

A MODEL PREDICTIVE CONTROL-BASED LANE CHANGING STRATEGY
FOR AUTONOMOUS HIGHWAY DRIVING

A Thesis

by

JU-HSUAN CHANG

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Reza Langari
Committee Members,	Won-jong Kim
	Shankar Bhattacharyya
Head of Department,	Bryan Rasmussen

December 2020

Major Subject: Mechanical Engineering

Copyright 2020 Ju-Hsuan Chang

ABSTRACT

This study proposes a highway driving strategy for autonomous vehicles.

First, a model predictive control (MPC)-based trajectory planner is built based on a kinematic model. A series of candidate strategies are created and form a strategy space. With the model and prediction of surrounding vehicles' movements, the MPC-based planner, according to the candidate strategies, generates feasible trajectories. Next, a decision-making payoff function is applied to select the best trajectory. The payoff function consists of four terms, including lane-changing incentive, cost of controls, cost of risk, and cost of a late lane-changing decision. This decision-making payoff function will select the best trajectory, but this trajectory only provides longitudinal acceleration information.

To maneuver a vehicle, the controller should involve lateral movement. We proposed a yaw rate profile approach as a strategy space for lateral controls. Given longitudinal acceleration, each yaw rate profile will lead the vehicles to a different lateral position, and the one that drives the vehicle to the center of the target lane is the best yaw rate profile. While the vehicle is changing to the target lane, the best yaw rate profile keeps updating. However, because the method to update does not consider the initial error so it fails in some cases. To cope with this issue, an MPC-based path tracking controller is introduced to minimize the error while making the vehicle operating within certain constraints.

Two simulations are created. The first simulation is to test the decision-making payoff function; with a larger weight designed for lane-changing incentive, the autonomous vehicle is more aggressive and more willing to take risks to achieve the lane with higher average speed. The second simulation is designed to show that with the MPC-based path tracking controller, the autonomous vehicle is able to overcome the problems caused by the errors and successfully changes to the target lane.

ACKNOWLEDGEMENTS

I would like to thank Dr. Reza Langari for his support throughout the course of my master's research. In the beginning, when I did not have experience in this discipline, he patiently discussed with me and inspired me; throughout the research, his guidance helps me to build skills for being a researcher. This work would not have been possible without him. I am also grateful to my committee members, Dr. Won-jong Kim and Dr. Shankar Bhattacharyya, for their willingness to help and to provide me different perspectives to improve this work.

In addition, I want to thank all my lab mates. It was a great time to work with them and to discuss critical problems together. Furthermore, I also want to express my gratitude to Dr. Qingyu Zhang who shared his experience and insightful comment with me so that I learned a lot during my research journey.

Most importantly, I must express my very profound gratitude to my family who have been supportive. Whenever I had a difficult time, they are always willing to give me advice and encourage me. This achievement would not have been possible without them. Thank you.

CONTRIBUTORS AND FUNDINGD SOURCES

Contributors

This work was supervised by a thesis committee consisting of Professor Reza Langari, Professor Won-jong Kim of the Department of Mechanical Engineering, and Professor Shankar Bhattacharyya of the Department of Electrical and Computer Engineering.

All other work conducted for the thesis was completed by the student under the advisement of Reza Langari of the Department of Mechanical Engineering.

Funding Sources

This graduate study was supported by Ford Motor Company under the grant URP 2016-8009R.

NOMENCLATURE

Acronyms

ACC	Adaptive cruise control
AV	Autonomous vehicle
IDM	Intelligent driver model
MPC	Model predictive control
PV	Preceding vehicle
SV	Subject vehicle
SRV	Surrounding vehicle
TTC	Time-to-collision
TV	Target vehicle
TPV	Target vehicle's preceding vehicle
V2V	Vehicle-to-vehicle

Symbols

C	Constant distance SV wants to maintain between itself and surrounding vehicles
$C_{\alpha f}$	Front tire's cornering stiffness
$C_{\alpha r}$	Rear tire's cornering stiffness
h	Slack variable
h_r	Minimal TTC during the time when SV changes lane

h_{thres}	Soft threshold of minimal TTC during the time when SV changes lane
H_r	Minimal time headway during the time when SV changes lane
H_{thres}	Soft threshold of minimal time headway during the time when SV changes lane
I_z	Moment of inertia of the vehicle
J	Cost function for MPC-based path tracking controller
J_0	Cost function for the MPC-based trajectory planner
k	MPC step index
l_f	Longitudinal distance from center of mass to the front tire
l_r	Longitudinal distance from center of mass to the rear tire
L	Lane width
m	Mass of the vehicle
N	MPC prediction horizon
N'	The step that SV crosses the lane marking and changes to the target lane
p_i	Vehicle i's longitudinal position
P	Peak values of yaw rate profiles
Q	Weight of deviation from reference outputs
Q_v	Weight of the SV's desired speed
R	Weight of increment of steering angle
R_h	Weight of the slack variable

R_u	Weight of SV's longitudinal acceleration
T_{des}	Desired time headway in IDM
T_{LC}	Time that lane changing takes
T_S	Sampling time
T_U	Time SV previews in the future to evaluate the lane-changing incentive
u_a	SV's longitudinal acceleration
u_{IDM}	Acceleration command of IDM
u_s	SV's steering angle
U_c	Cost of controls
U_{SV}	SV's total payoff
U_l	Cost of a late lane-changing decision
U_r	Cost of risks
U_v	Lane-changing incentive
v_{des}	SV's desired speed
v_i	Vehicle i 's velocity
V_{des}	Desired velocity in IDM
\dot{x}	Longitudinal velocity in the body frame
X	Global X axis coordinate, direction along the road
\dot{y}	Lateral velocity in the body frame
Y	Global Y axis coordinate, direction vertical to the road
z	Outputs

ϕ	Heading angle
$\dot{\phi}$	Yaw rate
Φ	Strategy space of yaw rate profiles
ξ	States

Note: The subscript, i , in p_i and v_i indicates the belonging of this variable. For example, if $i = SV$, p_{SV} is SV's longitudinal position.

TABLE OF CONTENTS

	Page
ABSTRACT	ii
ACKNOWLEDGEMENTS	iv
CONTRIBUTORS AND FUNDINGD SOURCES	v
NOMENCLATURE	vi
TABLE OF CONTENTS	x
LIST OF FIGURES	xii
LIST OF TABLES	xv
1. INTRODUCTION.....	1
1.1. Background and Motivation.....	1
1.2. Terminology	2
1.3. Literature Review	3
1.4. Organization of the Thesis and Objectives	8
2. MOTION PLANNING	10
2.1. Trajectory Planning	10
2.2. Decision Making	15
2.3. Summary	20
3. PATH TRACKING.....	21
3.1. Vehicle Model for Lateral Dynamics.....	21
3.2. Yaw Rate Profile Approach	23
3.3. MPC-based Path Tracking Controller	28
3.4. Summary	35
4. SIMULATIONS.....	36
4.1. Decision Making Simulation.....	37
4.2. Steering Control Simulation.....	42
4.3. Low Speed.....	49

4.4. Summary	54
5. CONCLUSIONS AND FUTURE WORK	55
5.1. Conclusions	55
5.2. Future Work	56
6. REFERENCES	57

LIST OF FIGURES

	Page
Figure 1.1 Vehicle naming	2
Figure 1.2 Schematic diagram of the frame work	9
Figure 2.1 Feasible region	12
Figure 2.2 Lane change timing	12
Figure 2.3 Traffic scene	14
Figure 2.4 Velocity incentive payoff	16
Figure 2.5 Distances considered for risk cost	17
Figure 2.6 Risk cost vs. Time headway for SV and PV	18
Figure 2.7 Risk cost vs. Time-to-collision	20
Figure 3.1 Diagram of vehicle model for lateral dynamics	22
Figure 3.2 Coordinate transformation	23
Figure 3.3 Yaw rate profiles and heading angle profiles example	25
Figure 3.4 Example of trajectories with different yaw rate profiles	25
Figure 3.5 Y displacement vs. Required time to finish lane changing	27
Figure 4.1 Initial condition	37
Figure 4.2 High lane-changing incentive case	38
Figure 4.3 SV's longitudinal velocity, high lane-changing incentive	39
Figure 4.4 SV's time headway, high lane-changing incentive	39
Figure 4.5 TV's time headway, high lane-changing incentive	39
Figure 4.6 Low lane-changing incentive case	40
Figure 4.7 SV's longitudinal velocity, low lane-changing incentive	41
Figure 4.8 SV's time headway, low lane-changing incentive	41

Figure 4.9 TV's time headway, low lane-changing incentive	42
Figure 4.10 Initial condition of the double lane changing scenario	43
Figure 4.11 Double-lane changing without implementing the MPC-based path tracking controller.....	44
Figure 4.12 SV's trajectory, double-lane changing, no MPC-based path tracking controller.....	45
Figure 4.13 SV's steering angle, double-lane changing, no MPC-based path tracking controller.....	45
Figure 4.14 SV's heading angle, double-lane changing, no MPC-based path tracking controller.....	46
Figure 4.15 Double-lane changing with implementing the MPC-based path tracking controller.....	47
Figure 4.16 SV's trajectory, double-lane changing, with the MPC-based path tracking controller.....	48
Figure 4.17 SV's steering angle, double-lane changing, with the MPC-based path tracking controller.....	48
Figure 4.18 SV's heading angle, double-lane changing, with the MPC-based path tracking controller.....	49
Figure 4.19 Initial condition of the low speed scenario	49
Figure 4.20 SV's trajectory, low speed, no MPC-based path tracking controller	50
Figure 4.21 Low speed, without implementing the MPC-based path tracking controller.....	50
Figure 4.22 SV's steering angle, low speed, no MPC-based path tracking controller.....	51
Figure 4.23 SV's heading angle, low speed, no MPC-based path tracking controller.....	51
Figure 4.24 Low speed, with the MPC-based path tracking controller.....	52
Figure 4.25 SV's trajectory, low speed, with the MPC-based path tracking controller.....	52

Figure 4.26 SV's steering angle, low speed, with the MPC-based path tracking controller.....	53
Figure 4.27 SV's heading angle, low speed, with the MPC-based path tracking controller.....	53

LIST OF TABLES

	Page
Table 1 Comparison of high and low lane-changing incentives	42
Table 2 Desired speed of the vehicles	43

1. INTRODUCTION

1.1. Background and Motivation

Autonomous vehicles (AV) are believed to assist humans, make roads safer, and improve traffic in the future. Currently, some related technologies are well developed and implemented on modern cars. These technologies are also known as advanced driver assistance systems (ADAS), including lane keeping, adaptive cruise control (ACC) [1], blind-spot monitor, and automatic emergency braking. The ultimate goal of autonomous vehicles is to drive without human intervention. Furthermore, through vehicle-to-vehicle (V2V) communications, AVs will be able to share traffic information and, in turn, to avoid collisions. However, before V2V is widely used, AVs should interact with human drivers. Interactions with human drivers involve intention estimation and decision making, which usually require prior experiences. Therefore, AVs need to be as intelligent as experienced drivers, proficient at making a correct decision and driving safely.

One of the common maneuvers is to change lane. To ensure safety, AVs have to act like human drivers, including making decisions and driving predictably. Three main tasks are involved during lane changing, which are prediction, evaluation, execution. Prediction is to predict the motion of surrounding vehicles. After prediction, drivers evaluate the feasibility of their strategies. Finally, a driver executes if it is feasible or abandons the intention if it is not. Those are what a human driver usually needs to consider while driving on the highway. Those tasks seem simple for humans,

but if we want to implement on computers used for AVs, we need well-organized logic in programming for AVs to mimic human behaviors.

1.2. Terminology

Some terms are defined in this article for convenience. First, the AV we want to design and control is called subject vehicle (SV). Target vehicle (TV) refers to the vehicle that SV is interacting with or SV may interact with. It is possible to have multiple TVs simultaneously. Other vehicles nearby but not necessarily strongly affecting SV are called surrounding vehicles (SRV). TV is one particular case of surrounding vehicles (SRV). The difference between TV and SRV is that TV and SV are strongly interacting with each other. For example, TV is trying or probably will try to enter the gap in front of SV or vice versa. The vehicle in front of SV is called PV (preceding vehicle), and the vehicle in front of TV is called TPV (target vehicle's preceding vehicle) as shown in Figure 1.1.

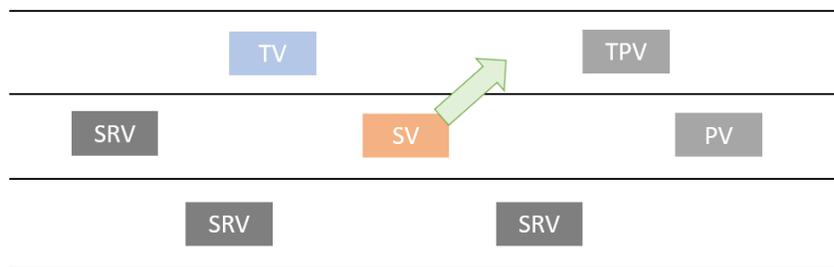


Figure 1.1 Vehicle naming

Two risk indicators are often used to evaluate how risky the situation is between two vehicles [2]. The first one is time headway. Time headway is the time between two

successive vehicles passing the same point or is represented as the distance between two vehicles divided by the rear vehicle's velocity. The second indicator is time-to-collision (TTC). TTC is the time that would take a rear vehicle to collide with a leading one if both vehicles' speeds are maintained. It equals relative distance divided by relative speed while the following vehicle is faster than the leading vehicle so that the value of TTC should always be positive.

1.3. Literature Review

This literature review includes trajectory planning, decision making, and control of the vehicle system.

1.3.1. Trajectory Planning

There are different kinds of trajectory planning approaches for autonomous vehicles, and they are categorized into different types.

Dijkstra's algorithm is a search-based approach to find a path [3]. To build the path, this algorithm first determines two nodes and then finds the shortest path between the two points by connecting a series of other nodes. This approach is useful in an unstructured environment, but it needs to evaluate all the nodes because it sees all the nodes as equal. If the environment is vast, it takes even more time. Therefore, [4] constructs an augmented graph for nodes to bounded the solving time.

A* is another search-based approach, and it introduces heuristic to the searching algorithm, which helps reduce the amounts of nodes needed to be evaluated [5]. However, it is hard to determine a heuristic rule suitable for different scenarios. That the resulting path is not continuous and that this algorithm takes much time to calculate

are two disadvantages of the search-based approach for autonomous vehicles. Although they can be designed to avoid obstacles, in a dynamic environment, that procedure is time-consuming.

A sampling-based approach, Rapid-Exploring Random Tree (RRT) [6, 7], evolves nodes by randomly searching in its search space, so it is adaptable in an unknown environment. Besides, this approach is capable of operating the vehicle within kinematic constraints. However, because it randomly searches the nodes, the resulting path is often too jerky for vehicles to track.

[8] uses discrete strategy space combined with a cost function to plan the trajectory. The authors first generate many speed profiles within a range of acceleration and then discretize the strategy space. A prediction engine will simulate all the scenarios and use a predefined cost function to calculate the cost of each trajectory. The cost function includes factors like progress cost (deviation of desired headway), comfort cost, safety cost, and fuel consumption cost.

Another technique to generate trajectories is curve-based algorithms. One application of curve-based trajectories is to combine segments of curves via a search-based approach to produce a smooth trajectory. In [9], the authors first find endpoints for future trajectories, connect them with polynomial functions, and finally, use boundary conditions on the points to derive the parameter in the polynomial function. The curve-based algorithm's advantage is that the speed and curvature are smooth since they are polynomial function. The other type is to generate a piecewise curve for executing lane changing. [10] applies polynomial functions, and [11] applies Bezier

curve. The advantage of piecewise curves is their continuity and simplicity. On the contrary, they require a predefined speed to track the curve. However, during highway driving, drivers need to adjust their speed sometimes even during lane changing to maintain the gap between the front and rear vehicles, especially when the front vehicle's speed is not constant.

Model predictive control (MPC) is also applied to motion planning for autonomous vehicles. [12] and [13] use MPC to generate multiple feasible trajectories and design payoff functions to evaluate the payoff of each candidate trajectory. Finally, the trajectories with the maximal payoff are used. The advantage of MPC is that it involves the vehicle model, allowing MPC to consider actuator constraints when designing the path. The main drawback of this approach is that MPC relies on the precision of the predefined model but when increasing the model fidelity, it requires high computing ability.

1.3.2. Decision Making

Some trajectory planning algorithms generate a series of feasible trajectories and then find the best trajectory among all. Some trajectories result in keeping driving on the current lane, and the others lead to change lanes. On the other hand, some trajectory planning approaches decide whether to change to the adjacent lane before planning the trajectory. The first type embeds lane selecting in the pathfinding algorithm itself, and the second type chooses a lane followed by pathfinding.

The first type of decision-making approach usually relies on a payoff function to select the best trajectory. Some essential factors in the payoff function are safety payoff, speed, smoothness, control effort (acceleration) and comfort (jerk) [14, 15].

The second type of decision-making approach determines to change lane based on the current situation. [16] develops a driving model to make decision whether to change lane. They build the driving model by using probability and support machine vector (SVM) [17] and then train the model with the Next Generation Simulation (NGSIM) data [18]. Knowing the distances between surrounding vehicles and TV as well as TV's movement, the model can show the probability of each maneuver (turn left, turn right, or keep straight) TV is going to perform in the next step. [19] proposes feasibility criteria for lane changing and merging situations. The feasibility depends on relative position, relative speed, and distance to the merging point (for merging case).

1.3.3. Control of the Vehicle System

To track the reference trajectory, controllers are designed for AVs to drive vehicles. Most of the controllers are built upon the bicycle model, a simplified vehicle dynamics model [20]. The inputs of this system are acceleration and steering angle. In [21], the acceleration is determined by feedforward desired velocity and velocity feedback. As for the steering angle, the author manage to minimize the lateral offset between the lookahead point and the trajectory by adopting backstepping control to ensure control robustness.

[22] constructs paths through points and smooth curves and select the final path through a reward function. It decouples the longitudinal and lateral control and forms a

simple high-level planner and a complex low-level controller. Higher level is used to send a steering angle command based on lookahead point and curvature of the reference path while low-level controller adopts artificial potential field to send an acceleration command. However, the longitudinal control does not consider the rear vehicle in the target lane.

On top of the bicycle model, a higher fidelity model is adopted by considering the details of the tire model. [23] builds a more complex model to derive more reliable input-output relations. The authors assume the vehicle travel at a constant speed, so there is no acceleration input. As for the steering angle, they use nonlinear MPC to send commands to the controller to track the desired heading angle and lateral position.

[24] compares several tracking methods and finds out that MPC has outstanding performance regarding lateral errors and angular errors since it can handle constraints and collision avoidance.

1.3.4. Model Predictive Control (MPC)

Model predictive control is widely used for control problems. For example, automotive, aerospace, and chemical plants apply MPC to control the systems. Given current states, MPC finds an optimal sequence of control inputs for a finite time horizon (also called receding horizon predictive control). The first input of the sequence is then sent to the model and calculating the optimal input series again for the next horizon. Therefore, although MPC does not apply feedback control, the correction is already built-in because it computes optimal inputs periodically and set the initial condition at each sampling time.

To determine optimal inputs, MPC requires two essential elements, a prediction model and a cost function [25]. Furthermore, due to several advantages of MPC, including constraints handling, preview capability, and the fact that it is able to apply to multi-input multi-output systems, it is used in studies of autonomous driving such as [12, 13, 23, 26].

1.4. Organization of the Thesis and Objectives

Because of the dynamic environment on highways, AVs are required to perform efficient algorithms and be able to adjust their own velocity to react to the surrounding vehicle's movements. Thus, this work proposes a two-stage model predictive controls. The first-stage MPC plans the path for AV's kinematic model. The second-stage MPC addresses longitudinal and lateral control to track the path. At this stage, the lateral vehicle dynamics model is linearized at each time step according to the longitudinal velocity, simplifying the model and reducing model mismatch due to varying velocity.

This thesis is organized as follows and a schematic diagram of the framework is shown in Figure 1.2.

presents the basic knowledge about autonomous vehicles and related research in the past.

Chapter 2 is about motion planning. After observing and estimating surrounding vehicles, MPC-based trajectory planner will generate a series of feasible trajectories. The trajectories generated are evaluated by the decision payoff function, and the one offering the best payoff is used as a reference trajectory. However, this trajectory only includes information about the longitudinal movement.

Chapter 3 is about path tracking. In the beginning, the vehicle dynamics model is introduced. Next, this study proposes a yaw rate profile approach. Combining the yaw rate approach and the reference trajectory generated in Chapter 2, the MPC-based path tracking controller controls the lateral movement and tracks the trajectory.

Chapter 4 shows the simulation results of the designs and validates if the proposed method is applicable.

Chapter 5 concludes this work and discusses future work.

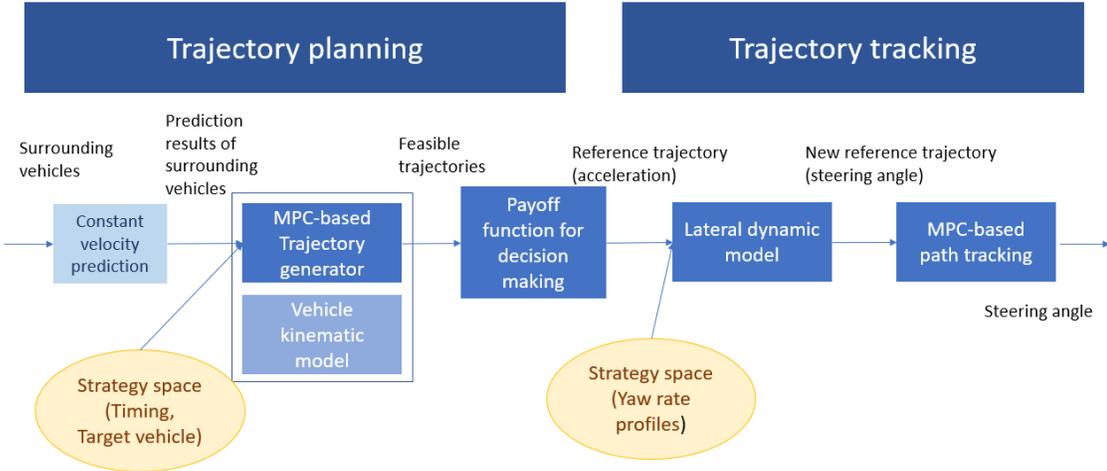


Figure 1.2 Schematic diagram of the frame work

2. MOTION PLANNING

In this section, a motion planning model will be developed. First, an MPC-based trajectory planner will give out several feasible trajectories considering the constraints. Next, a decision-making payoff function selects the best trajectory regarding some important factors.

2.1. Trajectory Planning

In section 2.1, the vehicle kinematic model and the strategy space of SV are first defined. Next, SV's trajectory candidates will be generated through model predictive control (MPC) for each strategy. An MPC-based trajectory generating method has been proposed in previous work [12]. Here, we reformulate it to match this work and solve it for the sake of completeness.

2.1.1. Vehicle Kinematic Model

The vehicle kinematic model is used for trajectory generation. To control the vehicle, we still need a vehicle dynamics model, which will be presented in chapter 3. The kinematic model (2.1) is the longitudinal kinematic of SV.

$$\begin{bmatrix} \dot{p}_{SV}(t) \\ \dot{v}_{SV}(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_{SV}(t) \\ v_{SV}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_a(t) \quad (2.1)$$

where p_{SV} , v_{SV} , and u_a are the longitudinal position, velocity, and acceleration of SV respectively. The model in the form of discrete time is as (2.2),

$$\begin{bmatrix} p_{SV}(k+1) \\ v_{SV}(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_{SV}(k) \\ v_{SV}(k) \end{bmatrix} + \begin{bmatrix} 0 \\ T_s \end{bmatrix} u_a(k) \quad (2.2)$$

where k is the discrete time index and T_s is sampling time.

As for SV's lateral movement, at the stage of trajectory planning, SV is assumed to drive at a constant speed to the target lane. The separation of longitudinal and lateral movement will temporally yield a non-smooth path. This will be addressed in chapter 3, where the control of vehicle dynamics is designed.

2.1.2. Strategy Space

The strategy space of SV involves two factors, to select a target vehicle (TV) from surrounding vehicles and to choose the best timing to change lane.

Before selecting a TV from SRVs, SV has to define the scope of the option. Although it is better to consider as more SRVs so that SV will not miss any chance to find the best gap to enter, we need to confine the strategy space to make the algorithm efficient. Therefore, a feasible region is defined, and SRVs located in that region will be seen as TV candidates. As shown in Figure 2.1, the upper limit of the feasible region is L_f meters ahead of SV, and there is no lower limit. Three closest vehicles within this range in the adjacent lane are the TV candidates.

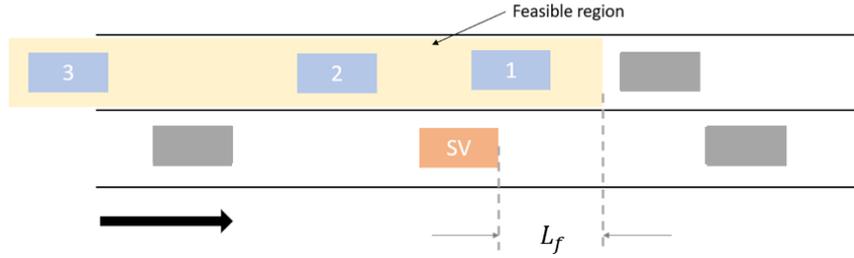


Figure 2.1 Feasible region

The other strategy space is the timing to change lane. After knowing the TV candidates, SV needs to decide the best timing to enter the gap for each of the TV. In section 2.1.3, the MPC-based trajectory planner will determine a trajectory for each timing option. There is also an upper limit timing to change lane, and that timing is the last moment in the prediction horizon of MPC. The prediction horizon is denoted as T_H , so the timing strategy space ranges from the current time to T_H . Furthermore, to make the duration of decision making small, we discretize the strategy space as shown in Figure 2.2.

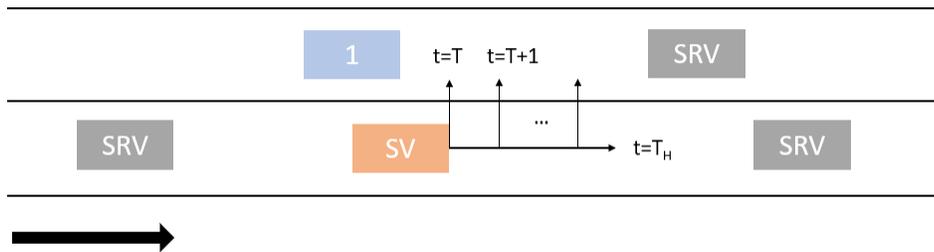


Figure 2.2 Lane change timing

2.1.3. MPC-based Trajectory Generator

The function of MPC is to generate the optimal trajectory based on each strategy. The way it defines optimal is to find a trajectory that minimizes a predefined cost function. In this work, the cost function is taken as

$$\min_{u_a} J_0(v_s, u_a, h) = \sum_{k=1}^N Q_v |v_s(k) - v_{des}|^2 + R_u |u_a(k)|^2 + R_h |h(k)|^2 \quad (2.3)$$

subject to

$$v_s(k) \geq 0 \quad (2.4)$$

$$u_{min} \leq u_a(k) \leq u_{max} \quad (2.5)$$

$$\Delta u_{min} \leq u_a(k) - u_a(k-1) \leq \Delta u_{max} \quad (2.6)$$

for $1 \leq k \leq N'$

$$p_s(k) \leq p_{PV}(k) + h(k) - C \quad (2.7)$$

for $N' + 1 \leq k \leq N$

$$p_s(k) \leq p_{TPV}(k) + h(k) - C \quad (2.8)$$

$$p_s(k) \geq p_{TV}(k) - h(k) + C \quad (2.9)$$

2.2. Decision Making

After the MPC-based trajectory planner generates a series of trajectories for strategies in the strategy space, we have to evaluate each trajectory's payoff and select the one which has the maximal payoff. In section 2.2, a payoff function is defined, and it consists of four terms, lane-changing incentive (U_v), cost of controls (U_c), cost of risks (U_{r1}, U_{r2}, U_{r3}), cost of a late lane-change decision (U_l), as shown in (2.10).

$$U_{SV} = w_v U_v + w_c U_c + w_r (U_{r1} + U_{r2} + U_{r3}) + w_l U_l \quad (2.10)$$

SV's total payoff (U_{sv}) has one positive term and three negative terms. Weighting these terms through weights (w_v, w_c, w_r, w_l) and summing them result in the payoff.

The first term is the lane-changing incentive, and it differentiates the payoff of each lane by evaluating the average speed it can derive from the chosen lane. As shown in Figure 2.4, if SV decides to keep driving on the current lane, it has to follow PV, while if it changes to the left lane, SV follows TPV, allowing SV to have a larger average velocity in T_U seconds because TPV (20 m/s) drives faster than PV (15 m/s). The statement above is valid only when SV's desired speed is greater than 15 m/s. Therefore, this lane-changing incentive payoff calculates the average speed SV can derive by knowing its desire speed and the PV's and TPV's speed and position.

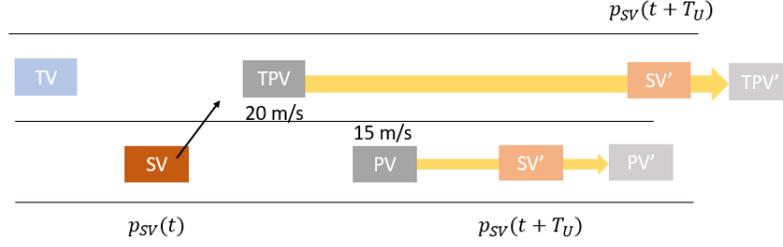


Figure 2.4 Velocity incentive payoff

This lane-changing incentive payoff is taken as the average speed (V_{avg}) minus the SV's desired speed (v_{des}) and then divided by the desired speed, as shown in (2.11) and (2.12). The nominator represents the deviation from the desired speed.

$$V_{avg} = \frac{(p_{SV}(t + T_U) - p_{SV}(t))}{T_U} \quad (2.11)$$

$$U_v = \frac{V_{avg} - v_{des}}{v_{des}} \quad (2.12)$$

The second term in the payoff function is cost of controls. It is related to SV's acceleration and deceleration, meaning how much control effort is required to follow the trajectory. The cost of control is taken as the summation of control input ($u_{a,k}$) within the prediction horizon from step $k = 1$ to $k = N$, as (2.13).

$$U_c = \sum_{k=1}^N -|u_{a,k}| \quad (2.13)$$

The third term is cost of risks. This term is related to a gap between two vehicles during lane changing. Cost of risks consists of three parts because we have to consider distances between SV and three vehicles, PV, TV, and TPV, as shown in Figure 2.5.

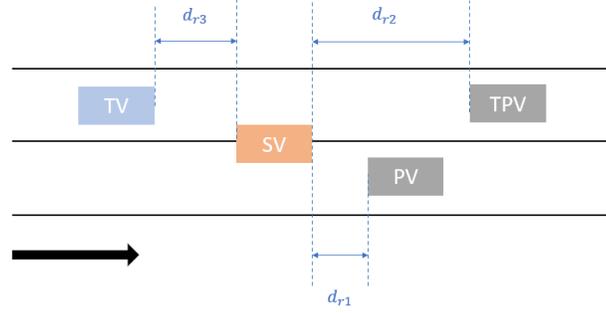


Figure 2.5 Distances considered for risk cost

In Figure 2.5, SV wants to change from the right lane to the left lane. First, the risk between SV and PV is evaluated. This evaluation is associated with the first-half trajectory. SV should always maintain a safe gap between itself and PV during the first half of lane changing. If SV ensures safety gaps, the path is seen as feasible, and the value of risk is defined next. Then the minimal value of the time headway during this first-half trajectory, which can be seen as the riskiest moment during lane changing, is used (4.2) and the value of risk cost is, in turn, defined as (4.2).

$$H_{r1} = \min_{t \in t_{pr}} \frac{d_{r1}(t)}{v_{SV}(t)} \quad (2.14)$$

$$U_{r1} = -C_1^{(H_{r1} - H_{thres})} \quad (2.15)$$

where $t \in t_{pr}$ means the first half of the lane changing process. H_{r1} is the minimal time headway during this process. d_{r1} is the distance between SV and PV over time. C_1 and H_{thres} are two constants. The relation between the minimal time headway and the value of risk cost is shown in Figure 2.6. With the time headway decreasing, the value of risk cost exponentially increases.

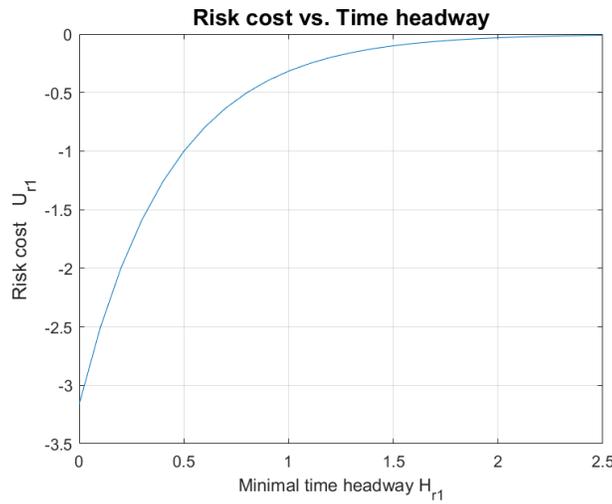


Figure 2.6 Risk cost vs. Time headway for SV and PV

The next step is to evaluate the risk in the rest of lane-changing process. The same safety criterium is applied here; only when the safety gap is ensured, will the trajectory be practical. However, the way to define the value of risk cost is slightly different here. Instead of the time headway, time-to-collision (TTC) at the moment when SV crossing the lane marking is used. Because once SV crosses the lane, TV behind it should adjust its velocity to create a safe distance between SV and TV. TTC at the moment of crossing can represent how much time SV and TV have for adjusting

their velocity and enlarge the gap in front of them. TTC and risk cost are shown in (2.16) to (2.19).

$$h_{r2} = \frac{d_{r2}}{v_{SV} - v_{TPV}}, \forall v_{SV} > v_{TPV} \quad (2.16)$$

$$h_{r3} = \frac{d_{r3}}{v_{TV} - v_{SV}}, \forall v_{TV} > v_{SV} \quad (2.17)$$

$$U_{R2} = \begin{cases} -c_2^{(h_{r2}-h_{thres})} & , v_{SV} > v_{TPV} \\ 0 & , v_{SV} \leq v_{TPV} \end{cases} \quad (2.18)$$

$$U_{R3} = \begin{cases} -c_2^{(h_{r3}-h_{thres})} & , v_{TV} > v_{SV} \\ 0 & , v_{TV} \leq v_{SV} \end{cases} \quad (2.19)$$

where c_2 and h_{thres} are two constants. d_{r2} is the distance between SV and TPV, and h_{r2} is the corresponding TTC. d_{r3} is the distance between TV and SV, and h_{r3} is the corresponding TTC. The relation between TTC and the risk is shown in Figure 2.7. With the TTC decreasing, the threat is huger, so the value of risk cost exponentially increases.

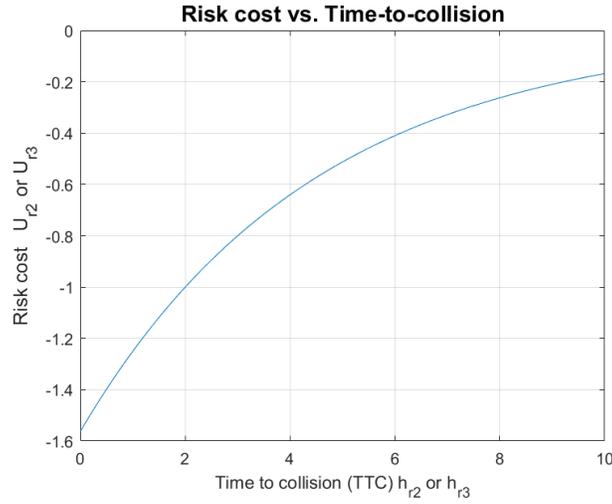


Figure 2.7 Risk cost vs. Time-to-collision

The final term in the payoff function is cost of a late lane-changing decision. Because the prediction of other vehicles' later states is more uncertain, this term encourages sooner lane changes over later ones. In (2.17) N' means that SV will change lane at the N' th step in the prediction horizon.

$$U_L = -N' \quad (2.20)$$

2.3. Summary

This chapter presents the trajectory planning method. First, different lane change timings and different target vehicles in the strategy space combined with the MPC-based trajectory planner create multiple trajectory candidates. Next, we define a decision-making payoff function, and it includes four terms, lane-changing incentive, cost of controls, cost of risks, and cost of a late lane-changing decision. The trajectory which gives the maximal payoff value is chosen to be the reference in the next chapter.

3. PATH TRACKING

This section introduces approaches to control the vehicle and to track the reference trajectory. The reference trajectory generated in the previous chapter only provides the longitudinal movement in the global frame at each step and the best timing to change lane. However, we need more information to control the vehicle such as reference heading angle and reference yaw rate, so that a controller can decide how much acceleration or steering angle it should put into the system to track the path.

In this chapter, a vehicle model for lateral dynamics is discussed. On top of that, we propose a yaw rate profile approach as a strategy space for lateral control. The best yaw rate profile is then adopted and refines the reference trajectory by specifying the heading angle at each step in the prediction horizon. To track the new reference trajectory within the system constraints, another MPC is used in this chapter to control the steering angle.

3.1. Vehicle Model for Lateral Dynamics

In this study, acceleration and steering angle are two inputs for controlling the vehicle system. A vehicle model for lateral control proposed in [20] is used for this control problem. As shown in Figure 3.1, \dot{x} is longitudinal velocity in the body frame and \dot{y} is lateral velocity in the body frame. ϕ and $\dot{\phi}$ are heading angle and yaw rate of the vehicle. $C_{\alpha f}$ and $C_{\alpha r}$ are cornering stiffness of front and rear tires respectively. l_f is the longitudinal distance from center of mass (C.O.M) to front tires and l_r is the longitudinal distance from center of mass to rear tires. m is the mass of the vehicle. I_z

is yaw moment of inertia of the vehicle. u_s is the steering angle. X is the direction along the road in the global frame and Y is the direction vertical to the road in the global frame.

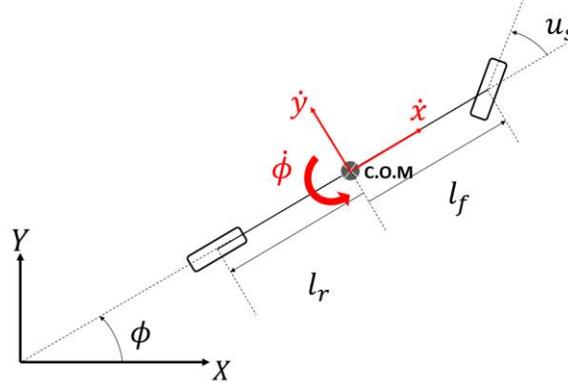


Figure 3.1 Diagram of vehicle model for lateral dynamics

The state space model of this vehicle model is as (3.1). The input of this model is steering angle, and the states are lateral velocity, heading angle, and yaw rate. Note that the states propagation depends on longitudinal velocity \dot{x} , and that lateral velocity \dot{y} and yaw rate $\dot{\phi}$ are highly affecting each other. That means when controlling to track one of the variables, the others are being affecting. Hence, the reference trajectory should be designed in a way that the three variables are associated.

$$\frac{d}{dt} \begin{bmatrix} \dot{y} \\ \phi \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{2C_{\alpha f} + 2C_{\alpha r}}{m\dot{x}} & 0 & -\dot{x} - \frac{2C_{\alpha f}l_f - 2C_{\alpha r}l_r}{m\dot{x}} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2l_f C_{\alpha f} - 2l_r C_{\alpha r}}{I_z \dot{x}} & 0 & -\frac{2l_f^2 C_{\alpha f} + 2l_r^2 C_{\alpha r}}{I_z \dot{x}} \end{bmatrix} \begin{bmatrix} \dot{y} \\ \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} \frac{2C_{\alpha f}}{m} \\ 0 \\ \frac{2l_f C_{\alpha f}}{I_z} \end{bmatrix} u_s \quad (3.1)$$

3.2. Yaw Rate Profile Approach

The coordinate of the trajectory derived from the previous chapter is in global frame. That only provides the desired position and velocity in X position at each step in the prediction horizon. However, to control a vehicle, we need to convert it to the body frame because the acceleration applied to vehicle aligns with its longitudinal direction. The relation between global frame and body frame is shown in the Figure 3.2 and equation (3.2).

$$\begin{aligned}\dot{X} &= \dot{x}\cos\phi - \dot{y}\sin\phi \\ \dot{Y} &= \dot{x}\sin\phi + \dot{y}\cos\phi\end{aligned}\tag{3.2}$$

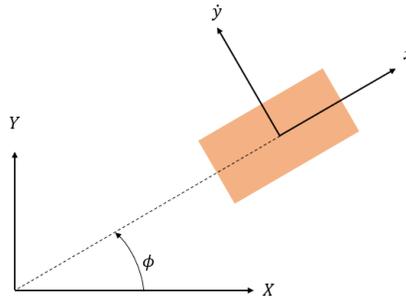


Figure 3.2 Coordinate transformation

In ordinary lane changing cases, heading angle ϕ and lateral velocity \dot{y} are relatively small so the first equation in (3.2) is simplified to (3.3).

$$\dot{X} = \dot{x}\cos\phi\tag{3.3}$$

By applying (3.3), longitudinal velocity \dot{x} within the prediction horizon is known once the $\phi(t)$ is determined, so the next step is to find an appropriate heading angle, $\phi(t)$ and yaw rate, $\dot{\phi}(t)$.

To make the trajectory smooth during lane changing, a vehicle's yaw rate over time should be continuous and thus results in a continuous heading angle. Several continuous yaw rate profiles with different peaks are created in advanced through the function (3.4) and they are in the profile strategy space, Φ .

$$\dot{\phi}(t) = P \sin\left(\frac{2\pi t}{T_{LC}}\right) \quad (3.4)$$

where P is the peak value, T_P is the time period that lane changing takes.

For example, in Figure 3.3 two profiles from the profile strategy space, Φ , are shown. The higher peak yaw rate profile (Strategy 1) leads to the higher peak heading angle profile, and so as the lower peak (Strategy 2) ones. We assumed lane changing takes T_{LC} seconds. During first $0.5T_{LC}$ seconds a vehicle's heading angle keeps increasing and the rest of time it keeps decreasing. Moreover, the higher peak profile leads to greater displacement in Y direction, which is shown in Figure 3.4.

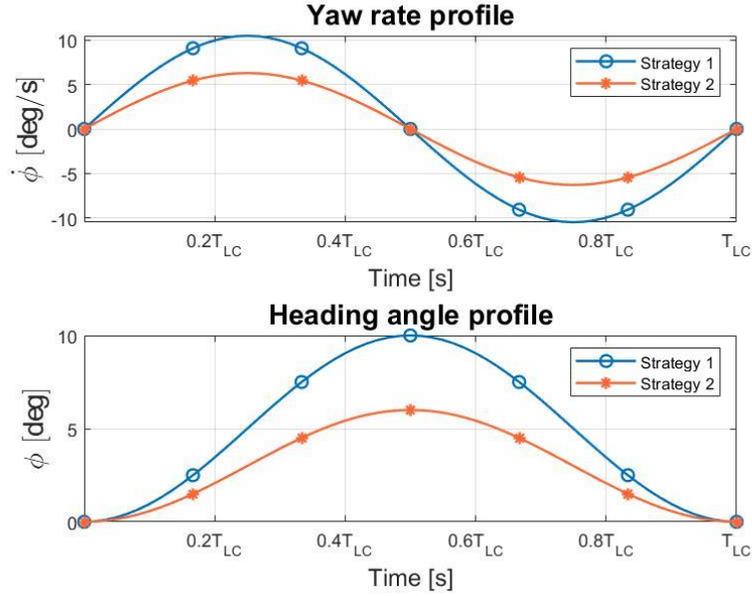


Figure 3.3 Yaw rate profiles and heading angle profiles example

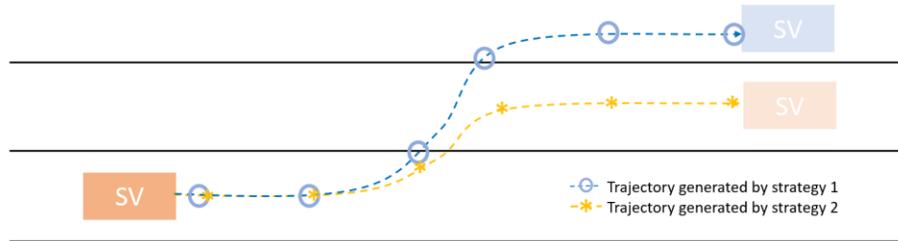


Figure 3.4 Example of trajectories with different yaw rate profiles

Applying (3.3), We derive corresponding longitudinal velocity, \dot{x} . Furthermore, the vehicle model (3.5) provides the relation among input u_s , lateral velocity \dot{y} , and yaw rate $\dot{\phi}$. For each yaw rate and heading angle profile, we can calculate the corresponding \dot{y} and u_s .

$$\frac{d}{dt} \begin{bmatrix} \dot{y} \\ \dot{\phi} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{2C_{\alpha f} + 2C_{\alpha r}}{m\dot{x}} & 0 & -\dot{x} - \frac{2C_{\alpha f}l_f - 2C_{\alpha r}l_r}{m\dot{x}} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2l_f C_{\alpha f} - 2l_r C_{\alpha r}}{I_z \dot{x}} & 0 & -\frac{2l_f^2 C_{\alpha f} + 2l_r^2 C_{\alpha r}}{I_z \dot{x}} \end{bmatrix} \begin{bmatrix} \dot{y} \\ \dot{\phi} \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} \frac{2C_{\alpha f}}{m} \\ 0 \\ \frac{2l_f C_{\alpha f}}{I_z} \end{bmatrix} u_s \quad (3.5)$$

Next, obtain \dot{Y} by using the coordinate transformation equation, and then integrate this variable from current time $t = T$ to the time when lane changing is finished $t = T_{LC}$ as shown in (3.6). The displacement in Y direction is known for each yaw rate profile.

$$\Delta Y = \int_{t=T}^{t=T_{LC}} \dot{Y} dt \quad (3.6)$$

In Figure 3.4, since the desired endpoint is at the center of the target lane, the trajectory generated by strategy 2 surpasses the trajectory generated by strategy 1. That means the lower peak yaw rate profile is the best option $\dot{\phi}_{opt}(t)$ in the profile strategy space, Φ . The above description can be written in an equation (3.7) and Y_{target} is the Y position of the center of the target lane.

$$\dot{\phi}_{opt}(t) = \min_{\phi \in \Phi} \left| \int_{t=T}^{t=T+T_{LC}} \dot{Y}(\dot{\phi}(t)) dt - Y_{target} \right| \quad (3.7)$$

After the optimal yaw rate $\dot{\phi}_{opt}(t)$ is determined, the corresponding reference steering angle $u_s(t)$ is also derived, which is the reference input to the vehicle system for lateral control.

During lane changing, the best yaw rate profile should also update all the time. The update method used in this work is based on the SV's Y position. This work first assumes that the relation between the position Y and required time to finish lane

changing follows the curve in Figure 3.5 generated by a function (3.8). In Figure 3.5, ΔY means the displacement the vehicle has made, L is the lane width, and T_{LC} is total lane changing time. For example, if the vehicle has pass $0.3065L$ [m] then it still needs $0.6T_{LC}$ [sec] to finish the rest. Therefore, the updated yaw rate profile is to extract the yaw rate profiles in strategy space from $0.6T_{LC}$ to the end of the profiles, and among all the profile strategies, the best yaw rate profile is determined by using the same way described before.

$$\frac{\Delta Y}{L} = -P \sin\left(\frac{2\pi t}{T_{LC}}\right) \times \frac{T_{LC}}{2\pi} + Pt \quad (3.8)$$

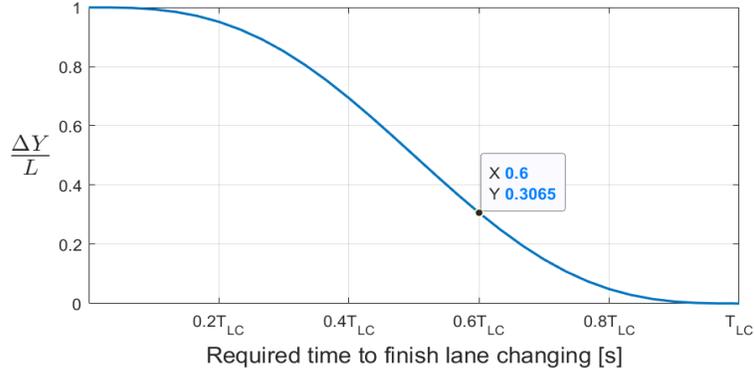


Figure 3.5 Y displacement vs. Required time to finish lane changing

However, this updating method introduces a problem. When updating the profiles, it calculates the final Y position by assume that the vehicle follows the heading angle profile, which is usually not the case. Therefore, the error of true heading angle accumulates and sometimes lead to failed lane changing. Hence, the MPC-based path tracking controller is designed to solve this problem.

3.3. MPC-based Path Tracking Controller

Although the reference trajectory and reference inputs are determined from section 0, the way does not consider the state errors. Therefore, model predictive control is applied to avoid errors accumulate while planning the new trajectory. Furthermore, MPC handles constraints, enabling variables to operate within the system constraints.

In section 3.3, an MPC-based path tracking controller is proposed to solve the control problem. First, state variables of the control problem are defined, and the equations of motion are linearized and discretized for MPC. Finally, MPC will yield the optimized inputs.

3.3.1. Defining the Linear State Space Model

The variables needed to be constrained are used as outputs for the state space model of the MPC. Those constraints include heading angle, yaw rate, and Y position, and the related dynamics equations of motion are written as (3.9) to (3.11).

$$\frac{d}{dt}\dot{y} = -\frac{2C_{\alpha f} + 2C_{\alpha r}}{m\dot{x}} \times \dot{y} - \left(\dot{x} + \frac{2C_{\alpha f}l_f - 2C_{\alpha r}l_r}{m\dot{x}} \right) \times \dot{\phi} + \frac{2C_{\alpha f}}{m} \times u_s \quad (3.9)$$

$$\frac{d}{dt}\dot{\phi} = -\frac{2l_f C_{\alpha f} - 2l_r C_{\alpha r}}{I_z \dot{x}} \times \dot{y} - \frac{2l_f^2 C_{\alpha f} + 2l_r^2 C_{\alpha r}}{I_z \dot{x}} \times \dot{\phi} + \frac{2l_f C_{\alpha f}}{I_z} \times u_s \quad (3.10)$$

$$\frac{d}{dt}Y = \dot{x} \sin \phi + \dot{y} \cos \phi \quad (3.11)$$

(3.9) and (3.10) are two equations from the vehicle model for lateral motion. The equation (3.11) is the coordinate transformation function and is nonlinear. Thus,

we need to linearize this model for the MPC problem. A compact form of equations (3.9) to (3.11) is written as (3.12).

$$\dot{\xi} = f(\xi, u_s) \quad (3.12)$$

where $\xi = [\dot{y} \ \phi \ \dot{\phi} \ Y]^T$, and $\dot{\xi}$ is function of ξ and u_s .

By applying Taylor's series expansion for linearization, (3.12) becomes

$$\xi - \hat{\xi} = \left. \frac{\partial f(\xi, u_s)}{\partial \xi} \right|_{\hat{\xi}, \hat{u}_s} (\xi - \hat{\xi}) + \left. \frac{\partial f(\xi, u_s)}{\partial u_s} \right|_{\hat{\xi}, \hat{u}_s} (u_s - \hat{u}_s). \quad (3.13)$$

$\hat{\xi} = [\dot{y} \ \phi \ \dot{\phi} \ Y]^T$ and \hat{u}_s are the operating points which are also the values used in reference trajectory derived from the best yaw rate profiles approach. Rewrite (3.13) in the matrix form will derive

$$\delta \dot{\xi}(t) = A_c(t) \delta \xi(t) + B_c \delta u_s(t) \quad (3.14)$$

where the state matrix is

$$A_c(t) = \begin{bmatrix} -\frac{2C_{\alpha f} + 2C_{\alpha r}}{m\dot{x}(t)} & 0 & -\dot{x}(k) - \frac{2C_{\alpha f}l_f - 2C_{\alpha r}l_r}{m\dot{x}(t)} & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{2l_f C_{\alpha f} - 2l_r C_{\alpha r}}{I_z \dot{x}(t)} & 0 & -\frac{2l_f^2 C_{\alpha f} + 2l_r^2 C_{\alpha r}}{I_z \dot{x}(t)} & 0 \\ \cos \hat{\phi}(t) & \dot{x}(t) \cos \hat{\phi}(t) - \dot{y} \sin \hat{\phi}(t) & 0 & 0 \end{bmatrix} \quad (3.15)$$

$$B_c = \begin{bmatrix} \frac{2C_{\alpha f}}{m} & 0 & \frac{2l_f C_{\alpha f}}{I_z} & 0 \end{bmatrix}^T. \quad (3.16)$$

In (3.14) those “ δ ” terms represent the amount of deviation from the operating point. For example, $\delta\xi = \xi - \hat{\xi}$, $\delta\dot{\xi} = \dot{\xi} - \hat{\dot{\xi}}$, and $\delta u_s = u_s - \hat{u}_s$. Note that the matrix A_c is time-varying, since it includes time-varying parameters, \dot{x} and $\hat{\phi}$.

Next, by applying Euler’s method, the model is discretized with sampling time T_s and becomes

$$\delta\xi(k+1) = A(k)\delta\xi(k) + B\delta u_s(k) \quad (3.17)$$

where $A = I + A_c T_s$ and $B = B_c T_s$.

A discretized and linear state space model is derived, so the MPC-based path tracking controller can be formulated. The variables used in MPC are not the states of the vehicle; instead, they are those “ δ ” terms, which are the amount of deviation from the reference trajectory and reference input defined earlier. Besides, the output of the system is taken as

$$\delta z_k = C\delta\xi_k = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta y_k \\ \delta\phi_k \\ \delta\dot{\phi}_k \\ \delta Y_k \end{bmatrix} \quad (3.18)$$

where the output matrix C is used to extract variables from the $\delta\xi$.

3.3.2. Solving the Control Problem

To optimize the reference input, this work defines a cost function (3.19) for the MPC based on state space model defined previously. The target of this optimization problem is to minimize the sum of weighted amount of deviation from the reference

output and weighted increment of control input (steering angle) while satisfying the constraints (3.20) to (3.24) within the prediction horizon from current state to step N .

$$J = \sum_{k=1}^N \|z_k - \hat{z}_k\|_Q + \|\Delta u_{s,k}\|_R \quad (3.19)$$

subject to

$$u_{s, \min} < u_{s,k} < u_{s, \max} \quad (3.20)$$

$$\Delta u_{s, \min} < u_{s,k} - u_{s,k-1} < \Delta u_{s, \max} \quad (3.21)$$

$$\phi_{\min} < \phi_k < \phi_{\max} \quad (3.22)$$

$$\dot{\phi}_{\min} < \dot{\phi}_k < \dot{\phi}_{\max} \quad (3.23)$$

$$Y_{\min} < Y_k < Y_{\max} \quad (3.24)$$

Because the variables used in the linear state space model is the amount of deviation from the reference trajectory, we need to substitute the cost function and constraints with those variables. Therefore, the optimization problem becomes (3.25) to (3.30).

$$J = \sum_{k=1}^N \|\delta z_k\|_Q + \|(\hat{u}_{s,k} + \delta u_{s,k}) - (\hat{u}_{s,k-1} + \delta u_{s,k-1})\|_R \quad (3.25)$$

Subject to

$$u_{s, min} < \hat{u}_{s, k} + \delta u_{s, k} < u_{s, max} \quad (3.26)$$

$$\Delta u_{s, min} < (\hat{u}_{s, k} + \delta u_{s, k}) - (\hat{u}_{s, k-1} + \delta u_{s, k-1}) < \Delta u_{s, max} \quad (3.27)$$

$$\phi_{min} < \hat{\phi}_k + \delta \phi_k < \phi_{max} \quad (3.28)$$

$$\dot{\phi}_{min} < \hat{\dot{\phi}}_k + \delta \dot{\phi}_k < \dot{\phi}_{max} \quad (3.29)$$

$$Y_{min} < \hat{Y}_k + \delta Y_k < Y_{max} \quad (3.30)$$

To solve the problem, it is necessary to know the relation between δz_k and $\delta u_{s, k}$. First, express all the δz_k with initial condition, $\delta \xi_0$, and the control inputs, $\delta u_{s, k}$, and derive (3.31).

$$\begin{aligned} \delta z_1 &= C \delta \xi_1 = C A_0 \delta \xi_0 + C B \delta u_{s, 1} \\ \delta z_2 &= C \delta \xi_2 = C A_1 A_0 \delta \xi_0 + C A_1 B \delta u_{s, 1} + C B \delta u_{s, 2} \\ \delta z_3 &= C \delta \xi_3 = C A_2 A_1 A_0 \delta \xi_0 + C A_2 A_1 B \delta u_{s, 1} + C A_2 B \delta u_{s, 2} + C B \delta u_{s, 3} \\ &\vdots \\ \delta z_N &= C \delta \xi_N = C \prod_{i=0}^{N-1} A_i \delta \xi_0 + C \prod_{i=0}^{N-1} A_i B \delta u_{s, 1} + \dots + C B \delta u_{s, N}. \end{aligned} \quad (3.31)$$

Write the (3.31) in the form of matrix and it becomes:

$$\delta Z = \Psi \Delta \xi_0 + \Theta \delta \mathcal{U}_s \quad (3.32)$$

where

$$\delta Z = [\delta z_1, \delta z_2, \dots, \delta z_N]^T \quad (3.33)$$

$$\Psi = [C A_0, C A_1 A_0, \dots, C \Pi_{i=0}^{N-1} A_i]^T \quad (3.34)$$

$$\Theta = \begin{bmatrix} C B & 0 & \dots & 0 \\ C A_1 B & C B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C \Pi_{i=0}^{N-1} A_i & C \Pi_{i=0}^{N-1} A_i B & \dots & C B \end{bmatrix} \quad (3.35)$$

$$\delta \mathcal{U}_s = [\delta u_{s,1}, \delta u_{s,2}, \dots, \delta u_{s,N}]^T \quad (3.36)$$

Next, the second term of cost function is expressed with $\delta \mathcal{U}_s$ which becomes:

$$M_1 \hat{\mathcal{U}}_s + M_2 \delta \mathcal{U}_s \quad (3.37)$$

where M_1 is the following $(N + 1) \times N$ matrix

$$M_1 = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & \dots & 0 & 0 \\ 0 & 0 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -1 & 1 \end{bmatrix} \quad (3.38)$$

$$\hat{\mathcal{U}}_s = [u_{s,0} \quad \hat{u}_{s,1} \quad \hat{u}_{s,2} \quad \dots \quad \hat{u}_{s,N}]^T \quad (3.39)$$

and M_2 is the following $N \times N$ matrix

$$M_2 = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ -1 & 1 & \dots & 0 & 0 \\ 0 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & -1 & 1 \end{bmatrix}. \quad (3.40)$$

Therefore, the cost function can be substitute with $\delta u_{s,k}$ and the cost function is rewritten as

$$J = (\Psi \delta \xi_0 + \Theta \delta \mathcal{U}_s)^T \mathbf{Q} (\Psi \delta \xi_0 + \Theta \delta \mathcal{U}_s) + (M_1 \hat{\mathcal{U}}_s + M_2 \delta \mathcal{U}_s)^T \mathbf{R} (M_1 \hat{\mathcal{U}}_s + M_2 \delta \mathcal{U}_s). \quad (3.41)$$

Finally, the cost function is formulated into a quadratic program problem, as shown from (3.42) to (3.44).

$$J = \delta \mathcal{U}_s^T H \delta \mathcal{U}_s + 2 \delta \mathcal{U}_s^T f + Const. \quad (3.42)$$

where

$$H = \Theta^T \mathbf{Q} \Theta + M_2^T \mathbf{R} M_2 \quad (3.43)$$

$$f = \Theta^T \mathbf{Q} \Psi \delta \xi_0 + M_2^T \mathbf{R} M_1 \hat{\mathcal{U}}_s \quad (3.44)$$

The $\delta \mathcal{U}_s$ that minimizes the cost function while satisfying the constraints is the optimal input $\delta \mathcal{U}_{s,opt}$ for the current prediction horizon as shown in (3.45).

$$\delta \mathcal{U}_{s,opt} = \min_{\delta \mathcal{U}_s} J \quad (3.45)$$

Note that $\delta \mathcal{U}_{s,opt}$ is the deviation from reference inputs and the actual inputs to send to the system is $\mathcal{U}_s = \hat{\mathcal{U}}_s + \delta \mathcal{U}_{s,opt}$.

The MPC-based path tracking controller includes errors of initial states in the cost function and panelizes the errors, so accumulation of state errors is minimized.

3.4. Summary

In this chapter, a path tracking strategy is proposed. The longitudinal acceleration is derived from chapter 2. As for the lateral control, this work proposes a yaw rate profile approach. Several profiles are predefined in the strategy space. The best yaw rate profile $\dot{\phi}_{opt}(t)$ leads the autonomous vehicle to desired Y position. Furthermore, the updating method for yaw rate profiles is introduced. However, because the model has uncertainty, and the updating method does not consider state errors, which leads to the accumulation of error and sometimes finally fails to change lane. An MPC-based path tracking controller is then introduced to solve this problem. Because MPC considers the vehicle states and can penalize the deviation from the desired trajectory, that avoids errors heading angle to increase. This MPC is linearized at each step to handle the varying longitudinal speed, reducing the model mismatch. This MPC-based path tracking controller finds optimized control inputs for the lateral motion and sends the steering angle command, u_s , to the autonomous vehicle for tracking the desired trajectory.

4. SIMULATIONS

In this chapter, simulations are constructed to show the results of the proposed strategy. The simulations in the section 4.1 present how the weight in the payoff function affects the decision. The simulations in the sections 4.2 and 4.3 are made to compare two control methods. One is to control the steering angle with only the yaw rate profile approach, and the other is to control the steering angle with the yaw rate profile plus the MPC-based path tracking controller.

In the simulation screenshot photos, SV is represented by an empty rectangle, while SRVs are the filled rectangles. SRVs all follow the same intelligent driver model (IDM) [27]. The driving style of IDM is affected by two parameters, desired velocity and desired time headway. The equation of IDM is:

$$u_{IDM}(v, \Delta v, s^*) = a \left[1 - \left(\frac{v}{V_{des}} \right)^\mu - \left(\frac{s^*(v, T_{des})}{s_\alpha} \right)^2 \right] \quad (4.1)$$

$$s^*(v, T_{des}) = s_0 + vT_{des} + \frac{v\Delta v}{2\sqrt{ab}} \quad (4.2)$$

where u_{IDM} is the control input (acceleration), v is current speed, Δv is the relative speed between the vehicle and its preceding vehicle, s_α is the distance between the vehicle and its preceding vehicle, a and b are maximum acceleration and comfortable deceleration respectively, μ characterizes how acceleration decreases with velocity, V_{des} and T_{des} are the vehicle's desired velocity and desired time headway.

4.1. Decision Making Simulation

A simulation scenario is designed to test the effect of changing the weight of lane-changing incentive, w_v in (2.10). Two simulations are made; one is with a higher weight for lane-changing incentive and the other is with a lower weight. There are three lanes in the simulation scenario, and SV is driving behind a very slow vehicle in the middle lane. SV may want to change to either the left or the right lane. The average speed on the left lane is higher than that on the right lane. SV's desired speed is 20 [m/s]. The initial condition of this simulation is shown in the Figure 4.1.

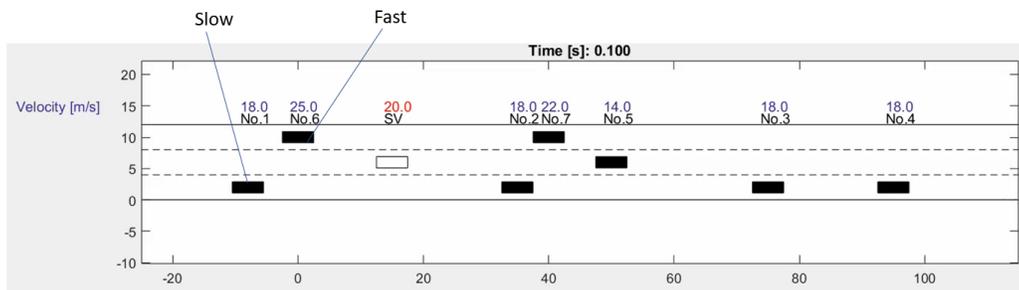


Figure 4.1 Initial condition

The result indicates that a higher lane-changing incentive drives the autonomous vehicle to the left lane, as Figure 4.2. Moving to the left lane allows SV to have higher speed, but this option is riskier and takes more effort.

Since the vehicles on the left lane are faster than SV in the beginning, SV needs to speed up to catch up on the gap between car No.6 and car No.7. SV's longitudinal velocity over time is shown in Figure 4.3. SV keeps accelerating until it enters the gap in front of car No.6, and then it gradually slows down to its desired velocity.

SV's time headway before changing lane is shown in Figure 4.4 ,and TV's time headway after SV's changing lane is shown in Figure 4.5. Because SV keeps moving forward and accelerating, time headway decreases continuously. SV's headway is 0.8856 [s] and TV's (No.6's) time headway is 0.4986 [s] at the moment it crosses the lane markings.

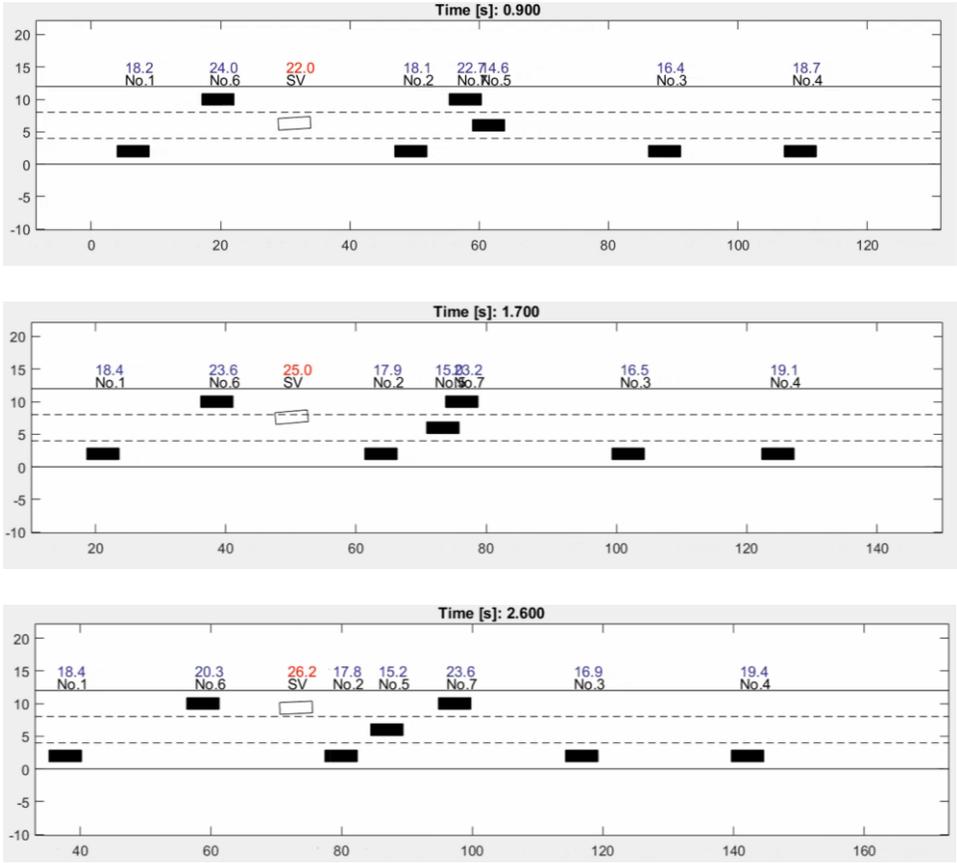


Figure 4.2 High lane-changing incentive case

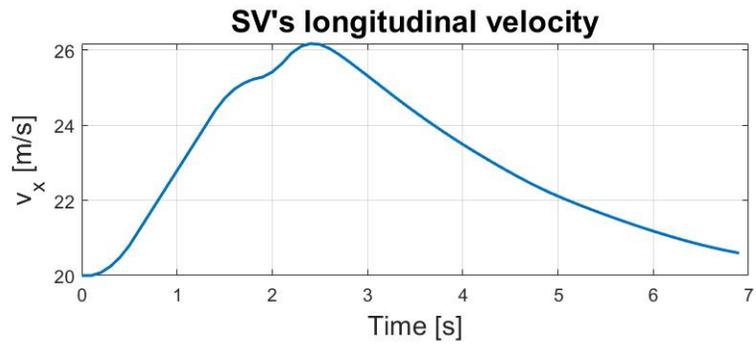


Figure 4.3 SV's longitudinal velocity, high lane-changing incentive

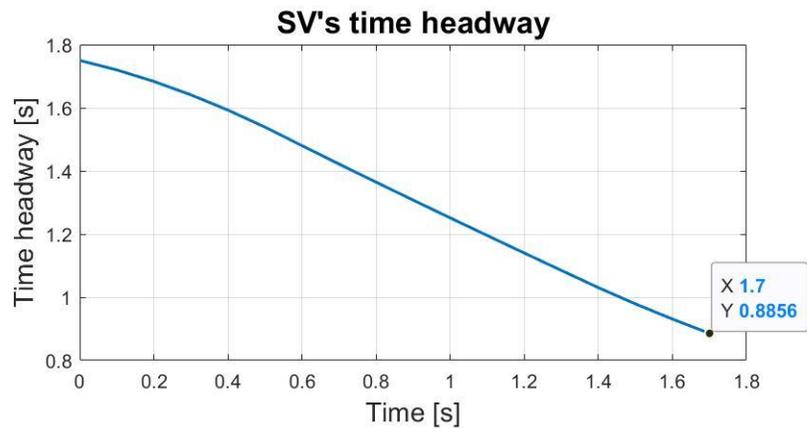


Figure 4.4 SV's time headway, high lane-changing incentive

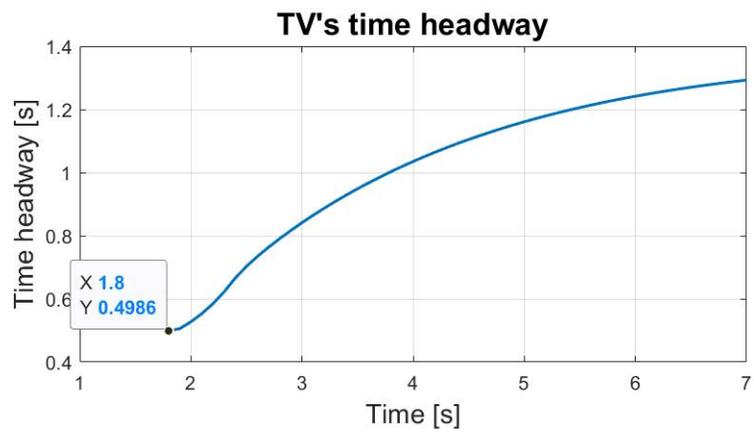


Figure 4.5 TV's time headway, high lane-changing incentive

The other simulation is created with the same situation but a lower lane-changing incentive, as shown in Figure 4.6. With a smaller weight of lane-changing incentive, SV will be less eager to drive fast, so it chooses to change to the right lane where SV can avoid being stuck by No. 5 while having a safer lane-changing process than the left lane.

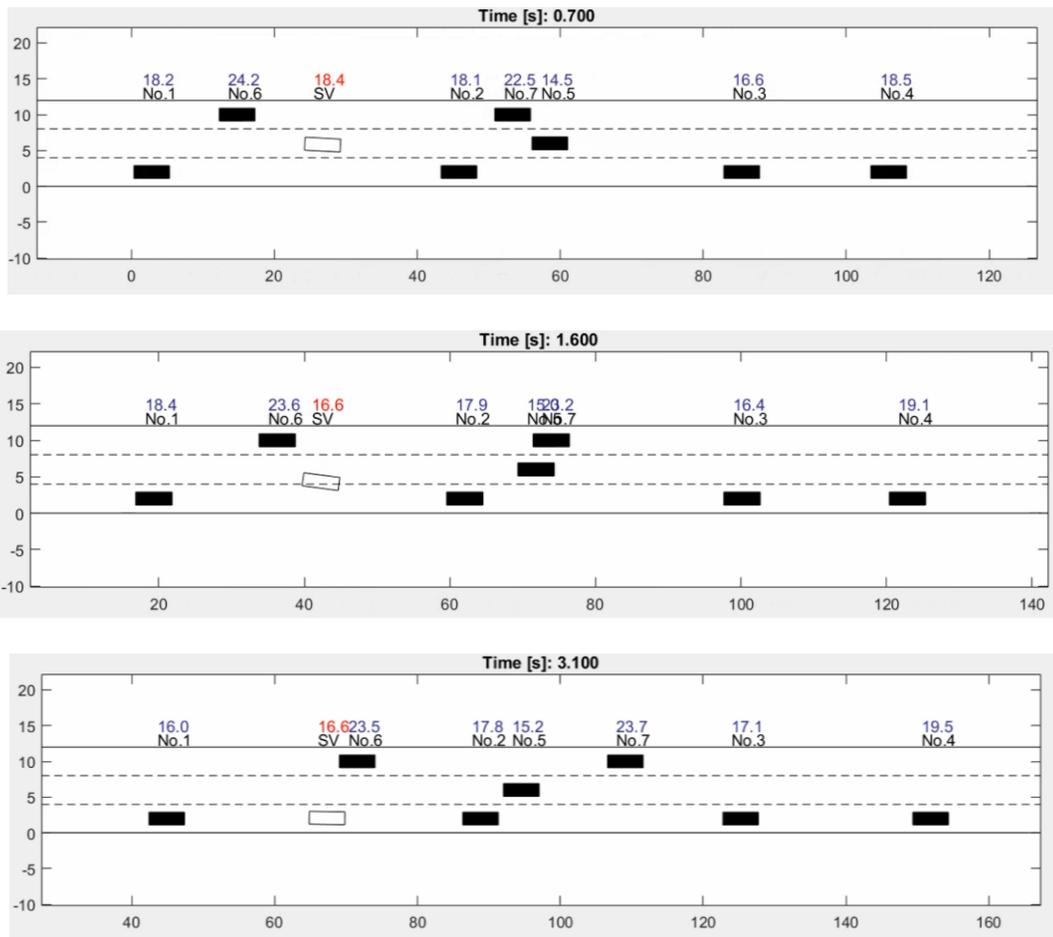


Figure 4.6 Low lane-changing incentive case

Figure 4.7 shows SV's velocity during lane changing. Compared to Figure 4.3, in Figure 4.7, SV has to decrease its current speed to enter the gap in front of car No.1.

On top of that, the minimal value in Figure 4.8 is larger than Figure 4.4, and the minimal value in Figure 4.9 is larger than Figure 4.5, meaning it is less risky for SV to choose the right lane, as shown in Table 1.

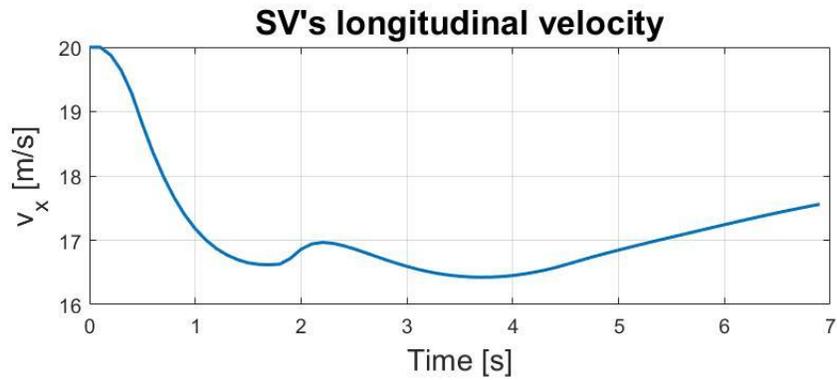


Figure 4.7 SV's longitudinal velocity, low lane-changing incentive

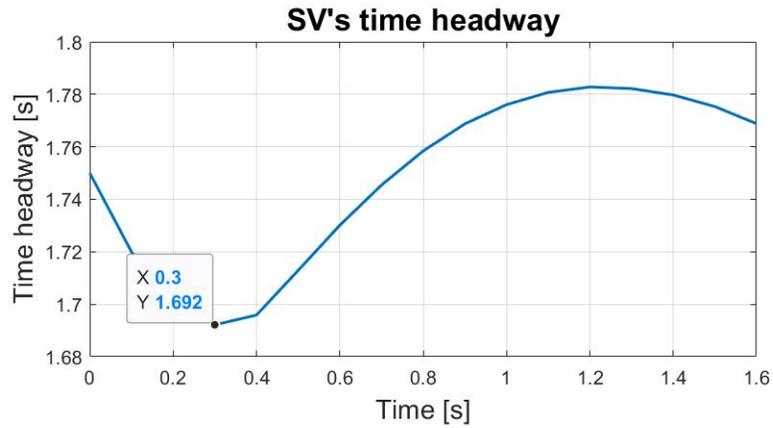


Figure 4.8 SV's time headway, low lane-changing incentive

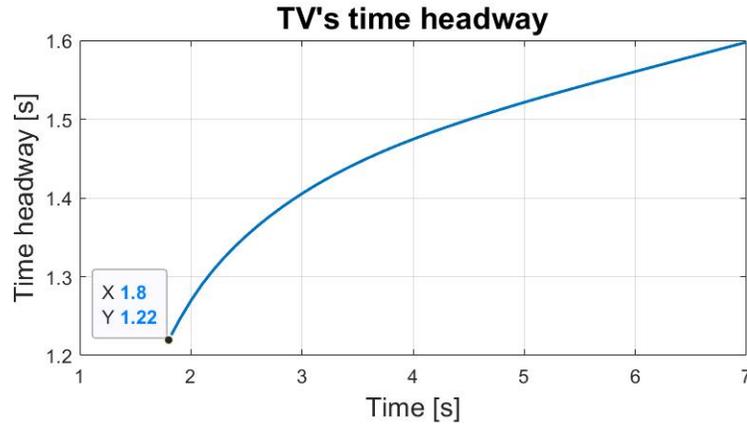


Figure 4.9 TV's time headway, low lane-changing incentive

Table 1 Comparison of high and low lane-changing incentives

	SV's minimal time headway	TV's minimal time headway
High	0.886	0.499
Low	1.692	1.22

4.2. Steering Control Simulation

Two scenarios are designed to test the performance of the MPC-based path tracking controller. For each scenario, we will use two different ways to control the SV, one is with the MPC-based path tracking controller, and the other is merely sending input command based on the best yaw rate profile.

4.2.1. Double-Lane Changing

In the first scenario, SV is on the right lane and driving behind a slow vehicle whose velocity is 20 [m/s]. SV's desired speed is 31 [m/s], so it may want to change to the adjacent lane to maintains its desired speed. The average speed on the left lane is

higher than that on the middle lane, as shown in the Figure 4.10, and each vehicle's desired speed is shown in Table 2.

Table 2 Desired speed of the vehicles

	SV	NO.1	NO.2	NO.3	NO.4	NO.5	NO.6	NO.7	NO.8
Desired speed [m/s]	31	20	31	28	28	28	28	35	30

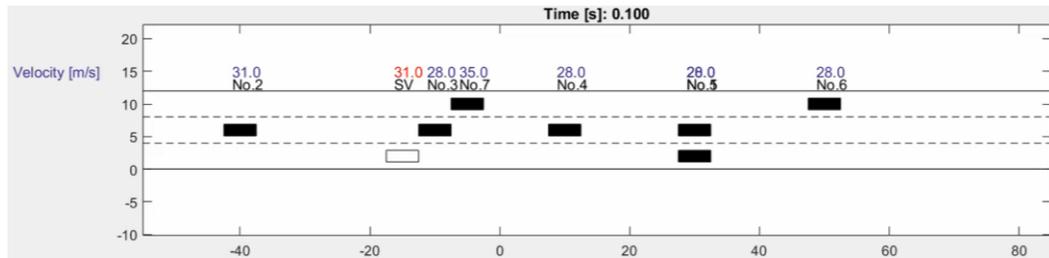


Figure 4.10 Initial condition of the double lane changing scenario

Two control approaches are implemented, controlling the steering angle with only the best yaw rate profile and controlling the steering angle with the best yaw rate profile modified by the MPC-based path tracking controller. The result of controlling with only the best yaw rate profile is shown in Figure 4.11. SV speeds up and moves into the gap between Noo.3 and No.4 successfully. However, it fails when it tries to change to the third lane. Because its heading angle is not zero when starting the second lane changing (the yaw rate profile approach assumes the initial heading angle is zero), it fails to achieve the target lane's center before the steering angle starts to decrease.

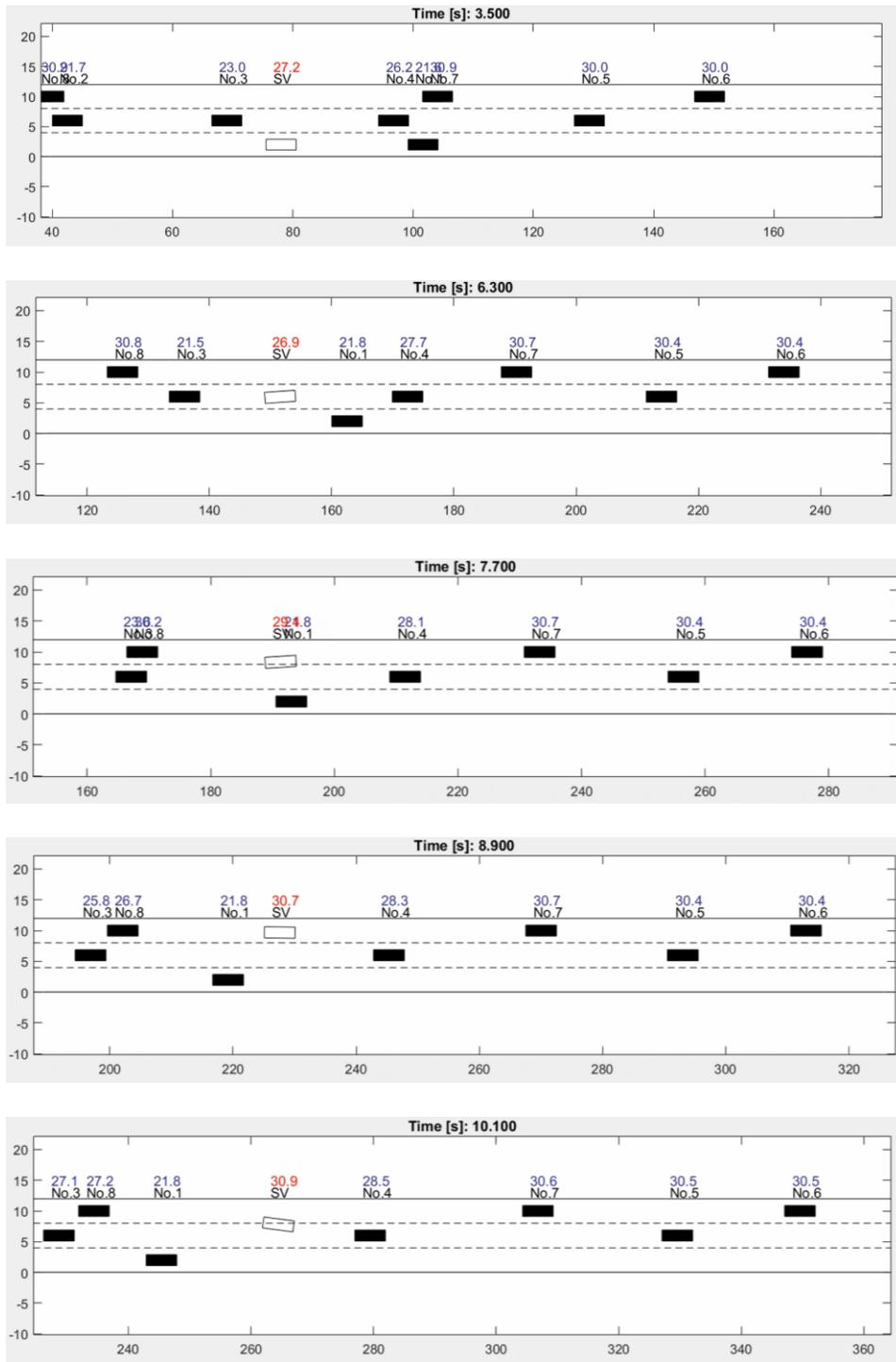


Figure 4.11 Double-lane changing without implementing the MPC-based path tracking controller

In Figure 4.12, SV fails to change to the left lane and turns back to the middle lane. Because the updating approach for the best yaw rate profile is based on SV's Y position (Figure 3.5), the approach starts to make the steering angle decrease upon SV crosses the lane marking, as shown in Figure 4.13.

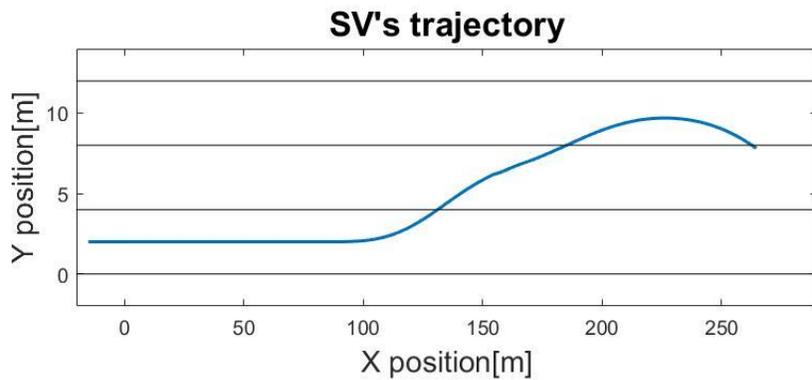


Figure 4.12 SV's trajectory, double-lane changing, no MPC-based path tracking controller

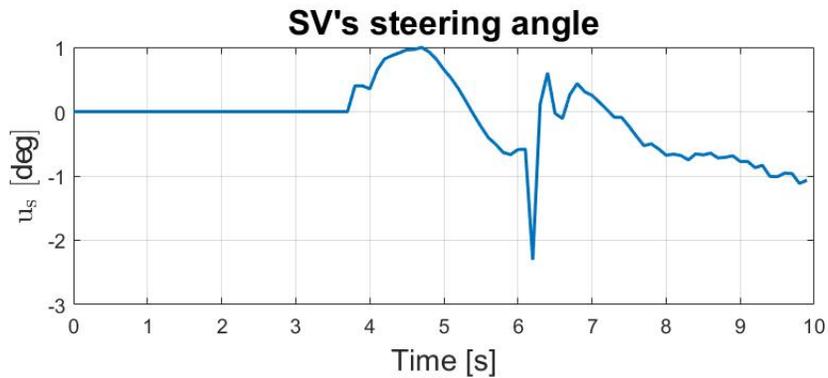


Figure 4.13 SV's steering angle, double-lane changing, no MPC-based path tracking controller

Decreasing steering angle causes that SV's yaw rate decreases, making the heading angle keep decreasing (Figure 4.14) and deviates from the center of the target lane (left lane).

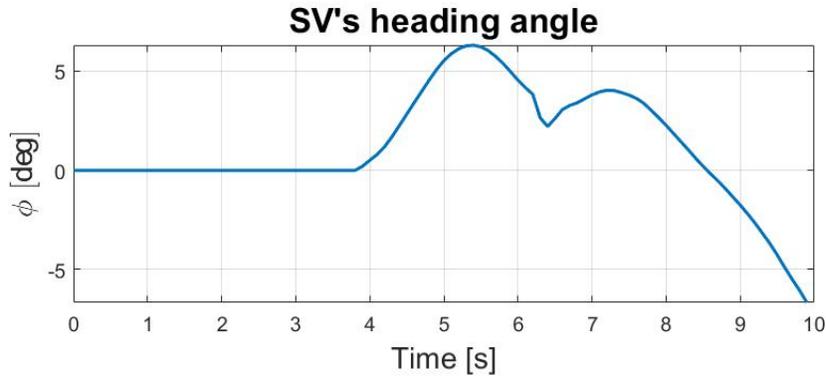


Figure 4.14 SV's heading angle, double-lane changing, no MPC-based path tracking controller

In a tracking problem, the desired heading angle is the state to be tracked. The best yaw rate profile generates the trajectory and corresponding inputs. This approach tracks that corresponding inputs but not the heading angle. Due to this characteristic, the best yaw rate profile cannot handle the cases where the heading angle is not as expected initially. Without considering the discrepancy, the error of heading angle accumulates, causing the failure of lane changing. Therefore, the MPC-based path tracking controller is implemented to cope with this problem because it is able to minimize the state errors and modify the inputs determined by the best yaw rate profile. The simulation result of applying the MPC-based path tracking controller is shown in Figure 4.15, and the trajectory is presented in Figure 4.16.

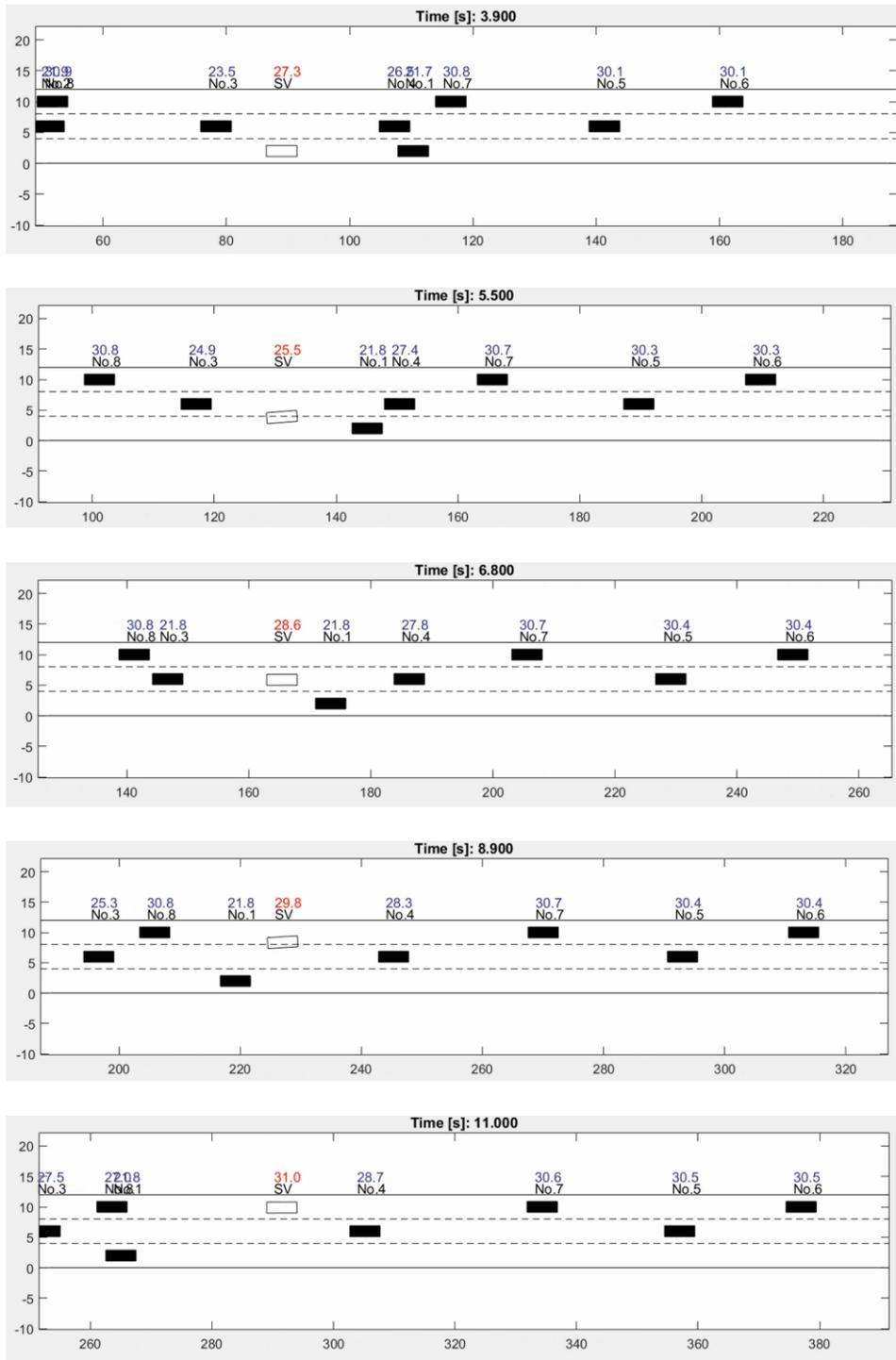


Figure 4.15 Double-lane changing with implementing the MPC-based path tracking controller

With the MPC-based path tracking controller, SV can consider the error of the heading angle so that SV does not merely follow the inputs yield by the best yaw rate profile. The controller adjusts the steering angle to reduce the heading angle error during lane changing, making it track the desired heading angle, as shown in Figure 4.17 and Figure 4.18.

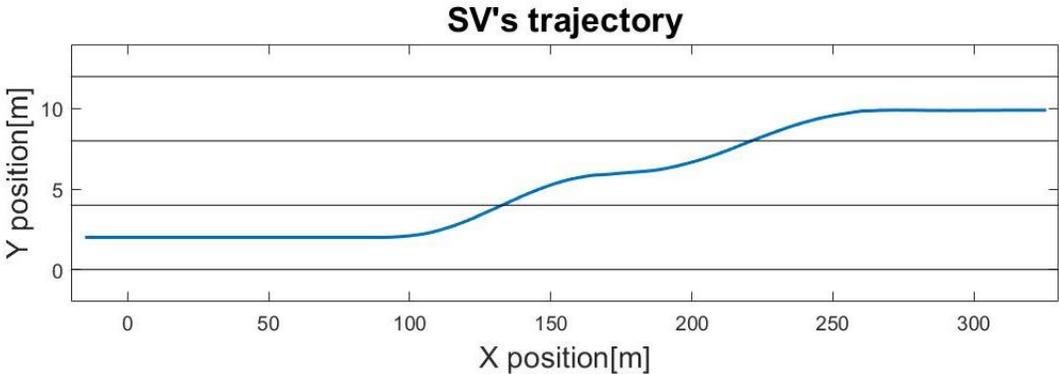


Figure 4.16 SV's trajectory, double-lane changing, with the MPC-based path tracking controller

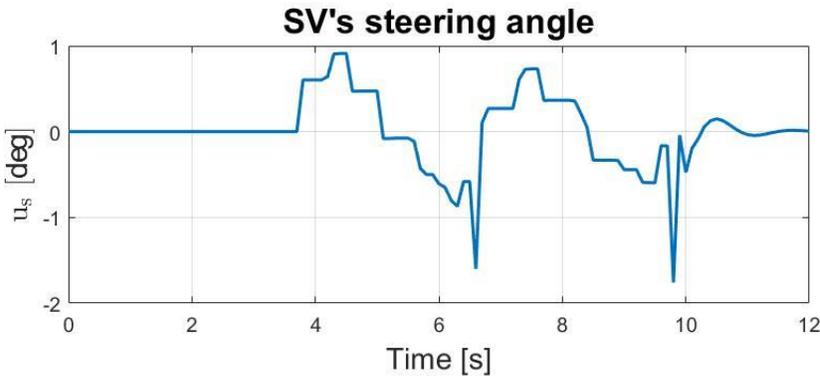


Figure 4.17 SV's steering angle, double-lane changing, with the MPC-based path tracking controller

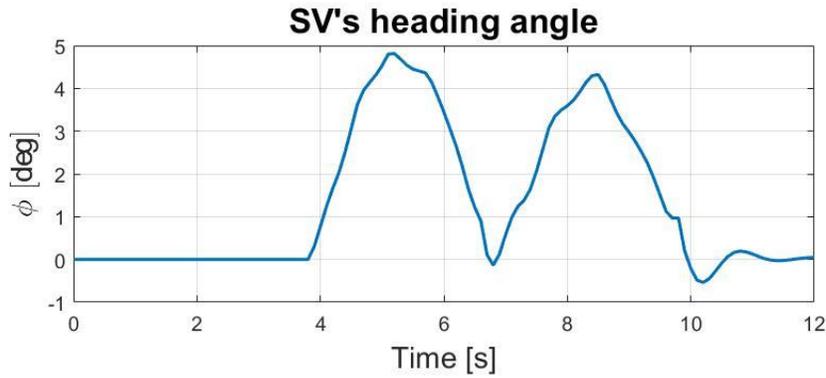


Figure 4.18 SV's heading angle, double-lane changing, with the MPC-based path tracking controller

4.3. Low Speed

The other simulation is constructed. In this scenario, the average speed of the whole traffic is low. The initial condition is shown in Figure 4.19. In this simulation, SV is driving behind an extremely slow vehicle whose speed is 8 [m/s] while other vehicles' speeds are 12 – 15 [m/s] on the adjacent lane. Therefore, SV will move to the adjacent lane to maintain its desired speed, 15 [m/s].

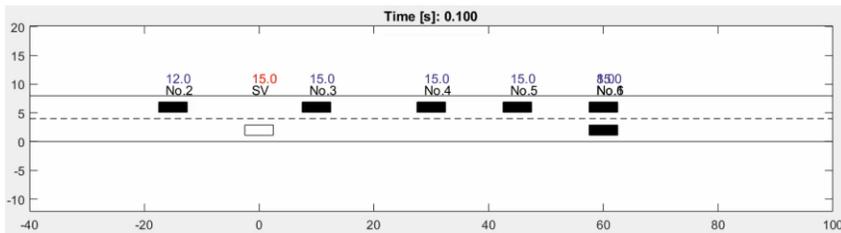


Figure 4.19 Initial condition of the low speed scenario

Because the updating method follows Figure 3.5, which is not the exact relation between Y position and the time left before reaching the center of the adjacent lane, it

fails, especially in low-speed cases. The simulation where SV drives without the MPC-based path tracking controller is shown in Figure 4.20 (trajectory) and Figure 4.21.

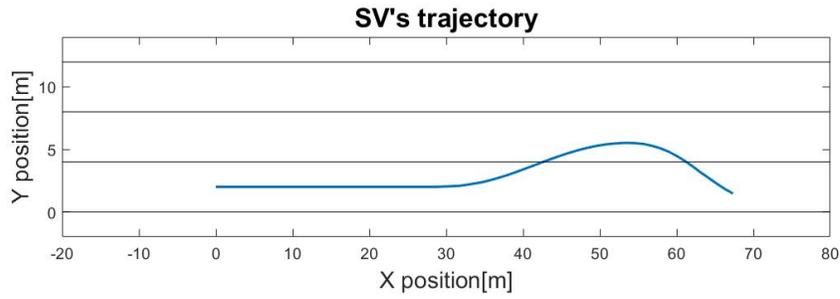


Figure 4.20 SV's trajectory, low speed, no MPC-based path tracking controller

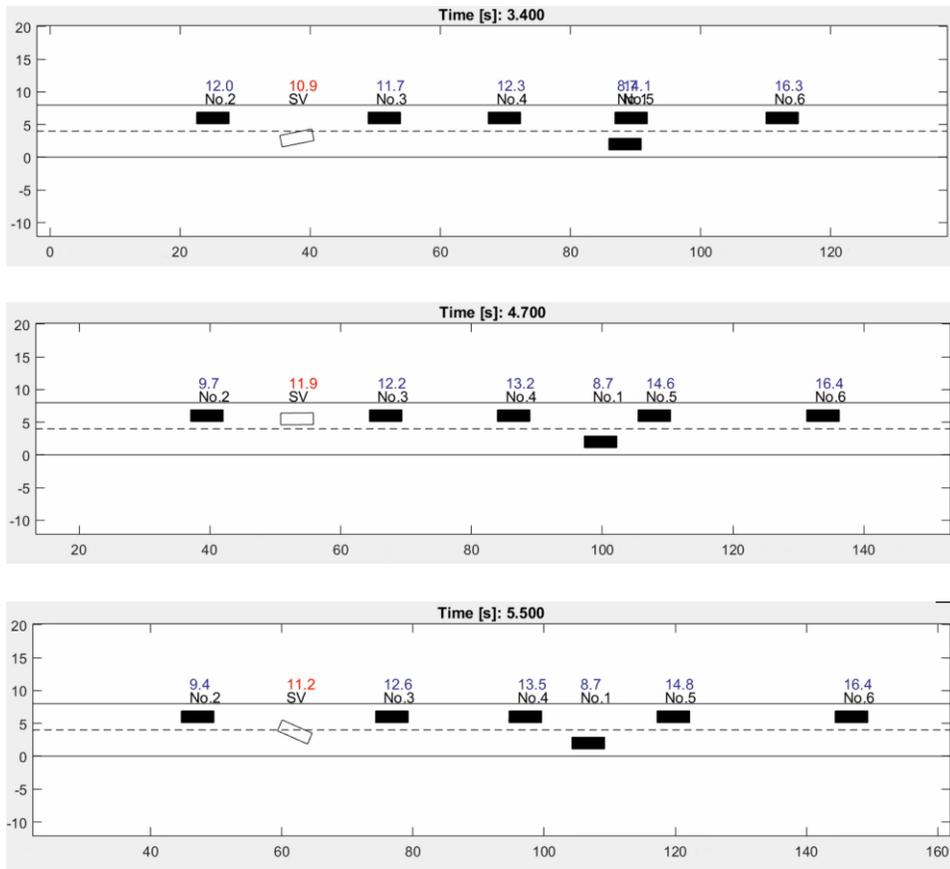


Figure 4.21 Low speed, without implementing the MPC-based path tracking controller

SV fails to achieve the target lane's center before its steering angle starts to decrease, making the heading angle decreases, as shown in Figure 4.22 and Figure 4.23.

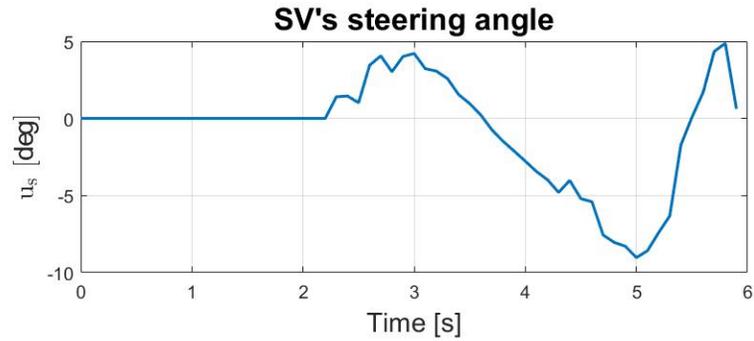


Figure 4.22 SV's steering angle, low speed, no MPC-based path tracking controller

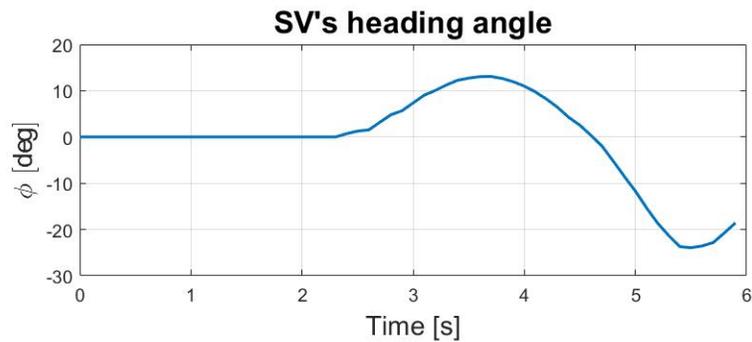


Figure 4.23 SV's heading angle, low speed, no MPC-based path tracking controller

Like the previous case, by implementing the MPC-based path tracking controller, SV overcomes the issue caused by the imperfect updating approach. The simulation result is shown as Figure 4.24, Figure 4.25, Figure 4.26, and Figure 4.27.

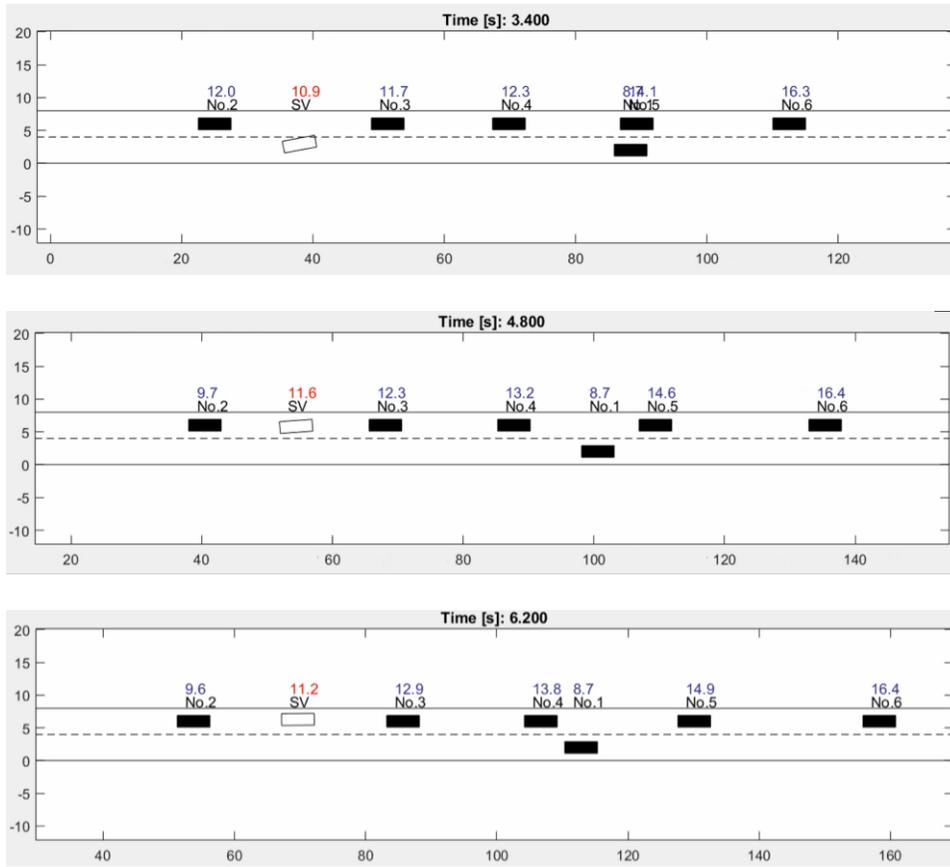


Figure 4.24 Low speed, with the MPC-based path tracking controller

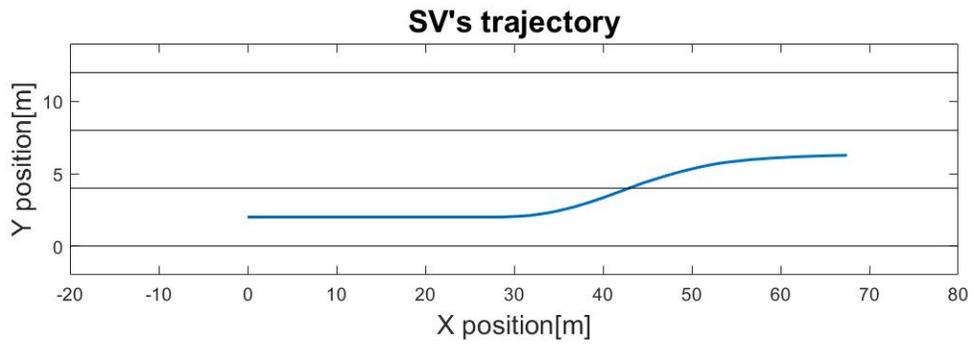


Figure 4.25 SV's trajectory, low speed, with the MPC-based path tracking controller

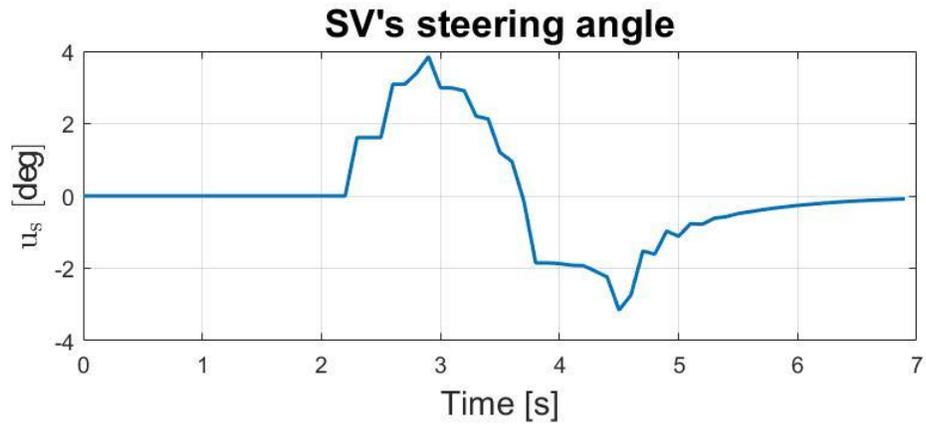


Figure 4.26 SV's steering angle, low speed, with the MPC-based path tracking controller

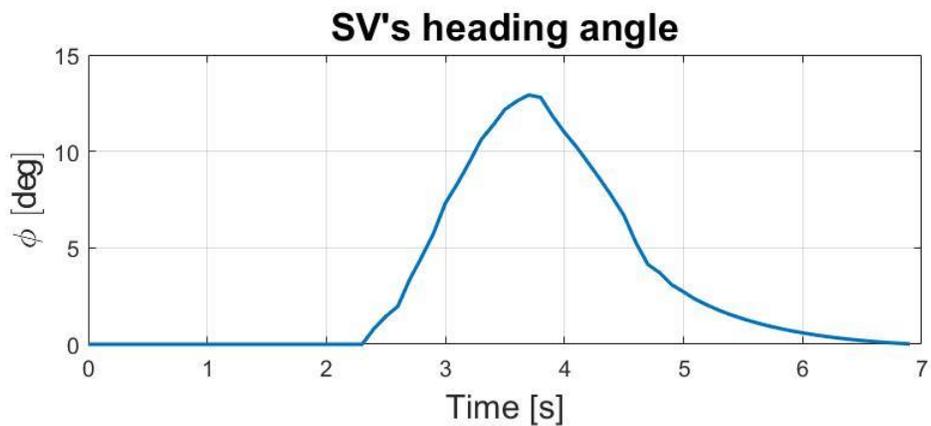


Figure 4.27 SV's heading angle, low speed, with the MPC-based path tracking controller

The proposed MPC-based path tracking controller modifies the input to fit the desired heading angle at each step. The simulations validate the feasibility of the proposed MPC-based path tracking controller. Furthermore, it is able to handle varying longitudinal velocity during lane changing, which is important since this prove the AV's adaptability when facing a situation where it needs to accelerate or decelerate during lane changing.

4.4. Summary

In this chapter, one simulation is made to test the decision-making payoff function, and the other two simulations are constructed to test the performance of the proposed MPC-based path tracking controller.

The first simulation suggests that the proposed payoff function can cooperate with the trajectory generated by the MPC-base trajectory planner and make reasonable decisions for the simple scenario designed in this work. Moreover, by adjusting the weight of the lane-changing incentive, it makes different decisions. Similar to human drivers, some of them are willing to take risks or put more control effort to derive higher speed, but some tend to drive conservatively.

The second and the third simulations both indicate that the yaw rate profile approach and its updating method are not perfect for autonomous vehicles since it assumes the initial heading angle is zero and does not consider the state errors, and Figure 3.5 is not the actual relation between Y position and required time to finish lane changing. By applying the MPC-based path tracking controller, the inputs are modified. It makes SV track the heading angle and Y position instead of the inputs determined by the best yaw rate profile to minimize the errors.

5. CONCLUSIONS AND FUTURE WORK

5.1. Conclusions

This study proposed a two-stage MPC to plan and track the trajectory. At the first stage, MPC generates trajectories for the vehicle kinematic model, and the predefined payoff function is applied to decide the best trajectory. The MPC-based trajectory planner allows SV to accelerate and decelerate during lane changing, which is able to handle the dynamic environment, meaning that SV can adjust its speed based on observing other vehicle's movement. However, this trajectory includes only information about longitudinal movements in the global frame. To control a vehicle, we need to transform that information into the body frame, and the vehicle lateral dynamic model is used to control the steering angle.

Here, this study proposes a yaw rate profile approach. Each strategy leads to different lateral displacement results, and the one that brings the autonomous vehicle to the center of the target lane is the best yaw rate profile. Although the yaw rate profile combined with the vehicle model can yield the reference steering angle, this approach does not consider the state errors and may lead to failure.

Therefore, the second-stage MPC controller is used to control the steering angle. Because MPC can take states into consideration and penalize state errors, it prevents errors from growing. In other words, the best yaw rate profile yields reference inputs and reference states. What we want to track is the reference states instead of the reference inputs. Hence, the reference inputs are modified according to the initial state

error and the reference inputs through implementing the MPC-based path tracking controller.

The way to build the MPC-based path tracking controller includes determining a vehicle model, linearizing the model, deciding a cost function, and solving that minimization problem. The model is built upon the vehicle lateral dynamics model and coordinate transformation function and then linearized at each step around the reference states. Since the lateral dynamic depends on longitudinal velocity, which varies at each step, the state space models at each step are different. The cost function consists of two terms: minimizing the state errors and minimizing the steering angle increment. Finally, this minimization problem is converted into a quadratic programming problem and solved to give the final steering angle command.

The simulation result shows that the proposed strategy for lane changing is feasible and can handle varying velocity during lane changing.

5.2. Future Work

Model predictive control strongly relies on the precision of the model. Therefore, if the model used in the controller is far from the ground truth, then the controller is not robust and can fail. Future works should handle model uncertainty and disturbances by applying stochastic MPC or robust MPC to address this issue.

The decision-making payoff function involves weights that can be tuned to match average human behavior. Further investigation is needed to adjust the weights of the payoff function.

6. REFERENCES

- [1] H. Winner, S. Witte, W. Uhler and B. Lichtenberg, "Adaptive Cruise Control System Aspects and Development Trends," *SAE Transactions*, vol. 105, pp. 1412-1421, 1996.
- [2] K. Vogel, "A comparison of headway and time to collision as safety indicators," *Accident Analysis & Prevention*, vol. 35, no. 3, pp. 427-433, 2003.
- [3] M. Bahram, A. Wolf, M. Aeberhard and D. Wollherr, "A prediction-based reactive driving strategy for highly automated driving function on freeways," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, Dearborn, MI, 2014.
- [4] T. Gu, J. M. Dolan and J.-W. Lee, "Runtime-Bounded Tunable Motion Planning for Autonomous Driving," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, Gothenburg, 2016.
- [5] M. Likhachev, G. Gordon and S. Thrun, "ARA*: Anytime A* with provable bounds on sub-optimality," in *Advances in Neural Information Processing Systems (NIPS)*, 2004.
- [6] L. Ma, J. Xue, K. Kawabata, J. Zhu, C. Ma and N. Zhang, "A fast RRT algorithm for motion planning of autonomous road vehicles," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Qingdao, 2014.
- [7] Y. Kuwata, G. A. Fiore, J. Teo, E. Frazzoli and J. P. How, "Motion planning for urban driving using RRT," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, 2008.
- [8] J. Wei, J. M. Dolan and B. Litkouhi, "Autonomous vehicle social behavior for highway entrance ramp management," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, Gold Coast, QLD, 2013.
- [9] W. Xu, J. Wei, J. M. Dolan, H. Zhao and H. Zha, "A real-time motion planner with trajectory optimization for autonomous vehicles," in *2012 IEEE International Conference on Robotics and Automation*, Saint Paul, MN, 2012.
- [10] F. You, R. Zhang, G. Lie, H. Wang, H. Wei and J. Xu, "Trajectory planning and tracking control for autonomous lane change maneuver based on the cooperative

- vehicle infrastructure system," *Expert Systems with Applications*, vol. 42, no. 14, pp. 5932-5946, 2018.
- [11] J. Chen, P. Zhao, T. Mei and H. Liang, "Lane change path planning based on piecewise Bezier curve for autonomous vehicle," in *Proceedings of 2013 IEEE International Conference on Vehicular Electronics and Safety*, Dongguan, 2013.
- [12] Q. Zhang, D. Filev, H. E. Tseng, S. Szwabowski and R. Langari, "A Game Theoretic Four-Stage Model Predictive Controller for Highway Driving," in *2019 American Control Conference (ACC)*, Philadelphia, PA, USA, 2019.
- [13] J. Nilsson, J. Fredriksson and E. Coelingh, "Trajectory planning with miscellaneous safety critical zones," *International Federation of Automatic Control*, vol. 50, no. 1, pp. 9083-9088, 2017.
- [14] W. Xu, J. Pan, J. Wei and J. M. Dolan, "Motion planning under uncertainty for on-road autonomous driving," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, 2014.
- [15] W. Schwarting, A. Pierson, J. Alonso-Mora, S. Karaman and D. Rus, "Social behavior for autonomous vehicles," in *Proceedings of the National Academy of Sciences*, 2019.
- [16] Y. Liu, X. Wang, L. Li, S. Cheng and Z. Chen, "A Novel Lane Change Decision-Making Model of Autonomous Vehicle Based on Support Vector Machine," *IEEE Access*, vol. 7, pp. 26543-26550, 2019.
- [17] C.-W. Hsu, C.-C. Chang and C.-J. Lin, "A Practical Guide to Support Vector Classification," 2003. [Online]. Available: <https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>. [Accessed 25 Aug. 2020].
- [18] "Next Generation Simulation (NGSIM) Vehicle Trajectories and Supporting Data," U.S. Department of Transportation Federal Highway Administration, 2016. [Online]. Available: <http://doi.org/10.21949/1504477>. [Accessed 25 Aug. 2020].
- [19] P. Hidas, "Modelling vehicle interactions in microscopic simulation of merging and weaving.," *Transportation Research Part C: Emerging Technologies*, vol. 13, no. 1, pp. 37-62, 2005.
- [20] R. Rajamani, *Vehicle dynamics and control*, Boston, MA, USA: Springer, 2012.

- [21] S. Hima, S. Glaser, A. Chaibet and B. Vanholme, "Controller design for trajectory tracking of autonomous passenger vehicles," in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Washington, DC, 2011.
- [22] S. Zhou, Y. Wang, M. Zheng and M. Tomizuka, "A hierarchical planning and control framework for structured highway driving," *International Federation of Automatic Control*, vol. 50, no. 1, pp. 9101-9107, 2017.
- [23] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng and D. Hrovat, "Predictive Active Steering Control for Autonomous Vehicle Systems," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 566-580, 2007.
- [24] A. Rupp and M. Stolz, *Survey on Control Schemes for Automated Driving on Highways*, Springer, 2016.
- [25] E. F. Camacho and C. B. Alba, *Model predictive control*, Media: Springer Science & Business, 2013.
- [26] J. Nilsson, Y. Gao, A. Carvalho and F. Borrelli, "Manoeuvre generation and control for automated highway driving," in *International Federation of Automatic Control Proceedings*, Cape Town, 2014.
- [27] A. Kesting, M. Treiber and D. Helbing, "Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity," *Philosophical Transactions of the Royal Society A*, vol. 368, no. 1928, pp. 4585-4605, 2010.