Centralized nonlinear model predictive control for highway traffic throughput maximization

Case study of on-ramp merging

Omer Khalid

Master of Science Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control

Centralized nonlinear model predictive control for highway traffic throughput maximization

Case study of on-ramp merging

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Omer Khalid

October 12, 2020

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Abstract

Traffic congestion on highways is a multi-sectoral phenomenon affecting society, the economy and the environment. It often takes place at specific locations such as on and off-ramps, weaving segments and intersections. The on-ramp merging procedure is considered as one of the main factors that causes traffic congestion on highways. The studies in literature show that the merging procedure can result in adverse traffic scenarios such as the buildup of the vehicles on the ramp which causes a downstream drop in capacity and subsequent blockage of upstream off-ramp traffic flow. Moreover, the on-ramp vehicles need to take the actions of leading and following mainlane vehicles into account during the merging process. On highly congested roads, this merging process becomes even more tedious and undesirable stop-and-go traffic behavior becomes unavoidable.

Connected and Autonomous Vehicles (CAVs) that can provide safe gaps between vehicles along with identifying appropriate merging speed profiles have the potential to reduce traffic accidents and improve traffic efficiency. This thesis introduces a Nonlinear Model Predictive Control (NMPC) based control strategy for autonomous merging control based on a cost function that tracks the desired inter-vehicular gaps for on-ramp and mainlane vehicles, and thus intends to fully exploit the capacity of the road in order to maximize the traffic throughput. The proposed controller aims to optimize both acceleration and steering rate profiles of vehicles, and to guide on-ramp vehicles to merge efficiently, without frequent slowdown or wait for merging gaps at the end of the ramp along with minimal disruption to the mainlane traffic flow. The controller is evaluated under different initial conditions, ranging from low to high traffic conditions. The performance of the controller is compared to that of a baseline scenario, and the results show that the proposed controller increases travel times in the range of 2.46% and 4.17% for different traffic conditions, without disrupting the mainline traffic operation. Additionally, average speed of vehicles is improved in the range of 8.2% and 4.5% under different traffic conditions.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank my supervisors dr.ir. Erik Steur and dr.ir. Meng Wang for their continuous support and assistance during my research. Thank you for guiding me through the thesis and helping me in every possible way, wherever I got stuck during the whole phase of the research work. This thesis would definitely not be of the same quality without your professional input, constructive feedback, and critical remarks. Also, I am thankful for your support during some difficult times where you showed great understanding and gave me the space I needed, which I really appreciate.

I also want to acknowledge the support of Delft University of Technology for the several opportunities I was given and for having accompanied me in my own personal growth. I am grateful to my friends and colleagues from the Delft Center for Systems and Control (DCSC) department with whom I spent valuable time, and had many fruitful interactions and discussions during the whole MSc program.

Lastly, I would like to thank my parents for supporting me and encouraging me throughout the course of my MSc study. It would not have been possible to complete this arduous, at times unassailable yet rewarding journey without your consistent support.

Delft, University of Technology
October 12, 2020

Omer Khalid

"What is the conclusion?" they said and asked,
It was said, "Returning to the beginning at last."
Mahmoud Shabestari
(translated from Persian)

Chapter 1

Introduction

1-1 Motivation

Traffic congestion on highways is a multi-sectoral phenomenon affecting society, the economy and the environment. It often takes place at specific locations such as on and off-ramps, weaving segments and intersections. The on-ramp merging procedure is considered as one of the main factors that causes traffic congestion on highways [2]. It has been identified in the relevant literature that the merging procedure can result in adverse scenarios such as the buildup of the vehicles on the ramp which causes a downstream drop in capacity and subsequent blockage of upstream off-ramp traffic flow [2], [15]. Speed variance at ramps due to different types of merging vehicles and driving behaviors (cooperative or adversarial) is another factor which adversely impacts the occurrence of crashes, termed as Crash Occurrence Likelihood (COL) [4], [22]. Moreover, a local perturbation such as a lane change can result in congestion which ultimately influences the overall traffic flow stability [5], [16].

Connected and Autonomous Vehicles (CAVs) that can provide safe gaps between vehicles along with identifying appropriate merging speeds have the potential to reduce traffic accidents and improve traffic efficiency [6], [7]. It has been shown that the merge bottleneck capacity increases quadratically as Cooperative Adaptive Cruise Control (CACC) market penetration becomes higher, and CACC vehicles can help to maintain a stable traffic flow free of stop-and-go waves [12]. Vehicle-to-Vehicle (V2V) based communication during merging can result in decreased travel times as compared to baseline case of manually driven vehicles [13]. To this end, a number of automobile companies and research institutions are dedicating their efforts to demonstrate and implement partially or highly automated features in modern vehicles such as the artificial intelligence based computational platforms for autonomous driving from NVIDIA [9] and the Autopilot from Tesla [10]. In addition to these, fully autonomous vehicles are also in advanced stages of development. In the context of this thesis, the use of the term 'autonomous' has been restricted to automation Level 4 or above as defined by the Society of Automotive Engineers (SAE) [11]. Level 4 pertains to 'high automation' where the human intervention is not required sans in limited circumstances. Level 5 automation does not require human intervention at all.

Research on the management and avoidance of congestion at on-ramps has been carried out since the development of motorway roads. The research work in this regard, for ease of understanding, can be described chronologically. In the **first** phase, most of the efforts to decrease and prevent congestion at merges were related to the physical improvement of the infrastructure layout. This included improvement in the road layout and identification of the optimal junction design [17], [18], [19]. This physical road network enhancement is expensive to undertake and often not desirable. In the **second** phase, Active Traffic Management (ATM) strategies were introduced to control the traffic flow in an efficient way. ATM employs technologies for real-time monitoring and management of traffic to respond to the continuously changing traffic conditions. These primarily included ramp metering, Variable Speed Limit (VSL) and route guidance [20], [21]:

- **Ramp metering** determines the rate of flow at which the vehicles enter the highway which is implemented by the use of traffic signals at the on-ramps.
- **Variable speed limit** is used to manage the traffic speeds on highways. Speed limit signs are displayed to mange the upstream vehicle speeds in order to mitigate the occurrence of congestion on merging areas.
- **Route guidance** involves the use of on-board devices that display information about the state of traffic such as waiting time or queue lengths and hence, assist drivers in planning their routes accordingly.

In the **third** and most recent phase, innovative control strategies based on Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication are being used in combination with other driving assistance systems to better manage the merging process. With the introduction of Connected and Autonomous Vehicles (CAVs) on the roads and the anticipated arrival of self-driving vehicles, the importance of V2V and V2I based strategies has grown manifold and it necessitates further research on the topic [2], [3].

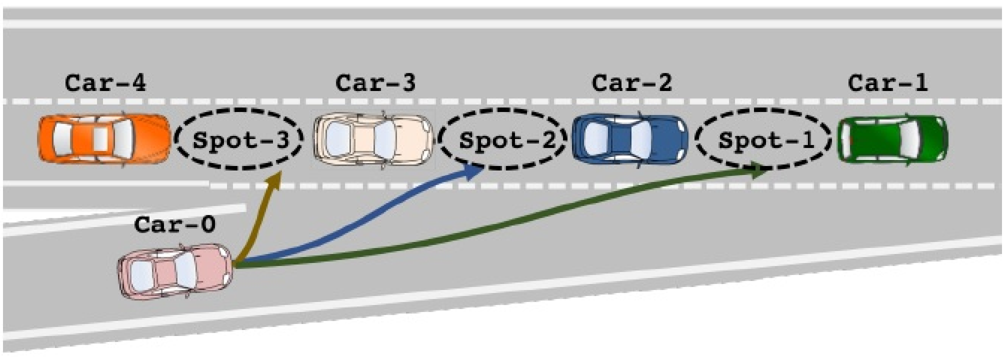


Figure 1-1: A typical highway on-ramp merging scenario [117].

A typical freeway merging situation is shown in Figure 1-1. Car-0 is the ego vehicle intending to merge from the on-ramp into the mainlane on the highway. There are three spot candidates where the it can be merged i.e. Spot-1, 2 and 3. The ego vehicle needs to determine a spot from these set of the candidates while performing the merging process in a safe and efficient manner. At first, the driver determines which spot is the best one based on certain criteria e.g. the safest, the first available and with a comfortable driving profile. The comfort pertains to the driver not being subjected to aggressive acceleration or braking commands. Once the spot is selected, the driver adjusts accordingly the speed of the vehicle to align with the targeted spots and mainlane vehicle speed; and, then performs the required maneuvers to merge. Moreover, the vehicle on the mainlane can either give way to the ego vehicle or decline and hence, the ego vehicle needs to take the actions of leading and following vehicles into account during the merging process. On highly congested roads, this merging process becomes even more tedious and undesirable stop-and-go traffic behavior becomes unavoidable.

The implementation of autonomous on-ramp merging presents many challenges. One challenge is that the merging vehicle should take the *long-term effects* into account while undertaking *current actions*. These *long term* effects pertain to the successful completion of smooth merging process (e.g. identifying gap, maintain merging speed and minimize distortion of mainlane traffic flow) while the *current actions* include acceleration, deceleration or steering commands. This also shows that the ramp merging is a time-sequential process as the completion of the process depends on a sequence of current actions which have an impact on the whole process. Another challenge is that the merging maneuver not only depends on the ego vehicle's own dynamic state but also depends on the surrounding vehicles' actions. The surrounding vehicles can either cooperate with the merging vehicle by giving way to them or take adversarial actions such as speeding up to maintain their own speed profile which have wider implications for the traffic throughput from the road segment. Moreover, the merging process usually includes two modules, a decision making module and a control execution module. The former works on a strategic or tactical level and advises to make a lane change based on either a planned route (e.g. making a detour to reach a specific destination) or a specific driving condition (e.g. overtaking a slow vehicle to maintain speed profile). The control execution module issues the necessary longitudinal and lateral commands to execute a safe and smooth lane change process.

1-2 Problem statement and scope

A literature study [1] has been done to gain a holistic understanding of the ramp merging problem and identify suitable research gaps for this research. One observation was that the focus of research in this domain has been gradually shifting from rule-based methods to optimization and control based approaches. The latter have advantages in terms of devising optimal solutions and achieving scalability, although the computational costs remain a challenge. Recent innovations in autonomous driving have offered the use of Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication capabilities to augment the design of control strategies, the feasibility of which needs to be explored further in complicated traffic networks such as ramp merging. It was also inferred that most of the previous works consider a single mainlane during merging and a rather simplified model, such as point mass is used to study the proof-of-concept approaches [2]. As shown in Figure 1-1, such a scenario includes a single mainlane with an on-ramp and varied number of vehicles. Furthermore, previous works generally focus on the design of control algorithm and its assessment while wider effects on the traffic performance and the subsequent evaluation of their metrics are not explicitly considered such as merging delays and upstream or downstream congestion.

In recent years, Model Predictive Control (MPC) based control strategies have seen a significant increase in usage for numerous new applications, particularly in the automotive industry [121]. The recent spike in interest to use MPC for new applications can be ascribed to the ever increasing processing power in modern day computing systems. This allows to solve complex optimization problems online, that were not deemed to be solvable before. In this thesis work, the applicability of the model predictive control and incorporation of lane changing in a multi-lane setting is researched in application to the throughput maximization of traffic on highways.

The road segment studied in the thesis is illustrated in Figure 1-2. It is pertinent to mention here that previous works in this domain have mainly focused on scenarios with one mainlane and an on-ramp [2], [3]. Moreover, if multi-lane scenario is considered, then a restriction on the lane changes within the mainlanes is placed [8]. In order to address these research gaps, and to portray a more realistic merging traffic scenario, two mainlanes, lane 1 and 2, and one on-ramp are considered. The number of vehicles (N) in the road segment are varied for different scenarios which are discussed later in the thesis. In a Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V) connected vehicle environment, highway vehicles inside the *Control Zone* are informed of the traffic conditions by the *Centralized Control Agent*. Based on the generated control commands, the vehicles are tasked to slow down, accelerate, or shift to the left lane to allow vehicles from the ramp to join the highway with the overall objective of maximizing the throughput from the entire road segment. The location where the ramp and the freeway connect is called the *merging point*. The *Merging Zone* is defined as a segment of the right-most lane of the highway where the vehicles from the on-ramp merge onto the mainlane. This is also the region where there is a potential lateral collision of the vehicles, hence the distinction with regards to the *Control Zone*. The *Acceleration Region* allows the vehicles on the ramp to accelerate/decelerate prior to merging onto the lane 2. In order to induce generalization into the problem scenario, the initial velocities and the entry time of vehicles in each lane are chosen subjectively and no limits are placed on the occurrence of lane changing in either *control* or *merging* zones.

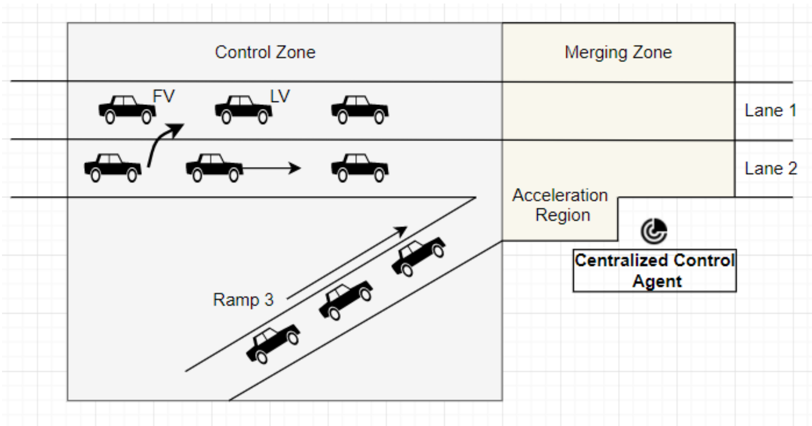


Figure 1-2: On-ramp merging road segment.

The implications of this approach are that vehicles do not have to come to a full stop at the merging point, thereby conserving momentum while improving throughput and travel time. The centralized controller is designed in way that controls the vehicles in the entire road segment i.e. the control and the merging zone. Moreover, by optimizing each vehicle's acceleration/deceleration, the transient engine operation can be better managed which can result in additional benefits such as reduced energy consumption. To this end, the thesis addresses this line of research and the overarching **objective** of the thesis is:

To design a centralized nonlinear model predictive control strategy for highway traffic throughput maximization.

The proposed controller would be able to:

- make use of a nonlinear vehicle prediction model to capture the system dynamics during merging of vehicles.
- coordinate the vehicles inside the control and merging zone so as to maximize the traffic throughput from the road segment.
- incorporate both longitudinal (acceleration command) and lateral control (steering rate command) for each vehicle.
- maintain safe inter-vehicular distances and attain safe trajectories for merging vehicles.

Scope

In Figure 1-3, the complete control environment for Connected and Autonomous Vehicles (CAVs) is presented and the focus of this thesis is highlighted with the blue block. It is assumed that the *Supervisory Control* has access to pre-processed data and is able to receive messages from the other vehicles e.g. via GPS data and Vehicle-to-Infrastructure (V2I) communication. The supervisory layer gives the *Centralized Control Agent* information of when to perform the different tasks.

The focus of this thesis will exclusively be on this *Centralized Control Agent* which generates the acceleration and steering rate commands for each vehicle, which are then transformed into respective actuator signals by the *Low-level Control*. For instance, it asks a specific vehicle to make a lane change while making sure that the safety distances to the other vehicles are not violated during the merging.

1-3 Major contributions

This thesis aims to include the following major contributions to the field:

- Incorporation of nonlinear vehicle kinematic prediction model for the MPC based control strategy.
- Formulation of a multi-lane highway setting with on-ramp merging.
- Integration of both longitudinal and lateral control of vehicles into the proposed control strategy.

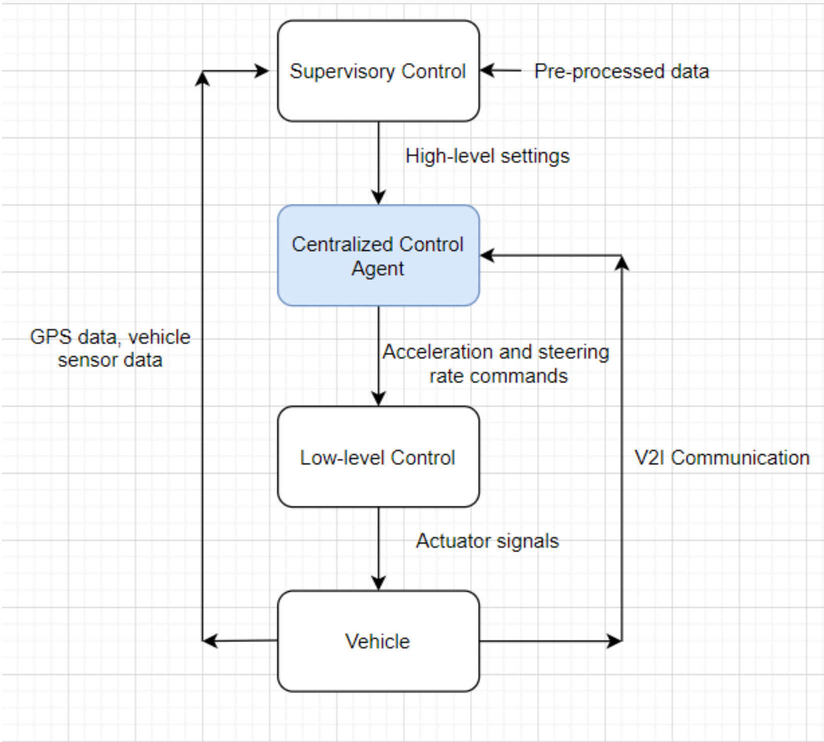


Figure 1-3: The complete control system environment for CAVs.

- Analysis of the possibilities and limitations of Nonlinear Model Predictive Control (NMPC) based strategy for different traffic scenarios, with a focus on comparison with baseline and state-of-the-art works.

- Evaluation of traffic performance metrics e.g. travel times and average speed, in addition to the controller assessment.

1-4 Thesis outline

Following the introduction as presented in this chapter, the remainder part of the thesis is structured as follows:

Chapter 2: Background information. This chapter gives the background information on the topic. The traffic modelling and the formulated ramp-merging road segment are presented. This is followed by the description of the vehicle model along with the respective model assumptions. Next to this, an overview of control strategies is presented which is followed by the motivation for the choice of MPC.

Chapter 3: Control design. This chapter introduces the control approach in detail. At first, an introduction to MPC is presented and the motivation for its use is discussed. This is followed by the formulation of objective function, constraints and further design choices. Lastly, the lane change methodology of vehicles is presented.

Chapter 4: Simulation results. In this chapter, the implementation aspects of the controller are discussed. The case studies of baseline scenario, low traffic, and high traffic scenario are presented. This is followed by the simulation results and discussion.

Chapter 5: Conclusion. In this chapter, the conclusions from the entire thesis work are summarized. Also, some recommendations and research directions for the future work in this domain are presented.

Chapter 2

Background information

In this chapter, background information regarding the formulation of ramp merging problem is presented. Section 2-1 starts with the description of the way the problem is formulated vis-a-vis traffic and the road segment. In Section 2-2-1, the governing equations for the kinematic single-track model are presented which form the fundamental basis for the proposed Model Predictive Control (MPC) method. Section 2-3 summarizes the findings from the literature survey conducted with regards to the different control strategies for the merging problem while the motivation for the choice of Model Predictive Control (MPC) is presented in Section 2-4.

2-1 Modeling of the traffic

Traffic models are, in general, divided into two distinct classes based on the behavior that they capture and the level of details that they provide; macroscopic, mesoscopic, and microscopic models. In **macroscopic** modeling, the traffic is described in aggregate terms based on average speed, flow and density of the highway sections instead of capturing the behavior of individual vehicles. The three important variables of macroscopic modeling are flow, density and speed of vehicles in a certain road segment. **Mesoscopic** models are another category of traffic models which partly use the characteristics of both microscopic and macroscopic models. The level of detail for a mesoscopic model is greater than a macroscopic model and less than that of a microscopic model [39]. The traffic flow dynamics are described in an aggregate way based on probability distribution functions. Three main classes of mesoscopic models include headway distribution models, cluster models, and gas-kinetic models. **Microscopic** modeling, on the other hand, is used to describe the behavior of vehicles at an individual level. The important aspects of microscopic models are longitudinal driving behavior (car following) and lateral driving behavior (lane changing). Car following and lane changing behavior is generally described as a function of the distance to and relative speed of the surrounding vehicles, and the desired speed. As the vehicles are modeled individually in microscopic traffic models, different characteristics can be assigned to each vehicle. These characteristics are related to the individual driving style of the driver, type of vehicle (e.g. car, truck), its destination, and route choice.

As the focus of the thesis will be to investigate the merging of vehicles at an individual level along with studying and incorporating the notions of trajectory generation and lane changing, so the microscopic traffic modeling is considered to be suitable for this purpose. In this way, each vehicle in the network can be characterized by parameters such as, the desired velocity, acceleration/deceleration, vehicle type, driving behaviour etc. The formulation of the road segment considered in this thesis is shown in Figure 2-1.

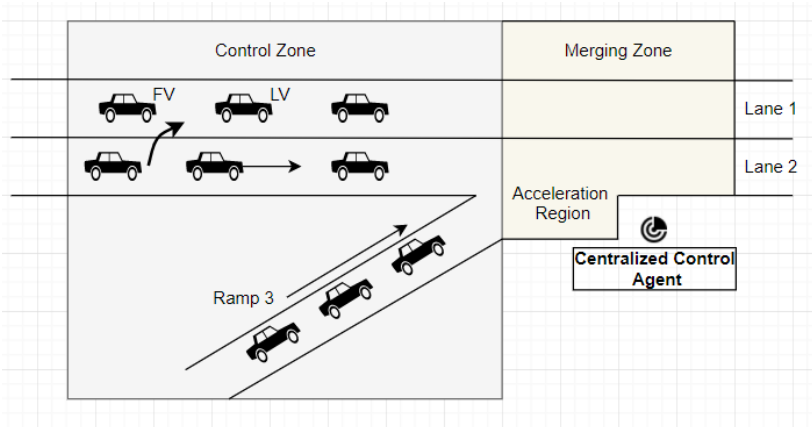


Figure 2-1: On-ramp merging road segment.

In this thesis, the proposed controller is designed with an aim to maximize the traffic throughput from the entire road segment. Throughput pertains to the overall travel efficiency, one way of measuring which is to calculate the total travel time that the vehicles spend inside the control and merging zones. A large travel time value means that the controller is conservative and might take more time than desired time to complete the merging task, resulting in lower traffic efficiency. Conversely, a smaller value could mean that the controller is assertive and makes proactive decisions to achieve higher traffic efficiency. Another criteria is to analyse the average speed of vehicles during the course of controller simulation. In particular, this gives an indication about the merging behavior of on-ramp vehicles and whether they have to slow down too much to merge into the lane 2, which could in turn, induce merging delays.

2-2 Vehicle models overview

The vehicle model plays a pivotal role in two aspects of the controller design. The first aspect is the future states prediction, which can be based on the mathematical representation of the vehicle model. The second aspect is the system simulation, where the vehicle model is used to simulate the vehicle behavior and the performance of the proposed control strategy is assessed. In general, three kinds of vehicle traffic models are presented in literature.

Point mass models treat the individual vehicle as a particle and describe the motion of the vehicle along a specified path in a simplified way. In these models, the vehicle is considered to be an object with a finite mass and the vehicle motion is described as a function of time only based on parameters such as position, velocity and acceleration. **Multi-body models** consider the vertical load of all 4 wheels due to roll, pitch, and yaw, along with their individual spin and slip, and nonlinear tire dynamics. Different tire models are used to describe the interaction between vehicle and the road. These models are able to include suspension forces while also capturing the vehicle dynamics in high tire slip conditions [44]. **Kinematic models** are based on the geometry of the vehicle, i.e. using differential drive kinematics to model wheeled robots. They are useful for representing the vehicle motion in low and medium speed conditions which do not involve highly dynamic maneuvers and tire force saturation. In case of kinematic bicycle model, the vehicle is modeled with only two wheels, where the front and rear wheels are each lumped into one wheel. This simplification is justified when the roll dynamics need not to be considered. An example, where the benefit of a kinematic bicycle model is evident, is parking: a point-mass model is not sufficient since it would not consider the non-holonomic behavior and in particular, the minimum turning radius. A comparison of three different vehicle models is shown in Table 2-1.

Table 2-1: Brief comparison of vehicle models

Vehicle Models	**Level of detail**	**Scalability and Computational tractability**	**Behavioral representation**
Point mass models [47]	2nd order dynamics, vehicle motion is described as a function of time only based on parameters such as position, velocity and acceleration	appropriate for higher number of vehicles, in cooperative control approaches, lower computational requirements, better for optimization based approaches	treat the individual vehicle as a particle and describe the motion of the vehicle along a specified path in a simplified way
Kinematic vehicle model [44]	depend on geometric parameters of the vehicle	need higher computational requirements than point mass models, model is usually simplified to point mass for reducing computational burden of optimization	useful for low speed maneuvers, provides accurate representation of the nonholonomic motion of vehicle
Dynamic vehicle model [44], [43]	describe interaction between vehicle (wheel) and road so higher accuracy	Number of agents modelled and simulated can be extended to a higher number	longitudinal behaviour (car following models), lateral behavior (lane changing), complex dynamic behavior can be captured

2-2-1 Choice of kinematic single-track model

In this thesis, the basis for the prediction model and plant model for the proposed control strategy is the kinematics single-track vehicle model. This model of the vehicle dynamics uses the approximation where the left and right wheels are lumped into a single wheel. This also explains the term *single-track model*. This is a common simplification approach in developing models suitable for the control design [44]. An illustration of this model is shown in Figure 2-2. l_r and l_f are distance from center of gravity to rear axle and front axle, respectively. In total, the length of the wheelbase l_{wb}=l_r+l_f is taken to be 2.5m. The state equations of the vehicle model are as follows:

$$\dot{X} = f^{KM}(X, U), \tag{2-1}$$

where X and U are the state and input vectors respectively. The basic assumptions of the single-track model are as follows:

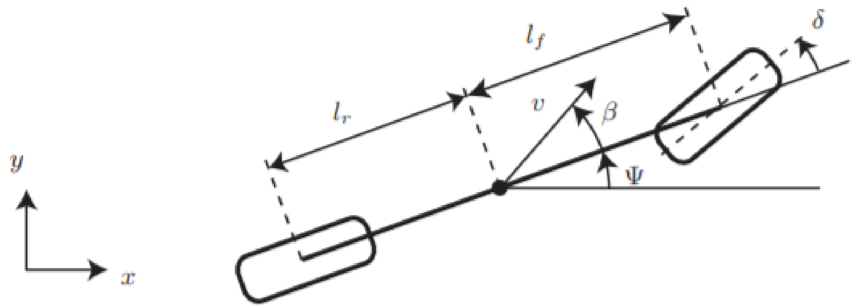


Figure 2-2: Kinematic single track model

- The wheels of the vehicle are assumed to be lumped into the center of each axle.
- The vehicle model is defined in a top-view 2D Euclidean space. 3D effects, such as roll-over are not modeled.
- Each unit of the vehicle is a rigid body mass.
- No suspension movements, chassis or cabin dynamics are modeled.
- The aerodynamic forces are negligible.

Kinematic models are based on the geometry of the vehicle. They are useful for representing the vehicle motion in low and medium speed conditions which do not involve highly dynamic maneuvers and tire force saturation. The differential equations describing the motion of the vehicle are as follows:

$$\dot{x} = v\ cos(\psi), \tag{2-2}$$
$$\dot{y} = v\ sin(\psi), \tag{2-3}$$
$$\dot{v} = a, \tag{2-4}$$
$$\dot{\delta} = v_\delta, \tag{2-5}$$
$$\dot{\psi} = \frac{v}{l_{wb}} tan(\delta), \tag{2-6}$$

where the notation ψ is the heading angle, δ is the steering angle and v_δ is the steering rate. v and a denote the velocity and acceleration of the vehicle's center of gravity, respectively. The velocity vector v is always aligned with the link between the front and rear wheel. Hence, the slip angle $\beta = tan^{-1}\left(\frac{l_r}{l_f+l_r} tan(\delta)\right)$ is assumed to be zero. As the scenario of on-ramp merging does not corresponds to high-speed maneuvers or sharp cornering, so the choice of neglecting the slip angle is acceptable in this case.

The state equations include 5 variables i.e. longitudinal position, lateral position, velocity, steering angle, and heading for each vehicle present in the road segment. These states or output variables are as follows:

$$X = \begin{bmatrix} x_i & y_i & v_i & \delta_i & \psi_i \end{bmatrix}^T.$$

The manipulated variables of acceleration and steering rate are as follows:

$$U = \begin{bmatrix} a_{long,i} & v_{\delta,i} \end{bmatrix}^T,$$

where the subscript i refers to each vehicle in the control and merging zone.

2-3 Overview of control

A literature study was conducted to study the state-of-the-art in terms of control strategies and their assessment for merging problems. Herein, the survey is summarized by highlighting the relevant works in literature along with their underpinnings and caveats. It is pertinent to mention here that the intersection control problem and the ramp merging control problem are very similar in nature and most of the approaches proposed for intersection control, can easily be adapted for merging coordination and vice versa. In the following subsections, the control approaches for intersection and ramp merging control are briefly presented. For the sake of brevity, the control methods are divided into heuristic, optimization-based and learning control approaches.

2-3-1 Heuristic and rule-based approaches

In heuristic and rule-based approaches, the general idea is that there is a centralized controller that coordinates the merging process based on scheduling requests and information received from the vehicles inside the merging zone. The solutions generated in these approaches are not guaranteed to be optimal [3]. Researchers have designed various approaches in this regard.

One approach is the reservation based method where a centralized controller or intersection manager is used to coordinate the merging schedule based on the requests and information received from the vehicles located inside the communication range. The merging area is divided into cells, or points, which are to be assigned, or reserved, for only one vehicle at each instant of time to avoid collisions. The main challenges in this approach are the heavy communication requirements and the possibility of deadlocks occurrence. The communication between merging vehicles becomes a critical issue particularly when they are required to communicate several times with the central controller until their reservation request is approved. Dresner *et al.* [52] proposed the use of the reservation scheme to control a single intersection of two roads with vehicles traveling with similar speed on a single direction on each road, i.e. no turns are allowed. Each vehicle is treated as a driving agent which requests the reservation of the cells to cross the intersection at a specific time interval. Once the centralized reservation system receives the request, it accepts if there is no conflict with the already accepted reservations. Otherwise, the request is rejected. In case of rejection, the driving agent is required to decelerate and send a new reservation request. This work was extended by Huang *et al.* [53] in which the the computation of the vehicle trajectories is centralized to minimize the possibility of reservation requests cancellation. Moreover, a a hierarchical processing of requests based on priority is designed and indicators relating to environmental benefits are assessed.

Uno *et al.* [72] presented the concept of virtual vehicle/platooning for efficient ramp merging. The concept of *virtual platooning* relates to shifting of the time instant for platoon forming

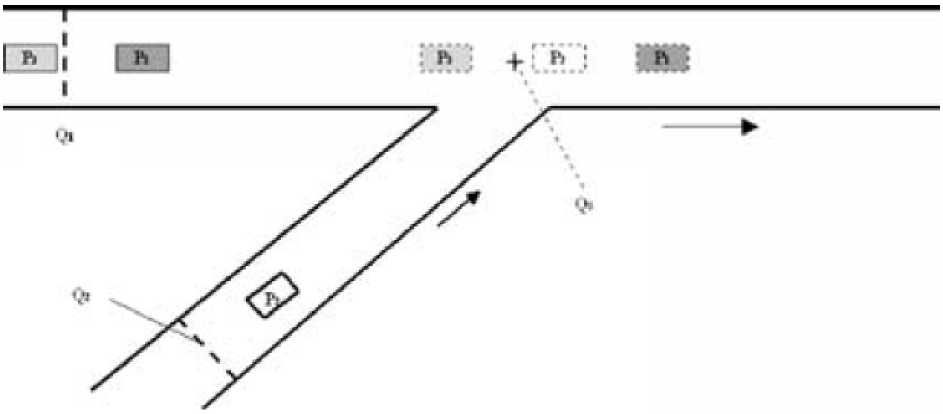


Figure 2-3: Virtual platooning merging mechanism [73].

forward before the real merging process takes place. A virtual vehicle is mapped onto the main road before the actual merging is supposed to occur in order to allow the vehicles to perform smoother and safer merging process. Figure 2-3 illustrates this mechanism. This concept is further explored by Lu *et al.* [73] where a platoon of vehicles is merged by considering it as a single vehicle. Furthermore, a merging algorithm using a "slot-based" approach for autonomous vehicles is proposed by Marinescu *et al.* (2012) [71]. In this case, each vehicle drives normally until the central Traffic Management System (TMS) detects that the traffic conditions require a more efficient use of the infrastructure. At this point, each vehicle is allocated to a virtual slot, and on-ramp vehicles are mapped into the empty slots on the mainline that are selected by the central controller called Road Side Unit (RSU). The slot's occupancy status is determined in two ways. The first is a hierarchical approach where the TMS maintains this information for all the slots on the motorway. Initially, all slots are empty. When a vehicle moves into a slot based on the VTS algorithm, it marks the slot as occupied. When a vehicle wants to change its slot, it asks the status of the target slot with TMS which assigns a target slot to the vehicle. The second method is a distributed approach. In this case, the vehicle coordinates the slot changing with its neighboring vehicle based on Vehicle-to-Vehicle (V2V) information. Microscopic simulation has been used to evaluate the algorithm performance, and results for medium and heavy traffic saturation show that this approach results in an increment in the on-ramp throughput and decrease in merging delay.

2-3-2 Optimization and control based approaches

In literature, many optimization and control based approaches have been utilized to study the merging of vehicles on ramps and intersections. The optimization based approaches involve the formulation of a cost function with a certain objective to achieve. This objective is, for instance, in terms of optimizing total ravel time, minimizing vehicles overlap or multi-objective problem formulation. Additional terms are added into the cost function to include safety constraints such as collision avoidance. One approach to generate trajectories for merging is by using Model Predictive Control (MPC), also known as receding horizon control. The main aim of MPC is to minimize a performance criterion in the future that is subjected to constraints on the manipulated inputs and outputs, where the future behavior is computed according to a model of the plant. The model predictive controller uses the system model and current plant measurements to calculate future direction in the independent variables that will result in an operation that honors all the independent and dependent variable constraints. The MPC then sends this set of independent variables to the corresponding regulatory controller set-points to be implemented in the process.

A proactive merging strategy for two streams of vehicles is proposed by Awal *et al.* [83]. The objective is to minimize the merging time on on-ramps and hence reduce the traffic bottlenecks. This is done by a recursive pruning algorithm that generates the optimal merging order based on the current traffic condition so that the on-ramp vehicles can merge to the main lane minimizing the merging delay. The vehicles are V2V enabled and the merging vehicle obtains the position, velocity and acceleration information of the surrounding vehicles. The results show improvement in merging time, energy consumption and average speed while the average travel time on the main highway was slightly increased [83]. One way of reducing the traffic congestion is by increasing the throughput at an intersection or the ramp merging area. This can be achieved through the optimization of the travel time for all the vehicles located inside the control zone. Raravi *et al.* [76] formulated the autonomous merging problem as an optimization problem with the objective to minimize the driving time to intersection. The literature such as Li *et al.* [61], Yan *et al.* [62] and Wu *et al.* [63] also involves optimization of travel time.

Another approach used in optimization is to minimize the overlap of vehicles inside the intersection zone. The objective is to determine the acceleration profiles of the vehicles such that only a limited number of vehicles are present inside the intersection at each instant of time. The total number of vehicles depends on different criteria such as the size of the vehicles, the length of the intersection area and the minimum safest following distance. Lee *et al.* [64] proposed this approach in which the potential overlaps of vehicle trajectories were eliminated. The simulation results showed that this approach only reduces the total travel time and delays but also minimizes the total fuel consumption during the procedure. This work was further extended in [65] by considering the case of an urban corridor and studying the safety and environmental aspects of the system.

Multi-objective optimization is based on including multiple performance criteria in the objective function. The assumption here is that the vehicles have already been assigned the merging schedule and the aim is to minimize the error between the actual and desired control commands i.e. speed or acceleration. The objective function can be formulated as follows:

$$min \sum_{t=1}^{T} \left(w^v \left(v_i(t) - v^d \right)^2 + w^u u_i^2(t) + w^c f_i^2 \left(t, u_i(t) \right) \right), \tag{2-7}$$

where T is the total number of horizon steps. Weighting factors are denoted by w and the superscripts v, u correspond to speed and acceleration respectively. $f(t, u)$ is an additional function that can be used to quantify the risk of collisions in the system. In general, the constraints found in literature include minimum headway (distance and/or time) between vehicles, speed and acceleration limits [3].

Campos *et al.* [66] formulated the problem in which speed tracking error and acceleration is incorporated in the objective function. The constraints include velocity and acceleration values that would make the crossing procedure safe. Model Predictive Control (MPC) is used in [67] to solve the problem that includes a risk factor function to quantify the collision risk at the intersection. In [68], the error between the desired speed and the actual speed of each vehicle is minimized while keeping a safe time headway between the vehicles crossing the intersection. Moreover, smaller acceleration values are assigned to heavier vehicles that those assigned to the lighter vehicles. This helps to make the trajectory of heavier vehicles smooth which in turn, improves the energy consumption of vehicles. This approach was compared with a intersection controlled by traffic lights. The results showed marked improvement in the safety and energy consumption aspects as compared to the latter approach.

A gap selection and path generation method is proposed by Cao *et al.* [69] where the vehicle on the main lane adjusts its speed to facilitate the merging process of the vehicle attempting to merge. The optimization problem is formulated by including four terms. These include: a) relative distances between the relevant vehicles b) bounds on the available moving area for the merging vehicle c) the desired traveling speed of vehicle and, d) bounds on the acceleration vehicles. The simulation results showed that for the specified initial conditions, the proposed method can generate trajectories for the vehicles during merging. Another multi-objective optimization problem is formulated by Rios-Torres *et al.* (2016) [77] and a closed form solution using Hamiltonian analysis is presented. The distinction of the work lies in the formulation of vehicle fuel consumption as a function of speed and control input. The evaluated solution results in decreased fuel consumption and total travel time as compared to baseline cases. The work of Chen *et al.* [87] presents a cooperative merging control strategy based on MPC which determines the optimal merging time along with the acceleration of the involved vehicles. Human-like behavior is incorporated by making use of lane change path model from [88]. This results in on-ramp vehicles accepting smaller gaps when approaching the end of the acceleration/merging lane. Simulation results show the feasibility of the designed controller for completing the merging process automatically and safely, and the capability to accept small gaps for merging.

2-3-3 Learning control methods

More recently, learning based techniques such as reinforcement learning has been explored for the control of traffic and autonomous vehicles. The goal is to find an optimal driving policy by maximizing a specific long-term reward. Reinforcement learning is a kind of machine learning algorithm which trains itself continually through trials and errors [104]. It has the potential to allow the vehicle agent to learn how to drive under different or previously unpredictable situations by training it to build up its pattern recognition capabilities. Reinforcement learning is different from standard supervised learning techniques, which need ground truth as input and output pairs. A reinforcement learning agent learns from past experience and then tries to capture the best possible knowledge to find an optimal action given its current state, with the goal of maximizing a long-term reward which is a cumulative effect of the current action on future states. An illustration of the reinforcement learning procedure is shown in Figure 2-4.

The system state is denoted by S_t, the control action is denoted by A_t while the goal is to learn the optimal policy π which is defined as $A_t = \pi_t(S_t)$. The interaction of the agent with the environment during the training process is as follows:

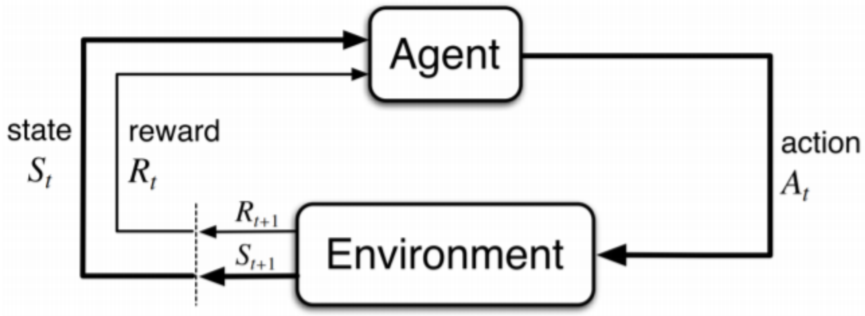


Figure 2-4: Reinforcement learning procedure. Image from Sutton [104].

- observe the system state S_t
- take action A_t
- observe the system new state S_{t+1}
- get the immediate reward from the interaction R_{t+1}

Reward is the feedback that the agent receives from its interaction with the environment for its immediate performance. It is a scalar function of the system state and control action. The sum of all the immediate rewards over the course of a training episode is defined as return. The agent learns a control policy by optimizing the reward obtained from the interactions with the environment.

An RL based framework is proposed by Ngai *et al.* [105] to study the problem of automated overtaking of vehicles. A quantization method is used to convert continuous state and action space into discrete spaces. The vehicle manages to complete the overtaking task although it cannot always turn to the desired direction due to the use of discrete steering angles. In literature, relatively fewer work has been done on continuous action space. Sallab *et al.* [107] formulated an RL algorithm for lane keeping assistance and compared two action spaces: discrete and continuous. The evaluation was done in TORCS simulator and it was shown that discrete action space resulted in abrupt steering commands while continuous action space gave better performance with smooth control. The use of reinforcement learning for lane change maneuvers is explored by Mukadam *et al.* (2017) [112]. The authors state that learning a full policy is relatively difficult as it includes learning tactical decisions as well as low-level control and collision avoidance. This is due to increase in complexity of the training network.

As part of Berkeley DeepDrive project, an autonomous ramp merge maneuver approach based on deep reinforcement learning is presented by Wang *et al.* in [113], [114]. The action space includes only the longitudinal acceleration while Q-function approximation is used to calculate the Q-values. The cumulative reward curves for the trained policy are evaluated which shows promising results for further investigation into the topic. The wider effects on the traffic network are not investigated. This work has been further extended to lane change maneuvers in [115] where both the state space and action space is treated to be continuous. A quadratic function is designed to be used as the Q-function approximator which leads to low computational complexity. Extensive simulations are done which show the convergence of loss and reward functions. A hierarchical reinforcement learning architecture is presented in [116], which decomposes the autonomous lane change procedure into decision layer and execution layer. Deep Q network is used to train the both layers separately and promising convergence results are shown. The authors note that further work is required to assess the viability of the approach by using other control decisions (e.g. overtaking), actor-critic methods and incorporating different road geometries.

Although, reinforcement learning based approaches are being applied to many control-related tasks with reasonable success but they also have drawbacks that need to be taken into account. A few of these are described below:

- Learning based approaches can be sample inefficient. In many cases, a reinforcement learning algorithm might need a very large amount of data to learn a policy. This is particularly important for large-scale and complex problems. This results in not only longer training time but also requires high processing power (GPUs) which is normally difficult to obtain for most of the researchers [118]. Moreover, as a consequence of sample inefficiency, many interactions with the environment are needed which can have an effect for the real-world applications. For example, a robotic manipulator would naturally suffer from wear and tear and for which maintenance costs would not justify the benefits of a learning controller [119].

- The formulation of reward function is a critical factor that determines the success of learning a desired policy. A misspecified reward function can add bias to the learning behaviour. As reported in literature [120], it is difficult to accurately formulate the reward function as per the desired behavioural requirements. A reward defined too loosely can improve the learning process time but can also result in learning an altogether different behaviour.

- One drawback regarding reinforcement learning based approaches is the *exploration vs exploitation* dilemma [104]. In control applications, the RL algorithm attempts to solve a nonlinear optimization problem by finding a policy that maximizes the return. As with all nonlinear optimization problems, there are often many different local optima, and the RL agent might get stuck in one of those points. Therefore, the agent needs to be able to *explore* its action space, while still *exploiting* what it has learned so far. On the other hand, if the exploration is increased, the agent might not be able to converge to a stable policy.

- Another drawback pertains to hyperparameter tuning as RL algorithms are known to be sensitive to changes in hyper-parameters (e.g. number of hidden layers, learning rate, discount factor) and other design choices such as topology of the designed neural network. The tuning of these parameters is not a trivial task, as it requires a lot of iterations and results in high computational effort.

2-4 Motivation for MPC

To summarize, in case of heuristic or rule based methods, the main challenges are associated with the heavy communication requirements and the scalability of the approach. The communication becomes a critical issue, particularly when vehicles are required to communicate several times with the central controller until their respective request is approved. These methods are conceptually comprehensible and rather easy to implement but are practically vulnerable due to their inability to adapt to unforeseen and complex traffic situations in the real world.

Predictive control methods such as MPC have become an attractive solution for automotive applications [121]. One reason is that MPC allows to include the state and actuator constraints along with incorporating complex models which gives it an edge over other methods [122]. As compared to reinforcement learning, less tuning effort is required which is mostly related to setting the weights in the specified cost function based on design requirements. Tuning the objective function provides a rather straight-forward and intuitive way of achieving the desired performance. MPC can also take into account model errors. The solution is calculated for each time step and when an external disturbance is added, then it is taken into account in the control input that is calculated for the next time step. Moreover, the Nonlinear Model Predictive Control (NMPC) has the same characteristics as the conventional linear MPC but the prediction model, objective function and the constraints can be nonlinear functions. This broadens a lot the possibilities by using a more realistic model of the system that accurately captures the underlying system dynamics, although this requires the use of non-convex optimization techniques to arrive at the optimal solution.

The use of Receding Horizon Control (RHC) in automotive control is very attractive for several reasons. Firstly, the cost function is continuously updated with the latest sensor measurements such as the vehicle configuration, road parameters, and obstacle information. This takes into account the limited range of the sensors. Moreover, updating and solving the problem incrementally introduces a feedback mechanism, offering the possibility to continuously adjust the control commands to account for any sudden changes in the surrounding environment. Lastly, the method based on receding horizon is similar to how humans drive in real world traffic situations [51]. MPC based controllers are also akin to adaptive controllers. This is because the notion of receding horizon accommodates for the rejection of disturbances. Adaptability is a desirable property in order to deal with various traffic conditions in the vicinity of the vehicle. Furthermore, an adaptive controller can also compensate for speed variations, inaccurate vehicle parameters, and other unforeseen circumstances. Hence, the RHC principle also provides MPC with some robustness.

A major drawback of MPC is its high computational complexity. Detailed system dynamics and often, a nonlinear optimization problem over an entire horizon need to be taken into account. But due to ever-increasing automation within vehicles and the inevitable arrival of self-driving cars, processors within vehicles are becoming more powerful with each iteration. This concurrent advance in the available computational power of vehicles could allow to cope with MPC's computational complexity in the next few years.

2-5 Concluding remarks

In this chapter, background information was given on the topic this research is focused on. The road segment with traffic was presented. The equations of motion for kinematic single-track vehicle model were presented. Next to this, a summary of the various vehicle merging control strategies was discussed based on the literature survey conducted prior to start of the thesis followed by the motivation for the choice of Model Predictive Control (MPC) as a chosen strategy for the controller design. Next chapter delves deeper into the formulation and implementation of the proposed MPC based approach.

Chapter 3

Control design

In this chapter, the controller design will be discussed in detail. Section 3-1 starts with the description of Model Predictive Control (MPC) framework followed by the formulation of objective function and the constraints equations. The designed strategy for lane changing of vehicles is presented in Section 3-4. Finally, the implementation of the controller is discussed in Section 3-5 followed by the concluding remarks.

3-1 Model Predictive Control

Model Predictive Control (MPC), also known as Dynamic Matrix Control (DMC) or Receding Horizon Control (RHC), is a type of control based on a model that predicts the behavior of the system over time. Model predictive control is utilised to devise an optimal control sequence based on a pre-defined objective function for a system over a time interval known as prediction horizon. In other words, an optimization algorithm is used in the control scheme to find the most suitable future values for the control inputs. During every iteration of the controller, a prediction is made of the behaviour from the system model, and an optimal control signal is formulated for the chosen prediction horizon. Instead of only using the current and past states of the system to adjust the input signals, an MPC attempts to predict the future using information from the current states and a model of how the system behaves.

The main aim of MPC is to minimize a performance criterion in the future that is subjected to constraints on the manipulated inputs and outputs, where the future behavior is computed according to a pre-conceived model of the plant. In this case, this model is as presented in Section 2-2-1. A block diagram of a model predictive control system is shown in Figure 3-1. The required control inputs are calculated at each sampling instance k, using the current state as initial conditions to solve a finite optimal control problem. The number of prediction steps for the control input are defined as the prediction horizon (N_p). This control input is implemented for the first step, and then the sequence of control inputs is calculated by repeating the optimization process at the next sampling instance. To reduce the complexity of the optimization problem, a control horizon (N_c) is introduced, which means that the input is taken to be constant beyond the sample step $k + N_c$. Along with a decrease in the number of optimization parameters and the computational time, a smaller value of N_c also results in a smoother control signal, which is often desired in practical situations.

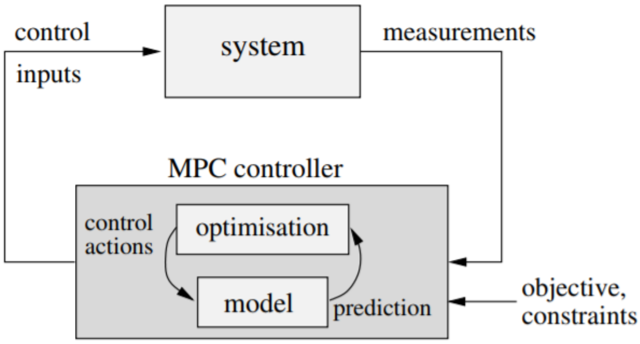


Figure 3-1: Model Predictive control block diagram [78]

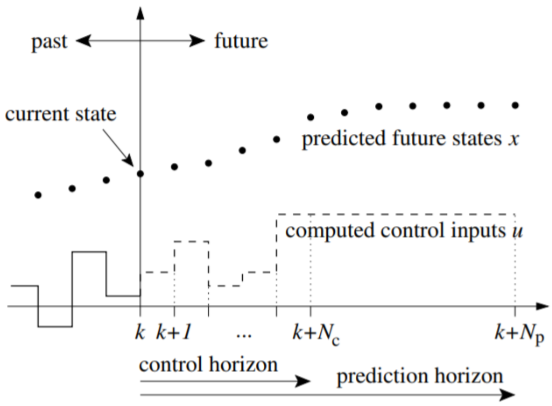


Figure 3-2: Model predictive control [78]

Essentially, an MPC control strategy is very much analogous to driving a car [121]. The driver has a reference trajectory in mind over a certain spatial and temporal horizon, with the intention to stay within a road lane and follow the lane centerline. Based on previous experiences of driving, the driver has a mental model of the car and know how the car will behave when applying a certain control input. Naturally, the control input is only applied at the current time step but is also planned ahead i.e. planning in advance to brake before an upcoming turn. This is in contrast to classical control strategies such as PID, where only past and current errors are taken into account.

In Figure 3-1, the system and the prediction model refers to the discretized equations of motions given by the Eq. (2-2) to Eq. (2-6). The continuous state equations Eq. (2-1) can be converted to discrete form by representing them as follows:

$$X_k = f(X_{k-1}, U_{k-1}) = X_{k-1} + \int_{t_{k-1}}^{t_k} f\left(X(\tau), U(\tau)\right) d\tau, \tag{3-1}$$

where $t_k = t_{k-1} + \Delta T$ and ΔT is referred to as the sampling time, the choice of which is motivated in Chapter 4.

3-2 Objective function

The previously described system setup, as shown in Figure 3-3, allows us to formulate the objective function centered around it. In this thesis, the main goal for the centralized control agent is to maximize the throughput of vehicles from the road segment. With respect to automotive control, the objectives typically deal with safe generation of safe trajectories of vehicles while respecting vehicular and environmental constraints. The following control objectives were defined to form a framework for the controller development. The controller needs to be able to:

- Maximize the throughput of the vehicles from the road segment.
- Generate safe and smooth trajectories for vehicles by maintaining control inputs of acceleration and steering rate accordingly.
- Maintain each vehicle's velocity as per the desired highway cruising speed.

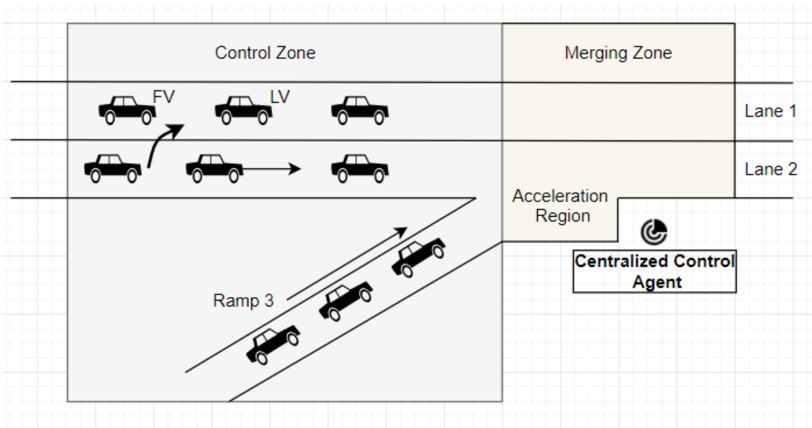


Figure 3-3: On-ramp merging road segment.

The control objectives described above broadly define the controller requirements and give a general guideline for the formulation of the objective function and constraints. In mathematical terms, the formulated objective function for the MPC consists of the following performance indices:

$$J = \sum_{j=1}^{N_p} \Bigg(\underbrace{\omega_1 \sum_{i=1}^{N} \|v_i(k+j|k) - v_i^{des}\|^2}_{\text{Desired velocity}} + \underbrace{\omega_2 \sum_{i=1}^{N} \|u_i(k+j|k)\|^2}_{\text{Manipulated variables}} + \underbrace{\omega_3 \sum_{i=1}^{N} \|\Delta d_i(k+j|k) - \Delta d_i^{des}\|^2}_{\text{Desired gap tracking}} + \underbrace{\omega_4 \sum_{i=1}^{N} \|\Delta u_i(k+j|k)\|^2}_{\text{Manipulated variables rate}}, \tag{3-2}$$

where $*(k+j|k)$ denotes the expected value of the variable at j-th step from k, with v_i^{des} being the the desired velocity of the vehicles. The control inputs, henceforth known as manipulated variables, refer to acceleration and steering rate of each vehicle. The subscript i refers to the index of each vehicle inside the control and merging zones, with the total number of vehicles equated to N. Δd_i is the longitudinal distance between leading and following vehicle. N_p refers to the prediction horizon, the choice of which is motivated further in the thesis. Weighting factors $w_{1,2,3,4}$ are introduced to prioritize the terms in the multi-objective cost function, since the cost function objectives are generally conflicting in nature and a tradeoff has to be made between them. The performance indices in the objective function are described as follows:

Desired velocity term (w_1)

The first term refers to the speed objective. It is important for the centralized control to set a desired speed for each vehicle inside the control zone based on the surrounding driving conditions. The reasons for this are twofold. One one hand, different road segments have a desired speed limit set in advance in order to be followed by the drivers. Moreover, the choice of a specific speed value for a specific vehicle can effect the comfort of the trajectory generated. Based on literature study of normal human driving conditions [124], the desired velocity v_i^{des} is chosen to be a nominal value of 30 m/s. Hence, this objective function term aims at minimising the difference between the current and desired velocity for each vehicle.

Manipulated variable term (w_2)

In order to restrict large control action in the prediction horizon, it is important to penalize the manipulate variables of acceleration and steering rate accordingly. The manipulated variable weight penalizes this term to ensure that there is no excessive acceleration, braking or steering rate commands. Furthermore the ratio of weights relative to other performance terms tells the controller the importance of achieving a certain performance metric. In this thesis, this term has been penalized with the intention to obtain smooth acceleration and steering rate profiles by tuning the weight accordingly.

Desired gap tracking term (w_3)

The third term pertains to maximizing the throughput from the road segment by tracking a desired gap between the vehicles. In a Connected and Autonomous Vehicles (CAVs) setting, the inter-vehicular gaps are an important consideration for the centralized control agent to manipulate in order to achieve system-wide traffic benefits. Δd_i is the longitudinal inter-vehicular gap while the desired inter-vehicular gap Δd_i^{des} is taken to be 10m. By tracking this desired value of gap between different vehicles in the two lanes and on-ramp, the vehicles minimize the inter-vehicular distances and can make better use of available capacity and thereby result in increasing throughput from the road segment.

Manipulated variable rate term (w_4)

Also known as the manipulated variable move suppression [80], this term indicates the implicit cost of the controller changing the manipulated variables per time increment. Abrupt transitions in the generated trajectory lead to rapidly varying control commands of acceleration and steering rates, which may negatively affect the smoothness and comfort of the trajectory. A high weight on this term would mean that the controller would consider it unfavourable to quickly change the values of two consecutive control inputs and hence, improve the trajectory profile of the vehicles. In the next chapter on Simulation Results, the weight tuning and the effect of changing weights is discussed.

3-3 Constraints

The automotive control design brings along several physical constraints that need to be taken into account due to the nature of the system. The constraints of the optimization problem are set up to consider the factors of safety, comfort, vehicle (actuator) limitations, and the aspects of human driving behavior in order to have a realistic representation of the designed scenario. The formulated state and manipulated variable constraints are as follows:

$$a_{min} \leq a_i \leq a_{max}, \tag{3-3}$$

$$v_{min} \leq v_i \leq v_{max}, \tag{3-4}$$

$$\dot{\delta}_{min} \leq \dot{\delta} \leq \dot{\delta}_{max}. \tag{3-5}$$

The first set of constraints bound the maximum acceleration and deceleration values, the second set of constraints put a limit on the vehicle speed, while the third set of constraints limit the steering rate. The vehicle speed is bounded by an upper speed limit of 40 m/s. A lower bound is set at 5 m/s to avoid stoppage or backward movement of the vehicle on the highway.

A statistical analysis of manual driving data of 125 drivers is presented in [124] which shows values of acceleration for different range of velocities. The results of this analysis are summarized in Table 3-1, where the 5 percentile, 25 percentile, 75 percentile, and 95 percentile values of the accelerations are presented. For instance, in the low velocity range of 0-40 $kmph$, 95% of the acceleration values are in the range between -2.17 and 1.27 m/s^2. At higher velocities, these values are relatively less, as the drivers not to accelerate or decelerate too much during high speed driving conditions [124]. Hence, an upper acceleration bound of 2 m/s^2 is chosen.

In general, most of the drivers and passengers feel uncomfortable when the vehicle is decelerating with values that are greater than 3 to 4 m/s^2. If the deceleration is of even higher

Table 3-1: Acceleration values based on speed profile

	Acceleration [m/s^2] analysis			
Velocity range/Percentile	**1%**	**5%**	**95%**	**99%**
0-40 [kmph]	-2.17	-1.42	1.27	1.77
40-70 [kmph]	-1.74	-0.85	0.81	1.09
Above 70 [kmph]	-0.88	-0.52	0.55	0.73

Table 3-2: Values of parameters

Parameters	**Value**	**Units**
l_{wb}	2.5	$[m]$
v_{min}	5	$[m/s]$
v_{max}	40	$[m/s]$
a_{min}	-4	$[m/s^2]$
a_{max}	2	$[m/s^2]$
$v_{\delta,min}$	-0.4	$[rad/s]$
$v_{\delta,max}$	0.4	$[rad/s]$

values, then it is to prevent the vehicle from crashing into other vehicles or obstacles. While the data shows the deceleration range of 2.5 m/s^2 for velocity profile of 40-70 km/h, a conservative value of 4 m/s^2 is chosen as the lower bound on acceleration. Similarly, the steering rate values human-driven vehicles are estimated to be in the range of 20 deg/s translating to 0.35 rad/s [127], [128]. Hence, the bounds on the steering rate are taken to be a slightly higher value of -0.4 rad/s and 0.4 rad/s. The values of the complete parameters are shown in Table 3-2.

Next to this, the safety constraints for the vehicles are formulated as follows:

$$|x_{1,l} - x_{1,l-1}| \geq S \quad \forall l = 1, 2, ..., i_1, \tag{3-6}$$
$$|x_{2,m} - x_{2,m-1}| \geq S \quad \forall m = 1, 2, ..., i_1, \tag{3-7}$$
$$|x_{3,n} - x_{3,n-1}| \geq S \quad \forall n = 1, 2, ..., i_1, \tag{3-8}$$

where S is a predefined safe value of inter-vehicle distance selected to be $5m$. The subscripts 1, 2, and 3 refer to the two mainlanes and on-ramp as shown in Figure 3-3. The subscripts l, m, and n are the index for vehicles on the mainlane 1, mainlane 2, and the on-ramp, respectively. The safety constraints are included such that there is collision avoidance for vehicles in the same lane. This is done by defining a distance between two consecutive vehicles in the same lane that must be greater than a minimum defined value, given by the set of constraints in Equations (3-6), (3-7), (3-8).

3-4 Lane change strategy

The centralized control agent incorporates both longitudinal and lateral control and as such, needs to devise lane changing trajectory for vehicles in the road segment. The lane changes are induced into the simulation primarily based on the distances between vehicles. This essentially means that the lane changing is incorporated in an over-the-top manner, where the system, as shown in Figure 3-4, is being simulated in a way that it not only traverses the motion of vehicles from the road segment but also executes lane changing based on a pre-defined strategy. The reason for including this strategy is to evaluate the Nonlinear Model Predictive Control (NMPC) in a more realistic scenario, and while also studying the lateral trajectories of vehicles. This does has a drawback as it underlies certain hard assumptions, such as the vehicles must make a lane change if the conditions are met. Nevertheless, incorporation of hard-coded lane changing in proof-of-concept studies can shed insights on the performance of controller under varied traffic conditions and later on, the lane changing strategy can be made part of the MPC formulation. In this thesis, the focus has been to accommodate the on-ramp vehicles onto the mainlane with the intention to maximize the throughput from the road segment. For an ego vehicle, if a safe distance is present with respect to the leading vehicle in an alternating lane and the safety conditions are satisfied, then the ego vehicle makes a lane change by merging into the gap between leading and following vehicle in the alternating lane. In this way, the vehicles on lane 2 make way for the vehicles from the ramp to merge into lane 2. As illustrated in Figure 3-3, the terms LV and FV refers to leading vehicle and following vehicle, respectively.

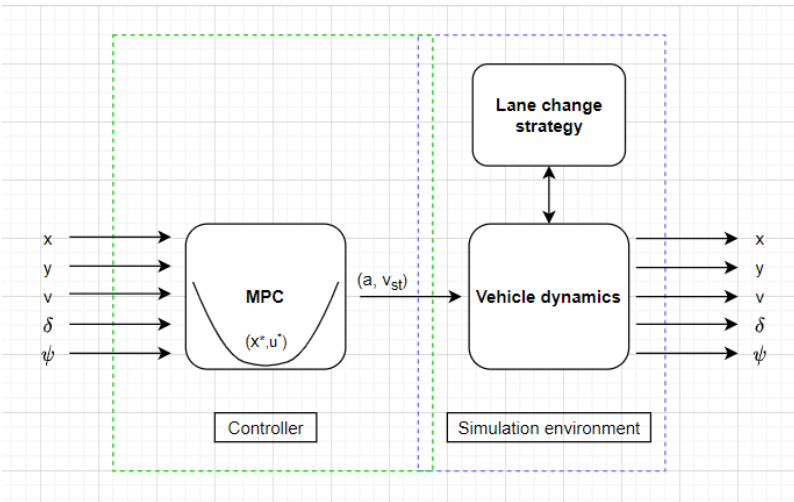


Figure 3-4: Schematic overview of the controller environment

Three scenarios are generated to distinguish between the three phases of driving. In the **normal scenario**, vehicles are made to traverse normally in their own lane while satisfying the safety constraints as presented before. While doing so, if there is a gap available in the alternating lane, with respect to leading vehicle in that lane while also satisfying safety constraints with the following vehicles, then the vehicle starts to make lane change. During the**lane-change scenario**, the vehicle tends to follow the leading vehicle in the target lane while also satisfying the lateral position constraint of following the lane centerline. When a vehicle is making a lane change, the other vehicles are put to a **waiting scenario**, in order to minimize the possibility of conflicts occurrence due to excessive lane changing.

The lane change maneuver is executed by providing the controller with an external lane change request. From an ego vehicle's perspective, the lane changing is only based on the distances with the leading vehicles in the alternating lanes. If a safe distance is present to make lane change, the decision maker approves to initiate the maneuver. Approximately 5 seconds after the lane change request, the lateral distance reference moves from the initial lane to the target lane.

The pseudo-code for lane change strategy is presented in Algorithm 1.

Algorithm 1 Pseudo-code for lane-changing strategy

procedure (i: a specific lane, j: alternate lane, FV: Following vehicle, LV: Leading vehicle)
NORMAL SCENARIO:
  initialize
  $x_{LV,i} - x_{ego,i} = GapAhead_{i,i}$
  $x_{LV,j} - x_{ego,i} = GapAhead_{i,j}$
  $x_{ego,i} - x_{FV,j} = GapBehind_{i,j}$
  if $GapBehind_{i,j} >$ SafeGap **then** POSSIBLE TO MAKE lane change
  if $GapAhead_{i,i} <$ SafeGap
    and $GapAhead_{i,j}$ - $GapAhead_{i,i} >$ GapDifference
    and vehicle ahead in this lane is in *Normal* scenario
    and it is POSSIBLE TO MAKE lane change **then**
    switch to *Lane-change* scenario
    if *following* vehicles are present in either lane **then**
      set their scenario to *Waiting*
  else set *leading* vehicle in current lane as target position
  close;
LANE-CHANGE SCENARIO:
  if if gap to lane centerline ≤ bound **then**
    change scenario form *Lane-change* to *Normal*
    set *leading* vehicle as target position
  else
    set *following* vehicles to *Waiting*
    set lane centerline as lateral target position
    close;
WAITING SCENARIO:
  set *leading* vehicle as target position
  set current lane centerline as lateral target position
  close;

3-5 Implementation

To solve the Model Predictive Control (MPC) problem, the optimization problem needs to be cast as a nonlinear programming problem (NLP). The general formulation of a nonlinear programming is as follows:

$$\begin{aligned} \underset{x(k),u(k)}{min} \quad & J\left(x(k), u(k)\right), && \text{(3-9)} \\ \text{subject to:} \quad & u(k) \in U, x(k) \in X, \\ & x(k+1) = f\left(x(k), u(k)\right), && \text{(3-10)} \\ & g\left(x(k), u(k)\right) \leq 0; \quad \text{for each } i \in \{1, 2, ..., m\}, \end{aligned}$$

where the cost function J constitutes the objective function as defined by Eq. (3-2), and the inequality constraints g refers to the set of inequality constraints as defined in Section 3-3. As part of MPC, the optimization problem (3-9) needs to be posed as a specific formulation that a dedicated NLP solver is able to solve. A transcription method transforms the problem into an NLP [123].

In general, transcription methods for optimal control can be categorised in two distinct groups. At first, there are direct and indirect methods. In direct methods, the problem is first discretized and then optimized. In indirect methods, the necessary conditions for optimality are calculated first and then, these conditions are discretized and solved. The transcription methods considered in this thesis, rely on direct transcription. Indirect transcription methods are generally more accurate and will have a more reliable error estimate but are hard to solve in practice as they require more accurate initial guesses to converge [123].

Direct methods can further be divided into single shooting and multiple shooting methods. In single shooting method, only the control trajectories are discretized, by a piecewise smooth approximation. The discretized control trajectories then enter the NLP. In other words, the trajectory is approximated using a simulation. A single shooting method can be compared to a shooting of cannon at a target. At first, an estimate is made of a good shooting angle. Then, it is checked whether the target has been hit. If not, the angle is adjusted based on the previous result and hence, the problem is solved in a sequential manner using multiple iterations. The resulting NLP using a single shooting method is generally a smaller sized problem, but highly nonlinear.

In multiple shooting method, instead of representing the entire trajectory as a single simulation, the trajectory is divided up into segments, and each segment is represented by a simulation [123]. As compared to a single shooting method, a multiple shooting method typically results in more robust performance and a higher degree of numerical stability i.e. less sensitive to errors due to e.g. poorly chosen initial guesses.

In this thesis, direct single shooting based on is used. The initial guess values for acceleration and steering rate of vehicles are taken to be 1 m/s^2 and 0 rad/s. Runge-Kutta methods are a class of methods to integrate ordinary differential equations. The order of the Runge-Kutta method refers to the amount of approximations of the slope within one time step. In this case, a second order Runge-Kutta method (RK2) is employed in order to get a balance between performance and computational complexity. RK2 method is reasonably simple and robust and is a good general candidate for numerical solution of differential equations. It approximates the solution to a differential equation $\dot{x} = f(x,t)$ with the associated initial conditions.

With respect to Figure 3-1, the system measurements are updated at each time step and serve as an input to the MPC controller. The optimal control inputs (manipulated variables) are the outputs of the MPC, which are then used as an input to the actuators of the vehicle. Moreover, the initial conditions for the states need to be defined within the MPC. This is done by choosing the initial values of for different traffic scenarios, as presented later in the thesis.

The two most established approaches to solve a nonlinear model predictive control problem are Sequential Quadratic Programming (SQP) and the Interior-Point Method. In SQP, an NLP is modelled as a Quadratic Programming (QP) subproblem at each iteration x_k. This subproblem is then solved to find the next iteration x_{k+1} [126]. This process is then iterated to find an optimal solution x^*. In contrast to the interior point method, in SQP the iterations do not require to be feasible. Interior point is a *large scale* algorithm. An optimization algorithm is termed as *large scale* when it uses linear algebra that does not need to store or operate on full matrices [125]. This may be done internally by storing sparse matrices, or by using sparse linear algebra for computations whenever possible. In this thesis, SQP method is employed as it is deemd to be the most suitable method for mid-range problems.

Optimization solver

The Sequential Quadratic Programming (SQP) is implemented using the **fmincon** function in MATLAB. The **fmincon** function from MATLAB is a solver that finds the minimum of constrained nonlinear multivariable function. The syntax to call the function to solve the nonlinear optimization problem is:

```
x = fmincon(fun,x0,A,b,Aeq,beq,lb,ub,nonlcon,options).
```

The term **fun** pertains to the objective function as defined by the Equation (3-2). The optimization is solved by starting from the initial conditions **x0**. The problem is subjected to the associated constraints where the **A** and **b** refer to linear inequality constraints. **Aeq** and **beq** refer to linear equality constraints, while `lb` and `lb` are lower and upper bound vectors, respectively. **nonlcon** is the function for nonlinear constraints as presented in Section 3-3. The input argument **options** is used to set specific solver settings for the optimization such as the type of algorithm and maximum number of iterations.

3-6 Concluding remarks

In summary, this chapter presented the prerequisites for controller development in terms of the formulation of objective function, and the comfort and safety constraints along with description of the relevant design choices. Hence, the problem setup of the thesis described in Chapter 2 is linked with the MPC framework. Moreover, the lane change strategy adopted for the vehicles is presented. This was followed by a discussion on the implementation of the MPC controller. The simulation setup, design of MPC parameters, weight tuning, and discussion on the obtained results is presented in the next chapter.

Chapter 4

Simulation results

The previous chapters have described the framework and the design fundamentals of the proposed centralized predictive control strategy for the throughput maximization of vehicles. This chapter will focus on the implementation aspects of the proposed controller in order to evaluate its performance. Furthermore, the obtained simulation results and the controller assessment is discussed. Section 4-1 starts with describing the simulation setup based on MATLAB. The design choices of Nonlinear Model Predictive Control (NMPC) such as sampling time, prediction, and control horizons are motivated in Section 4-2. This is followed by a discussion on the weight tuning of controller. Next to this, the simulated scenarios of low and heavy traffic are described. Finally, a discussion is done on the obtained results in Section 4-5.

4-1 Simulation setup

In order to inspect the proposed controller performance and to obtain insights into its behavior, a simulation framework is implemented in MATLAB. The multilane road segment as shown in Figure 4-1 is simulated which is able to handle varied number of vehicles with a specific set of initial conditions. To this end, three scenarios corresponding to the baseline case, low traffic, and high traffic scenarios are studied. On a holisitic level, this comparison helps to ascertain whether the controller can handle varied number of vehicles in a multilane framework and what are the caveats for incorporating a high number of vehicles in the NMPC scheme.

For a fair comparison with similar studies on the topic, it is pertinent to mention the system specifications upon which the simulations are done. The simulations are obtained by running MATLAB 2019a on an *HP Pavilion* notebook while other specifications are listed in Table 4-1. Also pertinent to mention here is that very often the computer simulations are accompanied by lots of tasks running simultaneously on the operating system. This implies that the effect of unrelated background programs on the computational time of the control algorithm need to be taken into account.

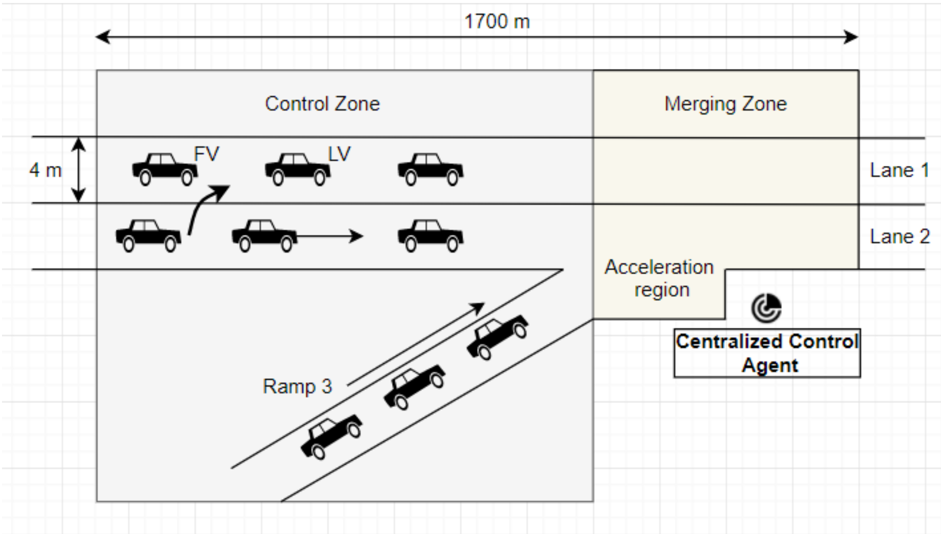


Figure 4-1: Simulated on-ramp merging road segment. (*The measurements are not to scale.*)

Table 4-1: Hardware and software specification of the computer system

Specification	**Details**
Processor	Intel Core i7-6500 @ 2.5GHz
Memory (RAM)	12GB
Operating System	Microsoft Windows 10, Build 18362

4-2 Design choices

The crux of the NMPC formulation lies in the appropriate values of its design parameters such as sampling time (T_s), prediction horizon (N_p), and the control horizon (N_c). For MPC tuning and especially for horizon adjustments, it is a recommended practice to first define the sampling time or control interval duration and then hold it constant during the process of further tuning the controller.

As the control signal will be updated every sample time, so we do not want the vehicle to drive too long a distance before it is updated. Qualitatively, as the control interval duration decreases the disturbance rejection properties of the controller improves. On the other hand, a lower value of sampling time will increase the computational time of the optimization problem. Hence, a trade-off between performance and computational effort is required. With a sampling time of 0.1 s and a vehicle driving at 20 m/s, the control signals would be updated every 2 m. If the speed is increased, the vehicle will travel longer distances between the updates e.g. if the velocity would increase to 40 m/s, the updates would come every 4 m instead. This means that the sampling time may be needed to decrease if the vehicle is to be driving faster.

After running simulations for different sampling times, a value of 0.5s was chosen to be the most suitable for our case. As the desired velocity is set to 30 m/s so the vehicles would not be traveling distances more than 15m before the next update. The vehicles would also not be making maneuvers which are aggressive or evasive, and hence, a higher sampling time would have a minimal effect on the controller performance. Furthermore, a minimum of 10 vehicles with two manipulated variables each, are being simulated and hence, the computational time increases significantly even for a slight increase in the sampling time.

Choosing a suitable value of the prediction horizon (N_p) is a matter of trade-off between an acceptable computational time and the desired performance to be attained. N_p is the number of future control actions the MPC evaluates by prediction while optimizing the manipulated variables. In other words, N_p tells the controller how many future time steps to take into consideration.

As the scenario primarily focuses on minimization of inter-vehicular distances in order to maximize the throughput along with lane changing and merging from the on-ramp, so the controller needs to be able to predict the states ahead into the future while attaining the desired performance. Different values were tested and it was concluded that an N_p value of 12 steps results in an appropriate controller performance along with limited computational time. Moreover, the effect of varying the prediction horizon is also studied by way of running simulations for different horizon values as depicted in Figure 4-2. Here, T_{solve} is defined as the as the time it took to perform one optimization iteration. An N_p value of 12 results in the T_{solve} value of 3.5 seconds. It can be seen that the computational time increases drastically as the number of prediction steps is increased. It was also observed that the computational time also increases proportionally with the number of vehicles in the road segment.

It is pertinent to mention here that, on the face of it, a time of 3.5 seconds for the controller to update the vehicle states is relatively high. This is due to the fact that the premise of the thesis rests on simulating the vehicles on the complete road segment and observing the advantages and limitations of employing an NMPC based strategy for traffic control. Furthermore, a few points of discussion and future research directions are presented in the chapter on Conclusion to mitigate the computational load of NMPC.

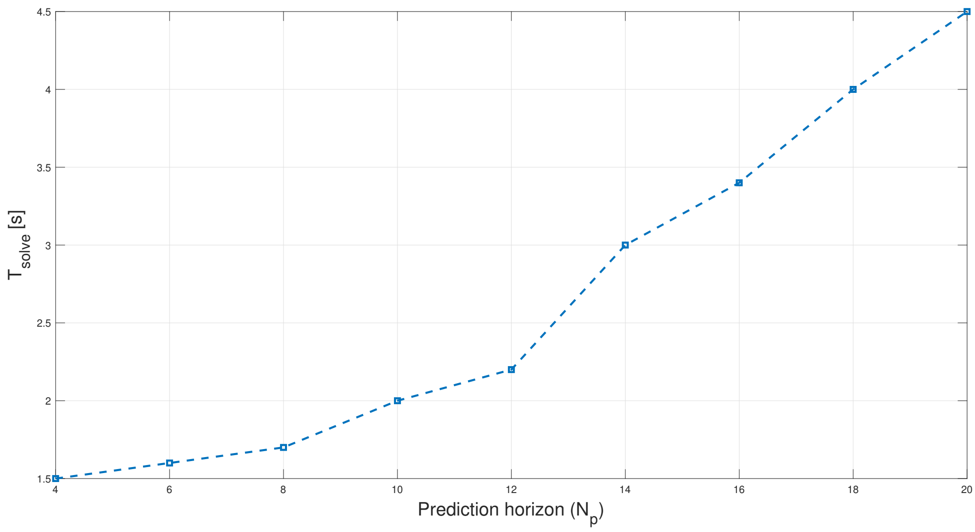


Figure 4-2: Comparison of computational time with prediction horizon

Correspondingly, the the control horizon N_c is the number of control actions that are op-

Table 4-2: Effect of varying gap minimization weight (w_3) on travel times

Weight (w_3)	Travel time
5	21573.8 s
7	20917.2 s
10	20784.5 s
15	20494.3 s
25	20257.3 s
35	20183.4 s

timized at each control interval. It is recommended to take the value of $N_c << N_p$ [125]. For the controller, a smaller value of N_c will result in a smaller number of variables to be computed by the solver at each control interval, thus improving computational time. An N_c value of 5 was deemed to be appropriate in this case. Moreover, manipulated variable blocking is employed where the control inputs are kept constant for $N_c > 5$. This has advantages in terms of tuning flexibility, and controller robustness. The complete lists of parameters is shown in the Table 4-3.

Tuning of weights

The tuning of weights in an appropriate manner is another critical factor in the formulation of an MPC based controller. The weights for the controller need to be carefully tuned in order to make the its behaviour and actions fit the control objective and the specified requirements. With tuning, the primary rule is that a smaller value of weight indicates that the controller should consider the corresponding variable as less important to the overall performance index of the controller. Conversely, putting more weight is akin to implicitly prefer that very variable term by the controller.

As discussed in Section 3-2, the primary objective is to maximize the throughput from the road segment which corresponds to the gap minimization term in the objective function, as shown in Eq. (3-2). Next to this is the desired velocity term which is tuned by observing the velocity plots and the convergence behavior of all the vehicles. In some cases, the control inputs calculated by the NMPC were found to be excessive. Hence, the manipulated variables a and v along with the rate terms namely Δv and Δa, were penalized by observing the velocity and acceleration plots for different simulations and the values were chosen that resulted in acceptable responses. Furthermore, hard constraints ensure that the manipulated variables stay within designated bounds. Regarding steering rates, most of the tuning effort was spent in making the merging trajectories of vehicles as smooth as possible.

In order to attain the maximization of throughput, the weight pertaining to the gap minimization term w_3 was tuned carefully. A comparison of different weights values for travel times was done in order to ascertain the effect of weight variation on the travel time of vehicles. This is shown in Table 4-2. It can be seen that an increase in the value of w_3 from 15 does not substantially increase the total travel time for vehicles and hence, a value of 15 is chosen for this term. It is also to be noted that this effect of varying weights is not exhaustive in nature, as different travel times were observed while running the simulations for various

Table 4-3: Values of parameters

Parameters	**Symbol**	**Value**
Prediction horizon	N_p	12
Control horizon	N_c	5
Sampling time	T_s (s)	0.5
Desired velocity weight	w_1	10
Acceleration weight	$w_{2,a}$	5
Steering rate weight	$w_{2,v_{st}}$	120
Gap minimization weight	w_3	15
Δa weight	$w_{4,a}$	5
Δv_{st} weight	$w_{4,v_{st}}$	40
Length of control zone	l_{CZ} (m)	1300
Length of merging zone	l_{MZ} (m)	400
Lane width	l_{width} (m)	4

traffic scenarios. Nevertheless, it gives an intuitive indication about the tuning of weights in the multi-objective optimization formulation.

4-3 Low traffic scenario ($i = 10$)

At first, a baseline case is designed in order to assess the performance of the proposed controller. In this case, the number of vehicles (i) is limited to 10. The gap tracking term is removed from the objective function (3-2) in order to mimic a *no-control* case, and hence, only the desired velocity along with manipulated variables and their rates are penalized. This baseline formulation also helps to compare the proposed NMPC controller to previous works (such as [83], [74]) where the focus has largely been on the smooth and collision-free ramp merging trajectories. Furthermore, to compare the performance of NMPC, a linear Model Predictive Control (MPC) is formulated based on the linearization of the vehicle prediction model. The linearization is done by using small angle approximation for the state equations described in Eq. (2-2) to Eq. (2-6). In a real world scenario, different vehicles enter and exit the road network at different time, and their initial speeds are also different, although within certain ranges. Here, the vehicle entry time pertains to the moment the vehicle enters the control zone, as shown in Figure 4-1. The velocities for vehicles have been randomly initialized and the initial lane index has also been specified for the vehicles. The specifics for this scenario are as follows:

$$\textit{vehicle entry time} = \{0, 0.5, 1, 3, 4, 6, 9, 12, 13, 14\},$$

$$\textit{initial velocity} = \{35, 25, 25, 30, 33, 27, 20, 35, 40, 25\},$$

$$\textit{initial lane index} = \{1, 2, 2, 1, 3, 2, 1, 2, 3, 3\}.$$

For the case of low traffic scenario, the number of vehicles are also taken to be 10. Here, the complete objective function as shown in Eq. (3-2) is included. The specification for vehicle entry time, initial velocities and lane indexes for this scenario are similar to that of the baseline case. More vehicles are placed on the lane 2 in order to observe the behavior of vehicles merging form the ramp onto lane 2.

4-3-1 Notable results

For the sake of brevity, only the most relevant plots are presented here, while the complete results are shown in Appendix. At first, the velocity plots for the vehicles are presented. Figure 4-3 shows the comparison between the velocity plots for NMPC and baseline case. Note that all the vehicles are represented with the same color code to distinguish between the two cases. It can be seen that the vehicles tend to achieve the desired velocity of 30 m/s for both cases. The velocity profile for baseline case is very smooth. This is expected, because the gap tracking is not being done and hence, the vehicles attain the desired velocity and then traverse with it during the course of the simulation. But the baseline has a drawback in terms of travel times which is discussed later. For the NMPC scenario, the jumps in the velocity profile are because the vehicles tend to achieve the desired inter-vehicular gap while simultaneously attaining the desired velocity. The gap tracking has been penalized more and hence, there are instances of vehicles increasing their velocity and then settling down to 30 m/s.

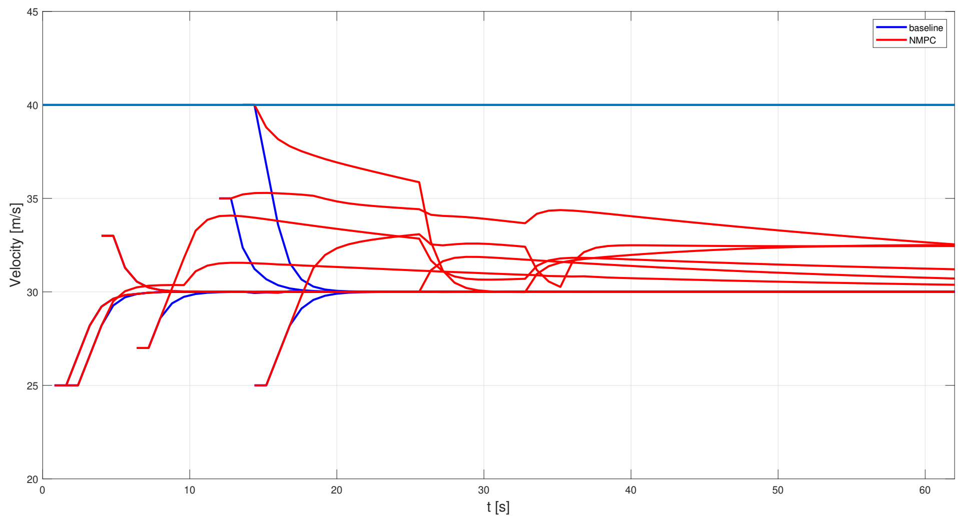
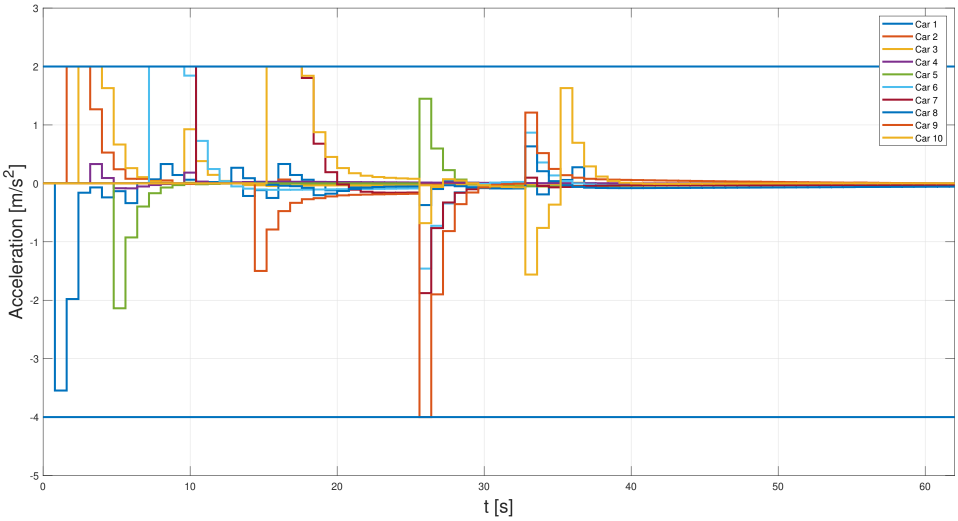
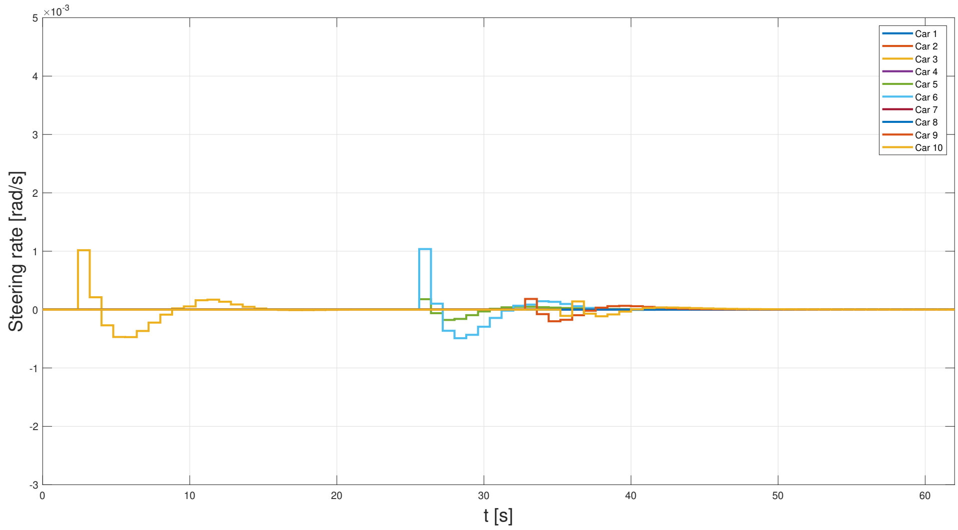


Figure 4-3: Comparison of velocity plots with baseline scenario

Figure 4-4a and Figure 4-4b show the plots for manipulated variables i.e. acceleration and steering rates, respectively for all the simulated vehicles. It can be seen that the manipulated variables of acceleration and steering rates remain within the designated bounds. The minimum and maximum values obtained for the manipulated variables are listed in Table 4-4. The reason for the vehicle to reach high longitudinal acceleration is partly because the vehicle intends to close the gap with the vehicle ahead in the lane and any desired acceleration rate is not specified. Furthermore, the term regarding gap minimization is being penalized more in comparison to other terms and hence, the acceleration values reach close to the upper bound.

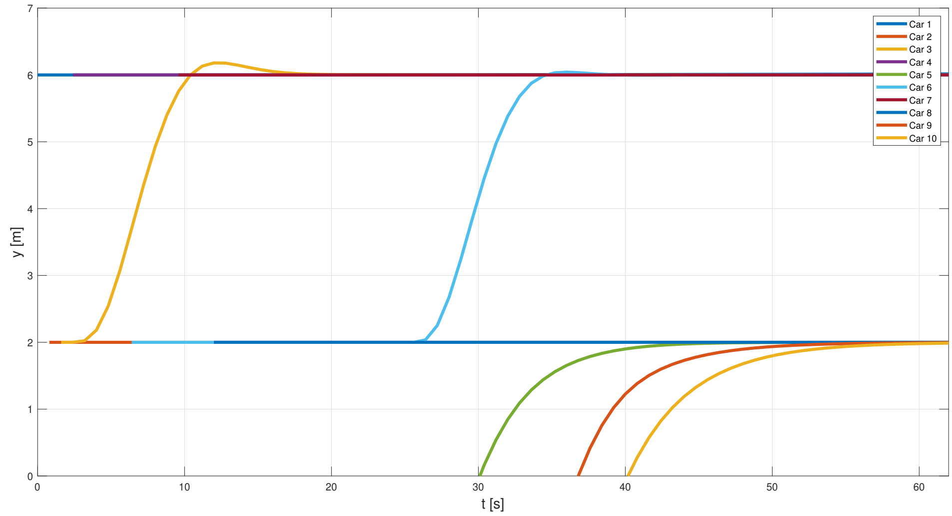


(a) Vehicular acceleration vs time plot

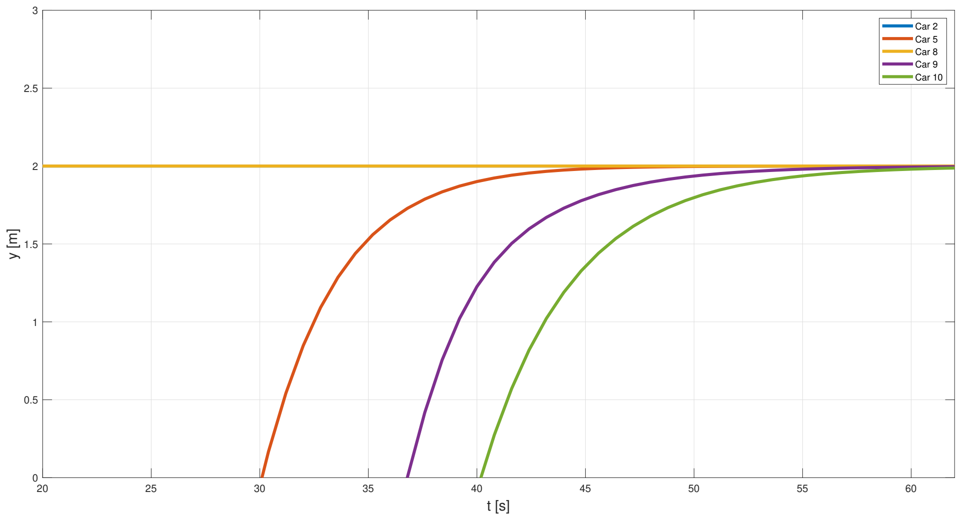


(b) Vehicular steering rate vs time plot

Figure 4-4: Manipulated variables i.e, acceleration and steering rate for the low traffic scenario based on the proposed NMPC controller

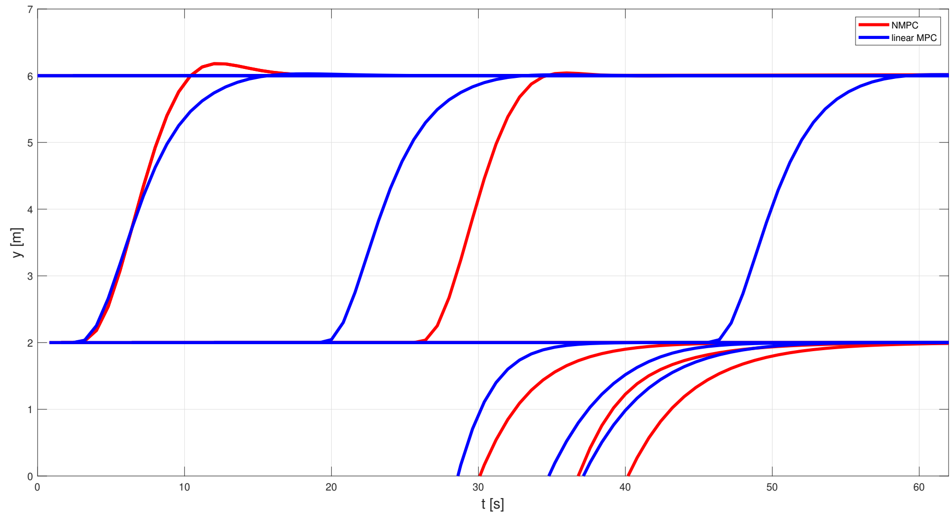


(a) Lateral trajectory between lanes 1 and 2

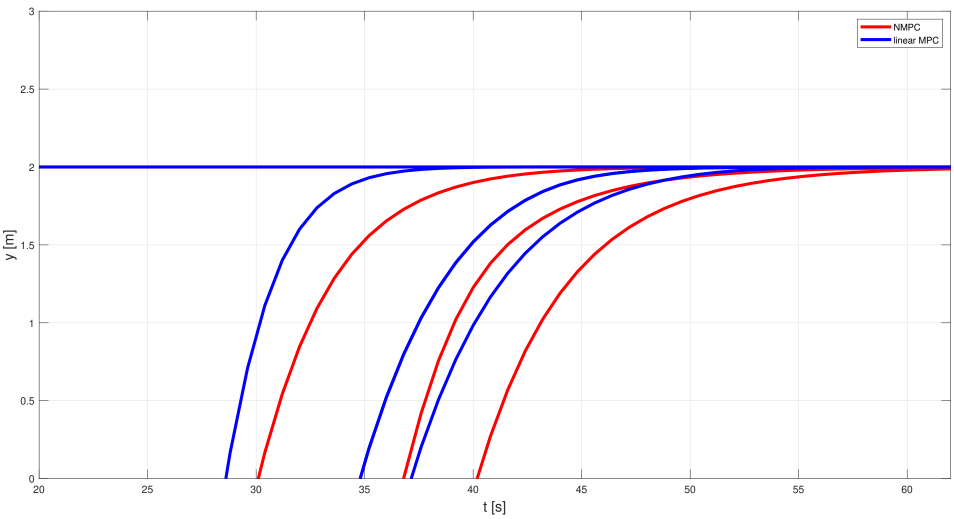


(b) Lateral trajectory between lanes 2 and on-ramp

Figure 4-5: Lateral trajectories of vehicles for the low traffic scenario based on the proposed NMPC controller



(a) Lateral trajectory between lanes 1 and 2



(b) Lateral trajectory between lanes 2 and on-ramp

Figure 4-6: Comparison of lateral trajectories of vehicles between NMPC and linear MPC

Nevertheless, the acceleration return to zero over time as the simulation progresses. The deceleration behavior shows that apart from a few vehicles reaching close to the lower bound, the other vehicles decelerate with reasonably small values. The values of the obtained steering rates are observed to be well within the bounds. This is due to the fact that the vehicles are not intended to make aggressive or evasive maneuvers which ask for abrupt steering rates.

The plots for the lateral trajectories for the vehicles are shown in Figure 4-5a and Figure 4-5b. The plots for lane 1-2 and lane 2-ramp are to be inspected next to each other as they complement each other to make the complete road segment. The lane changing and on-ramp merging behavior of vehicles show that there are smooth trajectories and a minimal value of overshoot for one vehicle. Furthermore, the trajectories for NMPC and linear MPC are plotted together in Figure 4-6. It is inferred that the NMPC performs better in case of lane changing between the two mainlanes (lane 1 and lane 2) as the lane changing duration of NMPC is shorter than that of linear MPC. This difference is almost negligible in case of on-ramp merging where both methods result in a similar duration of lane change.

4-4 High traffic scenario ($i = 20$)

For the case of high traffic scenario, the number of vehicles are taken to be 20, and divided into the three lanes as shown below. Six vehicles are placed on the on-ramp in order to observe the ramp merging behavior of vehicles under heavy traffic. The initialization values for simulation are as follows:

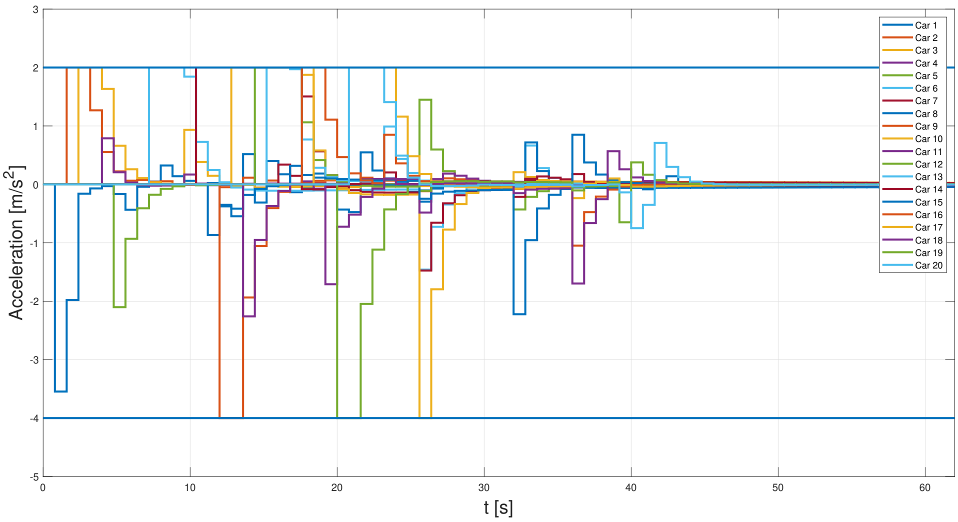
$$\textit{vehicle entry time} = \{0, 0.5, 1, 3, 4, 6, 9, 9, 9, 12, 12, 12, 12, 10, 8, 7, 5, 4, 12, 11\},$$
$$\textit{initial velocity} = \{35, 25, 25, 30, 33, 27, 20, 35, 40, 25, 35, 25, 25, 30, 33, 27, 20, 35, 40, 25\},$$
$$\textit{initial lane index} = \{1, 2, 2, 1, 3, 2, 1, 2, 2, 3, 1, 2, 2, 1, 3, 2, 1, 3, 3, 3\}.$$

4-4-1 Notable results

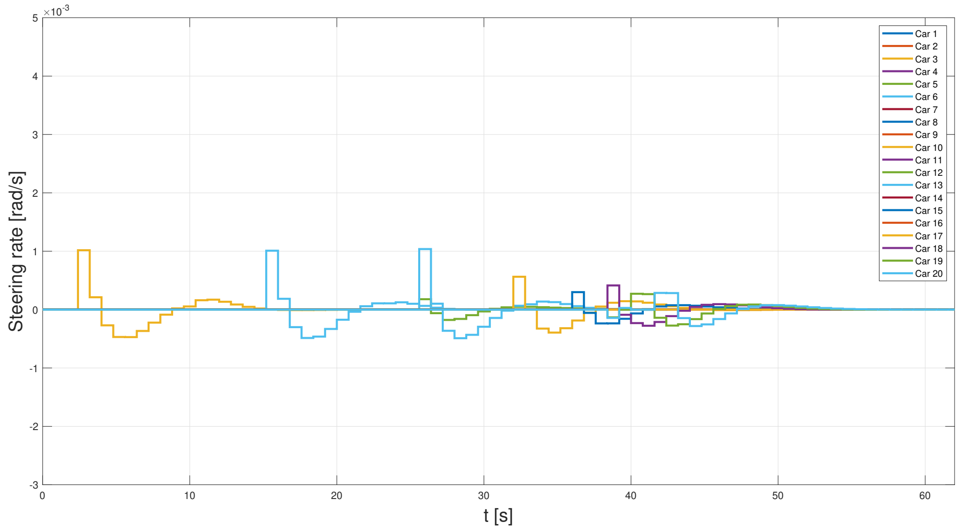
For this scenario, the plots for acceleration and steering rates of vehicles are shown in Figure 4-7a and Figure 4-7b, respectively. The acceleration values are similar to that of the low traffic scenario, with a few vehicles reaching the upper and lower bounds. Similarly, the steering rate values also remain well within the bounds. The plots for lateral trajectories are shown in Figure 4-8a and Figure 4-8b, respectively. It can be seen from Figure 4-8b that more vehicles on the ramp result in slightly higher merging time for the vehicles from the *acceleration region* to the lane 2. The interaction between *car 5* and *car 18* as shown in Figure 4-8b, highlights that the two vehicles have to wait for some while in the *acceleration region* while merging onto lane 2. This behavior was not observed in case of low traffic scenario, but was a consequence of increasing the number of vehicles on the ramp. Nevertheless, the merging profile of vehicles remain smooth with minimal overshoot achieved for one vehicle.

4-5 Discussion on results

The lateral trajectories of vehicles is shown in Figure 4-5 and Figure 4-8 for the scenarios of low and high traffic, respectively. It can be seen that the lane changing and ramp merging



(a) Vehicular acceleration vs time plot

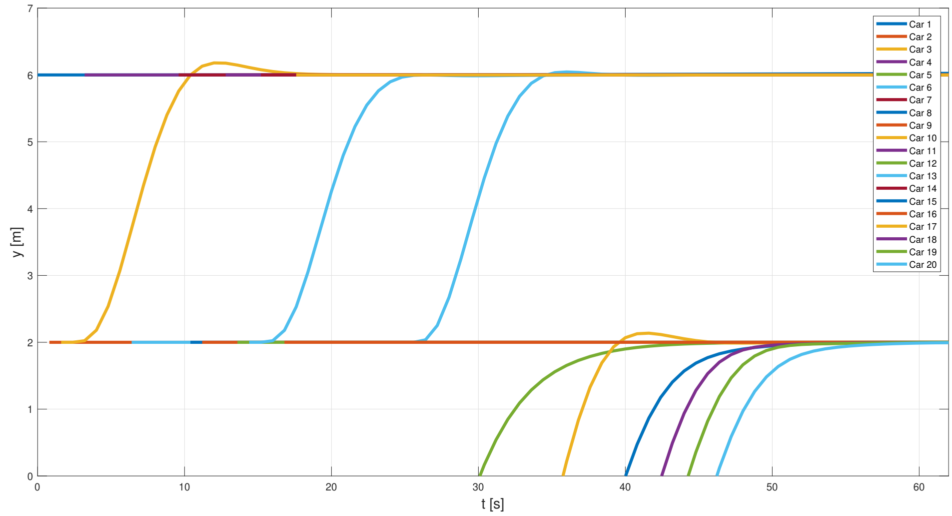


(b) Vehicular steering rate vs time plot

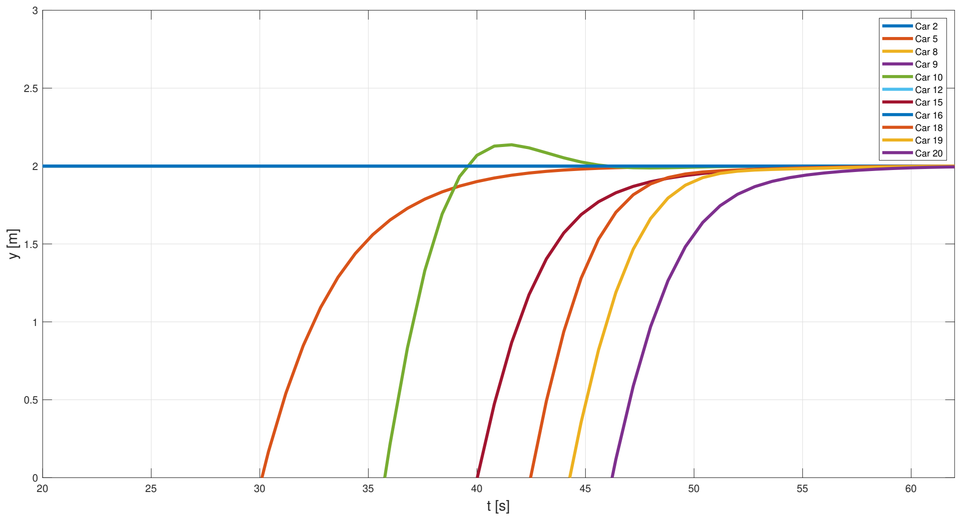
Figure 4-7: Manipulated variables i.e, acceleration and steering rate for the high traffic scenario based on the proposed NMPC controller

Table 4-4: Extremum values of the manipulated variables

Algorithm	a_{min}	a_{max}	$\dot{\delta}_{min}$	$\dot{\delta}_{max}$
baseline (low)	-4 m/s^2	2 m/s^2	$-1.5\text{x}10^{-3}$ rad/s	1x10^{-3} rad/s
NMPC (low)	-4 m/s^2	2 m/s^2	$-0.4\text{x}10^{-3}$ rad/s	1x10^{-3} rad/s
NMPC (high)	-4 m/s^2	2 m/s^2	$-0.4\text{x}10^{-3}$ rad/s	1x10^{-3} rad/s



(a) Lateral trajectory between lanes 1 and 2



(b) Lateral trajectory between lanes 2 and on-ramp

Figure 4-8: Lateral trajectories of vehicles for the high traffic scenario based on the proposed NMPC controller

Table 4-5: Performance metrics for the three scenarios

Algorithm	Travel time	Average speed	T_{solve}	Computation time
baseline (low)	21012.1 s	29.95 m/s	12.5 s	1205.6 s
baseline (high)	35173 s	30.7 m/s	21.7 s	2115.8 s
NMPC (low)	20494.3 s	32.9 m/s	14 s	1290.3 s
NMPC (high)	33705.4 s	32.4 m/s	23 s	2241 s
Linear MPC (low)	20644.6 s	32.5 m/s	11.5 s	590 s
Linear MPC (high)	33541.2 s	32.1 m/s	13 s	865 s
Comparison of NMPC (low) with baseline (low)	+2.46%	+8.18%	-12%	-7.2%
Comparison of NMPC (high) with baseline (high)	+4.17%	+4.5%	-10.6%	-14.8%
Comparison of NMPC (low) with linear MPC (low)	+0.72%	+1.2%	-21.7%	-118.5%
Comparison of NMPC (high) with linear MPC (high)	+0.48%	+0.93%	-76.9%	-158%

trajectories of vehicles are smooth in nature. However, small deviations for a few vehicles was observed. In case of high traffic scenario, the deviation of merging vehicles was a bit more apparent and the maximum overshoot was observed to be of 0.15 m.

An overview of the results of the three scenarios, obtained from simulation, is shown in Table 4-5. In order to quantify the performance of the proposed NMPC controller, two metrics of travel time and the average speed of the vehicles are calculated. Travel time refers to the total time that all the vehicles take to traverse the complete road segment as shown in Figure 4-1. Average speed is observed in order to evaluate the controller performance with respect to the baseline case. It is seen that the travel time and average speed of vehicles for low traffic scenario is improved by 2.46% and 8.18%, respectively. An improvement of 4.17% and 4.5% was observed for the travel time and average speed during the high traffic scenario. It was evident from the simulation that the average speed of vehicles is higher than that from the baseline case, as the vehicles tend to achieve the upper bound of velocity (40 m/s) while closing the inter-vehicular gaps.

Furthermore, the optimization time and computational time is also evaluated for the three scenarios. Here. T_{solve} refers to the time it takes for one optimization cycle to complete which gives an indication of the time required to update the states for the vehicles. Computational time, on the other hand, pertains to the total time required to perform the controller simulation for the complete road segment. The proposed controller results in higher values for both T_{solve} and the computational time, increasing them by 12% and 7.2% respectively. This can be explained by the addition of gap minimization term in the objective function and the fact that a mere addition of one term would have a substantial effect on the overall computational time because a large number of vehicles are being simulated. Apart from depending upon the formulation of controller, the computational time also proportionally varies with the length of the road segment. Here, a length of 1700 m was chosen in order to take into account the upstream and downstream effects of traffic due to the on-ramp merging. While the metrics of travel time and average were almost similar to that of the NMPC case, the linear MPC

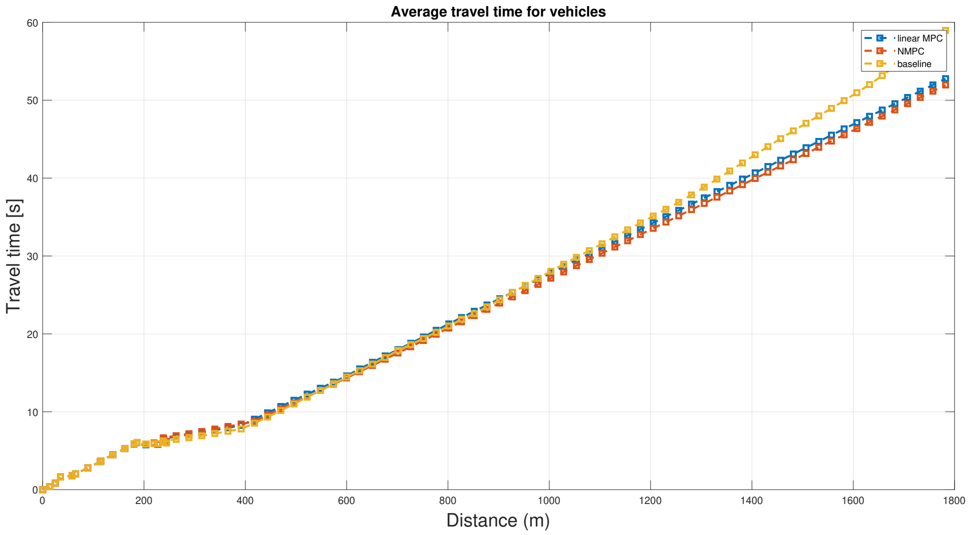


Figure 4-9: Comparison of total travel time for the baseline, NMPC, and linear MPC cases

showed a marked improvement in the T_{solve} and computational time of the algorithm. It was inferred that this improvement in the computational time was higher than that of simplifying the objective function by removing some of the sub-objectives, which shows that the incorporation of the nonlinear model greatly increases the computational effort of the controller. Another way to look at travel time is to study the average travel time for the vehicles as they traverse through the road segment. This gives an indication of the time that each vehicle would take to traverse a specific distance on the road segment, as shown in Figure 4-9 for low traffic scenario. It can be seen that the baseline case has higher travel time for vehicles. Furthermore, it can be concluded that the short plateau in the curve at the beginning is when there is no lane change taking place. This behavior can be correlated with Figure 4-5, where there is a brief interval after the first lane change and after that, there are subsequent lane changes with relatively shorter duration, which ultimately increases the travel time.

4-6 Concluding remarks

In summary, this chapter detailed the implementation aspects of the proposed NMPC controller along with presenting the obtained simulation results. The observations for the two scenarios of low and high traffic along with the baseline case were presented. It can be seen that the proposed NMPC controller results in maximizing the throughput of vehicles as indicated by the travel times and the average speed of vehicles. There exists some outlier aspects such as high computational time and the applicability of controller for different traffic scenarios. To this end, in the next chapter, conclusions derived from the work done in this thesis are presented along with highlighting a few threads of recommendations for future research.

Chapter 5

Conclusion

In this chapter an overall conclusion will be given on the research done in this thesis. Furthermore, a few recommendations will be presented together with a discussion on possible improvements and some future research threads.

5-1 Conclusion

In this thesis a Nonlinear Model Predictive Control (NMPC) based control strategy is devised to maximize the throughput of vehicles from an on-ramp merging road segment. It is shown that the proposed strategy works well to achieve the desired requirements although there exists a few caveats that need to be further evaluated.

The choice of MPC framework was motivated by the easy integration of the vehicle dynamics and system constraints into the MPC optimization. Moreover, equations from the kinematic bicycle model, which give a good representation of the vehicle dynamics for the ramp merging associated driving conditions were used. The nonlinear equations captures the most relevant nonlinearities associated with merging of vehicles in a highway setting. The objective function pertaining to the NMPC consisted of several different terms that each play their own role in maximizing the traffic throughput from the road segment along with controlling the vehicles' motion. Within the centralized framework, the central control agent determines the control inputs for the entire road network. These control inputs are found by optimizing the objective function using measurements of all the states. Hence, the goal was to devise a centralized controller that would lead to the optimal system performance in a receding horizon context.

Even though some simplifications were made in terms of an over-the-top formulation for lane changing, good performance was obtained. The lane changing was included primarily to make use of the space present in the alternating lane and by doing so, the on-ramp vehicles would have shorter delays during merging onto the mainlane. The centralized controller can accommodate varying traffic conditions such as initial velocities, lane indexes and the time instants of vehicles entering into the control zone.

As compared to baseline case, lower travel times and higher average velocities for the vehicles were achieved. The travel times improved by 2.46% while the average speed improved by 8.18% for the case of low traffic scenario. The simulation results show that the ramp merging and lane changing trajectories are smooth with minimal overshoot. This goes to show that the proposed controller scheme is a promising prospect for application in traffic control.

The main conclusions that are made in this thesis work, can be summarized as follows:

- Incorporation of nonlinear vehicle kinematic prediction model for Model Predictive Control (MPC) based control strategy works well for the designed scenarios. The optimized values of manipulated variables show that the values remain within bounds and a smooth merging profile for all the vehicles is obtained.
- The integration of both longitudinal and lateral control into the proposed control strategy works well to augment the traffic control. The lane changing makes use of the space available and keeps the lane 2 free enough so that the on-ramp vehicles can merge onto it. Although, this demands an assumption that any and all vehicles who meet the specified conditions must have to make a lane change.
- The formulation of a multi-lane highway setting with on-ramp merging provides with a more realistic traffic scenario and the centralized controller can successfully execute highway maneuvers such as lane changing and merging for all the vehicles. The trade off between different terms in the cost function can be incorporated as in this case, the minimization of inter-vehicular was penalized more than the other terms and the associated benefit in terms of throughput maximization was gained.
- The proposed NMPC based strategy works well in low and high traffic scenarios, although the trajectories of vehicles in high traffic show some deviations. The metrics of travel time and average speed of vehicles show an improvement in comparison with the baseline scenario.
- The computational effort of NMPC controller remains high. As is the case with the centralized control architecture, the size of the optimization problem scales with the size of the considered network. It was observed that the computational time varies with the number of sub-objectives in the objective function, along with the number of vehicles being simulated. The associated high computational expense associated with NMPC necessitates exploring different techniques to speed up the computation.

5-2 Recommendations

Validation of controller A prototype field testing or hardware-in-the-loop simulation is the first recommended next step to validate the proposed control strategy. After an extensive simulation phase, it is important to do real-world testing in order to evaluate the performance of controller. In this way, the real-time advantages and limitations could be studied such as the model mismatch along with the communication requirements on part of the Dedicated Short-Range Communication (DSRC) system.

Model mismatch The kinematic single-track model employed in this thesis assumes a vehicle with only two wheels, where the front and rear wheel pairs are each lumped into one wheel.

Furthermore, the tire slip is not included and hence, effects like oversteer and understeer are not considered. In future, a more detailed model, such as multi-body model could be used to include the effects of vehicles when they drive close to their physical capabilities.

Computational complexity An important direction in which future work can be done is to explore approaches for reducing the computational complexity of NMPC based controller. One example is to investigate the use of explicit MPC, which removes one of the main drawbacks of traditional MPC, namely the need to solve a mathematical program online to compute the control action. Explicit MPC allows to solve the optimization problem offline for a given range of operating conditions of interest. By exploiting multiparametric programming techniques, explicit MPC computes the optimal control action offline as an "explicit" function of the state and reference values, so that the online operations can be decreased and hence, the overall computational expense is reduced.

Traffic homogeneity In this thesis, the traffic was taken to be homogeneous. In future, a heterogeneous traffic could be introduced and the effects on wider traffic ought to be analyzed. The larger vehicles such as trucks have different model dynamics and as such, would behave differently in merging scenarios. Moreover, the introduction of vehicle platooning and mixed traffic (human-driven and autonomous) also needs to be analyzed as there will most likely be a transition phase. Future work could be done to study whether the centralized NMPC controller can equally work for such heterogeneous and/or mixed traffic flows.

Measurement errors This work assumes perfect measurements of the specified states in order to determine the optimal manipulated variables or control inputs. However, in real-world implementation, the measurements usually contain some errors. Future work could be done to examine the robustness of the method with respect to these measurement errors. In general, the vehicular trajectories should be insensitive to small and uncertain measurements. On the other hand, the vehicle should be reactive to certain deviating measurements, with the goal to have a consistent vehicle behavior without unnatural fluctuations. For instance, the use of robust MPC approach with application to traffic control have been explored in [129].

Communication delays This work assumes that there is no delay in the communication of vehicles with the centralized control agent. Moreover, it assumes that no errors occur with the communication. However, in real-world applications, there are always delays in the DSRC communication and errors are prone to occur. Hence, during implementation, the controller needs to be made robust against these types of delays and errors.

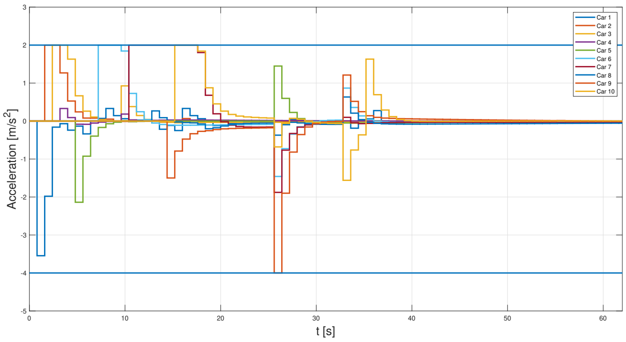
Motion prediction If the controller is equipped with a motion prediction model such as a simple curve fitting or regression method, it would help to better predict the dynamics of the surrounding traffic. However, equipping the controller with trajectory prediction comes at the expense of increased computational expense, since the trajectories need to be calculated at each time step.

Road geometries Another relevant research thread is to investigate the performance of centralized controller for different road geometries such as weaving sections, high curvature roads, and intersections. It would be interesting to apply the proposed controller for these scenarios and study the wider effect on the traffic flow.

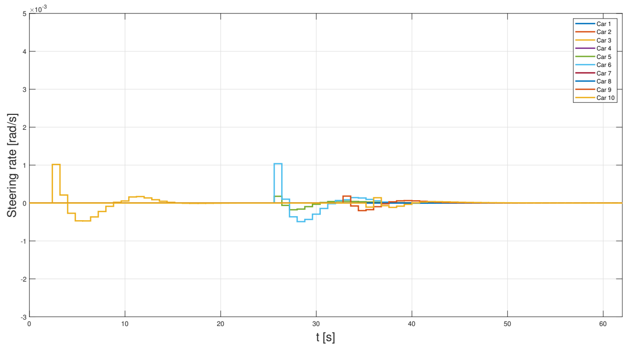
Chapter 6

Complete plot sets

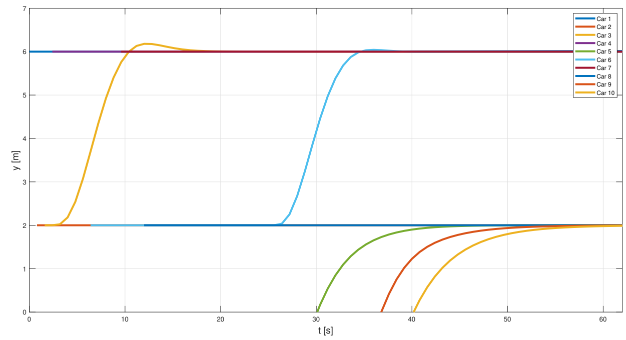
6-1 Manipulated variables and trajectory plots



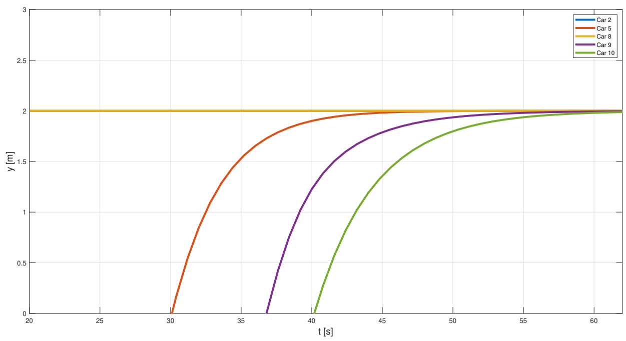
(a) Vehicular acceleration vs time plot



(b) Vehicular steering rate vs time plot

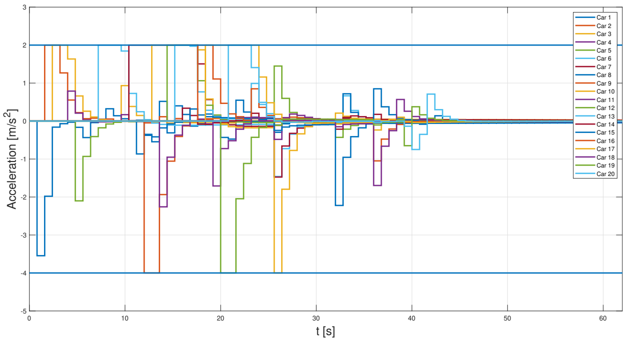


(c) Lateral trajectory between lanes 2 and on-ramp

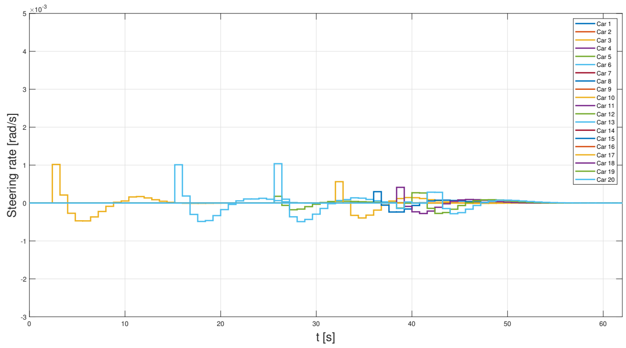


(d) Lateral trajectory between lanes 2 and on-ramp

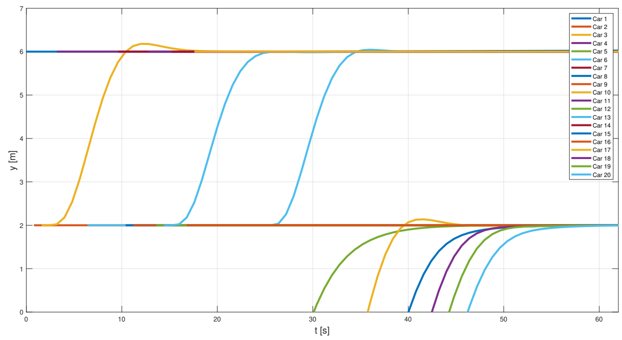
Figure 6-1: Manipulated variables and trajectories for low traffic scenario based on NMPC



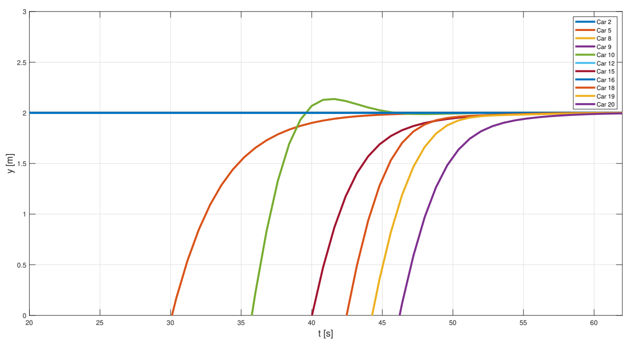
(a) Vehicular acceleration vs time plot



(b) Vehicular steering rate vs time plot

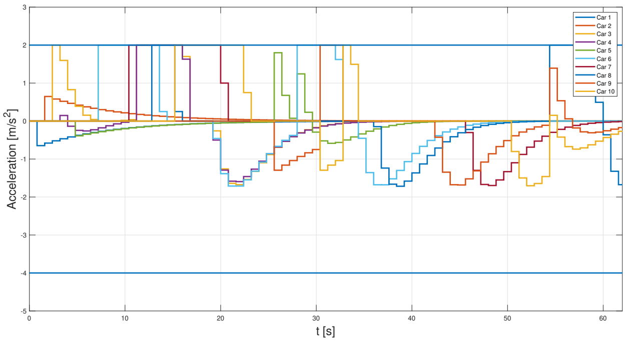


(c) Lateral trajectory between lanes 2 and on-ramp

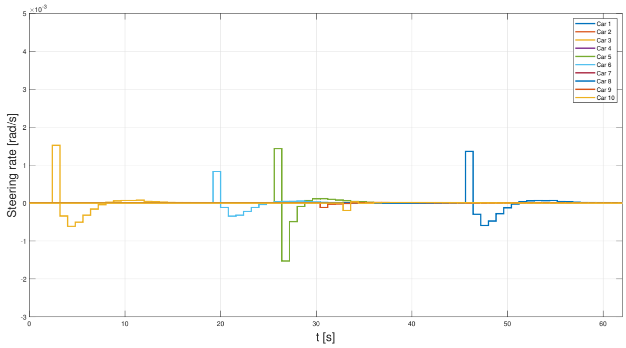


(d) Lateral trajectory between lanes 2 and on-ramp

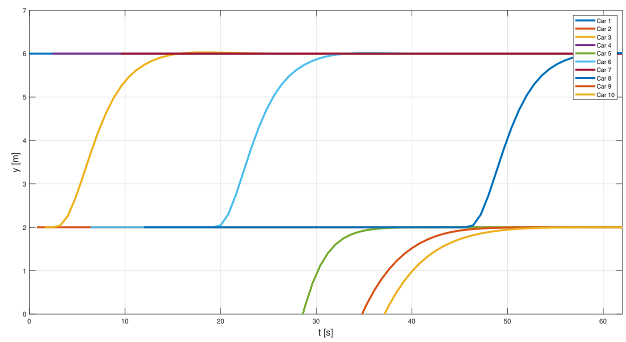
Figure 6-2: Manipulated variables and trajectories for high traffic scenario based on NMPC



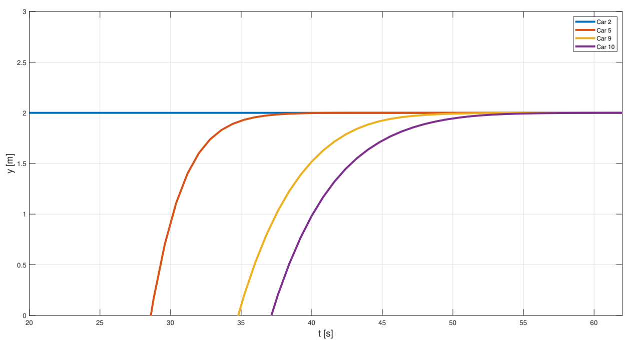
(a) Vehicular acceleration vs time plot



(b) Vehicular steering rate vs time plot

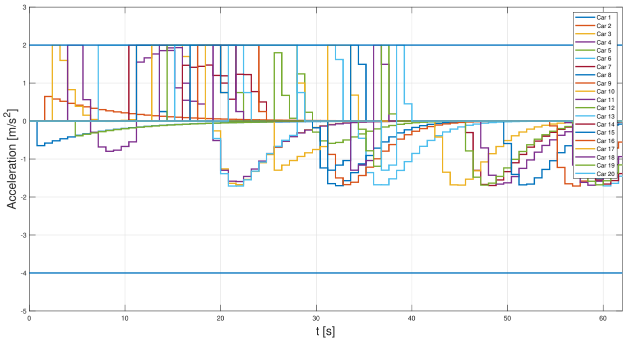


(c) Lateral trajectory between lanes 2 and on-ramp

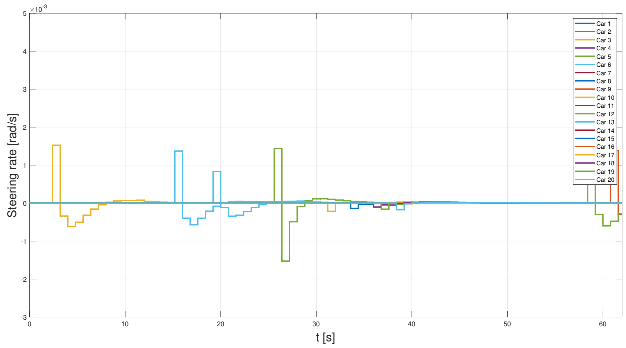


(d) Lateral trajectory between lanes 2 and on-ramp

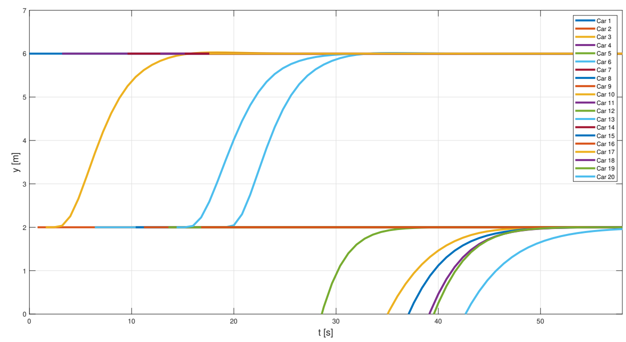
Figure 6-3: Manipulated variables and trajectories for low traffic scenario based on linear MPC



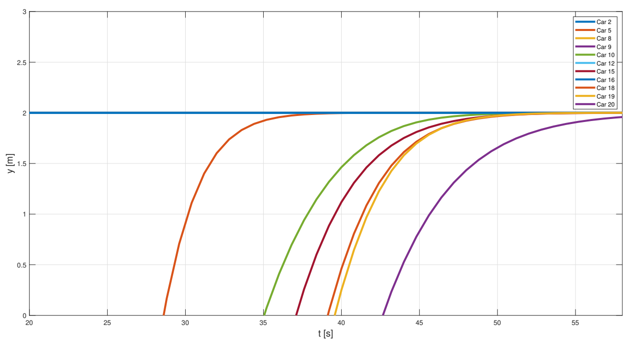
(a) Vehicular acceleration vs time plot



(b) Vehicular steering rate vs time plot



(c) Lateral trajectory between lanes 2 and on-ramp



(d) Lateral trajectory between lanes 2 and on-ramp

Figure 6-4: Manipulated variables and trajectories for high traffic scenario based on linear MPC

Bibliography

[1] O. Khalid, *Highway merging control strategies for autonomous driving*, Literature Study, Delft University of Technology, 2020.

[2] R. Scarinci, B. Heydecker, *Control Concepts for Facilitating Motorway On-ramp Merging Using Intelligent Vehicles*, Transport Reviews, 2014.

[3] J. Rios-Torres, and A. Malikopoulos, *A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps*, IEEE Transactions on Intelligent Transportation Systems, 2016.

[4] Z. Zheng, *Empirical Analysis on Relationship between Traffic Conditions and Crash Occurrences*, Procedia - Social and Behavioral Sciences, 43, 302-312. 2012.

[5] D. Chen, S. Ahn, *Capacity drop at extended bottlenecks: Merge, diverge, and weave*, Transportation Research Part B: Methodological, 108, 1-20, 2017.

[6] N. Hyldmar, Y. He, A. Prorok, *A Fleet of Miniature Cars for Experiments in Cooperative Driving*, International Conference on Robotics and Automation (ICRA), 2019.

[7] S. I. Guler, M. Menendez, L. Meier, *Using connected vehicle technology to improve the efficiency of intersections*, Transportation Research Part C: Emerging Technologies, 46, pp. 121-131, 2014.

[8] Y. Xie, H. Zhang, N. H. Gartnera, and T. Arsavac, *Collaborativemerging strategy for freeway ramp operations in a connected and autonomous vehicles environment*, Journal of Intelligent Transportation Systems, pp. 1-12, 2016.

[9] *http://www.nvidia.com/object/drive-px.html*

[10] *https://www.tesla.com/autopilot*

[11] *https://bit.ly/2QvC3DU*

[12] H. Liu, X. Kan, S. E. Shladover, X. Lu, and R. E. Ferlis, *Impact of cooperative adaptive cruise control on multilane freeway merge capacity*, Journal of Intelligent Transportation Systems, vol. 22 no. 2, 1-13, 2018.

[13] F. Chou, S. E. Shladover, X. Lu, and G. Bansal, *Coordinated Merge Control Based on V2V Communication*, IEEE Vehicular Networking Conference (VNC), 2016.

[14] P. Hidas, *Modeling vehicle interactions in microscopic simulation of merging and weaving*, Transportation Research Part C: Emerging Technologies, vol. 13, no. 1, 37-62, 2005.

[15] R. L. Bertini, and S. Malik, *Observed dynamic traffic features on freeway section with merges and diverges*, Transportation Research Record, 1867, 25-35, 2004.

[16] L. Leclercq, J. A. Laval and N. Chiabaut, *Capacity drops at merges: An endogenous model*, Transportation Research Part B: Methodological, 45, 1302-1313, 2011.

[17] M. Hunter, R. Machemehl and A. Tsyganov, *Operational evaluation of freeway ramp design*, Transportation Research Record: Journal of the Transportation Research Board, 1751, 90-100, 2001.

[18] I. Fukutome and K. Moskowitz, *Traffic behavior and on-ramp design*, Highway Research Board Bulletin, 235, 38-72, 1960.

[19] D. W. Harwood and J. M. Mason, *Ramp/mainline speed relationships and design considerations*, Transportation Research Record, 1385, 121-125, 1993.

[20] M. Papageorgiou, H. Hadj-Salem and J. M. Blosseville, *ALINEA:A local feedback control law for on-ramp metering*, Transportation Research Record, 1320, 58-64, 1991.

[21] M. Papageorgiou, and A. Kotsialos, *Freeway ramp metering: An overview*, IEEE Transactions on Intelligent Transportation Systems, 3, 271-281, 2002.

[22] Y. Qi, J. Liu, and Y. Wang. *Safety performance for freeway weaving segments*, Technical report, Texas Southern University, 2014.

[23] T. Nishi, P. Doshi and D. Prokhorov, *Freeway Merging in Congested Traffic based on Multipolicy Decision Making with Passive Actor Critic*, Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 2017.

[24] E. Camacho and C. Alba, *Model Predictive Control*, 2013.

[25] Z. Cong, *Efficient Optimization Methods for Freeway Management and Control*, PhD Thesis, 2015.

[26] P. H. L. Bovy, *Transportation modelling for tomorrow*, Delft University Press, 1996.

[27] P. I. Richards, *Shock waves on the highway*, Operations Research, 4:42-51, 1956.

[28] B. S. Kerner, *The Physics of Traffic: Understanding Complex Systems*, Springer, Berlin, Germany, 2004. ISBN: 978-3-540-20716-0

[29] M. Schonhof and D. Helbing, *Criticism of three-phase traffic theory*, Transportation Research, Part B-43, 784-797, 2009.

[30] S. Maerivoet and B. De Moor, *Cellular automata models of road traffic*, Physics Reports, 419(1): 1-64, November 2005.

[31] R. E. Chandler, R. Herman and E. W. Montroll, *Traffic dynamics: Studies in car-following*, Operations Research, 6(2), 165-184, 1958.

[32] P. G. Gipps *A behavioral car-following model for computer simulation*, Transportation Research, Part B: Methodological. 15(2), 105-111, 1981.

[33] M. Treiber, A. Hennecke and D. Helbing, *Congested traffic states in empirical observations and microscopic simulations*, Physical Review E: Statistical, Non-linear and Soft Matter Physics, 62(2), 1805-1824, 2000.

[34] A. Kesting, M. Treiber and D. Helbing, *General lane changing model MOBIL for car following models*, Transportation Research Record: Journal of Transportation Research Board, vol. 1999, 86-94, 2007.

[35] T. Toledo, H. N. Koutsopoulos and M. Ben-Akiva, *Integrated driving behavior modeling*, Transportation Research Part C: Emerging Technologies, 15(2), 96-112, 2007.

[36] W. J. Schakel, V. L. Knoop and B. van Arem, *Integrated lane change model with relaxation and synchronization*, Transportation Research Record: Journal of Transportation Research Board, 2316, 47-57, 2012.

[37] J. P. Lebacque, *The Godunov scheme and what it means for first order traffic flow models*, Proceedings of the 13th International Symposium on Transportation and Traffic Theory, pp. 647-678, 1996.

[38] H. J. Payne, *Models of freeway traffic and control*, Mathematical models of public systems, 28, 51-61, 1971.

[39] S. P. Hoogendoorn, and P. H. L. Bovy, *State-of-the-art of vehicular traffic flow modelling*, Proceedings of the Institution of Mechanical Engineers, Part 1: Journal of Systems and Control Engineering, 215(4):218-303, 2001.

[40] P. Falcone, F. Borrelli, H. E. Tseng, J. Asgari and D. Hrovat, *A Hierarchical Model Predictive Control Framework for Autonomous Ground Vehicles*, American Control Conference, 2008.

[41] Y. Gao, T. Lin, F. Borrelli, E. Tseng and Davor Hrovat, *Predictive Control of Autonomous Ground Vehicles With Obstacle Avoidance on Slippery Roads*, ASME Dynamic Systems and Control Conference, Vol.1, 2010.

[42] Y. Gao, *Model Predictive Control for Autonomous and Semiautonomous Vehicles*, PhD Thesis, University of California, Berkeley, 2014.

[43] J. Kong, M. Pfeiffer, G. Schildbach, F. Borrelli, *Kinematic and Dynamic Vehicle Models for Autonomous Driving Control Design*, IEEE Intelligent Vehicles Symposium (IV), 2015.

[44] A. Carvalho, S. Lefevre, G. Schildbach, J. Kong and F. Borrelli, *Automated driving: The role of forecasts and uncertainty - A control perspective*, European Journal of Control, pp. 14-32, 2015.

[45] R. Rajamani, *Vehicle Dynamics and Control*, Springer Science, 2006.

[46] A. Carvalho, Yiqi Gao, A. Gray, H. Tseng and F. Borrelli, *Predictive Control of an Autonomous Ground Vehicle using an Iterative Linearization Approach*, IEEE Annual Conference on Intelligent Transportation Systems, 2013.

[47] Y. Gao, *Model Predictive Control for Autonomous and Semiautonomous Vehicles*, PhD Thesis, University of California, Berkeley, 2014.

[48] A. Ferrara, S. Sacone, S. Siri, *Freeway Traffic Modelling and Control*, Advances in Industrial Control, ISBN 978-3-319-75961-6, 2018.

[49] A. Hegyi, T. Bellemans, and B. De Schutter, *Freeway traffic management and control*, in Encyclopedia of Complexity and Systems Science (R.A. Meyers, ed.), New York, New York: Springer, ISBN 978-0-387-30440-3, pp. 3943-3964, 2009.

[50] M. Papageorgiou, J. M. Blosseville, and H. Hadj-Salem, *Modelling and real-time control of traffic flow on the southern part of boulevard peripherique in Paris: Part I: Modelling*, In: Transportation Research A, vol. 24A, no. 5, pp. 345-359, 1990.

[51] A. Ollero and O. Amidi, "Predictive path tracking of mobile robots. Application to the CMU NavLab", International Conference on Advanced Robotics, vol. 2, pp. 1081-1086, 1991.

[52] K. Dresner and P. Stone, *Multiagent traffic management: A reservation based intersection control mechanism*, in Proc. 3rd Int. Joint Conf. Auton. Agents Multiagents Syst., pp. 530-537, 2004.

[53] S. Huang, A. W. Sadek, and Y. Zhao, *Assessing the mobility and environmental benefits of reservation-based intelligent intersections using an integrated simulator*, IEEE Trans. Intell. Transp. Syst., vol. 13, no. 3, pp. 1201-1214, Sep. 2012.

[54] C. Wuthishuwong and A. Traechtler, *Coordination of multiple autonomous intersections by using local neighborhood information*, in Proc. Int. Conf. Connect. Veh. Expo., 2013, pp. 48-53.

[55] Q. Jin, G. Wu, K. Boriboonsomsin, M. Barth, and S. Member, *Platoonbased multi-agent intersection management for connected vehicle*, in Proc. IEEE 16th Int. ITSC, 2013, pp. 1462-1467.

[56] G. K. Schmidt and B. Posch, *A Two-layer Control Scheme for Merging of Automated Vehicles*, 22nd IEEE Conference on Decision and Control, 1983.

[57] E. Onieva, V. Milanés, J. Villagrá, J. Pérez, and J. Godoy, *Genetic optimization of a vehicle fuzzy decision system for intersections*, Expert Syst. Appl., vol. 39, no. 18, pp. 13 148-13 157, Dec. 2012

[58] J. Alonso, V. Milanés, J. Pérez, E. Onieva, C. González, and T. de Pedro, *Autonomous vehicle control systems for safe crossroads*, Transp. Res. C, Emerg. Technol., vol. 19, no. 6, pp. 1095-1110, Dec. 2011.

[59] W. Wu, J. Zhang, A. Luo, and J. Cao, *Distributed mutual exclusion algorithms for intersection traffic control*, IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 1, pp. 65-74, Jan. 2015.

[60] J. Khoury and J. Khoury, *Passive, decentralized, and fully autonomous intersection access control*, in Proc. IEEE 17th Int. Conf. Intell. Transp. Syst., pp. 3028–3033, 2014.

[61] L. Li and F.-Y. Wang, *Cooperative driving at blind crossings using inter-vehicle communication*, IEEE Trans. Veh. Technol., vol. 55, no. 6, pp. 1712, 1724, Nov. 2006.

[62] F. Yan, M. Dridi, and A. El Moudni, *Autonomous vehicle sequencing algorithm at isolated intersections*, in Proc. IEEE 12th Int. Conf. Intell. Transp. Syst., pp. 1-6, 2009.

[63] J. Wu, F. Perronnet, and A. Abbas-Turki, *Cooperative vehicle-actuator system: A sequence-based framework of cooperative intersections management*, Intell. Transp. Syst., IET, vol. 8, no. 4, pp. 352-360, 2014.

[64] J. Lee and B. Park, *Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment*, IEEE Trans. Intell. Transp. Syst., vol. 13, no. 1, pp. 81-90, Mar. 2012.

[65] J. Lee, B. Park, K. Malakorn, and J. So, *Sustainability assessments of cooperative vehicle intersection control at an urban corridor*, Transp. Res. C, Emerg. Technol., vol. 32, pp. 193-206, Jul. 2013.

[66] G. R. De Campos, P. Falcone, and J. Sjoberg, *Autonomous cooperative driving: A velocity-based negotiation approach for intersection crossing*, in Proc. IEEE 16th Int. ITSC, pp. 1456-1461, 2013.

[67] M. A. S. Kamal, J. Imura, T. Hayakawa, A. Ohata, and K. Aihara, *A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights*, IEEE Trans. Intell. Transp. Syst., vol. 16, no. 3, pp. 1136-1147, Jun. 2015.

[68] L. Makarem and D. Gillet, *Fluent coordination of autonomous vehicles at intersections*, in Proc. IEEE Int. Conf. Syst. Man, Cybern., pp. 2557-2562, 2012.

[69] W. Cao, M. Mukai, T. Kawabe, H. Nishira, and N. Fujiki, *Gap selection and path generation during merging maneuver of automobile using realtime optimization*, SICE J. Control. Meas. Syst. Integr., vol. 7, no. 4, pp. 227-236, 2014.

[70] B. Ran, S. Leight, and B. Chang, *A microscopic simulation model for merging control on a dedicated-lane automated highway system*, Transp. Res. C Emerg. Technol., vol. 7, no. 6, pp. 369-388, 1999.

[71] Marinescu, D. *On-ramp traffic merging using cooperative intelligent vehicles: a slot-based approach*, IEEE Conference on Intelligent Transportation Systems, 2012.

[72] A. Uno, T. Sakaguchi, and S. Tsugawa, *A merging control algorithm based on inter-vehicle communication*, Proc. IEEE/IEEJ/JSAI Int. Conf. Intell. Transp. Syst. (Cat. No. 99TH8383), pp. 783-787, 1999.

[73] X. Lu, H. Tan, S. Shladover and K. J. Hedrick, *Automated Vehicle Merging Maneuver Implementation for AHS*, Vehicle System Dynamics, vol. 41, no. 2, pp. 85-107, Jan. 2004.

[74] J. Milanes, V. Godoy, J. Villagra, and J. Perez, *Automated on-ramp merging system for congested traffic situations*, IEEE Trans. Intell. Transp. Syst., vol. 12, no. 2, pp. 500-508, Jun. 2011.

[75] J. Milanes, J. Pérez and E. Onieva, *Controller for urban intersections based on wireless communications and fuzzy logic*, IEEE Trans. Intell. Transp. Syst., vol. 11, no. 1, pp. 243-248, Mar. 2010.

[76] G. Raravi, V. Shingde, K. Ramamritham and J. Bharadia, *Merge algorithms for intelligent vehicles*, Next Generation Design and Verification Methodologies for Distributed Embedded Control Systems pp. 51-65, 2007.

[77] J. Rios-Torres, et al., *Automated and Cooperative Vehicle Merging at Highway On-Ramps*, IEEE Transactions on Intelligent Transportation Systems, vol. 18, 2017.

[78] L. D. Baskar, *Traffic Management and Control in Intelligent Vehicle Highway Systems*, PhD Thesis, Delft University of Technology, 2009.

[79] P. Kachroo and Z. Li, *Vehicle merging control design for an automated highway system*, IEEE Conference on Intelligent Transportation Systems, pp. 224-229, 1997.

[80] Jay D. Schwartz, and Daniel E. Rivera *Simulation-Based Optimal Tuning of Model Predictive Control Policies for Supply Chain Management using Simultaneous Perturbation Stochastic Approximation*, Proceedings of the 2006 American Control Conference Minneapolis, Minnesota, USA, June 14-16, 2006

[81] R. Pueboobpaphan, F. Liu, and B. Arem, *The Impacts of a Communication based Merging Assistant on Traffic Flows of Manual and Equipped Vehicles at an On-ramp Using Traffic Flow Simulation*, 13th International IEEE Annual Conference on Intelligent Transportation Systems, 2010.

[82] J. Wei, J. M. Dolan and B. Litkouhi, *Autonomous Vehicle Social Behavior for Highway Entrance Ramp Management*, IEEE Intelligent Vehicles Symposium (IV), 2013.

[83] T, Awal, L. Kulik and K. Ramamohanrao, *Optimal traffic merging strategy for communication- and sensor-enabled vehicles*, IEEE Annual Conference on Intelligent Transportation Systems, 2013.

[84] Z. Wang, G. Wu and M. Barth, *Distributed Consensus-Based Cooperative Highway On-Ramp Merging Using V2X Communications*, SAE Technical Paper, doi:10.4271/2018-01-1177, 2018.

[85] A. Assad, et al., *Heavy-Duty Vehicle Platooning for Sustainable Freight Transportation: A Cooperative Method to Enhance Safety and Efficiency*, IEEE Control Systems, vol. 35, 2015.

[86] Lu, X. et al., *Automated Vehicle Merging Maneuver Implementation for AHS*, Vehicle System Dynamics, vol. 41, no. 2, pp. 85-107, 2004.

[87] N. Chen, M. Wang, T. Alkin, and B. Van Arem, *A Flexible Strategy for Efficient Merging Maneuvers of Connected Automated Vehicles*, Proceedings of the 18th COTA International Conference of Transportation Professionals. American Society of Civil Engineers (ASCE), pp. 46-55, 2018.

[88] Saimee, et al., *Towards a Decision-Making Algorithm for Automatic Lane Change Manoeuvre Considering Traffic Dynamics*, PROMET – Traffic and Transportation, 28(2), 2015.

[89] J. Rios-Torres, et al., *Automated and Cooperative Vehicle Merging at Highway On-Ramps*, IEEE Transactions on Intelligent Transportation Systems, Vol. 18, 2017.

[90] Z. Zhao, G. Wu, Z. Wang, and M. J. Barth, *Optimal Control-Based Ramp Merging System for Connected and Automated Electric Vehicles*, arXiv preprint, 2019.

[91] Z. Zhao, G. Wu, Z. Wang, F. Ye, and M. J. Barth, *The State-of-the-Art of Coordinated Ramp Control with Mixed Traffic Conditions*, IEEE Intelligent Transportation Systems Conference (ITSC), 2019.

[92] I. H. Zohdy and H. Rakha, *Game theory algorithm for intersection-based Cooperative Adaptive Cruise Control (CACC) systems*, in Proc. IEEE 15th Int. Conf. Intell. Transp. Syst., Sep. 2012, pp. 1097–1102.

[93] M. Wang, S. P. Hoogendoorn, W. Daamen, B. van Arem, and R. Happee, *Game theoretic approach for predictive lane-changing and car-following control*, Transportation Research Part C: Emerging Technologies, vol. 58, pp. 73-92, 2015.

[94] Z. Wang, G. Wu, and M. Barth, *Distributed Consensus-Based Cooperative Highway On-Ramp Merging Using V2X Communications*, SAE Technical Paper 2018-01-1177, 2018, doi:10.4271/2018-01-1177.

[95] I. Ntousakis, K. Porfyri, I. Nikolos and M. Papageorgiou, *Assessing the Impact of a Cooperative Merging System on Highway Traffic Using a Microscopic Flow Simulator*, Proceedings of the ASME International Mechanical Engineering Congress and Exposition, 2014.

[96] M. Antoniotti, A. Desphande and A. Girault, *Microsimulation analysis of multiple merge junctions under autonomous AHS operation*, Proceedings of Conference on Intelligent Transportation Systems, 2002.

[97] P. G. Gipps, *A model for the structure of lane-changing decisions*, Transportation Research Part B: Methodological vol. 20(5), 403-414, 1986.

[98] P. Hidas, *Modelling vehicle interactions in microscopic simulation of merging and weaving*, Transportation Research Part C 13, 37-62. doi: 10.1016/j.trc.2004.12.003, 2005.

[99] C. F. Choudhury, *Modeling driving decisions with latent plans*, Massachusetts Institute of Technology.

[100] A. Kesting, M. Treiber, and D. Helbing, *General Lane-Changing Model MOBIL for Car-Following Models*, Transportation Research Record: Journal of the Transportation Research Board, no. 1999, 86-94. doi: 10.3141/1999-10, 2007.

[101] W. J. Schakel, V.L. Knoop, and B.v. Arem, *LMRS: An integrated lane change model with relaxation and synchronization*, 2012.

[102] A. S. Polydoros and L. Nalpantidis, *Survey of Model-Based Reinforcement Learning: Applications on Robotics* Journal of Intelligent and Robotic Systems, 86(2), 153-173, 2017.

[103] L. Lin, *Reinforcement learning for robots using neural networks*, Carnegie Mellon University, 1993.

[104] R. S. Sutton and A.G. Barto, *Introduction to reinforcement learning*, Vol. 2, No. 4, MIT Press Cambridge.

[105] D. Ngai, N. Yung, *Automated vehicle overtaking based on a multiple goal reinforcement learning framework*, Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference, USA, 2007.

[106] A. Yu, R. Palefsky-Smith, R. Bedi, *Deep reinforcement learning for simulated autonomous vehicle control*, Stanford University. StuDocu. 2016.

[107] A. Sallab, M. Abdou, E. Perot, S. Yogamani, *End-to-end deep reinforcement learning for lane keeping Assistance*, 30th Conference on Neural Information Processing Systems (NIPS), Barcelona, Spain, 2016.

[108] S. Shalev-Shwartz, S. Shammah, A. Shashua, *Safe, multi-agent, reinforcement learning for autonomous Driving*, arXiv:1610.03295v1, 2016.

[109] J. C. Medina, *Agent-based Traffic Management and Reinforcement Learning in Congested Intersection Network*, PhD Thesis, University of Illinois at Urbana-Champaign, 2012.

[110] V. Mnih, K. Kavukcuoglu, et al. *Human-level control through deep reinforcement learning*, Nature, 518(7540):529-533, 2015.

[111] Y. Zhu et al., *Reinforcement and Imitation Learning for Diverse Visuomotor Skills*, ArXiv180209564 Cs, Feb. 2018.

[112] M. Mukadam, A. Cosgun, A. Nakhaei and K. Fujimura, *Tactical Decision Making for Lane Changing with Deep Reinforcement Learning*, 31st Conference on Neural Information Processing Systems (NIPS), USA, 2017.

[113] P. Wang, C. Chan, *Formulation of Deep Reinforcement Learning Architecture Toward Autonomous Driving for On-Ramp Merge*, IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), 2017.

[114] P. Wang, C. Chan, *Autonomous Ramp Merge Maneuver Based on Reinforcement Learning with Continuous Action Space*, arXiv preprint, 2018.

[115] P. Wang, C. Chan, A. Fortelle, *A Reinforcement Learning Based Approach for Automated Lane Change Maneuvers*, IEEE Intelligent Vehicles Symposium (IV), 2018.

[116] T. Shi, P. Wang, X. Cheng, C. Chan, *Driving Decision and Control for Autonomous Lane Change based on Deep Reinforcement Learning*, arXiv preprint, 2019.

[117] T. Nishi, P. Doshi and D. Prokhorov, *Freeway Merging in Congested Traffic based on Multipolicy Decision Making with Passive Actor Critic*, Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 2017.

[118] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, *Mastering the game of go without human knowledge*, Nature, 550 (7676), 354, 2017.

[119] J. Kober, J. A. Bagnell, and J. Peters, *Reinforcement learning in robotics: A survey*, The International Journal of Robotics Research, 32 (11), 1238-1274, 2013.

[120] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, Dan Mane, *Concrete Problems in AI Safety*, CoRR abs/1606.06565. URL http://arxiv.org/abs/1606. 06565.

[121] E. Camacho and C. Alba, *Model predictive control*, 2013.

[122] T. Shim, G. Adireddy, and H. Yuan, *Autonomous vehicle collision avoidance system using path planning and model-predictive-control-based active front steering and wheel torque control*, Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, vol. 226, no. 6, pp. 767-778, 2012.

[123] M. Kelly, *An Introduction to Trajectory Optimization: How to Do Your Own Direct Collocation*, SIAM Review, Society for Industrial and Applied Mathematics, Vol. 59, No. 4, pp. 849-904, 2017.

[124] S. Moon and K. Yi, “Human driving data-based design of a vehicle adaptive cruise control algorithm”, Vehicle System Dynamics, vol. 46, no. 8, pp. 661-690, 2008.

[125] “[https://www.mathworks.com/help/optim/ug/choosing-the-algorithm.htmlbrppuoz]”

[126] P. T. Boggs, W. J. Tolle, and J. W. Tolle, “Sequential Quadratic Programming,” Acta Numerica, vol. 4, no. November 2008, pp. 1-52, 1996.

[127] K. Han, G. Park, G.S. Sankar, K. Nam, and S.B. Choi, "Model Predictive Control Framework for Improving Vehicle Cornering Performance Using Handling Characteristics", IEEE Transactions on Intelligent Transportation Systems, pp. 1-11, 2020.

[128] F. Borrelli, P. Falcone, T. Keviczky, J. Asgari, and D. Hrovat "MPC-Based Approach to Active Steering for Autonomous Vehicle Systems", International Journal on Vehicle Autonomous Systems Vol. 3, pp. 265-291, 2005.

[129] S. Liu, A. Sadowska, J. R. D. Frejo, and A. Núñez, E. F. Camacho, J. Hellendoorn, and B. De Schutter, “Robust receding horizon parameterized control for multi-class freeway networks: A tractable scenario-based approach,”International Journal of Robust and Nonlinear Control, vol. 26, no. 6, pp. 1211–1245, 2016.

Glossary

List of Acronyms

DCSC	Delft Center for Systems and Control
DSRC	Dedicated Short-Range Communication
RSU	Road Side Unit
ATM	Active Traffic Management
TMS	Traffic Management System
CACC	Cooperative Adaptive Cruise Control
CAVs	Connected and Autonomous Vehicles
MPC	Model Predictive Control
NMPC	Nonlinear Model Predictive Control
DMC	Dynamic Matrix Control
RHC	Receding Horizon Control
V2V	Vehicle-to-Vehicle
V2I	Vehicle-to-Infrastructure
VSL	Variable Speed Limit
COL	Crash Occurrence Likelihood
CACC	Cooperative Adaptive Cruise Control
SQP	Sequential Quadratic Programming