 10.5 sec for the ego vehicle to stop after recognizing the stationary target.

As shown in Fig. 6.9(b), the initial control input when using proposed algorithm

is -1.1 m/s^2 . After that, the deceleration gradually increases, reaching -1.9 m/s^2 from 4.5 sec. The deceleration is then slowly decreased again, and the ego vehicle is stopped at 7.5 sec, as shown in Fig. 6.9(a). It takes 6.8 sec for the ego vehicle to stop after recognizing the stationary target when using the proposed algorithm.

It can be seen from Fig. 6.8 and Fig. 6.9 that the maximum deceleration of the proposed algorithm was reduced by 24% compared to that of the conventional algorithm. The time taken to reach the target position when using the proposed algorithm was reduced by 35% compared to that when implementing the conventional algorithm. The results show that the ride comfort and traffic flow efficiency of the proposed algorithm are improved for the stationary target.

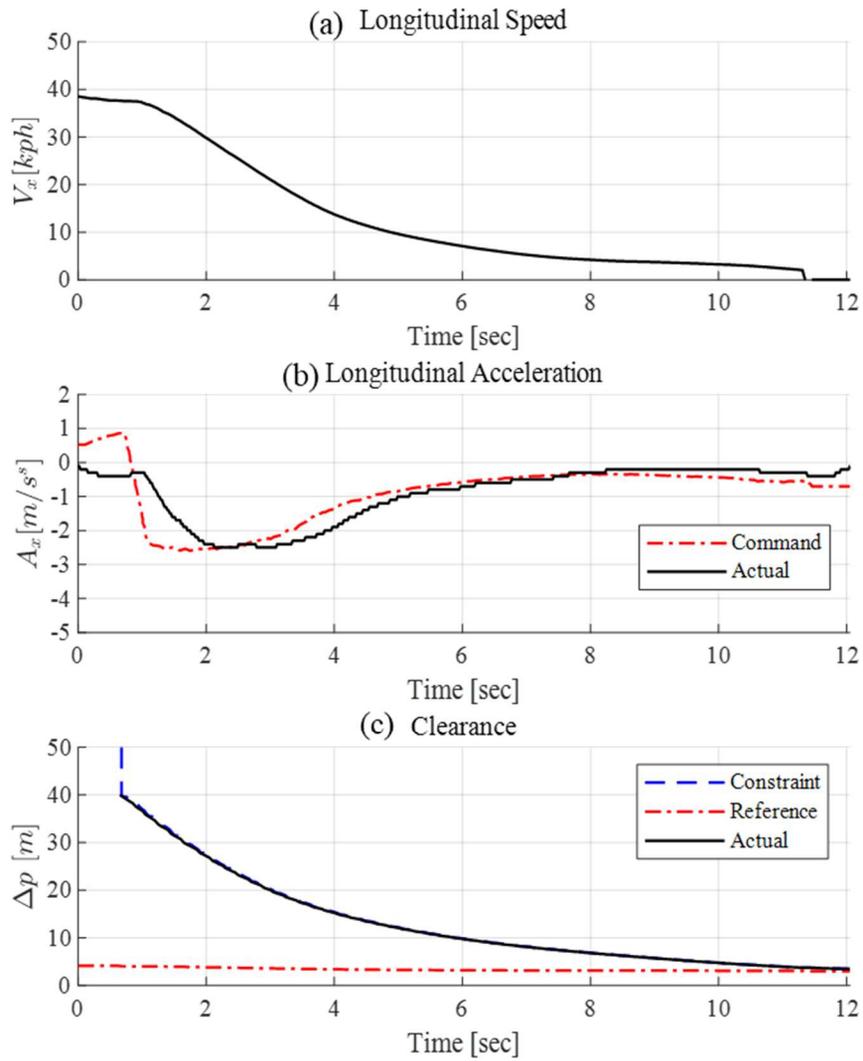


Fig. 6.8. Vehicle test results when implementing the conventional algorithm for a stationary target detected at the initial clearance of 40 m.

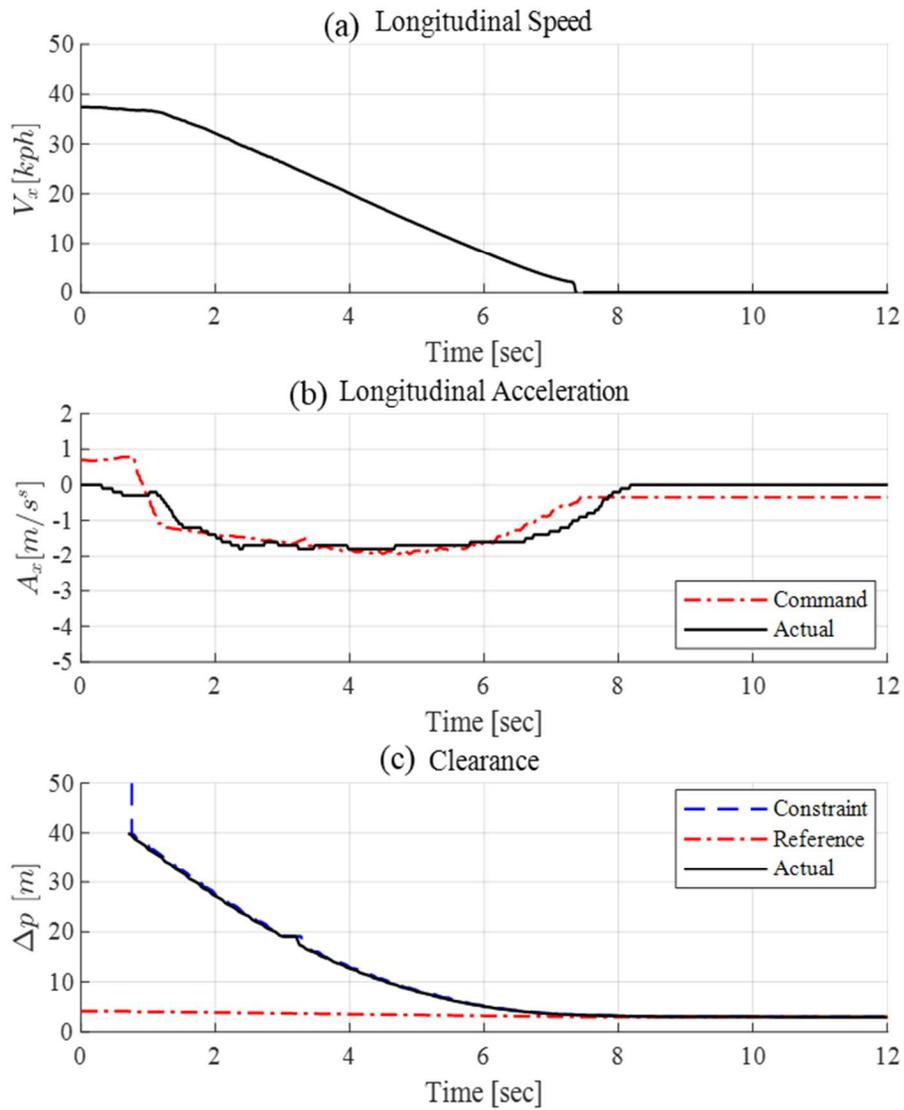


Fig. 6.9. Vehicle test results when implementing the proposed algorithm for a stationary target detected at the initial clearance of 40 m.

Fig. 6.10 shows the vehicle test results to confirm the applicability of the proposed algorithm against slow moving targets. In this test, the detection range was

set as 30 m. Fig. 6.10(a) shows that the vehicle initially travels at 40 kph. Then, as shown in Fig. 6.10(c), a preceding vehicle traveling at a low speed of 20 kph was recognized. At this moment, the desired acceleration was obtained as -0.8 m/s^2 , as depicted in Fig. 6.10. The deceleration profile was calculated almost evenly without a peak. Therefore, it can be observed that the proposed algorithm improves ride comfort for low-speed vehicles.

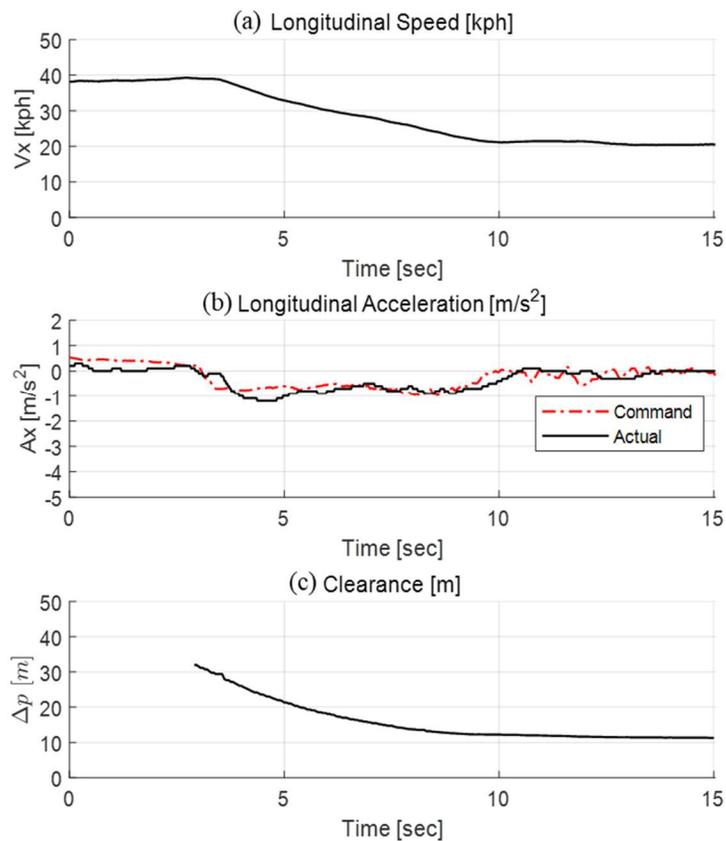


Fig. 6.10. Vehicle test results of the conventional algorithm for a slowly moving target (20 kph) detected at the initial clearance of 30 m.

6.2. Performance Evaluation of Lateral Motion Planning

6.2.1. Vehicle Test Results in FMTC Track

Vehicle test was performed in FMTC to evaluate the feasibility of the proposed algorithm. It was difficult to obtain the actual value of the bias in heading error. Therefore, by adding a known bias, the performance of the proposed disturbance estimation using MHE was examined. As a reference, disturbance estimation using an extended Kalman filter (EKF) was developed and compared with the estimation implementing MHE.

Fig. 6.11 and Fig. 6.12 show the vehicle test results when implementing the proposed and reference algorithms with additional, constant heading bias of -1 deg and -2 deg, respectively. Fig. 6.11(a) shows that the measured heading error is biased at about -2 deg. It can be seen from Fig. 6.11(b) that the lateral position error occurred at about 0.3 m when the heading bias of 1 deg is added. The additional bias was estimated using the EKF or MHE. Fig. 6.11(c) shows the estimated heading bias. As shown in Fig. 6.12(a), the measured heading error was about -3 deg; this is a gap of 1 deg compared with that in Fig. 6.11(a). The lateral position error was generated at about 0.6 m when the heading bias of 2 deg was added. This bias was also estimated using the EKF or MHE. Fig. 6.12(c) depicts the estimated heading bias. As a result, it can be confirmed that the additional heading bias can be attenuated when implementing the offset-free MPC with EKF or MHE. Among them, the lateral position error is kept smaller when implementing MHE than when using EKF, as

shown in Fig. 6.11(b) and Fig. 6.12(b). It can be seen that the convergence rate of MHE is faster than that of EKF.

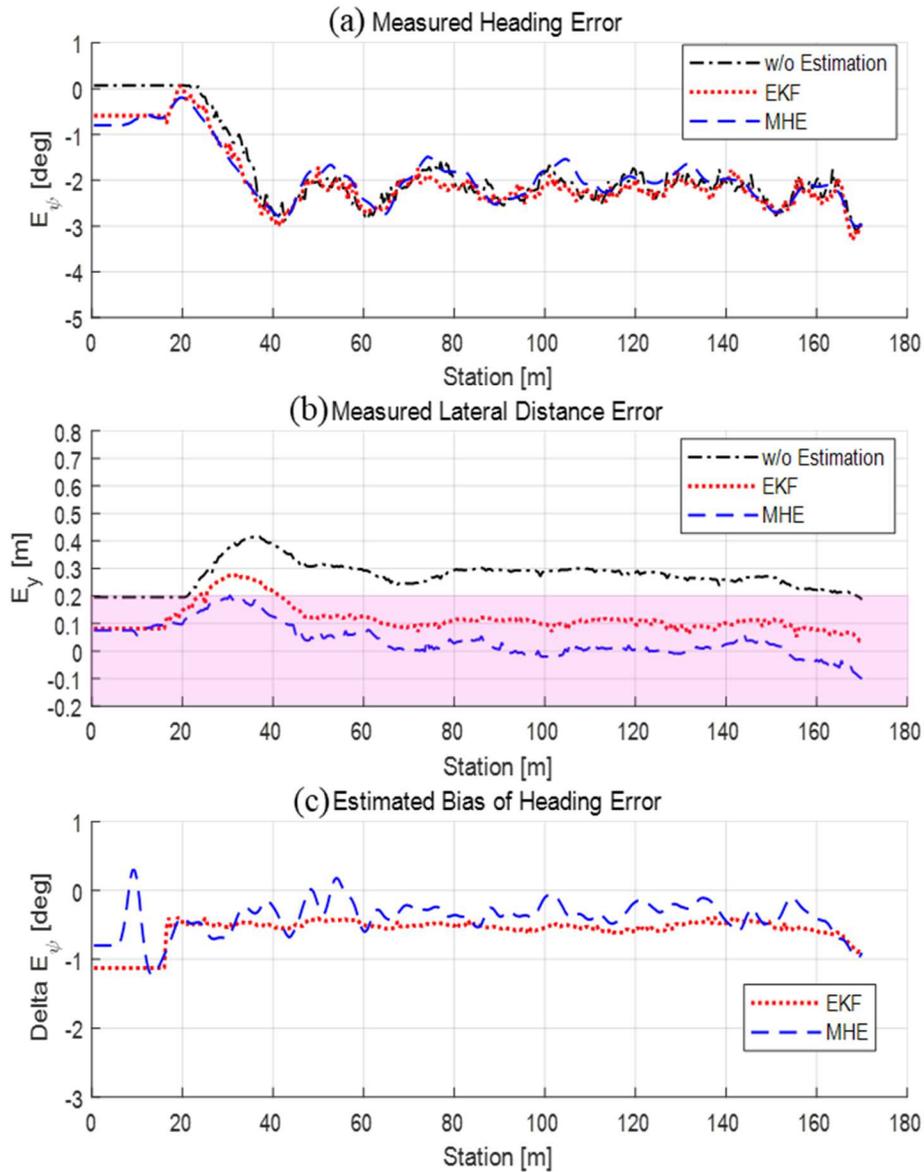


Fig. 6.11. Vehicle test results when implementing the proposed and reference algorithms with an additional heading bias of -1 deg.

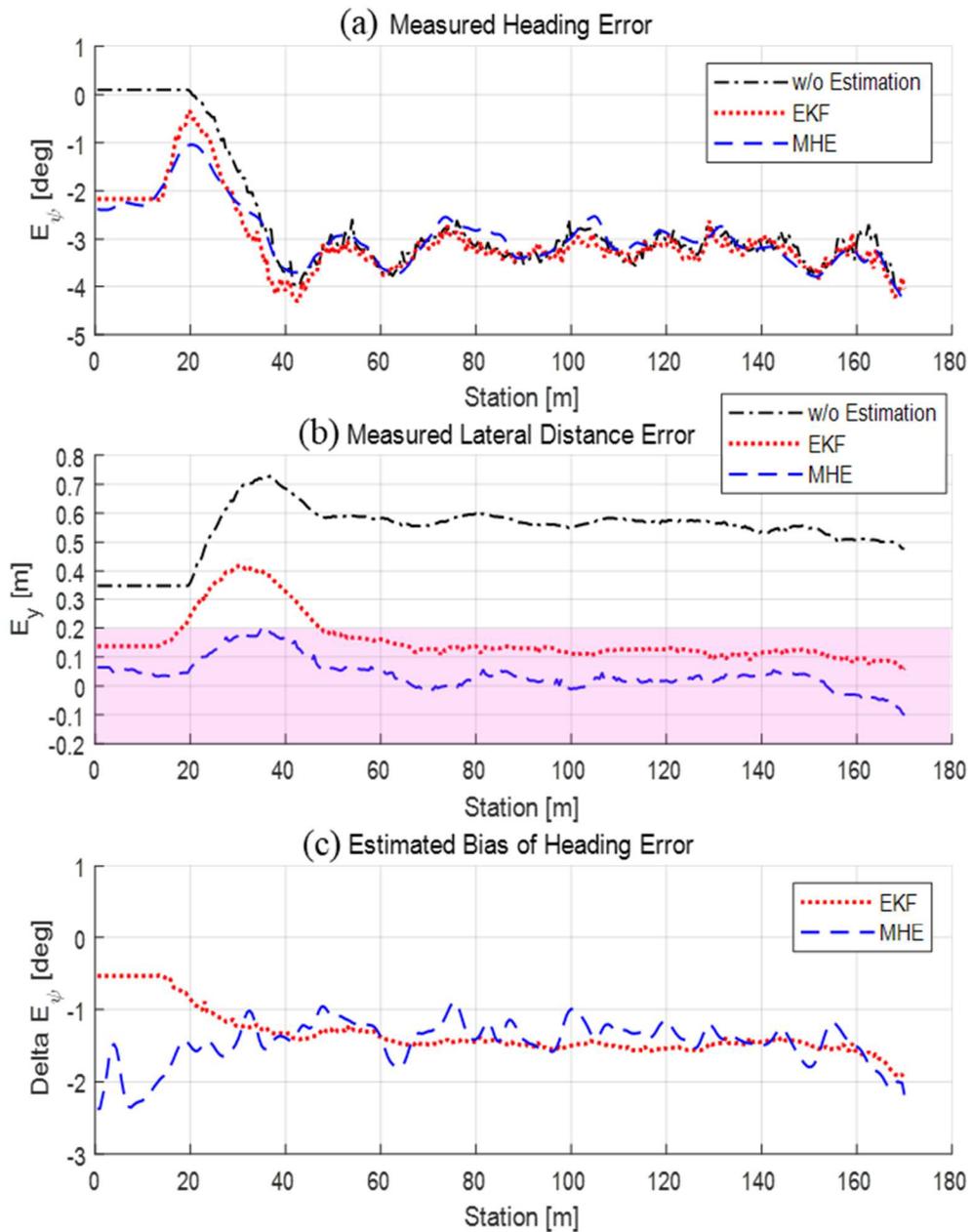


Fig. 6.12. Vehicle test results when implementing the proposed and reference algorithms with an additional heading bias of -2 deg.

Fig. 6.13 shows the vehicle test results when implementing the proposed algorithm on curved roads and straight roads in the FMTC track. The test was carried out by adding an oscillatory bias with an amplitude of 1 deg. The vehicle test when using the offset-free MPC with EKF was dangerous in actual driving conditions; therefore, only the offset-free MPC with MHE was investigated. From the test results, it can be confirmed that the proposed algorithm is robust to the changing disturbance. Fig. 6.13(c) shows that the oscillating bias can be estimated using MHE. As shown in Fig. 6.13(b), the lateral position error is smaller than the allowable gap (0.2 m) of the BRT course.

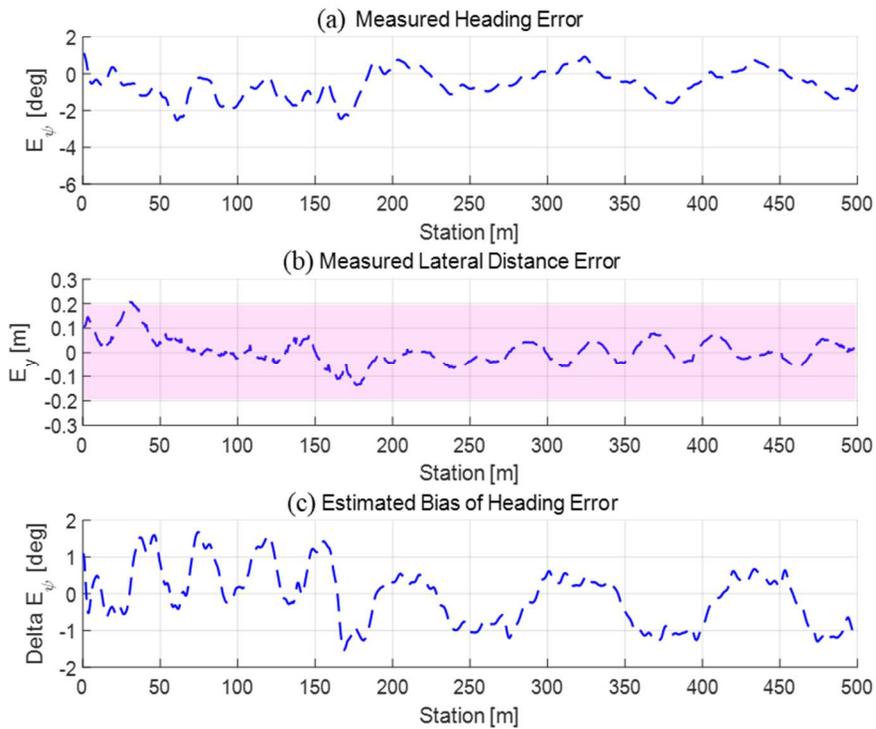


Fig. 6.13. Vehicle test results when implementing the proposed and reference algorithms with an additional heading bias of ± 1 deg.

6.2.2. Vehicle Test Results in Urban Bus-only Lanes

The applicability of the algorithm was examined by the vehicle tests in the BRT course. The test course includes two underpasses, as previously described in Fig. 2.5. The test results for the entire driving course are shown in Fig. 6.14. In Fig. 6.14, the regions marked with green shade is the section where the road shape is bent, and the errors along the road could be chattering. When disregarding these sections, the lateral position error was smaller than the allowable gap (0.2 m) of the BRT lanes. The lateral position error when using the conventional MPC algorithm with the drivable corridor determination is distributed from -0.1793 m to 0.2662 m, excluding the outliers. The lateral position error when implementing the proposed algorithm is distributed from -0.1123 m to 0.1820 m. Through these results, it can be seen that the proposed algorithm can ensure better performance in the urban driving environments with localization uncertainty.

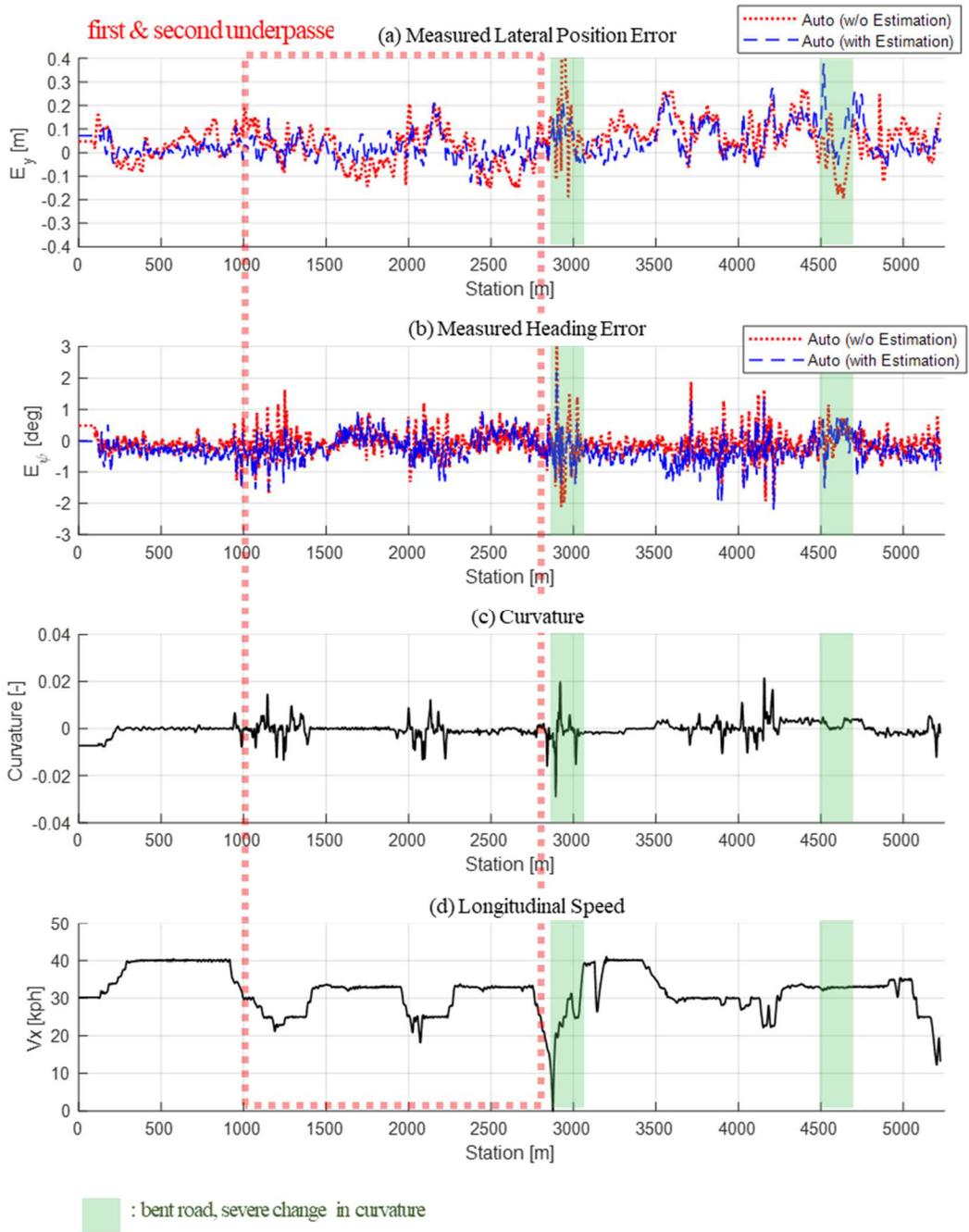


Fig. 6.14. Vehicle test results when implementing the proposed and reference algorithms in BRT course.

Fig. 6.15 shows the vehicle test results in the first and second underpasses among the three underpasses in the driving course. Fig. 6.15(a) shows the lateral position error. The vehicle test results using both algorithms were smaller than the allowable gap (0.2 m) of the BRT lanes. When entering the underpass, it can be observed from Fig. 6.15(c) that the estimated bias of heading error is changed from -0.5 deg to 0.4 deg. In the underpass, this heading bias could cause the vehicle to shift closer to the sidewall on the right side of the bus. To avoid the collision with the wall, the drivable corridor determination was utilized for both algorithms in the vehicle tests. As a result, it was possible to drive the course using the previous algorithm without using the disturbance estimation. However, the motion in the lateral direction showed very dangerous. On the other hand, the vehicle test result when implementing the proposed algorithm shows that the lateral position error is maintained within -0.1 m to 0.1 m in the two underpasses, as shown in Fig. 6.15(a). It can be seen from the results that despite the uncertainty in the localization module under actual driving conditions, the effect of the heading bias and lateral position error on the lateral motion planning could be attenuated.

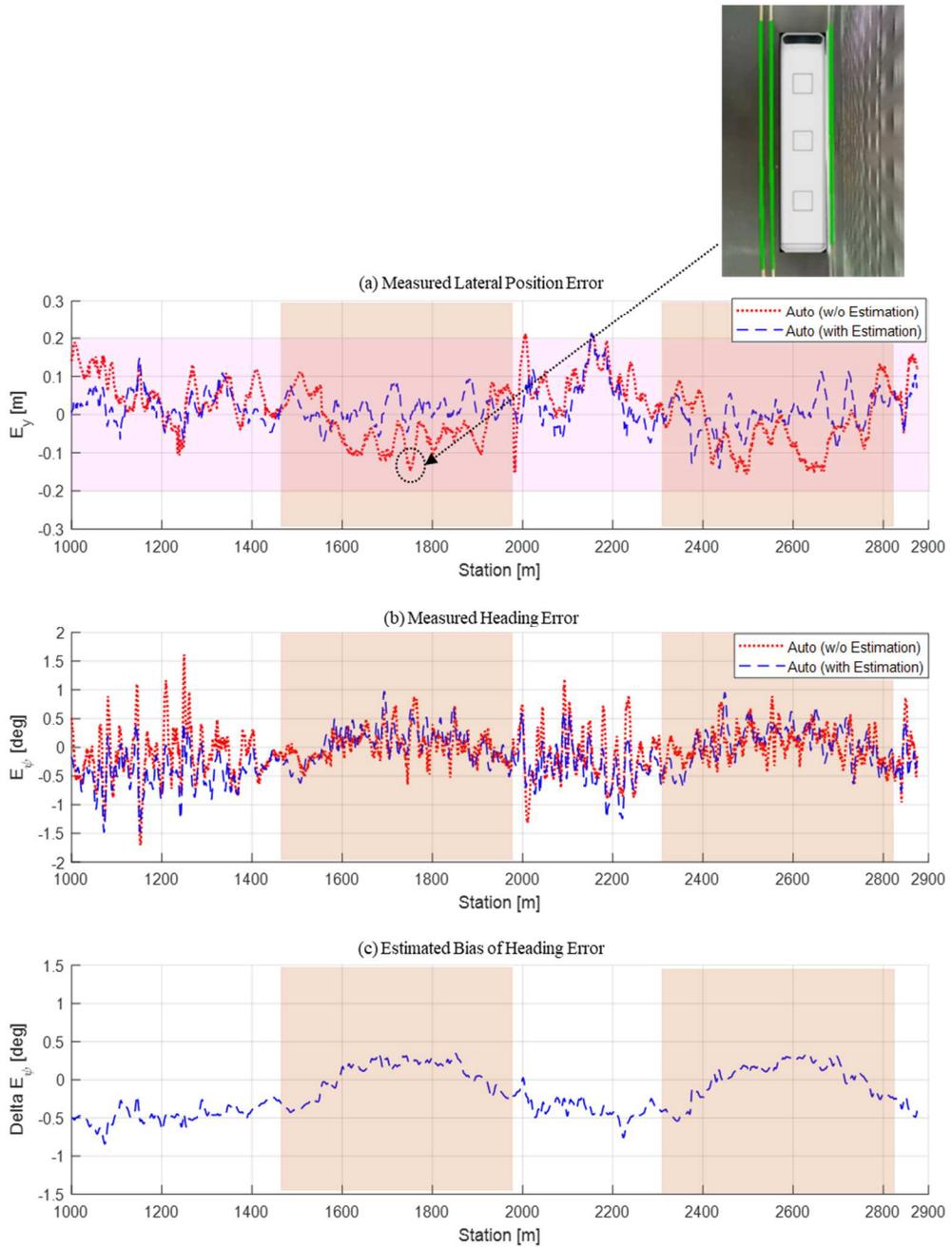


Fig. 6.15. Vehicle test results when implementing the proposed and reference algorithms in the underpasses of BRT course.

6.3. Performance Evaluation of Longitudinal Motion

Control

6.3.1. Computer Simulation Results

In this section, computer simulations were conducted to evaluate the robustness of the proposed longitudinal motion control algorithm. As outlined in Fig. 6.16, the simulation environment consists of the controller implemented using MATLAB/Simulink and a vehicle model using Mechanical Simulation Corporation's TruckSim software [78]. In the simulation, a full-sized tour-bus model with an engine torque/brake pressure interface was implemented. The vehicle mass is 12,285 kg, and the maximum engine power is 175 kW with 7-speed automatic transmission.

To construct a simulation model similar to the actual dynamics of the vehicle, the throttle control and brake system models were added to the vehicle model of the software. These two systems were represented using a first-order delay model, which considered the response delay for the control commands. The gain of each model is 1, and the time constants are 0.433 and 0.085, respectively. The brake pressure input ranges from 0 to 0.7 MPa. The brake torque to pressure gain is 10,700 Nm/MPa. In the brake system model, the non-dimensionalized brake command is converted to the brake pressure.

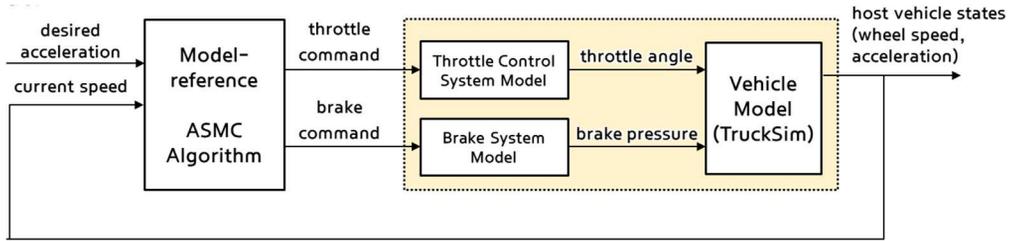


Fig. 6.16. Outline of the computer simulation for the proposed longitudinal motion control algorithm.

The profile of reference longitudinal speed was defined for the simulation. The speed profile was modeled considering the actual vehicle's acceleration range to maintain the relative speed and distance between the host and target vehicles. The speed ranges from 10 to 60 kph (Fig. 6.17 (a)), which is a general urban driving situation.

To investigate the performance of the proposed algorithm, PID, model-reference adaptive control (MRAC), and SMC were selected as reference controllers. Because this research aimed to improve the performance of the lower-level controller, the upper-level controller adopted the same MPC logic presented above.

The PID controller used P-gain and I-gain, and their values were 0.1, respectively. Instead of D-gain, the feed-forward control input corresponding to the desired acceleration was additionally used to improve the response speed, and it was 0.35. The MRAC controller used the same adaptation algorithm and parameters as the proposed controller, except for the switching control input of the SMC scheme. Similarly, the SMC controller did not implement the adaptation parameter, and the other parameters used the same value, except for the RBFNN-based upper bound estimation. Through these comparison groups, the effect of the adaptation algorithm and the bound estimation based on RBFNN could be investigated. All controllers

used equivalent control input to compensate for the zero-throttle acceleration and applied the same mode switching algorithm.

The controller design parameters used in the computer simulations and vehicle tests are summarized in Table 6.2. Since the environmental disturbances change frequently, the adaptation gains for $\hat{\eta}_2$ and $\hat{\eta}_3$ corresponding to the disturbances were set to be large. Conversely, the gain for $\hat{\eta}_1$ corresponding to the vehicle parameter was set to be relatively small. The parameters of the basis functions for RBFNN were selected considering the range of speed errors that occur in general driving conditions.

The adaptation parameters, $\hat{\eta}_1(t)$, $\hat{\eta}_2(t)$, and $\hat{\eta}_3(t)$, were converted into \hat{f}_{Rv} , \hat{f}_{R0} , and \hat{k}^i using (5.11) to help with the investigation of the results. These parameters represent the adaptation for the environmental forces and changes in vehicle parameters. Since \hat{f}_{Rv} and \hat{f}_{R0} are the parameters of environmental disturbances and uncertainties, they were initially set as zero. \hat{k}^i for throttle and brake modes are normalized control gain values, which were initially set as 1.

The simulation scenarios consisted of four; the first scenario was driving on flat ground, and the vehicle parameters were all nominal conditions as initial assumptions. The second scenario was driving on a slope, which ranges from -5 to +5%. The third scenario was when the mass was changed compared to the initial value; the mass was increased by 25%. Lastly, the fourth scenario was a change in the equivalent control input by 100%. The simulation results are summarized in Table 6.3.

In addition, a sensitivity analysis was conducted to examine the robustness of the controllers against parameter changes. As for the scenarios, the four scenarios

were also adopted. The simulation was performed when the parameter was increased compared to the initial condition. Then, the robustness of each controller according to the parameter change was reviewed by analyzing the RMS of speed tracking error and the derivative of acceleration (jerk).

Table 6.2. Design parameters of the proposed longitudinal motion control algorithm.

Symbol	Parameters	Value		Units
		Computer simulation	Vehicle test	
γ_1	Adaptation gain for $\hat{\eta}_1$	0.0005	0.0003	sec ² /m ³
γ_2	Adaptation gain for $\hat{\eta}_2$	0.005	0.008	sec ² /m ³
γ_3	Adaptation gain for $\hat{\eta}_3$	0.015	0.015	1/m
γ_4	Adaptation gain for RBFNN weights	0.01	0.01	1/m
μ_i	Center of the basis function, $h_i(s)$.	[-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5]	[-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5]	m/sec
ζ	Width of the basis function, $h_i(s)$	3.2	3.2	m/sec
Δt	Time step	0.01	0.01	sec
a_m	Gain of the reference model	0.1	0.1	1/sec
λ	Damping coefficient of the reference model	0.01	0.01	1/sec
ε_1	Positive constant for finite-time convergence	0.1	0.1	-

Table 6.3. Summary of the tracking performance from computer simulations when using the proposed longitudinal motion planning algorithm.

	Scenario	RMS of v_x error [kph]	RMS of a_x error [m/s ²]
Proposed	Flat ground	0.718	0.116
	Road slope (-5~5 deg)	0.796	0.119
	Mass change (+25%)	0.701	0.115
	Equivalent control input (+100%)	0.744	0.124
PID	Flat ground	1.594	0.154
	Road slope (-5~5 deg)	2.148	0.187
	Mass change (+25%)	1.959	0.174
	Equivalent control input (+100%)	2.086	0.188
MRAC	Flat ground	1.676	0.180
	Road slope (-5~5 deg)	1.677	0.189
	Mass change (+25%)	1.589	0.170
	Equivalent control input (+100%)	1.793	0.197
SMC	Flat ground	0.857	0.098
	Road slope (-5~5 deg)	1.566	0.121
	Mass change (+25%)	0.888	0.098
	Equivalent control input (+100%)	1.514	0.126

6.3.1.1. Performance on flat ground

To evaluate the performance of the proposed algorithm, the simulation was performed in normal driving conditions. Fig. 6.17 shows the control performance of the proposed method and the reference controllers on flat ground. Fig. 6.17 (a) shows the speed tracking performance. The model speed is the modified desired speed calculated from the reference speed model. As shown in Fig. 6.17 (b), the maximum

values of the speed tracking error of the proposed algorithm, PID, MRAC, and SMC, are about 1.293, 3.481, 3.133, and 1.382 kph, respectively. The RMS error of the proposed algorithm is 0.718 kph, which is improved by 54, 57, and 16 % compared to PID, MRAC, and SMC. Since the target acceleration value varies according to the speed tracking performance, the longitudinal acceleration of each controller is depicted in Fig. 6.17 (c). The jerk shown in Fig. 6.17 (c) is due to the change in the gear ratio (Fig. 6.17 (f)). It can be seen from Fig. 6.17 (a), (b), and (c) that all controllers show good tracking performance in all speed ranges. This is because, as described above, MRAC and SMC used the same design parameters as the proposed controller, and PID gain is designed to obtain similar performance to the proposed algorithm.

Fig. 6.17 (d) shows the parameter adaptation. The change of the adaptation parameters at 60 s is due to the vehicle model uncertainties; the change in the gear ratio (Fig. 6.17 (f)). The control input required to maintain a constant speed at the low-speed range is calculated differently, even for the same speed. In Fig. 6.17 (a), the speed at 130 s is similar to that at 10 s. Fig. 6.17 (e) shows that the control input at 130 s is different from that at 10 s. This model error causes significant tracking error at 85 s and 130 s as shown in Fig. 6.17 (b). PID and MRAC show considerable tracking errors.

Fig. 6.17 (e) shows the control input of the proposed algorithm. The control mode is changed at 50 s, 80 s, 102 s, and 144 s. As shown in Fig. 6.17 (a) and (c), the oscillatory motion is not observed at the control mode change. It can be seen that the control input is changed smoothly, even when switching from throttle to brake mode.

Fig. 6.18 shows the speed tracking performance for 50–60 s and 80–90 s. As illustrated in Fig. 6.18 (a), the speed decelerates from 50 s, and the proposed algorithm, MRAC, and SMC have smaller phase delays than PID. Similarly, as shown in Fig. 6.18 (b), the phase delays of the proposed algorithm and SMC are smaller than that of the MRAC and PID during acceleration since 80 s. It can be inferred from Fig. 6.17 and Fig. 6.18 that the SMC schemes of the proposed algorithm can attenuate the model uncertainty using the switching control input and improve tracking speed.

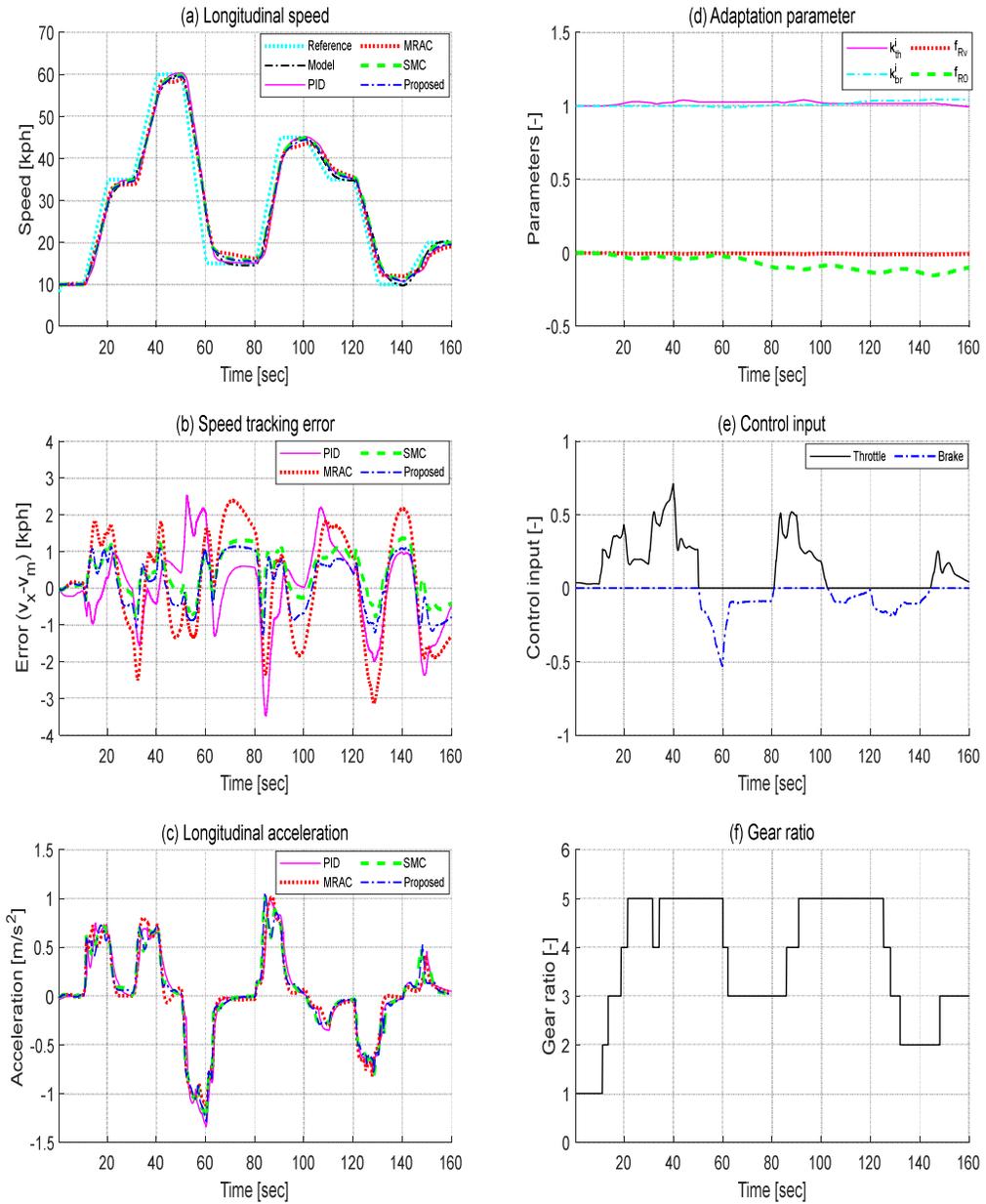


Fig. 6.17. Control performance of the proposed and reference longitudinal motion control algorithms on flat ground.

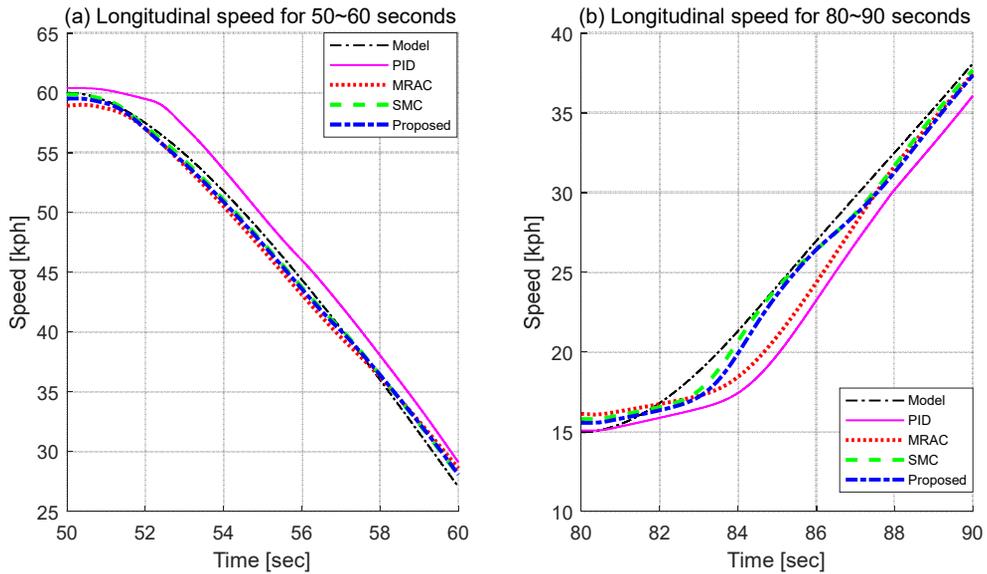


Fig. 6.18. Reference speed tracking delays on flat ground.

6.3.1.2. Sensitivity analysis

The sensitivity analysis was performed to analyze whether the proposed control logic is robust against the disturbances and uncertainties. The objects of the analysis were selected as follows: road slope, change in mass, and equivalent control input. With the RMS error of speed, the RMS of acceleration jerk was also plotted in Fig. 6.19. Through the change of the acceleration jerk, we wanted to investigate whether the ride comfort was degraded by excessive control input to compensate for the disturbances and uncertainties.

First, the simulation for the variance of road slope was performed. The same slope profile in the second scenario above was used; instead, the amplitude was set to $\pm 10\%$ in the second case. As demonstrated in Fig. 6.19 (a), the performance degradation of the proposed algorithm is the least. In contrast, as the uncertainty

increases, the acceleration jerk is also increased, though insignificantly. The degradation of MRAC is considerable in the second case; also, the degradation of PID and SMC is significant in the two cases.

Next, it was simulated that the mass was increased by 25% and 50%, as depicted in Fig. 6.19 (b). The results show that the proposed algorithm and MRAC had insignificant performance changes. This result is the influence of the parameter adaptation corresponding to the mass. On the other hand, the errors of PID and SMC increased insignificantly.

Lastly, it was investigated that the equivalent control input was increased by 100% and 200%, as presented in Fig. 6.19 (c). As discussed above, the change in the equivalent control input could be compensated using an adaptive control scheme. As a result, the errors of the proposed algorithm and MRAC were insignificantly increased. However, the performance of PID and SMC deteriorated.

In conclusion, it can be confirmed that the proposed algorithm is robust to both environmental disturbances and model uncertainties. If the magnitude of the disturbances and uncertainties is large, the ride comfort is dropped insignificantly due to the acceleration jerk, as depicted in Fig. 6.19.

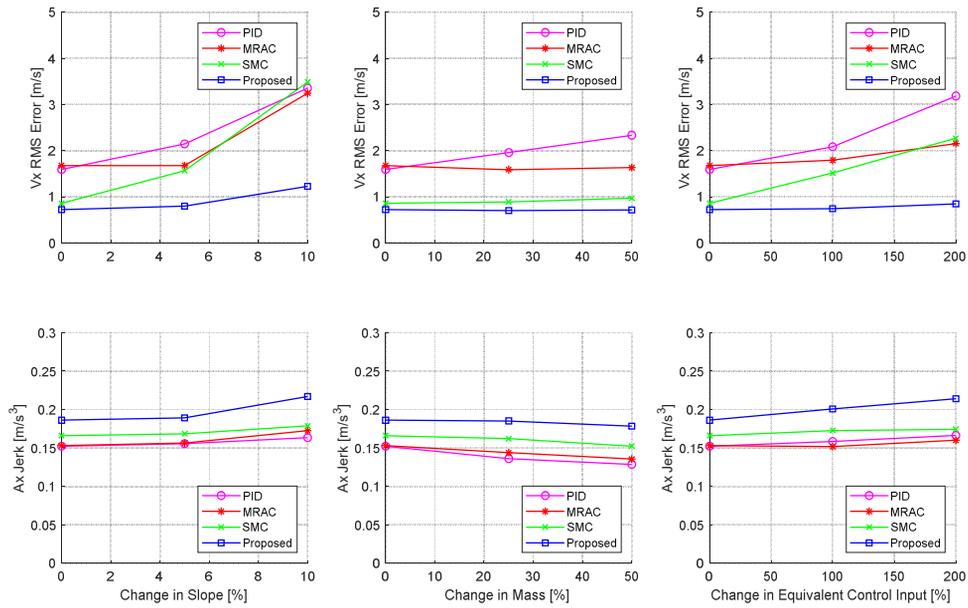


Fig. 6.19. Results of sensitivity analysis for (a) change in road slope (b) change in mass (c) change in equivalent control input when using the proposed and reference longitudinal motion control algorithms.

6.3.2. Vehicle Test Results in FMTC Track

To investigate the performance of the proposed algorithm, the suggested algorithm was compared with the PID controller. As illustrated in Fig. 6.20, the driving route for testing the longitudinal motion control algorithm in the test track includes a curved road, a straight road, and a slope. Since there were no other vehicles and signalized intersections, the reference speed profile was determined considering the curvature and speed limit. The shaded section in Fig. 6.20 is a hill, of which the slope ranges from -3 to 5%.

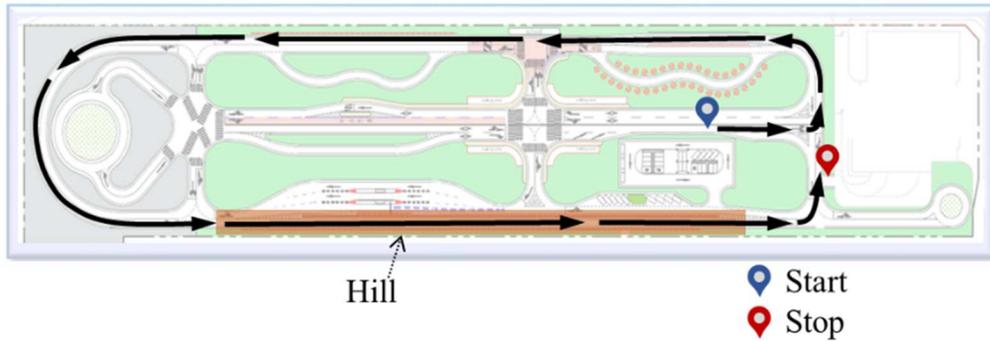


Fig. 6.20. Driving route for testing the proposed longitudinal motion control algorithm in FMTC track.

Fig. 6.21 (a) shows the speed tracking performance of the proposed algorithm and PID control. The desired speed is the model speed obtained from the reference speed model. The acceleration occurs after passing the left-turn path at 50 s and entering the straight hill at 750 s. The deceleration occurs while passing the left-turn path with large curvature at 1100 s. The peak of the tracking error is occurred at the moment, as in Fig. 6.21 (b). It is observed that the model error, which was neglected while assuming a linear model in Eq.(5.4), occurs due to the wheel slip. In Fig. 6.21 (b), the maximum speed tracking errors of the proposed and reference algorithms are 1.817 and 4.090 kph, respectively. The error of the proposed algorithm ranges from -1.817 to 1.017 kph, and that of the PID control ranges from -2.854 to 4.090 kph. It can be seen that the proposed algorithm gives a faster convergence rate for the model error.

The disturbance is also caused by the change in slope at 900 s and 1200 s, as in Fig. 6.21 (f). It can be seen that the disturbance was mainly compensated by the adaptation parameters, as in Fig. 6.21 (d). The error of the proposed controller keeps 25-75% at the maximum compared to that of the PID control. The proposed

algorithm exhibits superior performance in tracking the desired speed than the PID control, especially in the uphill and downhill sections. Fig. 6.21(c) shows the longitudinal acceleration measured by the GPS. The peak values of the vehicle controlled by the proposed algorithm and the PID control are about -1.039 and 0.424 m/s², and -1.352 and 0.517m/s², respectively. It can be seen from Fig. 6.21(a), Fig. 6.21(b), and Fig. 6.21(c) that the proposed scheme is robust to the model error and the environmental disturbances.

Fig. 6.21(d), Fig. 6.21(e), and Fig. 6.21(f) show the adaptation parameters, control input, and road slope, respectively. It can be observed from Fig. 6.21(d) that the adaptation parameters, \hat{f}_{Rv} , and \hat{f}_{R0} , corresponding to the environmental disturbances, are changed to compensate for the road slope in Fig. 6.21(f). The speed of parameter adaptation is faster than that of the simulation. Since the test vehicle is driven by the electric motor with a constant gear ratio, the model uncertainty of the vehicle test is less than that of the computer simulation. The adaptation parameters, \hat{k}_{th}^i and \hat{k}_{br}^i , corresponding to the model uncertainties, are changed less than the other parameters because of the small model uncertainty in the vehicle test. Fig. 6.21(e) shows that the control input is changed continuously because the parameters of the environmental disturbances and uncertainties are not changed during the control mode switch.

The vehicle test results demonstrate that the speed tracking performance of the proposed algorithm has been improved compared to that of the PID control. The RMS error of the proposed algorithm is 0.414 kph, which is improved by 53% compared to that of the PID control. The proposed algorithm achieved expected tracking performance on the flat ground and the uphill and downhill sections.

Moreover, the chattering phenomenon in the speed and acceleration of the vehicle controlled by the proposed algorithm was not observed because of the RBFNN-based SMC scheme [41]. Therefore, the proposed algorithm can be used as a speed tracking controller in various driving situations.

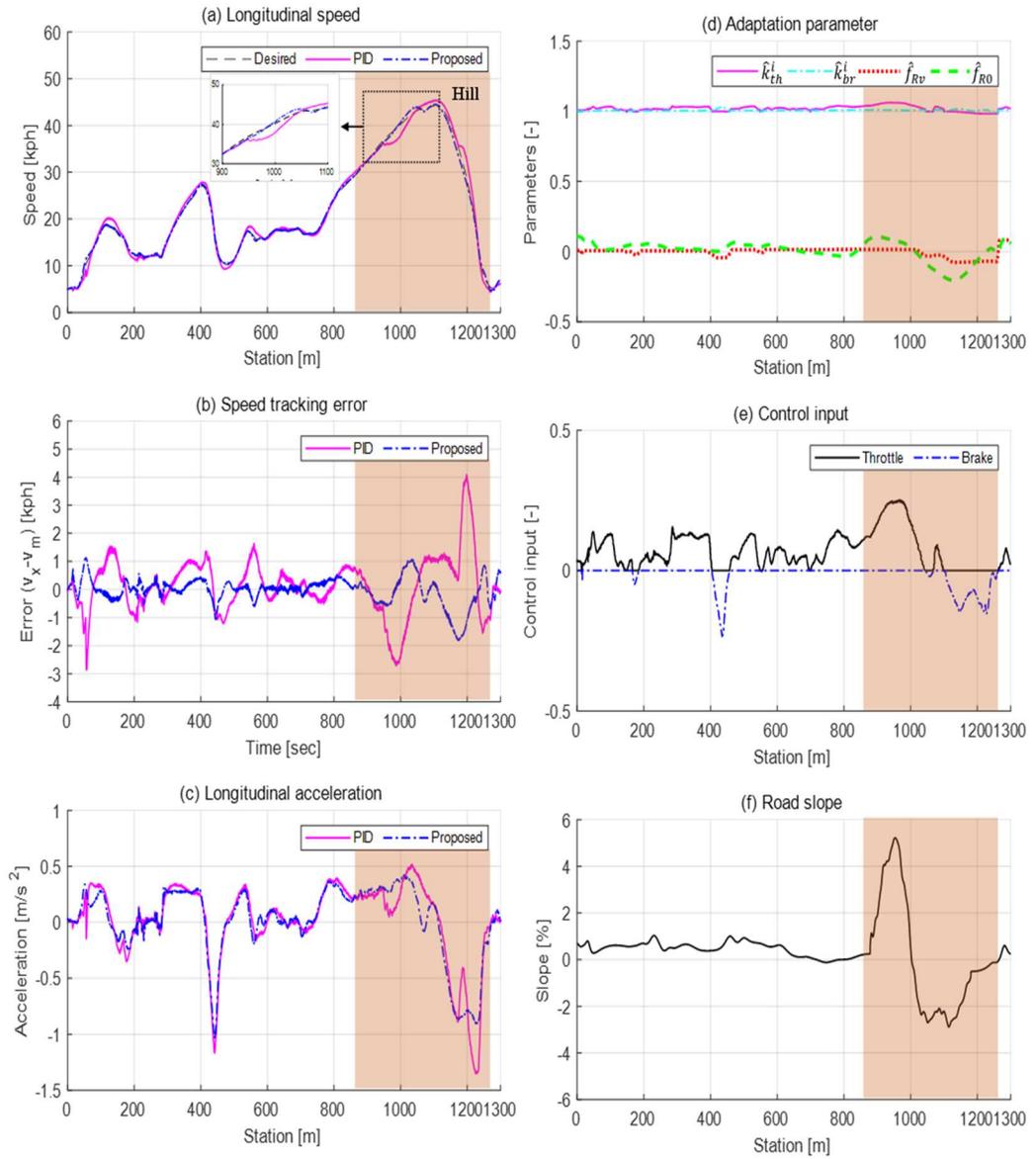


Fig. 6.21. Vehicle test results of the proposed and reference longitudinal motion control algorithms on the test track.

6.3.3. Vehicle Test Results in Urban Bus-only Lanes

Fig. 6.22 shows the vehicle test results on the urban road. The vehicle passed the underpasses at 120–180 s and 220–290 s, which contain the slope ranges from -6 to 6%, as depicted in Fig. 6.22 (f).

The reference speed profile was determined as shown in Fig. 6.22 (a), considering the curvature, remain-time at a signalized intersection, relative speed, clearance with other vehicles, and speed limits. The speed limits in 0–80 and 80–350 s are 45 and 35kph, respectively. The rapid changes in the desired speed at 50 and 340 s in Fig. 6.22 (a) occurred due to the front vehicles. The respective acceleration ranges from -1.218–1.437 m/s², as in Fig. 6.22 (c).

The test results of the stop-and-go situations are shown at 80 s, 300 s, and 320 s in Fig. 6.22 (a). Wheel slip occurred when accelerating from a complete stop at the intersections. The maximum error at 320 s is about 3.731 kph. Fig. 6.22 (d) shows the adaptation parameters. The adaptation parameters, \hat{f}_{RV} , and \hat{f}_{R0} , corresponding to the environmental disturbances in equations (5.11) and (5.28), show a similar pattern to the road slope in Fig. 6.22 (f). The adaptation parameters show more sharp changes in the accelerating situation at 80 and 320 s. The model and estimation errors by the wheel slip were compensated as the disturbances. The change in the adaptation parameters converges to near zero again after the vehicle accelerates.

The test results in the road with high curvature and the slope are observed between 110 and 290 s. The vehicle passed two underpasses, and there was a road with high curvature between the two underpasses. The most significant errors in this section are observed at 223 and 282 s, as shown in Fig. 6.22 (c). The errors are -1.169 and 1.342 kph, respectively. Fig. 6.22 (e) shows the control input. The input

value of about 223 s differs from the tendency of the adaptation parameters in Fig. 6.22 (d). The disturbance that occurred in this section is due to the lateral motion with high curvature, and it can be seen that the sliding mode control scheme affects to attenuate the disturbance. The disturbance that occurred at 282 s is due to the road slope, which is attenuated by the adaptation scheme.

The vehicle test results in the urban road environments demonstrate that the proposed algorithm quickly compensates for the disturbances due to both model errors and environmental disturbances. The peak values of the speed tracking error of the proposed algorithm are about -3.647 and 3.731 kph, respectively. The RMS error of the algorithm is 0.838 kph. The performance in the urban road is similar to that in the test track, except for the intersections. It is difficult to quantitatively compare the errors of the proposed algorithm in the urban road to that in the test track. Both the maximum and average errors of the proposed algorithm in the urban road are similar to those of the PID control in the test track. Moreover, it is confirmed that the model error with wheel slip can be compensated by the proposed algorithm.

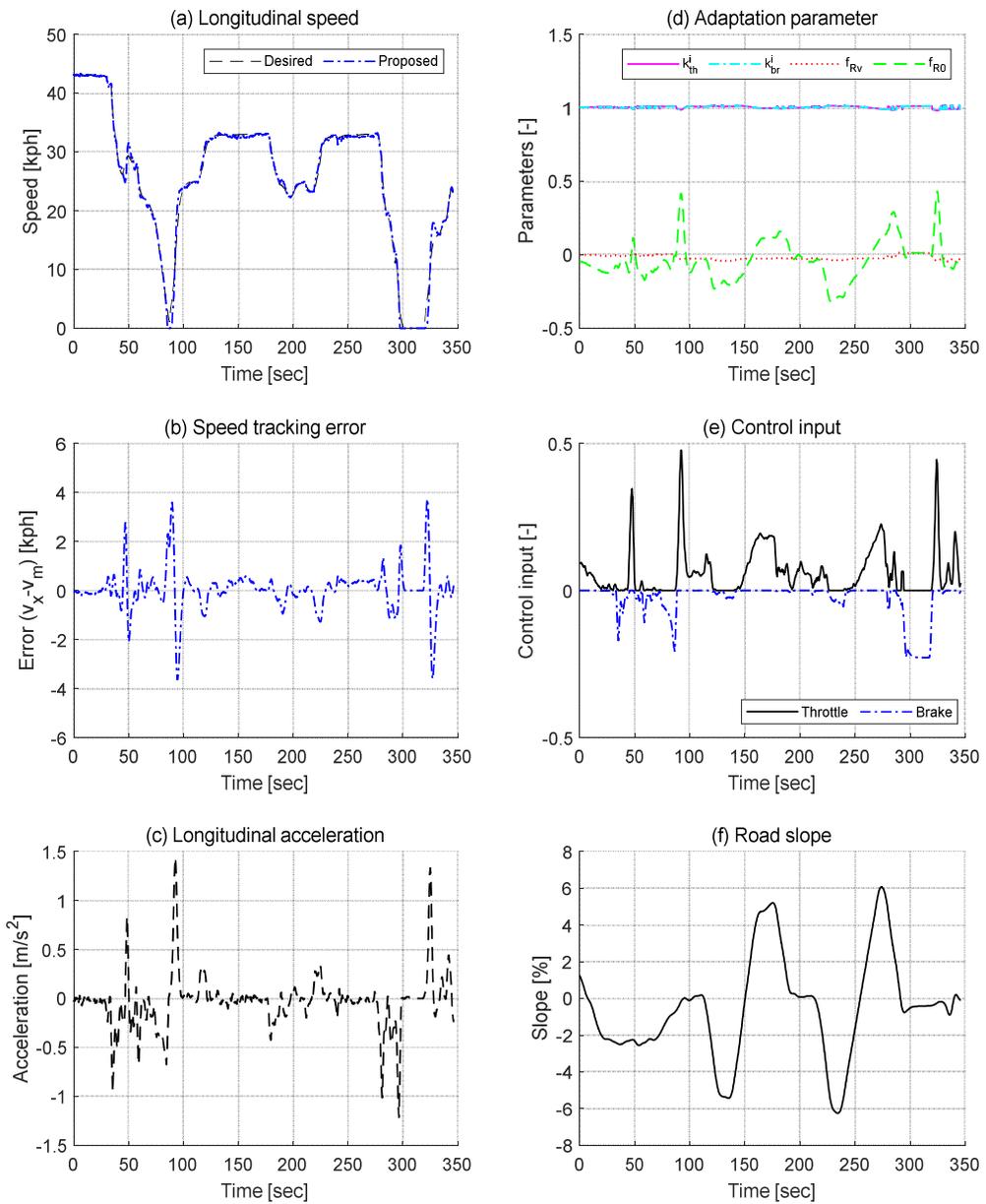


Fig. 6.22. Vehicle test results of the proposed longitudinal motion control algorithm on the urban bus-only lanes.

6.4. Performance Evaluation of Lateral Motion Control

6.4.1. Vehicle Test Results in FMTC Track

The vehicle test in the FMTC track was performed to compare the path tracking performance of the proposed lateral motion control algorithm. During the test, the maximum speed was set as 45 kph. There were changes in the speed limit due to the curvature of the route. In the vehicle test results, the lateral position error when implementing the proposed lateral motion control algorithm was smaller than the allowable gap (0.2 m). On the other hand, the maximum error when not implementing the proposed control algorithm was observed as 0.35 m. Consequently, it can be seen that the maximum error is reduced by 45 % when using the proposed algorithm.

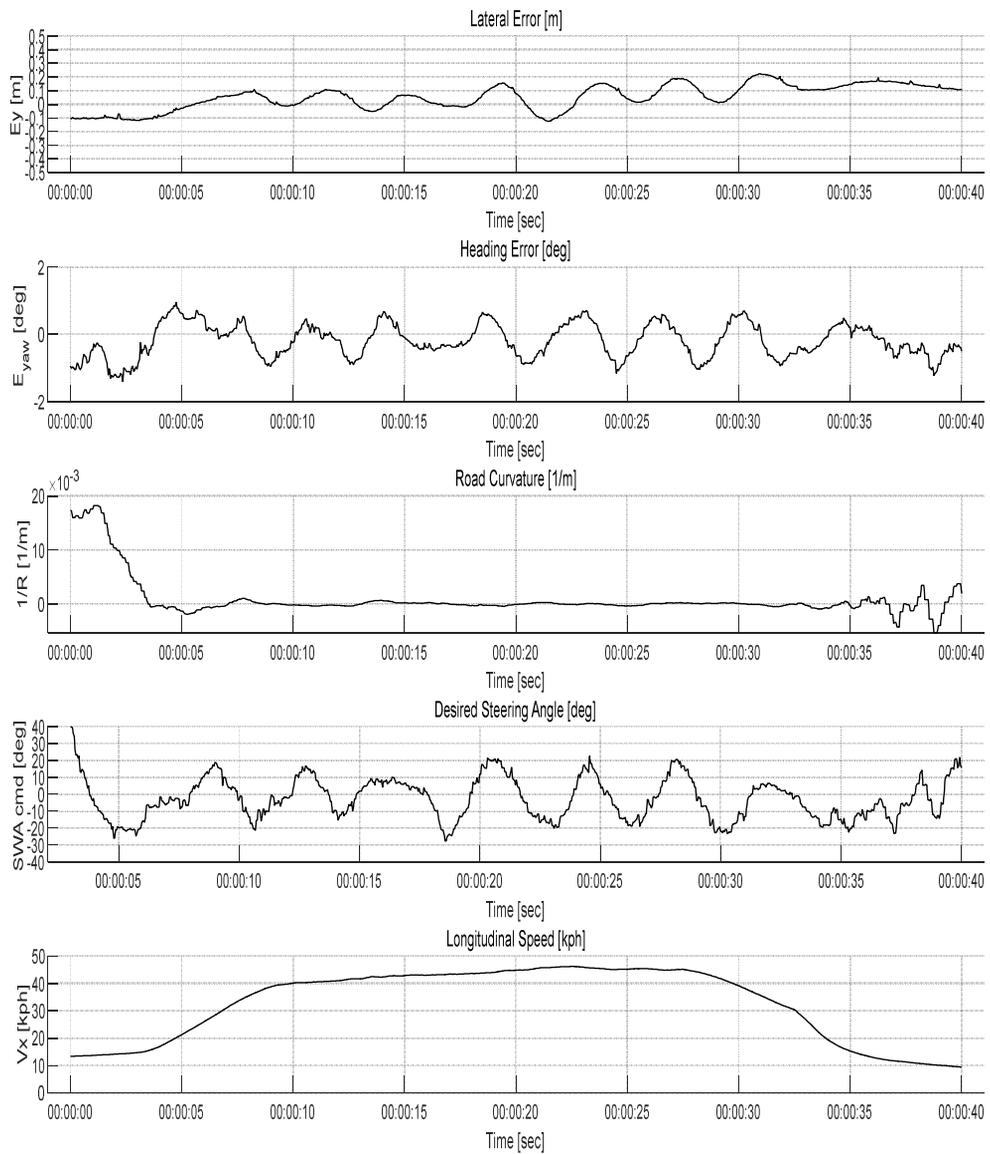


Fig. 6.23. Vehicle test results when using the proposed lateral motion control algorithm in the FMTC track

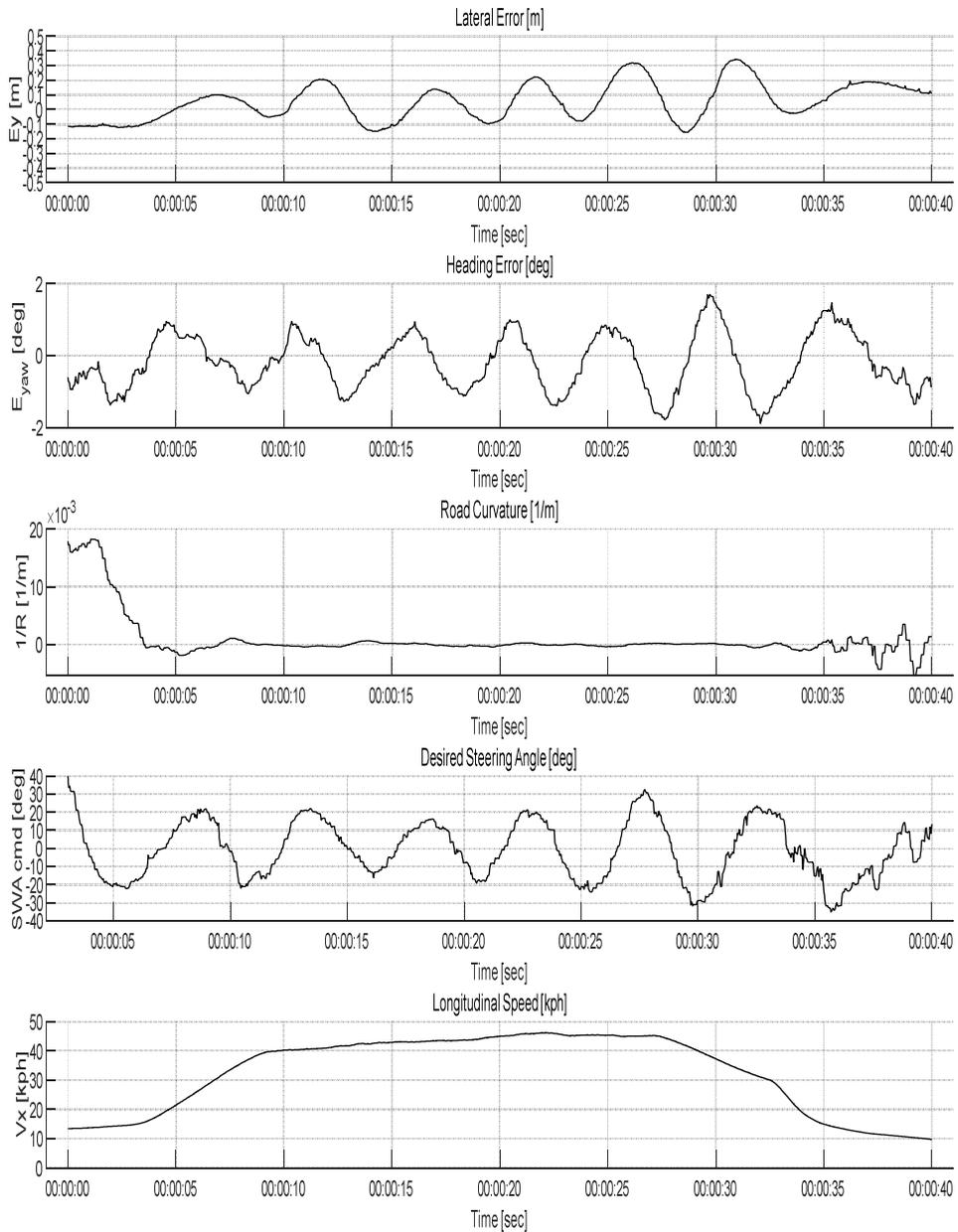


Fig. 6.24. Vehicle test results when implementing the conventional lateral motion control algorithm in the FMTC track

Chapter 7. Conclusion & Future Works

7.1. Conclusion

This dissertation proposes novel, robust hierarchical motion planning and motion control algorithms for an automated bus in urban bus-only lanes. Three challenges in autonomous driving of the full-sized bus in urban bus-only lanes were recognized: (1) improve ride comfort and safety with limited FOV and detection range, (2) cope with localization uncertainty, and (3) secure robustness against environmental disturbances and model uncertainties.

7.1.1. Contributions for Ride Comfort and Safety with Limited FOV and Detection Range

Ensuring ride comfort and safety with limited FOV and detection range was accomplished in the suggested longitudinal motion planning algorithm. The algorithm was constructed as a hierarchical framework: long time-horizon longitudinal motion planning and short time-horizon longitudinal motion planning. The long time-horizon planning was carried out for optimizing the reference motion with limited FOV and detection range. The optimal solution could be acquired using the DP method since the long time-horizon could cover the detection range with limited FOV and detection range. The reference motion profile showed a gradual reaction against the stationary and slowly moving targets when implementing the proposed algorithm. The development of the algorithm was accomplished according

to the initial purpose of ensuring ride comfort and safety since the tendency of the motion profile was well-matched with that of the human driving pattern.

The short time-horizon motion planning was also conducted for ride comfort and safety by replicating the human driving patterns. The risk index-based weights acquired from the fuzzy inference system considering newly constructed rules and parameters, which were determined from the analysis of human driving data in the urban bus-only lanes. The parameters indicate the correlations between control effort and risk index and between control effort and ego vehicle's speed. The optimization problem of the Fuzzy-tuned MPC was formulated considering control effort, jerk, and penalties for the travel distance and speed limit violation. For safety, localization uncertainty was additionally taken into account by designing the state constraint for travel distance.

After the feasibility of the suggested approach was evaluated from computer simulations, the applicability was confirmed from actual vehicle tests in the test track and BRT course. The results of simulations against stationary, slowly moving, and cut-in targets show the effectiveness of the proposed algorithm for the automated bus with limited FOV and detection range. Monte-Carlo simulation was additionally carried out for securing the credibility of the approach according to detection ranges. Significant decreases in the maximum deceleration and jerk were observed. Finally, the proposed algorithm was confirmed as applicable for fully autonomous driving of the full-sized bus in urban bus-only lanes.

7.1.2. Contributions for Coping with Localization Uncertainty

Potential dangers due to localization uncertainty were coped with through the suggested longitudinal and lateral motion planning algorithms. Localization uncertainty made heading, lateral position, and longitudinal position errors with respect to the reference path. The longitudinal position error was taken into account as the chance constraint in the longitudinal motion planning algorithm. As a result, the minimum clearance in front of stop lines could be secured. The heading error and lateral position error were coped with in the suggested lateral motion planning algorithm. Desired yaw rate was calculated from the offset-free MPC considering the desired path from the drivable corridor. After the bias in heading error was eliminated by designing the disturbance estimation using MHE, the corrected heading error was used as the current state of the MPC. The drivable corridor was determined to prevent collisions with static targets. As a result, the effect of the bias on the lateral motion planning could be attenuated.

The feasibility of the proposed algorithm was evaluated from vehicle tests in FMTC track by adding a constant and oscillatory bias in heading error. Regardless of the additional heading bias, the lateral position error was smaller than the allowable gap (0.2 m) of the BRT lanes. The applicability of the algorithm was confirmed from vehicle tests in the BRT course. Despite the uncertainty in localization module under actual driving conditions, the lateral position error was smaller than the allowable gap (0.2 m) of the BRT lanes as well. Finally, the proposed algorithm was confirmed as applicable with the localization uncertainty for fully autonomous driving of the full-sized bus in urban bus-only lanes.

7.1.3. Contributions for Robustness against Environmental

Disturbance and Model Uncertainty

Robustness against environmental disturbance and model uncertainty was achieved through the suggested longitudinal and lateral motion control algorithms with limited feedback information. The model-reference ASMC scheme with RBFNN-based uncertainty bound estimator was implemented for the longitudinal motion control. From the sensitivity analysis of the suggested longitudinal motion control algorithm, robustness against changes in road slope, mass, and model parameters could be recognized. The driving performance of the proposed longitudinal motion control algorithm was confirmed from vehicle tests in FMTC track and BRT course even when using wheel speed sensors only. The SMC scheme was implemented for the lateral motion control. Using the suggested lateral motion control algorithm, delay from the input of steering wheel angle to the response of the yaw rate could be attenuated.

7.2. Future Works

By suggesting long time-horizon planning in the longitudinal motion planning algorithm, it was possible to take into account additional terms from the optimization problem for ride comfort and control effort minimization. Potential applicability for the constructed approach to the ecological motion planning problem considering road grade, signalized intersection status, and traffic flow was recognized. The future topic of the research is the ecological longitudinal motion planning algorithm, which is applicable for an automated bus along a pre-determined route.

The lateral motion planning algorithm was constructed for the driving condition

where localization uncertainty was dominant regardless of diverse uncertainties, including environmental disturbance, model uncertainty, and localization uncertainty. The potential applicability for decoupling a variable uncertainty from uncertainties in the spatial domain was recognized by implementing the suggested approach. A learning-based motion planning is one of the future topics; The model uncertainty will be dealt with by implementing the learning-based approach. Errors represented by bias due to localization uncertainty or based on specific conditions in the spatial domain will be coped with using the proposed disturbance estimation. It will be possible to reduce the error in MPC, which is based on a nonlinear time-varying model, and to conduct a motion control with accuracy. In addition, the drivable corridor can be stochastically determined from the error-bound estimation.

Bibliography

- [1] M. Azad, N. Hoseinzadeh, C. Brakewood, C. R. Cherry, and L. D. Han, “Fully Autonomous Buses: A Literature Review and Future Research Directions,” *J. Adv. Transp.*, vol. 2019, no. Figure 1, 2019.
- [2] T. Ando, W. Kugimiya, T. Hashimoto, F. Momiyama, K. Aoki, and K. Nakano, “Lateral Control in Precision Docking Using RTK-GNSS/INS and LiDAR for Localization,” *IEEE Trans. Intell. Veh.*, vol. 6, no. 1, pp. 78–87, 2021.
- [3] W. Gao, J. Gao, K. Ozbay, and Z. P. Jiang, “Reinforcement-Learning-Based Cooperative Adaptive Cruise Control of Buses in the Lincoln Tunnel Corridor with Time-Varying Topology,” *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3796–3805, 2019.
- [4] R. Oliveira, P. Lima, M. Cirillo, and B. Wahlberg, “Autonomous Bus Driving: A Novel Motion-Planning Approach,” *IEEE Veh. Technol. Mag.*, vol. 16, no. 3, pp. 29–37, 2021.
- [5] K. Hilgarter and P. Granig, “Public Perception of Autonomous Vehicles: A Qualitative Study based on Interviews after Riding an Autonomous Shuttle,” *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 72, pp. 226–243, 2020.
- [6] D. Paddeu, G. Parkhurst, and I. Shergold, “Passenger Comfort and Trust on First-time Use of a Shared Autonomous Shuttle Vehicle,” *Transp. Res. Part C Emerg. Technol.*, vol. 115, no. March, p. 102604, 2020.
- [7] S. Wei, Y. Zou, X. Zhang, T. Zhang, and X. Li, “An Integrated Longitudinal and Lateral Vehicle Following Control System with Radar and Vehicle-to-vehicle Communication,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1116–1127, 2019.

- [8] Q. Xin, R. Fu, W. Yuan, Q. Liu, and S. Yu, "Predictive Intelligent Driver Model for Eco-driving using Upcoming Traffic Signal Information," *Phys. A Stat. Mech. its Appl.*, vol. 508, pp. 806–823, 2018.
- [9] F. Acciani, P. Frasca, G. Heijenk, and A. A. Stoorvogel, "Stochastic String Stability of Vehicle Platoons via Cooperative Adaptive Cruise Control with Lossy Communication," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–11, 2021.
- [10] S. Tak, J. Yoon, S. Woo, and H. Yeo, "Sectional Information-based Collision Warning System using Roadside Unit Aggregated Connected-vehicle Information for a cooperative Intelligent Transport System," *J. Adv. Transp.*, vol. 2020, 2020.
- [11] H. Chae and K. Yi, "Virtual Target-based Overtaking Decision, Motion Planning, and Control of Autonomous Vehicles," *IEEE Access*, vol. 8, pp. 51363–51376, 2020.
- [12] J. M. Yoo, Y. Jeong, and K. Yi, "Virtual Target-based Longitudinal Motion Planning of Autonomous Vehicles at Urban Intersections: Determining Control Inputs of Acceleration with Human Driving Characteristic-based Constraints," *IEEE Veh. Technol. Mag.*, vol. 16, no. 3, pp. 38–46, 2021.
- [13] M. Naumann, L. Sun, W. Zhan and M. Tomizuka, "Analyzing the Suitability of Cost Functions for Explaining and Imitating Human Driving Behavior based on Inverse Reinforcement Learning," *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 5481-5487.
- [14] S. Yoon, H. Jeon, and D. Kum, "Predictive Cruise Control using Radial Basis Function Network-based Vehicle Motion Prediction and Chance Constrained Model Predictive Control," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3832–3843, 2019.
- [15] L. Li, Y. Miao, A. H. Qureshi, and M. C. Yip, "MPC-MPNet: Model-predictive

- Motion Planning Networks for Fast, Near-optimal Planning under Kinodynamic Constraints,” *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4496–4503, 2021.
- [16] L. Chen, X. Hu, B. Tang, and Y. Cheng, “Conditional DQN-based Motion Planning with Fuzzy Logic for Autonomous Driving,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–12, 2020.
- [17] Z. Gao, X. Yan, F. Gao, and L. He, “Driver-like Decision-making Method for Vehicle Longitudinal Autonomous Driving based on Deep Reinforcement Learning,” *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.*, no. 5988, pp. 1–11, 2021.
- [18] M. Zhu, Y. Wang, Z. Pu, J. Hu, X. Wang, and R. Ke, “Safe, Efficient, and Comfortable Velocity Control based on Reinforcement Learning for Autonomous Driving,” *Transp. Res. Part C Emerg. Technol.*, vol. 117, no. April, p. 102662, 2020.
- [19] H. Dong, W. Zhuang, B. Chen, G. Yin, and Y. Wang, “Enhanced Eco-Approach Control of Connected Electric Vehicles at Signalized Intersection with Queue Discharge Prediction,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 6, pp. 5457–5469, 2021.
- [20] Lin, J. McPhee, and N. L. Azad, “Comparison of Deep Reinforcement Learning and Model Predictive Control for Adaptive Cruise Control,” *IEEE Trans. Intell. Veh.*, vol. 6, no. 2, pp. 221–231, 2021.
- [21] J. Suh, H. Chae, and K. Yi, “Stochastic Model-predictive Control for Lane Change Decision of Automated Driving Vehicles,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 4771–4782, 2018.
- [22] D. Fan, A. Agha, and E. Theodorou, “Deep Learning Tubes for Tube MPC,” in *Robotics Science Systems*, 2020.
- [23] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, “A Survey of Autonomous

Driving: Common Practices and Emerging Technologies,” *IEEE Access*, vol. 8, pp. 58443–58469, 2020.

[24] S. Xu *et al.*, “System and Experiments of Model-Driven Motion Planning and Control for Autonomous Vehicles,” *IEEE Trans. Syst. Man, Cybern. Syst.*, pp. 1–14, 2021.

[25] J. M. Yoo, Y. Jeong, and K. Yi, “Virtual Target-Based Longitudinal Motion Planning of Autonomous Vehicles at Urban Intersections: Determining Control Inputs of Acceleration with Human Driving Characteristic-Based Constraints,” *IEEE Veh. Technol. Mag.*, vol. 16, no. 3, pp. 38–46, 2021.

[26] A. Artunedo, J. Villagra, J. Godoy, and M. D. Del Castillo, “Motion Planning Approach considering Localization Uncertainty,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 5983–5994, 2020.

[27] L. Ge, Y. Zhao, F. Ma, and K. Guo, “Towards Longitudinal and Lateral Coupling Control of Autonomous Vehicles using Offset Free MPC,” *Control Eng. Pract.*, vol. 121, no. January, p. 105074, 2022.

[28] J. Marzbanrad and I. Tahbaz-zadeh Moghaddam, “Self-tuning Control Algorithm Design for Vehicle Adaptive Cruise Control system through Real-time Estimation of Vehicle Parameters and Road Grade,” *Veh. Syst. Dyn.*, vol. 54, no. 9, pp. 1291–1316, 2016.

[29] Y. Chen and J. Wang, “Adaptive Vehicle Speed Control with Input Injections for Longitudinal Motion Independent Road Frictional Condition Estimation,” *IEEE Trans. Veh. Technol.*, vol. 60, no. 3, pp. 839–848, 2011.

[30] J. Guo, Y. Luo, and K. Li, “Adaptive Neural-network Sliding Mode Cascade Architecture of Longitudinal Tracking Control for Unmanned Vehicles,” *Nonlinear*

Dyn., vol. 87, no. 4, pp. 2497–2510, 2017.

[31] M. Fliess and C. Join, “Model-free control,” *Int. J. Control*, vol. 86, no. 12, pp. 2228–2252, 2013.

[32] L. Menhour, B. D’Andréa-Novel, M. Fliess, D. Gruyer, and H. Mounier, “An Efficient Model-free Setting for Longitudinal and Lateral Vehicle Control: Validation through the Interconnected pro-SiVIC/RTMaps Prototyping platform,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 461–475, 2018.

[33] J. Liu, M. Zhao, G. Li, and J. Chen, “Multiple Linear Regression Prediction and Wavelet Neural Network based Intelligent Online Tuning Control Method,” *IEEE Access*, vol. 8, no. 2, pp. 160684–160696, 2020.

[34] P. Polack, B. D’Andréa-Novel, M. Fliess, A. de La Fortelle, and L. Menhour, “Finite-time Stabilization of Longitudinal Control for Autonomous Vehicles via a Model-free Approach,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12533–12538, 2017.

[35] H. Kim, D. Kim, I. Shu, and K. Yi, “Time-varying Parameter Adaptive Vehicle Speed Control,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 2, pp. 581–588, 2016.

[36] A. Raffin, M. Taragna, and M. Giorelli, “Adaptive Longitudinal Control of an Autonomous Vehicle with an Approximate Knowledge of its Parameters,” *11th Int. Work. Robot Motion Control. RoMoCo 2017 - Work. Proc.*, pp. 1–6, 2017.

[37] K. D. Young, V. I. Utkin, and Ü. Özgüner, “A Control Engineer’s Guide to Sliding Mode Control,” *IEEE Trans. Control Syst. Technol.*, vol. 7, no. 3, pp. 328–342, 1999.

[38] R. Wai, “Fuzzy Sliding-mode Control using Adaptive Tuning Technique,” *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 586–594, 2007.

[39] A. Amirkhani, M. Shirzadeh, N. Tork, and S. B. Shokouhi, “Path Tracking of an

Autonomous Vehicle by Means of an Indirect Adaptive Neural Controller,” *ICRoM 2019 - 7th Int. Conf. Robot. Mechatronics*, no. ICRoM, pp. 398–403, 2019.

[40] J. Fei and H. Ding, “Adaptive Sliding Mode Control of Dynamic System using RBF Neural Network,” *Nonlinear Dyn.*, vol. 70, no. 2, pp. 1563–1573, 2012.

[41] H. Du, L. Wang, J. Chen, H. Huang, and Y. Wang, “Adaptive Fuzzy Radial Basis Function Neural Network Integral Sliding Mode Tracking Control for Heavy Vehicle Electro-hydraulic Power Steering Systems,” *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.*, vol. 234, no. 2–3, pp. 872–886, 2020.

[42] Wang, Y. Hui, X. Sun, and D. Shi, “Neural Network Sliding Mode Control of Intelligent Vehicle Longitudinal Dynamics,” *IEEE Access*, vol. 7, pp. 162333–162342, 2019.

[43] M. A. D. Alves *et al.*, “A Modified Algorithm for Training and Optimize RBF Neural networks applied to Sensor Measurements Validation,” *IEEE Sens. J.*, vol. 21, no. 17, pp. 18990–18999, 2021.

[44] J. Kim and S.-C. Kee, “A Research on the ODD and OEDR Guidelines Based on the Demonstration Case of Autonomous Driving in Sejong City,” *Trans. Korean Soc. Automot. Eng.*, vol. 28, no. 10, pp. 659–668, Oct. 2020.

[45] S. Moon and K. Yi, “Human Driving Data-based Design of a Vehicle Adaptive Cruise Control Algorithm,” *Veh. Syst. Dyn.*, vol. 46, no. 8, pp. 661–690, 2008.

[46] Van Der Horst, Richard, and Jeroen Hogema. "Time-to-collision and Collision Avoidance Systems." (1993): 129-143.

[47] H. Lee, H. Lee, D. Shin, and K. Yi, “Moving Objects Tracking Based on Geometric Model-Free Approach With Particle Filter Using Automotive LiDAR,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–10, 2022.

- [48] J. So, J. Hwangbo, S. H. Kim, and I. Yun, "Analysis on Autonomous Vehicle Detection Performance according to Various Road Geometry Settings," *J. Intell. Transp. Syst. Technol. Planning, Oper.*, vol. 0, no. 0, pp. 1–12, 2022.
- [49] B. Samani and A. H. Shamekhi, "Multi-objective Adaptive Cruise Controller Design using Nonlinear Predictive Controller with the Objective Function with Variable Weights determined by Fuzzy Logic Controller," *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.*, vol. 236, no. 1, pp. 142–154, 2022.
- [50] H. Chu, L. Guo, Y. Yan, B. Gao, and H. Chen, "Self-Learning Optimal Cruise Control Based on Individual Car-Following Style," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 10, pp. 6622–6633, 2021.
- [51] S. Cheng, L. Li, M. M. Mei, Y. L. Nie, and L. Zhao, "Multiple-Objective Adaptive Cruise Control System Integrated with DYC," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4550–4559, 2019.
- [52] H. Zhou, C. Yuan, N. Dong, S. C. Wong, and P. Xu, "Severity of Passenger Injuries on Public Buses: a Comparative Analysis of Collision Injuries and Non-collision Injuries," *J. Safety Res.*, vol. 74, pp. 55–69, 2020.
- [53] M. Farina, L. Giulioni, and R. Scattolini, "Stochastic linear Model Predictive Control with chance constraints - A review," *J. Process Control*, vol. 44, pp. 53–67, 2016.
- [54] M. C. Campi, S. Garatti, and M. Prandini, "The Scenario Approach for Systems and Control Design," *Annu. Rev. Control*, vol. 33, no. 2, pp. 149–157, 2009.
- [55] A. Artunedo, J. Villagra, and J. Godoy, "Jerk-Limited Time-Optimal Speed Planning for Arbitrary Paths," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–15, 2021.

- [56] T. P. Lillicrap *et al.*, “Continuous Control with Deep Reinforcement Learning,” *4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc.*, 2016.
- [57] Y. Jeong, “Self-Adaptive Motion Prediction-Based Proactive Motion Planning for Autonomous Driving in Urban Environments,” *IEEE Access*, vol. 9, no. Cv, pp. 105612–105626, 2021.
- [58] N. Akai, Y. Akagi, T. Hirayama, T. Morikawa, and H. Murase, “Detection of Localization Failures Using Markov Random Fields With Fully Connected Latent Variables for Safe LiDAR-Based Automated Driving,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 4, pp. 1–13, 2022.
- [59] J. Al Hage, P. Xu, P. Bonnifait, and J. Ibanez-Guzman, “Localization Integrity for Intelligent Vehicles Through Fault Detection and Position Error Characterization,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 4, pp. 2978–2990, 2020.
- [60] Rajamani, R., 2012, *Vehicle Dynamics and Control*. 2nd edition, Springer, pp.87-111.
- [61] G. Pannocchia and J. B. Rawlings, “Disturbance Models for Offset-free Model-Predictive Control,” *AIChE J.*, vol. 49, no. 2, pp. 426–437, 2003.
- [62] U. Maeder and M. Morari, “Offset-free Reference Tracking with Model Predictive Control,” *Automatica*, vol. 46, no. 9, pp. 1469–1476, 2010.
- [63] U. Maeder, F. Borrelli, and M. Morari, “Linear Offset-free Model Predictive Control,” *Automatica*, vol. 45, no. 10, pp. 2214–2222, 2009.
- [64] M. D. James Blake Rawlings, David Q. Mayne, *Model Predictive Control: Theory, Computation, and Design*. Nob Hill Publishing, 2017.
- [65] C. V. Rao, J. B. Rawlings, and D. Q. Mayne, “Constrained State Estimation for Nonlinear Discrete-time Systems: Stability and Moving Horizon Approximations,”

- IEEE Trans. Automat. Contr.*, vol. 48, no. 2, pp. 246–258, 2003.
- [66] W. Zhang, Z. Wang, C. Zou, L. Drugge, and M. Nybacka, “Advanced Vehicle State Monitoring: Evaluating Moving Horizon Estimators and Unscented Kalman Filter,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5430–5442, 2019.
- [67] H. Liu, P. Wang, J. Lin, H. Ding, H. Chen, and F. Xu, “Real-Time Longitudinal and Lateral State Estimation of Preceding Vehicle Based on Moving Horizon Estimation,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 8755–8768, 2021.
- [68] U. Maeder, F. Borrelli, and M. Morari, “Linear offset-free Model Predictive Control,” *Automatica*, vol. 45, no. 10, pp. 2214–2222, 2009.
- [69] U. Maeder and M. Morari, “Offset-free Reference Tracking with Model Predictive Control,” *Automatica*, vol. 46, no. 9, pp. 1469–1476, 2010.
- [70] G. Pannocchia, “Offset-free Tracking MPC: A Tutorial Review and Comparison of Different Formulations,” *2015 Eur. Control Conf. ECC 2015*, vol. 527, no. 3, pp. 527–532, 2015.
- [71] Willy Klier, Gerd Reimann and Wolfgang Reinelt, “Concept and Functionality of the Active Front Steering System”, SAE 2004-21-0073.
- [72] A. Jo, H. Lee, D. Seo, and K. Yi, “Model-reference Adaptive Sliding Mode Control of Longitudinal Speed Tracking for Autonomous Vehicles,” *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.*, 2022.
- [73] J. C. Gerdes and J. K. Hedrick, “Vehicle Speed and Spacing Control via Coordinated Throttle and Brake Actuation,” *Control Eng. Pract.*, vol. 5, no. 11, pp. 1607–1614, 1997.
- [74] K. Yi, J. Hong, and Y. D. Kwon, “A Vehicle Control Algorithm for Stop-and-go Cruise Control,” *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.*, vol. 215, no. 10,

pp. 1099–1115, 2001.

[75] V. Stepanyan and K. Krishnakumar, “Adaptive Control with Reference Model Modification,” *J. Guid. Control. Dyn.*, vol. 35, no. 4, pp. 1370–1374, 2012.

[76] S. K. S. Christopher Edwards, *Sliding Mode Control: Theory and Applications*. CRC Press, 1998.

[77] T. Z. Shuzhi S. Ge, Chang C. Hang, Tong H. Lee, *Stable Adaptive Neural Network Control*. Springer, Berlin, 2002.

[78] Mechanical Simulation – Trucksim. [Online]. Available: <https://www.carsim.com/products/trucksim/>

초 록

자율주행 버스를 위한

강건한 계층적 거동 계획 및 제어

본 논문은 자율주행 버스의 버스전용차로 자율주행을 위한 강건한 계층적 거동 계획 및 제어 알고리즘을 제안한다. 제안한 거동 계획 및 제어 알고리즘은 도심 버스전용차로의 주행 환경과 대형 버스 차량의 특성을 고려하여, 자율주행 인지 범위 제한 조건에서 승차감과 안전성을 향상하고, 측위 로직의 불확실성을 극복하며, 환경 외란과 차량 모델의 불확실성에 대한 강건성을 확보하는 것을 목표로 하였다.

자율주행 버스의 인지 범위 제약에도 승차감과 안전성을 확보하기 위해 종 방향 거동 계획(Longitudinal Motion Planning) 알고리즘이 개발되었다. 제안한 알고리즘은 계층적 구조로서 장시간 구간(Long Time-horizon)의 거동 계획과 단시간 구간(Short Time-horizon)의 거동 계획으로 구성하였다. 장시간 구간의 거동 계획은 인지 범위 제한 조건에서 참조 거동(Reference Motion)을 최적화하기 위해 설계되었으며, 단순한 운동학 모델과 동적 계획 기법(Dynamic Programming)을 사용하였다. 단시간 구간의 거동 계획은 운전자의 주행 특성을 반영하여 승차감과 안전성을 향상하기 위해 개발하였다. 이 거동 계획의 최적화 문제는 차량 동역학 모델과 모델 예측 제어기(Model Predictive Control)를 사용하였다. 모델

예측 제어기는 장시간 구간의 계획에서 결정된 참조 거동의 추종, 제어 입력 및 변화량, 이동 거리와 속도 제한을 위배하는 것에 대한 페널티로 구성하였다. 이 최적화 함수의 가중치는 위험 지수를 기반으로 한 퍼지 추론 시스템(Fuzzy Inference System)으로 결정하였다. 제안한 알고리즘은 안전성을 향상하기 위하여 측위 및 인지 모듈의 추정 오차를 반영한 차량의 이동 위치에 대한 기회 제약 조건을 추가로 고려하였다. 그 결과 제안한 종 방향 거동 계획 알고리즘은 인지 범위 제한에도 승차감과 안전성을 확보할 수 있었다. 또한, 최근 연구 동향과 연계하여 제안한 알고리즘을 심층 강화 학습을 기반으로 동일한 목적으로 개발한 알고리즘과 비교하였다. 심층 강화 학습 기반의 종 방향 거동 계획 알고리즘은 계층적 구조로 개발하였으며, 그 성능을 동적 계획 기법을 이용한 알고리즘과 비교하였다. 대표적인 테스트 조건에서 시뮬레이션을 수행하여, 시뮬레이션 결과와 연산 시간을 포함한 장/단점을 비교 분석하였다.

도심 버스전용차로의 GPS 음영에 따른 측위 로직의 불확실성을 고려한 거동 계획 알고리즘이 제안되었다. 본 문제는 종 방향 거동 계획 및 횡 방향 거동 계획 알고리즘에서 고려되었다. 측위 로직의 불확실성은 주행 도로를 기준으로 헤딩(heading) 오차, 횡 방향 위치 오차, 종 방향 위치 오차로 구분하였다. 종 방향 위치 오차는 종 방향 거동 계획 알고리즘에서 이동 거리에 대한 기회 제약 조건으로 고려하였다. 횡 방향 거동 계획 알고리즘은 헤딩 오차와 횡 방향 위치 오차를 고려하였다. 헤딩 오차가 경로를 추종하는 성능에 미치는 영향을

저감하기 위한 이동 수평 추정(Moving Horizon Estimation)을 이용한 외란 추정기를 개발하였다. 횡 방향 위치 오차는 주행 가능 영역 내에서 목표 경로가 장애물과의 충돌을 회피하도록 해당 경로를 갱신함으로써 고려하였다. 편차 제거(Offset-free) 모델 예측 제어기는 주행 가능 영역으로부터 결정된 목표 경로와 추정된 상태를 이용해 목표 요 속도를 결정한다. 그 결과, 측위 로직의 불확실성이 경로 추종 성능에 미치는 영향을 저감하고, 실제 도로에서 안정적으로 주행할 수 있는 목표 거동을 계산할 수 있었다.

중 방향 및 횡 방향 거동 제어 알고리즘들은 피드백 정보가 제한된 상황에서도 환경 외란과 차량 모델의 불확실성에 대한 강건성을 확보하기 위해 설계되었다. 중 방향 거동 제어 알고리즘으로는 다양한 외란에서 강건함을 확보하기 위하여, 모델 추종 기반의 적응형 슬라이딩 모드 제어(Adaptive Sliding Mode Control) 기법에 방사형 기저 함수 신경회로망(Radial Basis Function Neural Network)으로 유계 불확실성의 경계를 추정하는 방법을 적용하였다. 횡 방향 제어 알고리즘으로는 조향각 입력에 대한 요 속도의 응답 지연 현상을 저감하기 위한 슬라이딩 모드 제어 기법이 제안되었다.

제안한 알고리즘들은 컴퓨터 시뮬레이션을 통해 타당성을 평가하였으며, 테스트 트랙과 버스전용차로에서 테스트 차량으로 주행 테스트를 수행하여 그 적용성을 확인하였다. 그 결과, 제안한 알고리즘으로 실제 버스전용차로에서 대형 버스 차량의 자율주행이 가능함을 확인할 수 있었다.

주요어: 자율주행 버스, 동적 계획법, 심층 강화 학습, 모델 예측 제어, 이동 수평 추정, 편차 감쇄 모델 예측 제어, 적응 슬라이딩 모드 제어

학 번: 2018-37061