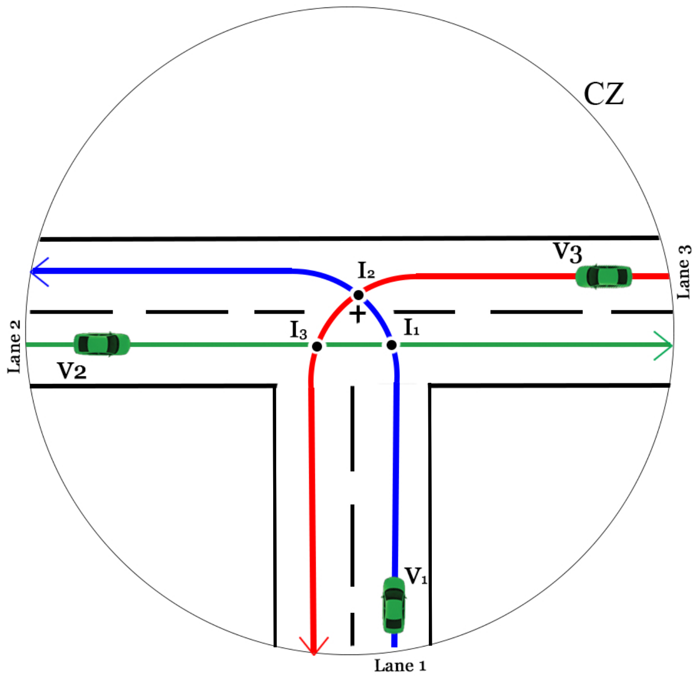
CHALMERS
UNIVERSITY OF TECHNOLOGY



Cooperative Collision Avoidance Strategies at Intersections

Master's thesis in Systems, Control and Mechatronics Programme

Arun Shenbaga Priyan Vijayan

Department of Signals and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2016

Master's thesis EX099/2016

Cooperative Collision Avoidance Strategies at Intersections

Arun Shenbaga Priyan Vijayan

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Signals and Systems
Mechatronics Research Group
Chalmers University of Technology
Gothenburg, Sweden 2016

Cooperative Collision Avoidance Strategies at Intersections

Arun Shenbaga Priyan Vijayan

Supervisor and Examiner: Paolo Falcone, Department of Signals and Systems

Master's Thesis EX099/2016
Department of Signals and Systems
Mechatronics Research Group
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Image in the cover was an Intersection Scenario Layout described in Figure 2.10 from Section 2.4.

Typeset in LaTeX
Gothenburg, Sweden 2016

Cooperative Collision Avoidance Strategies at Intersections
Arun Shenbaga Priyan Vijayan
Department of Signals and Systems
Chalmers University of Technology

Abstract

'Cooperative Driving' in autonomous vehicles is the capability of vehicles to drive by itself while communicating with other vehicles and road infrastructure. Cooperative Driving is the next level of challenge in autonomous cars. Maintaining a safety distance from preceding vehicle, platooning, overtaking, intersection crossing are some of the self organizing behaviours that can be achieved with cooperative driving. Collision avoidance is a major benefit of cooperative driving while also adding to reduction in total travel time, reduced aggregate fuel consumption etc. In this thesis the problem of cooperative driving in intersections while avoiding collision and driving as fast as possible are the major objectives that are solved. Control of longitudinal states in a vehicle can ensure these objectives of collision avoidance and fastest finish time. Model Predictive Control (MPC) framework is applied for control of longitudinal states, since it fulfills the requirement of imposing time varying constraints on the states, prediction of states and attaining the desired set value on the states.

Keywords: Cooperative Driving, Intelligent Transportation Systems, Collision Avoidance, Longitudinal Control, Model Predictive Control, Hardware In Loop Simulation, GCDC

Acknowledgements

I would like to thank Professor Paolo Falcone for providing this exciting opportunity to do a master thesis for one year academic period. His genius guidance and technical expertise helped to me solve the problems at a faster pace in both theoretical formulation and practical implementation parts of the project. I am greatly pleased by his decision for letting me showcase the master thesis work in a medium level research competition, Grand Cooperative Driving Challenge (GCDC). He was also instrumental in making sure that I get right help, at right moment from other students throughout the course of thesis. I am glad to thank Robert Hult for this timely help and guidance on designing and implementing the controller in simulation as well as in real car. Robert Hult and Marco Di Vaio's work on ego and participant sensor fusion for the real car reduced a longer idle time (of mine) during the thesis. During GCDC work, I was really amazed by the leadership and team building skills showcased by Robert, which motivated me deliver my part in GCDC project seamlessly. I would like to thank Albin Severinson for helping out in the communication part of the project and providing a fully functional communication system earlier in the time line of the project. Albin's guidance on technical aspect of communication module eased the load on me and let me focus less on communication and more on controller part. I would like to thank Marco Di Vaio for his extended support in sensor fusion of the vehicles in intersection scenario and also helping out in testing the controller in real car. The support that I got in driving the car reduced my handicap in driving and also feel little less tensed while testing the controllers in real car. In this aspect, I would also like to thank Jacob Brickner, Jasper Spanjaards and Lars Sjöberg for rendering their help on testing on real roads.

I am extremely grateful to my parents and sister for supporting me to pursue a master's degree in Sweden. I would like to mention the help rendered by my flat mates, friends in MPSYS and master thesis room who motivated me from time to time in the past one year.

Arun Shenbaga Priyan Vijayan, Gothenburg, October 2016

Contents

List of Figures

List of Tables

1
Introduction

Cooperative driving helps a group of autonomous vehicles to solve complex traffic situations, by their coordination. This thesis focuses on cooperative driving in an intersection scenario by solving the longitudinal control problem. The longitudinal controller for an autonomous vehicle in an intersection scenario will control its position, speed and acceleration so that collisions between vehicles inside the intersection area are avoided and the manoeuvre is carried out as fast as possible. Design, implementation, testing of a longitudinal controller for the intersection scenario is the primary aim of this thesis. Secondly, this thesis provided solution for the longitudinal control in the intersection crossing scenario, for the **Chalmers Car** team's Volvo S60 semi-autonomous vehicle which participated in Grand Cooperative Driving Challenge (GCDC)-2016. The solution proposed is also applicable to an intersection infrastructure of higher complexity, say for example an intersection of any geometry including a four way intersection. The test results of the longitudinal controller implemented in virtual vehicles and later on in a real vehicle are presented in this thesis report. The performance of the longitudinal controller in the Volvo S60 at GCDC intersection scenario is presented where other autonomous vehicles in the vicinity communicate with our car, demonstrating a cooperative behaviour.

1.1 Problem Statement

The problem considered in this thesis is to develop and test the longitudinal control algorithm for an autonomous vehicle in a T-intersection infrastructure like in the streets of an urban area or in the roads of the countryside. An intersection crossing scenario from Grand Cooperative Driving Challenge (GCDC) is as shown in Figure 1.1

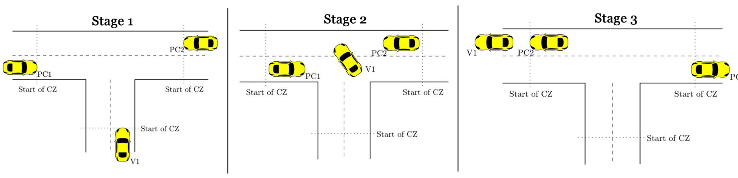


Figure 1.1: Intersection Crossing Scenario (Courtesy : GCDC)

There are three autonomous vehicles cooperating via wireless communication approaching the center of an intersection. Since these are cooperative vehicles they can communicate and exchange their intentions and other vital information required for cooperative driving. The vehicles should decide on a common understanding among themselves so that collision is avoided between them inside the intersection and all the vehicles exit the intersection at maximum speed possible. The solution for fulfilling the objective of a vehicle in such a intersection is by longitudinal control of the vehicle. Some parameters (e.g) maximum velocity of a vehicle inside intersection zone, minimum safety distance to be maintained between vehicles are acquired from the GCDC competition rules and regulations.

1.2 Literature Review

This section on literature review describes the articles, projects and papers which were aiding this thesis. Samuel Scheidegger, Martin Holder, Mathias Ernst, Paolo Falcone solved the problem of vehicle control for the GCDC intersection scenario as a scheduling problem and presented the simulation results of the scenario with virtual vehicles in [16]. For this thesis, their work helped in understanding the influence of longitudinal control of an autonomous vehicle in intersection scenario and also proposed a methodology to use the GPS data of the real world in intersection scenario simulation. Robert Hult, Gabriel R.Campos, Paolo Falcone et. al in [15] demonstrated the solution for optimal coordination of autonomous vehicles in an intersection scenario considering communication in the loop. This paper showed the usage of information about vehicles communicated through wireless network, in vital functions of autonomous vehicle (e.g) intersection coordination, collision avoidance. Technical requirements and judging criteria documents [2], [6] and [8] were released by the GCDC organisers. They describe the safety, speed limits, judging criteria and other important parameters required for an autonomous vehicle participating in the intersection scenario. The documents released by GCDC organisers on intersection coordination problem does not constraint an autonomous vehicle only to competition scenario of T-intersection but can also be applied to intersection scenario of higher complexity. Gabriel R.Campos, Paolo Falcone, Jonas Sjöberg in [14] applied a receding horizon control strategy to solve the problem of intersection crossing. A similar approach of using receding horizon control, with change in cost function and introduction of time varying constraints are used in this thesis. In [19] Paolo Falcone et al demonstrated testing of a vehicle controller or an active safety functionality with an autonomous vehicle as a hardware in the loop while virtually generated autonomous vehicles are present in the simulation. Their work was developed further by insertion of real world map data into the simulation environment and was used in testing the longitudinal intersection control on virtual vehicles. This work laid a foundation for using HIL simulation testing for the intersection scenario.

In [17] Paolo Falcone, Lei Ni applied MPC for controlling the lateral states of an autonomous vehicle and drive a vehicle along a predefined path expressed in terms of a polynomial function. In this thesis lateral control was implemented only on virtual vehicles inside simulation environment and follow the predefined path of continuous GPS points.

1.3 Chapters and their Contents

The chapters in the thesis and the aspect covered in them are shown in the table below

Table 1.1: Chapter and its contents

Chapter	Contents
Chapter 2	Basic principle of MPC and it's application for longitudinal control
Chapter 3	Problems in driving cooperatively in an intersection and solutions proposed
Chapter 4	Formulation of longitudinal controller for a vehicle inside intersection Zone
Chapter 5	Description of simulation environment, for testing the longitudinal intersection controller
Chapter 6	Discussion of simulation results
Chapter 7	Discussion of HIL simulation results
Chapter 8	GCDC competition results presentation
Chapter 9	Conclusion

2

Cooperative Driving in Intersections

Cooperative driving in intersections, where there are no traffic signals, has problems of *Vehicle Prioritization*, *Vehicle Characteristics* assignment to be solved among the vehicles, before a controlled behaviour is achieved. This chapter investigates the demand from autonomous vehicles in the intersections in terms of *Performance Measures* and *Safety Requirements* in order to achieve effective cooperative behaviour. The stages of problem solving in intersection crossing like *Vehicle Prioritization*, *Vehicle Characteristics* assignment for the vehicles are done based on the *Performance Measures* and *Safety Requirements*. The intersection considered for the study is a closed test region which is free from external disturbances like other vehicles (other than participants), road obstacles, objects on road or people. A T-intersection as in the Figure 2.1 is the chosen road layout for further study and implementation of control algorithms in this thesis. The intersections considered in our study has a secondary road joining a primary/main road. The secondary road is recognised as *Lane 1*, and one lane of the primary road is named as *Lane 2* and the other side is named as *Lane 3* as in Figure 2.1. There are three autonomous vehicles, one vehicle in every lane approaching the center of intersection.

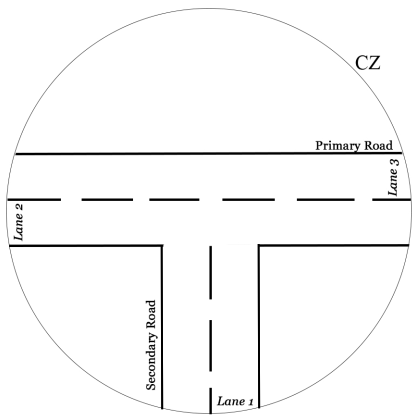


Figure 2.1: T-Intersection

The challenges in cooperative driving in an intersection does not vary with respect to the geometry of the intersection, so the cooperative driving strategies derived for an autonomous vehicle in a T-intersection will be valid for a four way intersection (and also more complex geometries) or even vehicle follow trajectories different from GCDC. The solution proposed in this thesis is primarily applied and validated experimentally in the GCDC intersection scenario and also applied to two other T-intersection scenarios in which the vehicles intend to follow different trajectories which are named **Example Scenario 2** and **Example Scenario 3**. Vehicle Prioritization and Vehicle Characteristics assignment are applied in the scenarios declared above for better understanding.

2.1 Performance Measures

An autonomous vehicle crossing an intersection cooperatively is expected to fulfill some commonly listed performance measures, for all the vehicles in the intersection. These performance measures are listed such that the cooperative vehicles exhibit better characteristics as compared to conventional crossing done by human drivers. The features that are expected from the cooperative autonomous vehicles are

- **Faster Crossing** - the vehicle should cross the intersection at a **faster rate** than a human driver (or) travel at a speed closer to the **maximum speed** allowed in that intersection. The cooperative driving vehicle should maximise its speed in crossing the intersection, when compared to a human driver.
- **Fuel** - travelling at maximum allowed speed limit, while expending minimum fuel as possible. Expending minimum fuel can be translate to expending **minimum acceleration** as possible.
- **Comfort - ensuring comfort** of the passengers inside the vehicle by avoiding any sudden jerks due to faster acceleration and deceleration changes in fulfilling the expectations on faster crossing.

2.2 Safety Requirements

Safety is a critical requirement from cooperative driving autonomous vehicles, when they are allowed to carry out an action by deciding among themselves. The safety demands from vehicles in an intersection scenario would be

- **Safety Region** Non-violation of safety distances between vehicles. In short a region around an autonomous vehicle where the other vehicles are not allowed to enter.The safety region is a precaution to avoid collision between vehicles in the intersection zone.
- **Dynamics Limits** Avoid exceeding the maximum speed limit allowed for a vehicle in the intersection zone.

2.3 GCDC Intersection Scenario

Grand Cooperative Driving Challenge (GCDC) is a competition for demonstrating the cooperative automation of vehicle driving, with communication established between vehicle to vehicle (V2V) and vehicle to infrastructure (V2X). GCDC 2016 is conducted by i-Game project which is supported by European Commission [1]. GCDC 2016 was held at Helmond, Netherlands. Competitors from various institutes and universities participated in GCDC, with their own autonomous vehicles abiding to the Requirements of GCDC [9]. There are three scenarios - Platooning Scenario, Intersection Scenario and Emergency Vehicle Scenario. In GCDC intersection scenario, a T-intersection is considered where a secondary road meets with a primary/main road. There are three lanes one in secondary and one in either side of the primary road as shown in Figure 2.2 which are named as *Lane 1*, *Lane 2* and *Lane 3*.

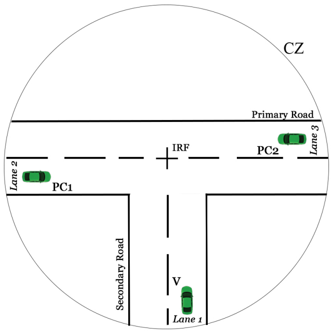


Figure 2.2: GCDC Intersection Scenario

2.3.1 Objectives

The intention of vehicle V is to take a *left* turn, vehicle PC_1 and PC_2 and intend to go *straight* in their respective lanes as in Figure 2.2. Vehicle V will be operated by GCDC organisers. PC_1 will be the vehicle of one of the competitors and PC_2 will be the vehicle of one another competitor. The sequence of events specified for the intersection scenario by GCDC Organisers as in [2] are

1. The front of the three vehicles V, $PC1$ and $PC2$ should enter the competing zone CZ within a certain time window. This time window was 20 seconds from the instant the start signal is provided via V2X hardware.
2. All the three vehicles should enter the CZ at the maximum speed limit of 30 *kmph*. This is the speed limit of the vehicles throughout the scenario.
3. All the three vehicles start communicating (V2V) after entering into CZ, there will be no communications between vehicles before entering CZ.
4. Vehicles $PC1$ and $PC2$ should allow vehicle V to enter into the primary road from the secondary road.
5. Vehicles $PC1$ and $PC2$ should maintain a safety distance from vehicle V, during the intersection crossing. This is part of the *safety requirements* in cooperative driving which are discussed in the following sections along with few other requirements.

6. When vehicle V has entered the primary road (or crossed *End of evaluation* line), all three vehicles should leave CZ at the earliest time possible.

2.3.2 Judging Criteria

Judging Criteria document of GCDC [8] was satisfying the **Performance Measures** and **Safety Requirements** described in this thesis. The participants in the intersection scenario are judged based on individual judging criteria, which are

1. Minimum Euclidean Distance (Safety Requirement)
2. Desired Distance (Safety Requirement)
3. Fastest Finish Time (Performance Measure)
4. Maximum Speed Limit (Safety Requirement)

The **score** for a competitor vehicle in the intersection scenario of GCDC is the average of the score secured by the vehicle in individual judging criteria.

2.3.2.1 Minimum Euclidean Distance

The *minimum Euclidean distance* judging criteria evaluates a competitor vehicle in position PC_1, when the vehicle V circle enters the *start of CZ / start of evaluation* until the vehicle V circle is out of the *End of evaluation* line as shown in Figure 2.3. The circle for vehicles are defined in such a way that the length and breadth of the vehicle is lying within the circle itself.

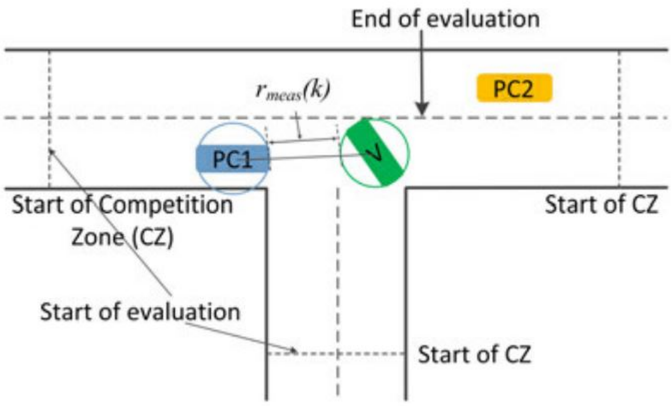


Figure 2.3: Minimum Euclidean Distance V and PC_1 (www.gcdc.net)

A competitor's vehicle in PC_1 position should not violate the minimum euclidean distance r_{euc} from the GCDC organiser's vehicle V

Error Computation

For a competitor vehicle PC_1, the error $e_i(k)$ is the measure of violation of *minimum Euclidean distance* judging criteria. The error $e_i(k)$ is the difference between the measured euclidean distance at time instant k and a constant minimum euclidean distance r_{euc}, which is given by

$$e_i(k) = r_{meas}(k) - r_{euc} \tag{2.1}$$

The constant minimum Euclidean distance, target value provided by the organisers during the competition was 15 m. $r_{euc} = 15\ m$.

Heat Score

As long as the error value $e_i(k)$ is positive in a time instant k, a competitor vehicle gets a maximum score of 10. But when the error value is negative a competitor's score is penalised by a scoring function Q_1 as shown in Figure 2.4.

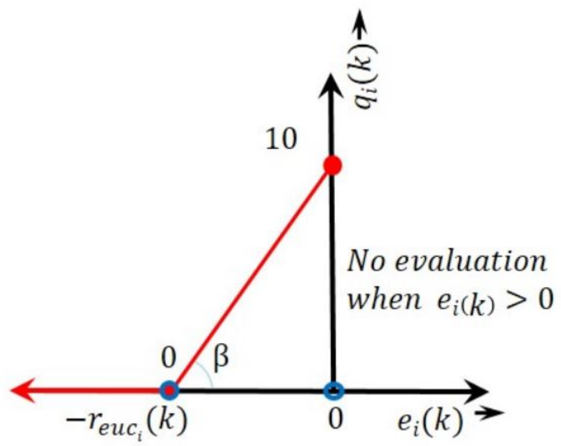


Figure 2.4: Q_1 Scoring Function (www.gcdc.net)

In the Figure 2.4 $q_i(k)$ is the score at the time instant k, which depicts that at k time instant if $e_i(k) \geq 0$, $q_i(k) = 10$. At k time instant if $e_i(k) < 0$, value of $q_i(k)$ is determined by the red line. $q_i(k)$ is the score of the competitor at time instant k. $q_i(k)$ gets smaller with smaller $e_i(k)$.
If K denotes the set of time instants k for which a competitor vehicle has violated the minimum euclidean distance or $e_i(k)$ is negative between the *start of CZ* and *End of evaluation*, then M denotes the size of K. The heat score $s_i(k)$ for a competitor vehicle i in heat h is calculated as

$$s_i(h) = \begin{cases} \min\limits_{k \in K} Q_1(e_i(k)), & M > 0 \\ Q_1(0), & M = 0 \end{cases} \tag{2.2}$$

If H is the number of heats for the vehicle i in the same position as PC_1, the average of the scores $score_{ind_i}(c)$ is taken as the final individual score of vehicle i for the criteria c

$$score_{ind_i}(c) = \frac{\sum\limits_{h=1}^{H} s_i(h)}{H} \tag{2.3}$$

Here c is the minimum euclidean distance criteria. $score_{ind_i}(c)$ is the final score of the vehicle i.

2.3.2.2 Desired Distance

The *desired distance* criteria is an individual evaluation criterion for a vehicle in the *platooning scenario* of GCDC, the same criterion applies for a competitor vehicle in PC_2 position in an intersection scenario. The desired distance criterion for a competitor vehicle in position PC_2 is as shown in Figure 2.5.

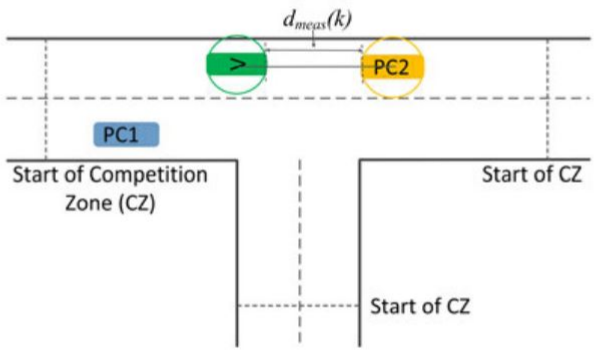


Figure 2.5: Desired Distance between V and PC_2 (www.gcdc.net)

After the *End of evaluation* line of vehicle V and until the PC_2's *Exit of CZ*, both the vehicles are in a stature similar to the highway platooning scenario. According to the desired distance criterion in the intersection scenario, the front of vehicle PC_2 should maintain a distance d_{des_i} from the back preceding vehicle V, once the vehicle V has crossed the *End of evaluation line.*

$$d_{des_i} \; = \; r \; + \; (h \times v_i(k)) \tag{2.4}$$

where r is the standstill safety distance from front vehicles, h is the headway time constant, the values of r, h provided by the GCDC organisers during the competition were $r = 15\ m$ and $h = 0.5\ s$, $v_i(k)$ is the velocity of the host vehicle i at the time instant k. In GCDC intersection scenario the position of host vehicle i is always PC_2. $d_{meas_i}(k)$ is the measured distance between vehicle V and PC_2 at the time instant k.

Error Computation

The error between the measured and desired distances between the vehicles V and PC_2 is given by

$$e_i(k) \; = \; d_{meas_i}(k) \; - \; d_{des_i}(k) \tag{2.5}$$

Scoring Function

The error computed in Equation 2.5 is required to be non-negative, which means at a time instant the measured distance $d_{meas_i}(k)$ should be greater than the desired distance $d_{des_i}(k)$. The computed error determines the score of a competitor in a heat h. The threshold value th is calculated as

$$th \;=\; d_{des_i}(k) - d_{safe_i}(k) \tag{2.6}$$

where $d_{safe_i}(k)$ is the minimum safety distance required to be maintained by the vehicle PC_2 from V. The value of $d_{safe_i}(k)$ is approximately

$$d_{safe_i}(k) \;=\; (0.7).d_{des_i}(k) \tag{2.7}$$

The scoring function $q_i(k)$ for the vehicle i at time instant k is given as

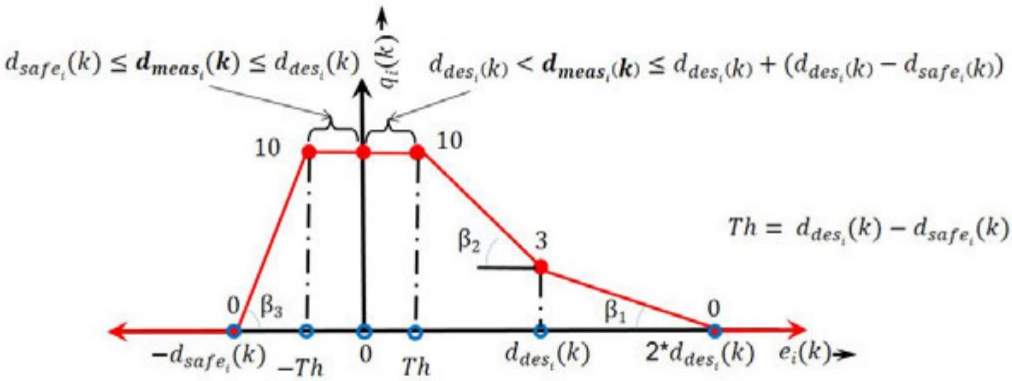


Figure 2.6: Q_2 Desired Distance Scoring Function (www.gcdc.net)

$$q_i(k) \;=\; Q_2(e_i(k)) \tag{2.8}$$

where Q_2 is the scoring function as shown in Figure 2.6 with error $e_i(k)$ on the X-axis and scoring function $q_i(k)$ on the Y axis. According to the scoring function Q_2 when the measured distance $d_{meas_i}(k)$ of the vehicle i is in the range of $[d_{meas_i}(k) - th,\ d_{meas_i}(k) + th]$ then the vehicle gets the score of 10 for that time instant k. If the measured distance d_{meas_i} is out of range $[d_{meas_i}(k) - th,\ d_{meas_i}(k) + th]$, the scores are calculated asymmetrically for the values on either of the range. If the error $e_i(k)$ is positive and the measured distance $d_{meas_i}(k)$ is more than $(d_{meas_i}(k) + th)$ then the scores are calculated as in the positive X axis with slopes β_1 and β_2. If the error $e_i(k)$ is negative at a time instant k then the penalization of the scores are heavy as determined by the slope β_3. When the error value $e_i(k)$ is lesser than the value $d_{meas_i}(k) - th$ the scenario is perceived to be life threatening and there is additional penalizing function Q_3 introduced as in Figure 2.7, which is valid for measured values $d_{meas_i}(k)$ at time instants k lesser than $d_{meas_i}(k) - th$.
This extra penalty score $q'_i(k)$ for vehicle i at time instant k is given by

$$q'_i(k) = Q_3(e_i(k)) \tag{2.9}$$

Heat Score

The final heat score for the vehicle i in the desired distance criteria at a heat h is given by

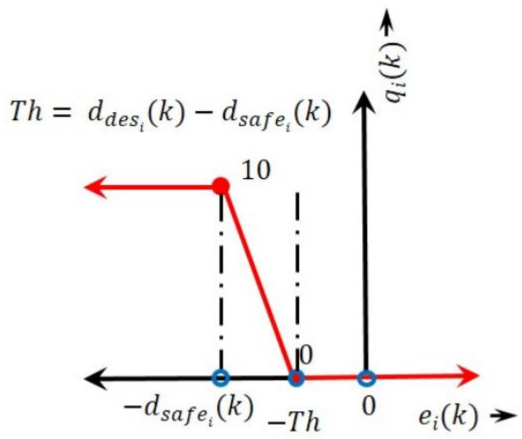


Figure 2.7: Further Penalty for Q_3 function (www.gcdc.net)

$$s_i(h) = \frac{\sum\limits_{k \in K} q_i(k)}{M} - \max_{k \in K} q_i'(k) \tag{2.10}$$

where K is the set of time instances K when the vehicle has got a score other than 10. The heat score involves deducing the maximum value of $q_i'(k)$ that a vehicle has received by violating the $d_{safe_i}(k)$, so the net value on this criterion can be even negative.
The final individual score $score_{ind_i}(c)$ for the vehicle i in this criterion c is the average of scores obtained in all the heats H held, which is given by

$$score_{ind_i}(c) = \frac{\sum\limits_{h=1}^{H} s_i(h)}{H} \tag{2.11}$$

2.3.2.3 Fastest Finish Time

Finish time t_{finish_i} for a competitor vehicle i in intersection scenario is the time spent by the vehicle inside the CZ, from start of the scenario until the end of the scenario as in Figure 2.8. The start of scenario is indicated by front of one of the vehicles crossing the CZ and the end of scenario is indicated by the front of the last vehicle leaving the CZ. Fastest finish time criteria is an individual criteria for a competitor vehicle.

Heat Score
Heat score $s_i(h)$ of the vehicle i in a specific heat h is given as

$$s_i(h) = t_{finish_i} \tag{2.12}$$

Criterion Score
In the case of multiple heats, with H being the number of heats, the average score sc_i for vehicle i is given as

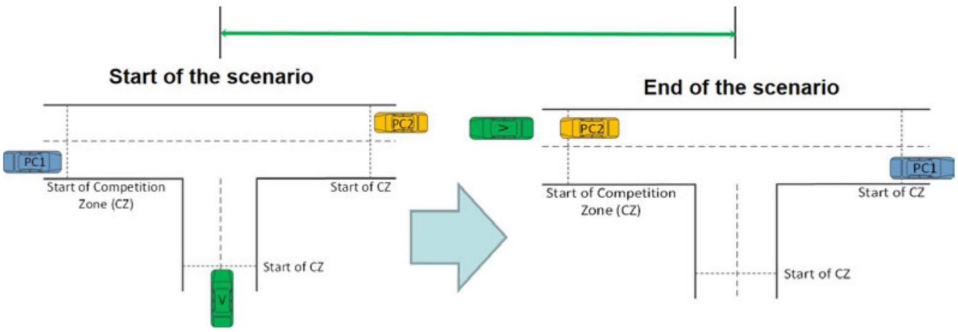


Figure 2.8: Entering and Exiting CZ (www.gcdc.net)

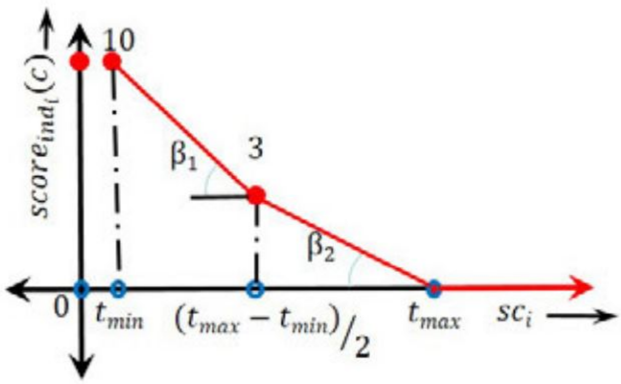


Figure 2.9: Q_3 Scoring Function (www.gcdc.net)

$$sc_i = \frac{\sum_{h=1}^{H} s_i(v)}{H} \tag{2.13}$$

The criterion score for individual vehicle i is the normalised score determined by the scoring function Q_3 as in Figure 2.9 taking in to account the overall minimum finish time t_{min} and maximum finish time t_{max}

$$t_{min} = \min_{1 \leq i \leq N} \min_{1 \leq h \leq H} s_i(h) \tag{2.14}$$

$$t_{max} = \min_{1 \leq i \leq N} \max_{1 \leq h \leq H} s_i(h) \tag{2.15}$$

The individual vehicle i score for the fastest finish time criterion c is given by

$$score_{ind_i}(c) = Q_3(t_{min}, t_{max}, sc_i) \tag{2.16}$$

2.3.2.4 Maximum Speed Limit

All the three vehicles V, PC_1 and PC_2 can speed up to a maximum limit of 30 $kmph$. The vehicles will be penalised if this maximum speed limit is violated. The method of calculating the score according for Maximum speed limit criteria is same as the minimum euclidean distance criteria.

Error Computation
The error $e_i(k)$ in maximum speed limit at a time instant k for vehicle i is given as

$$e_i(k) \ = \ 30 \ - \ v_i(k) \tag{2.17}$$

where $v_i(k)$ is the velocity of the vehicle at time instant k. The score for maximum speed limit criterion is evaluated only when the error value $e_i(k)$ is negative, which happens when the value of speed $v_i(k)$ at time instant k is higher than 30 *kmph*.

Heat Score
For a specific heat h according to the maximum speed limit criterion, the scoring function used is Q_1 which is same as minimum euclidean distance criterion. So the heat score for the criterion is given as

$$s_i(h) = \begin{cases} \min\limits_{k \ \in \ K} Q_1(e_i(k)), & M \ > \ 0 \\ Q_1(0), & M \ = \ 0 \end{cases} \tag{2.18}$$

where K is the set of time instants k, when the speed $v_i(k)$ is higher than 30 *kmph*. M is the size of the set K.
The individual vehicle score for this criterion c after a number of heats H is defined as the average of individual heat score $s_i(h)$ of the vehicle, which is given as

$$score_{ind_i}(c) = \frac{\sum\limits_{h=1}^{H} s_i(h)}{H} \tag{2.19}$$

2.4 Example Scenario 2

The longitudinal control of an autonomous vehicle should be valid for all kinds of intersection scenario. So in addition to GCDC Intersection scenario, two other T-intersection scenarios are defined in this thesis, in order to check the working of the intersection controller in different scenarios. In this subsection 2.7.2 a scenario, namely Example scenario 2 as described in the Figure 2.10 is considered. Note: The nomenclature of vehicles in each of the intersection scenarios are different.
In this Example Scenario 2 the T-intersection road geometry is same as the GCDC intersection scenario, but the vehicles parameters (like intentions and sizes) are different. The vehicle V_1 in *Lane 1* takes a left turn, vehicle V_2 in *Lane 2* intends to go straight and the vehicle V_3 in *Lane 3* intends to take a left turn from the primary road to secondary road.
The objectives and judging criteria for this *Example Scenario 2* are same as the GCDC intersection scenario as in sections 2.3.2 and 2.3.1 respectively.

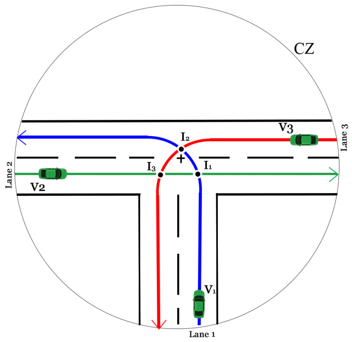


Figure 2.10: Example Scenario 2 Layout

2.5 Example Scenario 3

Example Scenario 3 also makes use of the road T-intersection geometry as in the previous 2 scenarios in sections 2.3 and 2.4. The vehicle V_1 in *Lane 1* takes a left turn, vehicle V_2 in *Lane 2* intends to take a right turn from primary road to secondary road and the vehicle V_3 in *Lane 3* intends to take a left turn from the primary road to secondary road. The vehicles V_1 and V_3 are passenger cars and the vehicle V_2 is a truck.

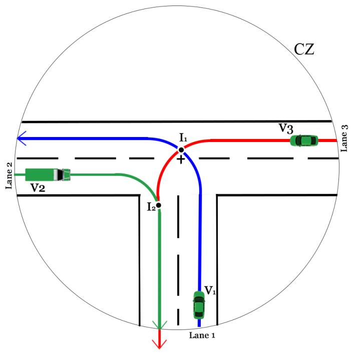


Figure 2.11: Example Scenario 3 - Layout

The objectives and judging criteria for this *Example Scenario 3* are same as the GCDC intersection scenario as in sections 2.3.2 and 2.3.1 respectively.

2.6 Stages in Intersection Crossing

Every vehicle participating in the cooperative driving at the T-intersection is required to go through the following stages before deciding the control input for the longitudinal motion.

1. **Entering CZ** The vehicles participating in the cooperative driving in the intersection should be physically present inside a Competing Zone (CZ) at the same time. The competing Zone (CZ) is a circle of radius (say fifty meters) with center at the geometrical center of the intersection as indicated by **IRF** in Figure 2.12. The three Vehicles V_1, V_2 and V_3 should enter into CZ on their respective lanes *Lane 1*, *Lane 2* and *Lane 3* at exactly same time and at intended maximum allowed velocity. Since the vehicles are forced to be inside CZ at the same time if the vehicles don't work cooperatively they will be colliding at the center of intersection (or) in best case will be violating the minimum safety distance between them when they are advancing towards the center.

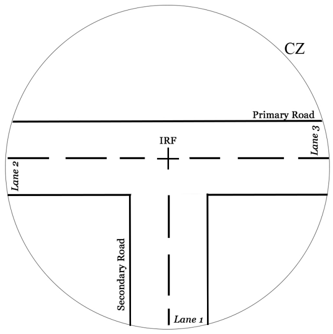


Figure 2.12: CZ and Lanes

2. **Vehicle to Vehicle (V2V) Communication** The vehicles start exchanging states in the form of communication messages once they are inside CZ. The exchanged states - position, velocity, acceleration, path intention (straight, left turn or right turn), travelling lane are important of the information communicated between vehicles. So each vehicle inside CZ is aware of the states of other two vehicles. The communication protocol and exchanged states are elaborated in section 2.6.1
3. **Vehicle Prioritization** All the three vehicles inside CZ, decide priorities for themselves based on the communicated states. The priorities are decided based on a common priority assigning methodology as in section 2.6.2. As a result of prioritization, every vehicle inside CZ knows which vehicle can cross the intersection first, second and third/last, the priorities being high, medium and low.
4. **Vehicle Characterization** Characteristics of a vehicle are decided between two vehicles, who have their paths intersecting. *Vehicle Characteristics* assigns between that pair of vehicles, a higher priority vehicle as *leader* and a lower priority vehicle as *follower*.
5. **Vehicle Control** The vehicles control their motion longitudinally fulfilling the objectives as described in Section 2.3.1. The longitudinal controller is designed, such that a *follower* vehicle allows the *leader* vehicle to cross the

intersection first. The lateral control prevents the vehicles from deviating their predefined path.

6. **Exiting CZ** When all the vehicles have crossed the intersection safely, they should leave the CZ as early as possible indicating the end of cooperative manoeuvre between them.

2.6.1 Vehicle to Vehicle (V2V) Communication

Vehicle to Vehicle (V2V) communication is necessary, such that a vehicle is aware of the states of other vehicles inside intersection zone. In this thesis, the states we are interested in are the ones important for longitudinal control of a vehicle (e.g) position, velocity and acceleration. V2V communication in our case is wireless communication and is established as per the requirements of GCDC communication protocol [3] [4] [5]. The Vehicular communication application in our study case intersection scenarios establishes only Vehicle to Vehicle(V2V) communication. The communication software and hardware used in the intersection scenarios of our study is the work of Albin Severinson, Paolo Falcone et al [10]. The communication layout between vehicles is as shown in Figure 2.13. From the simulink sending subsystem of the computer, the data are packed as Cooperative Awareness (CAM) messages, Decentralized Environment Notification (DENM) messages, iGame Cooperative Lane Changing (iCLCM) messages and sent to the API for broacasting the messages wirelessly. Similarly data of the other vehicles are broadcasted as messages wirelessly. The receiving side of our API receives those wireless messages from other vehicles and sends them to the simulink receiving side subsystem. There are other information exchanged between vehicles as per the Section A.2 in Appendix which are required by both Platooning Scenario and intersection scenario of GCDC. For the intersection scenario, we acquire only the signals from the Table 2.1. These signals are used by a vehicle for Vehicle Prioritization calculation, vehicle characteristics assignment and longitudinal control of the vehicles.

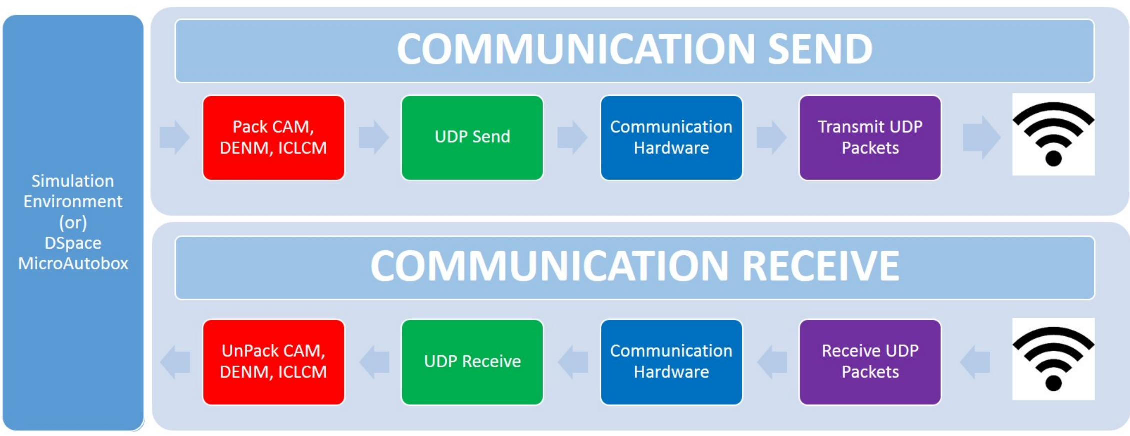


Figure 2.13: Communication Scheme

CAM - Cooperative Awareness Message, Decentralized Environmental Notification

Message (DENM) and iGAME Cooperative Lane Change Message (iCLCM) are the types of messages which are transmitted from a vehicle at a frequency of 25 Hz, 10 Hz and 25 Hz respectively. The messages receiving module of a vehicle can be run at a maximum frequency possible to receive messages from all the vehicles in the vicinity, here a frequency of 100 Hz is used.
Signals/data in the Table 2.1 are required for cooperative driving in T-intersection scenario, the description column briefs the importance of the signal/data

Table 2.1: Signals/data used by vehicles in Intersection Scenario

Signal ID	Signal	Message	Description
9	Vehicle Length	CAM	l_i
11	Vehicle Width	CAM	w_i
15	Vehicle Position, easting	CAM	x_i
16	Vehicle Position, northing	CAM	y_i
18	Vehicle Heading	CAM	θ_i
20	Longitudinal Velocity	CAM	v_i
24	Longitudinal Acceleration	CAM	$\alpha = \dot{v}_i$
44	Distance inside CZ	iCLCM	d_i
45	Intention	iCLCM	$\eta(i)$ (left, right or straight)
46	Lane of vehicle	DENM	$\kappa(i)$ (Lane 1, Lane 2 or Lane 3)
47	Vehicle Counter	DENM	m

2.6.2 Vehicle Prioritization

Vehicle Prioritization assigns *high*, *medium* or *low* priorities for the vehicles inside CZ. The *high* priority vehicle crosses the intersection first, the vehicle with *medium* priority crosses second and the vehicle with *low* priority crosses last. The vehicles can have same level of priorities if their paths of travel are not intersecting. If the paths of a pair of vehicles are intersecting, the two vehicles should have different priority. Every vehicle inside CZ strictly follows priority assignment methodology, based on say for example - *Lane* of travel, *intention* of vehicle, *Size Of Vehicle* and *Time of Arrival* inside CZ. The methodology used in this thesis is explained in Table 2.2 with three prioritizing parameters *Vehicle Path*, *Vehicle Size* and *Time of Arrival* in CZ. The parameters in the Table 2.2 describe the following.

1. The vehicle travelling from a secondary road to primary road gets a higher priority than the one travelling from primary to secondary road or primary to primary road.
2. Vehicle Size parameter decides that the heavier vehicle gets higher priority advantage in crossing the intersection than the lighter ones
3. Time of Arrival parameter considers that the vehicle which arrives in the CZ early gets higher priority than the vehicles arriving late.

For every vehicle in the scenario each parameter stated above is evaluated in the order as mentioned in the Table 2.2. Priorities are assigned between vehicles which have their paths intersecting. If the first parameter $Vehicle\ Path$ results in a tie in priority levels between two vehicles then the second parameter $Vehicle\ Size$ is taken into consideration for assigning priorities. If the second parameter also results in a tie in priority levels between two vehicles then the third parameter *Time of Arrival* is taken into consideration and so on. The number of parameters for assigning priorities can be increased/changed suiting different needs like medical emergencies of vehicles, tax paying criteria etc.

Table 2.2: Vehicle Prioritization

Parameter	**Priority**		
	High	**Medium**	**Low**
Vehicle Path	Secondary to Primary Road	Primary to Secondary Road	Primary to Primary Road
Vehicle Size	Heavy Vehicles	Medium sized Vehicles	Light Commercial Vehicles
Time of Arrival	Entered CZ First	Entered CZ Second	Entered CZ Third

2.6.3 Vehicle Characteristics

Vehicle Characteristics are decided between a pair of vehicles, if

1. The pair of vehicles have their paths intersecting inside CZ
2. If the priorities between those two vehicles are not the same like *high-medium*, *medium-low* and *high-low*.

The characteristics are decided such that one vehicle is called *leader* and the other vehicle is called *follower*. When the paths of two vehicles intersect, the vehicle with higher priority becomes the *leader* and the vehicle with lower priority becomes the *follower*. The *leader* vehicle crosses the intersection first. The *leader* controls its longitudinal dynamics such that it leaves the intersection at the maximum velocity, possible. The *follower* should choose a behaviour such that the *leader* is allowed to cross the intersection first, with the *follower* not violating the minimum safety distance between the *leader* and *follower* vehicles.

2.7 Application to Intersection Scenarios

Vehicle Prioritization and *Vehicle Characteristics* methodology discussed in previous Sections will be implemented for individual intersection scenarios in this Section.

2.7.1 GCDC Intersection Scenario

2.7.1.1 Vehicle Prioritization

The GCDC intersection scenario has priorities for vehicles which were predetermined by GCDC organisers. *Vehicle Prioritization* as in section 2.6.2 is not necessary to be applied for GCDC intersection scenario . Even though, the *Vehicle Prioritization* methodology yields the same priority results. According to the parameter *Vehicle Path* vehicle V is travelling from a secondary road to primary road, so the priority is *high*. Similarly vehicles PC_1 and PC_2 travel from primary to primary road so their priorities are *medium*. Since the paths of vehicles PC_1 and PC_2 do not intersect the prioritization function can be terminated here.

Table 2.3: Vehicle Prioritization - GCDC Scenario

Parameter	**Priority**			**Discussion**
	High	**Medium**	**Low**	
Vehicle Path	V	PC_1,PC_2		Paths of PC_1 & PC_2 do not intersect. Prioritization is done.

The priorities as in Table 2.3 are same as the fixed priorities specified by GCDC organisers.

2.7.1.2 Vehicle Characteristics

GCDC do not insist competitors to assign the vehicle characteristics. But vehicle characteristics is required for simplifying the problem of longitudinal vehicle control. So the characteristics assignment for the GCDC vehicles is as in Table 2.4

Table 2.4: Vehicle Characteristics - GCDC Scenario

Vehicle Pair	**Vehicle Characteristics**		**Discussion**
	leader	follower	
V - PC_1	V	PC_1	
V - PC_2	V	PC_2	
PC_1 - PC_2	-	-	Priorities are not related

2.7.2 Example Scenario 2

2.7.2.1 Vehicle Prioritization

If the 'Path of the vehicles' parameter is applied to this Example Scenario 2, V_1 gets *high* priority to cross the intersection, since it is joining the primary road from the secondary road. V_3 gets the *medium* priority since it intends to take a left turn into secondary road. Vehicle V_2 has *low* priority since it is travelling straight on the primary road.

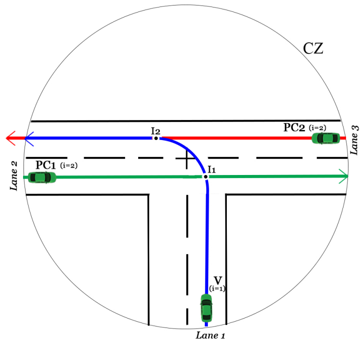


Figure 2.14: GCDC Intersection Scenario

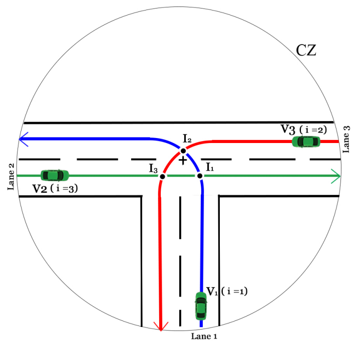


Figure 2.15: Priority and Characteristics Assignment - Example Scenario 2

Table 2.5: Vehicle Prioritization - Example Scenario 2

Parameter	Priority			Discussion
	High	**Medium**	**Low**	
Vehicle Path	V_1	V_3	V_2	Three vehicles have different priorities.

2.7.2.2 Vehicle Characteristics

The paths of vehicle $V1$ and $V3$ intersect at I_2 point. $V1$ is of *high* priority, so it is the leader and $V3$ of *medium* priority is the follower in crossing the intersection point I_2. Recalling the *Vehicle Characteristics* assignment from section 2.6.3 applies to the vehicles with adjacent priorities, there is no *leader*, *follower* assignment between V_1 and V_2, since they are of same priority levels which is *medium*. But the vehicle V_3 is of *medium* priority, V_2 is of *low* priority and have their paths intersecting at I_3, so V_3 is the leader and V_2 is the follower.

Table 2.6: Vehicle Characteristics - Example Scenario 2

Vehicle Pair	Vehicle Characteristics		Discussion
	leader	follower	
V_1 - V_2	-	-	Same priorities
V_2 - V_3	V_2	V_3	
V_1 - V_3	V_1	V_3	

2.7.2.3 Judging Criteria

Example Scenario 2, is considered in the thesis for applying, validating and testing of the longitudinal intersection control in intersection scenarios where the parameters of the vehicles are more complex than GCDC intersection scenario. For the example scenario 2, the *fastest finish time* criterion and *maximum speed limit criterion* are retained for individual vehicles. The *minimum euclidean distance* criterion is valid between vehicles V_1 and V_2, in which V_1 is the *leader* and V_2 is the *follower*, until V_1 is within the *End of evaluation* line. Similarly, *minimum euclidean distance* criterion is valid between vehicles V_2 and V_3, where V_2 is the *leader* and V_3 is the *follower*, until V_2 crosses the *End of evaluation* line.*End of evaluation* for individual vehicles are described more in next chapter. Scoring methods for individual criterion are same as explained in the sections of Judging Criteria 2.3.2.

2.7.3 Example Scenario 3

2.7.3.1 Vehicle Prioritization

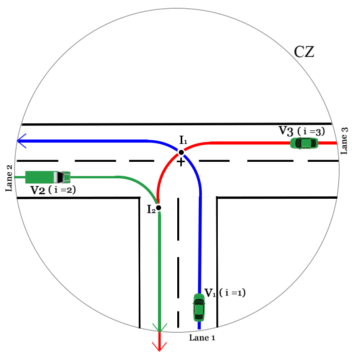


Figure 2.16: Priority and Characteristics Assignment - Example Scenario 3

According to 'Path of the vehicles' parameter V_1 gets *high* priority to cross the intersection, since it is joining the primary road from the secondary road. V_2 gets the *medium* priority since the vehicle travels from primary road to secondary road, so does vehicle V_3 with *medium* priority. The paths of vehicles V_2 and V_3 are intersecting at intersection point I_2, so the vehicles cannot remain in the same priority levels according to *Vehicle Prioritization* rule, section 2.6.2. So we consider the parameter

Size of Vehicle for splitting the priorities. Since V_2 being a heavy vehicle - truck, it gets higher priority level of *medium* and V_3 being a light vehicle - a passenger car gets lower priority level of *low*.

Table 2.7: Vehicle Prioritization - Example Scenario 3

Parameter	**Priority**			**Discussion**
	High	**Medium**	**Low**	
Vehicle Path	V_1	V_2,V_3		Two vehicles with *medium* priorities and their paths are intersecting
Size of Vehicle		V_2	V_3	

2.7.3.2 Vehicle Characteristics

The paths of vehicle $V1$ and $V3$ intersect at I_1 point. Vehicle $V3$ is of *low* priority and its immediate higher priority level is *high* as vehicle V_1, so V_3 is the *follower* and V_1 is the *leader* in crossing the intersection point I_1. The path of vehicles V_2 and V_3 intersect at point I_2, the priority of V_3 is *low* and the priority of V_2 is *medium*, so the vehicle V_3 is the *follower* and V_2 is the leader .

Table 2.8: Vehicle Characteristics - Example Scenario 3

Vehicle Pair	**Vehicle Characteristics**		**Discussion**
	leader	follower	
V_1 - V_2	-	-	Paths are not intersecting
V_1 - V_3	V_1	V_3	
V_2 - V_3	V_2	V_3	

2.7.3.3 Judging Criteria

For the *Example Scenario 3*, all the four Judging Criteria as in section 2.3.2 applies for determining the scores. The *minimum euclidean distance* criterion applies between vehicles V_1 and V_3, also applies between vehicles V_3 and V_2 until V_1 and V_3 cross their respective *End of Evaluation* line. The *desired distance* criterion applies between vehicles V_2 and V_3, once V_2 has crossed the *End of evaluation* line.

2.8 Summary

In this chapter the following are the summarising points:

- Discussion of *Performance Measure* and *Safety Requirements* for an autonomous vehicle in cooperative intersection scenario.
- Description in detail about the various cooperative intersection scenarios viz GCDC Intersection Scenario, Example Scenario 2, Example Scenario 3
- Discussion about *Judging Criteria* of GCDC as a tool for quantization of the *Performance Measure* and *Safety Requirement* for an Cooperative intersection scenario.

- Detailed description of *Stages in Intersection Crossing* for an autonomous vehicle inside an intersection scenario.
- Application of cooperative driving strategies discussed in this chapter to the three intersection scenarios described.

3
Vehicle Control

This chapter deals with the core of the thesis, which is the longitudinal control of the individual vehicle inside the intersection scenario. This chapter elaborates the conceptualisation and formulation of longitudinal controller. Acceleration is the output of longitudinal controller.

3.1 Optimal Control Problem

An autonomous vehicle, inside the intersection zone CZ, fulfilling the cooperative strategical factors mentioned in previous chapters would

- **Minimize** the **velocity error** of vehicle to the **maximum allowed velocity** as a *Performance Measure* and finish the intersection crossing.
- **Minimize input** to the autonomous vehicle, which means **zero or negative acceleration**, a *Performance Measure* to *reduce fuel consumption* inside CZ
- Constrain the position and motion of a vehicle, so that the **minimum safety distance** between two vehicles is not violated, which is a *Safety Requirement* to avoid collision between vehicles
- Impose constraints on the speed of the vehicle such as not to exceed **maximum velocity limit**, a *Safety Requirement* to speed limit of the vehicles in the intersection

Considering all the above points, the problem is to control the dynamics of the vehicle by **minimizing the error** to set values (or **optimizing the values**), while imposing **limits or constraints** for physical quantities affecting the motion of the vehicle.

3.2 Model Predictive Control (MPC)

Model predictive control (MPC) determines the control input of a system for the present time instant by taking into account the behaviour of the system in future time instants. Moreover, MPC also constrains the physical quantities of the system. In the past, MPC has been applied to a number of longitudinal and lateral vehicle control problems in Mechatronics Research Group of Chalmers. It is also a viable solution for the intersection longitudinal control problem at hand. Next section gives a quick introduction about MPC.

3.2.1 Problem Formulation

A MPC problem can be formulated as follows

- Model the system or process to be controlled in terms of state space equations. The parameters to be controlled should be designed as states and suitable inputs.

  If X are the states of the system and U the input to the system. The discrete state space equation calculates the value of states at time step $(k+1)$ from the present time step k, with the equation

$$X(k+1) \;=\; AX(k) \;+\; BU(k) \tag{3.1}$$

- Predict the states of the system for a prediction horizon of N steps

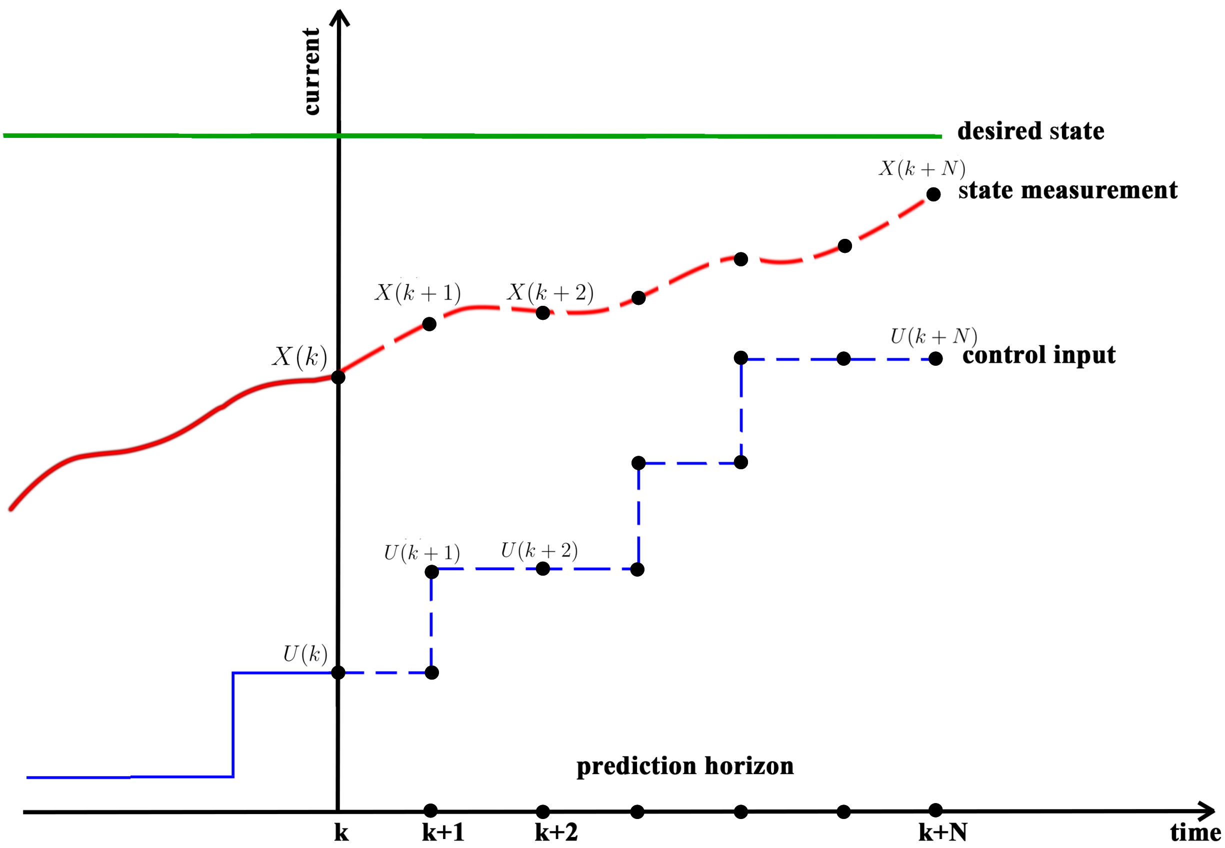


Figure 3.1: Model Predictive Controller

- Define the cost function for the MPC problem over the prediction horizon. A cost function takes into account both the input and states of the system as given below.

$$J_k(X(k), U(k)) \\ = \sum_{k=T+N} ||X(k)||_P^2 + \sum_{k=T}^{T+N-1} ||X(k)||_Q^2 + ||U(k)||_R^2 \tag{3.2}$$

On the right hand side of the equation, the first summation part is the final cost and the second summation part is the stage cost. $J_k(X(k), U(k))$ is the cost function for a horizon N. N is the number of steps of control Horizon/prediction horizon. P, Q, and R are the penalties for Final state cost, stage state cost and input cost respectively.

- The optimization problem considered here in our case is a minimization problem which is given by

$$\begin{aligned} &\text{minimize} \quad J_k(X(k), U(k)) \\ &\text{subject to } X(k) \ \in \chi, \ U(k) \ \in \ \upsilon \end{aligned} \tag{3.3}$$

where χ is the set representing state constraints and υ is the set representing input constraints.

- The optimal input from the current time step k is the control input to the system. The states of the system are measured at the next sample time and fed back to MPC.

3.3 Longitudinal Control

The components of longitudinal controller for the intersection scenarios is detailed in this section in terms of MPC requirements.

3.3.1 Longitudinal Dynamics

A double integrator model is used to represent the longitudinal dynamics of the vehicle. The states are longitudinal position $P(t)$, longitudinal velocity $v(t)$ along the intended path and the input acceleration $a(t)$ for the vehicle. So the dynamics of the vehicle in state space form is represented as

$$X(t) \ = \ [P(t) \ v(t)]^T \tag{3.4}$$

which can be expanded into

$$\begin{bmatrix} \dot{P}(t) \\ \dot{v}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} P(t) \\ v(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} a(t) \tag{3.5}$$

In Equation (3.5), the demanded input acceleration is assumed to be obtained in the real system, but this is not the case in real vehicle. The relation between the demanded input acceleration $a^d(t)$ and the actual acceleration of the vehicle is given as

$$\dot{a}(t) = \frac{-1}{\tau}a(t) + \frac{1}{\tau}a^d(t) \tag{3.6}$$

The value of τ in Equation (3.6) is experimentally found in [20] for speeds up to 80 $kmph$ as

$$\tau \;=\; 0.5\ s \tag{3.7}$$

Therefore Equation (3.5) can be re-written as

$$\begin{bmatrix} \dot{P}(t) \\ \dot{v}(t) \\ \dot{a}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \frac{-1}{\tau} \end{bmatrix} \begin{bmatrix} P(t) \\ v(t) \\ a(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{\tau} \end{bmatrix} a^d(t) \tag{3.8}$$

From Equation (3.8), actual acceleration $a(t)$ of the vehicle is added as a longitudinal state

$$X(t) \;=\; [P(t)\ v(t)\ a(t)]^T \tag{3.9}$$

and the input becomes

$$U(t) \;=\; a^d(t) \tag{3.10}$$

The state space equation for longitudinal control of a vehicle can now be represented as

$$\dot{X}(t) = AX(t) + BU(t); \tag{3.11}$$

where A and B matrices are in continuous time which are defined as

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \frac{-1}{\tau} \end{bmatrix} B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{\tau} \end{bmatrix} \tag{3.12}$$

The state space Equation (3.11) is discretized by Euler's method for the purpose of longitudinal control, which is given as

$$X(k+1) = A_d X(k) + B_d U(k) \tag{3.13}$$

where A_d is A matrix in discrete time and B_d is B matrix in discrete time.

3.3.2 Cost Function

Let the variable $J_k(X(k), U(k))$ denote the cost function, which considers the objectives of reducing the error in reaching maximum velocity and expended input acceleration, which is given by

$$\begin{aligned} &J_k(X(k), U(k)) \\ &= \sum_{k=T+N} ||v_{max} - v(k)||_P^2 + \sum_{k=T}^{T+N-1} ||v_{max} - v(k)||_Q^2 + ||U(k)||_R^2 \end{aligned} \tag{3.14}$$

and P, Q, and R are the penalties for Final state cost, stage state cost and input cost respectively. T is the present time instant for which the control input is calculated. N is the number of steps of control Horizon/prediction horizon. v_{max} is the desired velocity for the controller.
The cost function for Longitudinal MPC Controller satisfies the objectives of attaining desired velocity v_{max} by every vehicle inside CZ to cross the intersection at the earliest time possible by expending minimum input acceleration $U(k)$
The objective of the MPC longitudinal controller is to *minimize* the cost function $J_k(X(k), U(k))$, which is given as

$$\text{minimize} \quad J_k(X(k), U(k)) \tag{3.15}$$

This cost function is same for all the intersection scenario considered in this thesis which are GCDC intersection scenario, Example Scenario 2 and Example Scenario 3
The longitudinal dynamics is as represented in equation 3.13

$$X_i(k+1) = A_d X_i(k) + B_d U_i(k); \tag{3.16}$$

where k takes values in the range of $k = \{T,\ T+1,\ ....,\ T+(N-1)\}$.

3.3.3 Constraints

The constraints for the state variables $X(k)$ and input variables $U(k)$ in the MPC problem are as defined below.

3.3.3.1 Position Constraints

Position constraints $P(k)$ define the allowed limits for a vehicle along its intended path inside the CZ of an intersection scenario at every time instant k. Let the position $P(k)$ denote the geometrical centre of the vehicle at time instant k. Position constraints vary for every vehicle in the intersection scenario. Between a pair of vehicles, leader l and follower f, the leader does not have to constrain its position based on the follower vehicle, but the follower vehicle have to constraint its position so that longitudinal control objectives of vehicle f are fulfilled, and collision inside CZ is avoided.

leader
If a vehicle is a *leader* then it is denoted by l. The *leader* l should cross the intersection point between the paths of vehicles f and l and clear way for vehicle f. So this position constraint $P_l(k)$ for *leader* can be written as

$$P_{l,CZ\ Entry} \ < \ P_l(k) \ \leq \ P_{l,CZ\ Exit} \tag{3.17}$$

Equation (3.17) means that the vehicle l can be at any position on its path between the entry of CZ ($P_{l,CZ\ Entry}$) and the exit of CZ ($P_{l,CZ\ Exit}$). This position constraint for a leader vehicle means that it is free to occupy whatsoever position along its path inside CZ. It is possible that a vehicle is a *leader* between one pair of vehicles and is a *follower* between another pair of vehicles, then the vehicle should constraint it's position considering itself as a *follower*.

follower
As per the longitudinal control requirements for a *follower* inside CZ, the position constraints should ensure that either the *minimum Euclidean distance* criteria or the *desired distance* criteria is fulfilled as mentioned in Section 2.3.2.1 **??**.

$$\begin{aligned} & d_{f,l}(k) > r_{euc}, \\ \text{if,} \quad P_{l,CZ\ Entry} \leq & \hat{P}_l(k|T) \leq P_{l,End\ of\ evaluation} \\ & f \neq l, \quad f,l \in i \end{aligned} \tag{3.18}$$

$$\begin{aligned} (d_{des_f}(k) - d_{safe_f}(k)) \leq & d_{f,l}(k) \leq (d_{des_f}(k) + d_{safe_f}(k)), \\ \text{if,} \quad P_{l,End\ of\ evaluation} \leq & \hat{P}_l(k|T) \leq P_{l,CZ\ Exit} \\ & f \neq l, \quad f,l \in i \end{aligned} \tag{3.19}$$

where $d_{f,l}(k)$ is the measured euclidean distance between the circle of leader vehicle l and the circle of follower vehicle f at the time instant k. The circle of a vehicle is the circle around vehicle with radius as half the length of the vehicle and center at the geometrical center of the vehicle which is same as position of the vehicle $P_i(k)$. The circle of vehicle is a notation used to approximate the geometry of a competitor vehicle to a circle.
The equation (3.18) should be written in terms of constraints for the follower position variables $P_f(k)$ which are represented as

$$\begin{aligned} P_{f,CZ\ Entry} \leq & P_f(k) \leq P_{f,Critical\ point}, \\ if, \quad P_{l,CZ\ Entry} \leq & \hat{P}_l(k|T) \leq P_{l,End\ of\ evaluation} \\ & f \neq l, \quad f,l \in i \end{aligned} \tag{3.20}$$

$P_{f,Critical\ point}$ is a point along the path of the vehicle f, such that when $P_f(k)$ is lesser than or equal to $P_{f,Critical\ point}$, the measured distance $d_{f,l}(k)$ between the vehicles leader l and follower f is always greater than minimum euclidean distance r_{euc} as defined in equation (3.18). Similarly equation (3.19) can be written in terms of position of the follower vehicle $P_f(k)$ as

$$\begin{aligned} (d_{des_f}(k) - d_{safe_f}(k)) \leq & \hat{P}_l(k|T) - P_f(k) \leq (d_{des_f}(k) + d_{safe_f}(k)), \\ if, \quad P_{l,End\ of\ evaluation} \leq & \hat{P}_l(k|T) \leq P_{l,CZ\ Exit} \\ & f \neq l, \quad f,l \in i \end{aligned} \tag{3.21}$$

The *follower* vehicle f predicts the position $\hat{P}_l(k|T)$ of *leader* vehicle l, in the intended path of *leader* over the prediction horizon N. This position prediction information of a leader vehicle l is necessary for a follower vehicle f to calculate its position constraints accordingly outside MPC problem formulation. A constant acceleration double integrator model is predicting the position of leader vehicle Application of position constraints from generic form to more specific intersection scenarios are shown below

3.3.3.1.1 GCDC Intersection Scenario

In this section, the position constraints for different vehicles in the GCDC intersection scenario are discussed and detailed. The position constraints explained in the previous section 3.3.3.1 are applied in this section for GCDC intersection scenario. The priorities of the vehicles in the GCDC intersection scenario are predefined by the organizers where vehicle V is the *leader*, vehicles PC_1 & PC_2 are termed as *follower*.

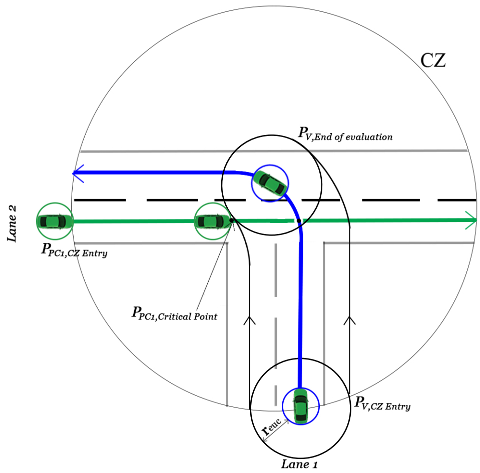


Figure 3.2: $PC1$ Position Constraints - GCDC Intersection Scenario

***leader* - V**

The leader vehicle V in GCDC intersection scenario will always be the benchmark vehicle assigned by the GCDC organisers. Here the vehicle exhibits Cruise Control (CC) behaviour trying to attain maximum velocity without any constraints on its position from *CZ Entry* till *CZ Exit*

$$P_{V,CZ\ Entry} < P_V(k) \leq P_{V,CZ\ Exit} \tag{3.22}$$

$P_V(k)$ denotes the position variable, $P_{1,CZEntry}$ is the point of entry of vehicle V into CZ and $P_{V,CZExit}$ is the point of exit of leader vehicle V from CZ as shown in Figure 3.2.

***follower* - PC_1**

Follower vehicle PC_1 is one of the *follower* vehicles with *medium* priority or a *competitor* vehicle as pre-defined by GCDC organisers. The *follower* vehicle should not violate the *minimum safety distance* with *leader* vehicle V. So the *follower* vehicle PC_1 has to constrain its motion until the *leader* vehicle V can no longer cause collision with vehicle PC_1 or vehicle V has not crossed the *End of evaluation* line. Position constraints for Follower Vehicle PC_1 is given by

$$\begin{aligned} P_{PC_1,CZ\ Entry} \ < \ P_{PC_1}(k) \ < \ P_{PC_1,Critical\ Point} \\ if, \quad P_{V,CZ\ Entry} \ \leq \ \hat{P}_V(k|T) \ \leq \ P_{V,End\ of\ evaluation} \end{aligned} \tag{3.23}$$

$P_{PC_1}(k)$ denotes the position variable for vehicle $PC1$, $P_{PC_1,CZ\ Entry}$ denotes position of entry of follower vehicle PC_1 into CZ. The physical significance of PC_1 critical point $P_{PC_1,Critical\ Point}$, is that beyond this point the vehicle violates the *minimum euclidean distance* criterion with vehicle V, provided when V is between $P_{V,CZ\ Entry}$ and $P_{V,End\ of\ evaluation}$. If the position of vehicle V is between $P_{V,End\ of\ evaluation}$ and $P_{V,CZ\ Exit}$, the vehicle would be completely inside the *Lane 3* and cleared way for vehicle PC_1. At this position of V, the position constraint for vehicle PC_1 changes to

$$\begin{aligned} P_{PC_1,CZ\ Entry} \ < \ P_{PC_1}(k) \ \leq \ P_{PC_1,CZ\ Exit} \\ if, \quad P_{V,End\ of\ evaluation} \ \leq \ \hat{P}_V(k|T) \ \leq \ P_{V,CZ\ Exit} \end{aligned} \tag{3.24}$$

***follower* - PC_2**

Follower vehicle PC_2 is a *follower*, *medium* priority vehicle, whose *leader* vehicle is V as predefined by the GCDC organisers. After the vehicle V and vehicle PC_2 are on the same path with vehicle PC_2 following vehicle V; PC_2 has to constrain its position over time such that, it is within *minimum or maximum desired distance* from vehicle V. Position constraints for the vehicle PC_2 is given by

$$\begin{aligned} (d_{des_{PC_2}}(k) \ - \ d_{safe_{PC_2}}(k)) \ \leq \ \hat{P}_V(k|T) \ - \ P_{PC_2}(k) \\ \leq \ (d_{des_{PC_2}}(k) \ + \ d_{safe_{PC_2}}(k)), \\ if, \quad P_{V,End\ of\ evaluation} \ \leq \ \hat{P}_V(k|T) \ \leq \ P_{V,CZ\ Exit} \end{aligned} \tag{3.25}$$

where $d_{des_{PC_2}}(k)$ is the desired distance of vehicle PC_2 from vehicle V at time instant k, $d_{safe_{PC_2}}(k)$ is the minimum safety distance of vehicle PC_2 from vehicle V at time instant k. During the course of training and GCDC competition week, the equation 3.25 was simplified by the GCDC organisers as equation 3.26

$$\begin{aligned} r_{euc} \ \leq \ \hat{P}_V(k|T) \ - \ P_{PC_2}(k) \\ \leq \ (r_{euc} \ + \ (h.\hat{v}_V(k|T)), \\ if, \quad P_{V,End\ of\ evaluation} \ \leq \ \hat{P}_V(k|T) \ \leq \ P_{V,CZ\ Exit} \end{aligned} \tag{3.26}$$

$P_{V,CZ\ Exit}$ is the point where vehicle V exits CZ, $P_{V,End\ of\ evaluation}$ is the point on the path of vehicle V where it crosses the *End of evaluation* line as shown in Figure 3.3.

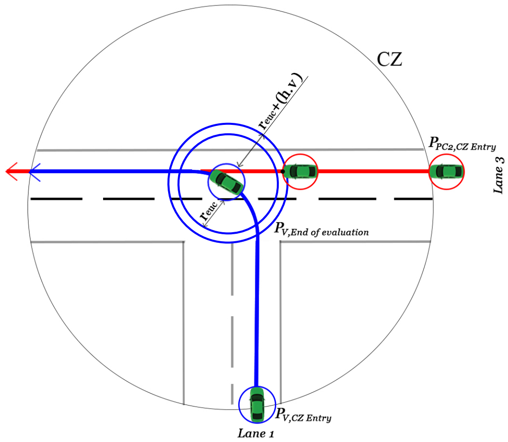


Figure 3.3: $PC2$ Position Constraints - GCDC Intersection Scenario

The position prediction model $\hat{P}_{PC_2}(k|T)$ of the *leader* vehicle V is the constant acceleration prediction with double integrator model as given in equation **??**
$P_{PC_2}(k)$ denotes the position variable of vehicle PC_2, $P_{PC_2,CZ\ Entry}$ denotes the position of entry of $PC2$ into CZ. The physical significance of $PC2$'s critical point $P_{PC_2,Critical\ Point}$, is that, beyond this point the vehicle violates the lowest euclidean distance constraint $(d_{des_{PC_2}}(k) - d_{safe_{PC_2}}(k))$ from vehicle V, provided V, is between $P_{V,CZ\ Entry}$ and $P_{V,End\ of\ evaluation}$. If the position of vehicle V is between $P_{V,End\ of\ evaluation}$ and $P_{V,CZ\ Exit}$, the vehicle would be completely inside the *Lane 3* and inline the path of vehicle PC_2.
Once the vehicle V has crossed the position $P_{V,End\ of\ evaluation}$ and is between $P_{V,End\ of\ evaluation}$ and $P_{V,CZ\ Exit}$, the difference in position or distance between the leader V and the follower PC_2 is calculated as shown in the figure 3.4

3.3.3.1.2 Example Scenario 2

The position constraints for the vehicles in an intersection scenario is applied below to the Example Scenario 2, briefed in section 2.7.2 of the previous chapter. In the scenario, V_1 is the vehicle with *high* priority, V_3 is the vehicle with *medium* priority and V_2 is the vehicle with *low* priority. Between the pairs of vehicle $V_1 - V_3$, V_1 is the *leader* and V_3 is the *follower*, similarly between pairs of vehicles with intersecting trajectories, V_2 and V_3, V_3 is the *leader* and V_2 is the follower. The position constraints of individual vehicles are discussed as follows

***leader* - V_1**

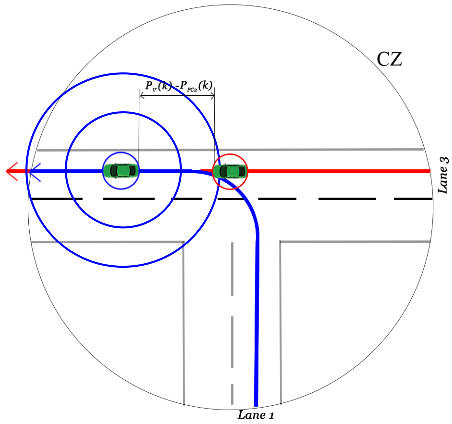


Figure 3.4: $PC2$ Position Constraints - GCDC Intersection Scenario

Vehicle V_1 is the *leader* vehicle with *high* priority, so it can cruise from its entry in to CZ $P_{V_1,CZ\ Entry}$ till the exit of CZ, $P_{V_1,CZ\ Exit}$.

$$P_{V_1,CZ\ Entry} \ < \ P_{V_1}(k) \ \leq \ P_{V_1,CZ\ Exit} \tag{3.27}$$

***follower* - V_3**

Vehicle V_3 is the vehicle with *medium* priority and it is the *follower* of vehicle V_1. The path of vehicle V_1 and V_3 intersect as shown in Figure 3.5. Vehicle V_3 has to constrain it's position over time, such that the *minimum euclidean distance* criteria with *leader* vehicle is fulfilled with V_1 and *minimum safety distance* is not violated with vehicle V_1, until V_1 can no longer collision with vehicle V_3 (or V_1 has crossed *End of evaluation* line). The position constraints of vehicle V_3 until V_1 crosses the *End of evaluation* line as in Figure 3.5 is given as

$$\begin{aligned} P_{V_3,CZ\ Entry} \ &< \ P_{V_3}(k) \ < \ P_{V_3,Critical\ Point} \\ \text{if,} \quad P_{V_1,CZ\ Entry} \ &\leq \ \hat{P}_{V_1}(k|T) \ \leq \ P_{V_1,End\ of\ evaluation} \end{aligned} \tag{3.28}$$

Once the vehicle V_1 has crossed or predicted to cross the *End of evaluation* line, then the vehicle V_3 will become unconstrained at that time instant k.

$$\begin{aligned} P_{V_3,CZ\ Entry} \ &< \ P_{V_3}(k) \ < \ P_{V_3,CZ\ Exit} \\ \text{if,} \quad P_{V_1,End\ of\ evaluation} \ &\leq \ \hat{P}_{V_1}(k|T) \ \leq \ P_{V_1,CZ\ Exit} \end{aligned} \tag{3.29}$$

Between the pair of vehicles V_3 and V_2, V_3 is the *leader*, when its position is unconstrained, which is after the vehicle V_1 has crossed the $P_{V_1,End\ of\ evaluation}$ point.

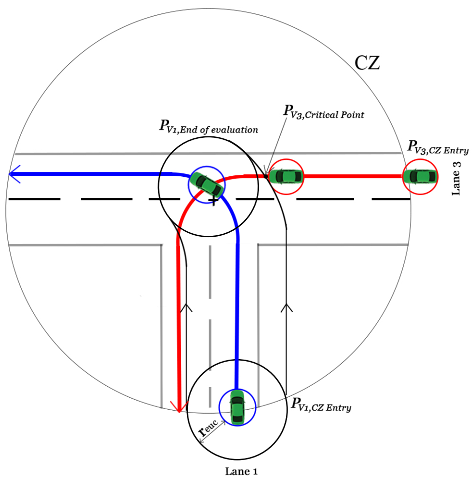


Figure 3.5: V_3 Position Constraints of Example Scenario 2

follower **-** V_2

Vehicle V_2 is of *low* priority, the pair of vehicles $V_1 - V_2$ is not considered for *leader - follower* assignment since their trajectories are not intersecting. V_3 is a vehicle with *medium* priority and also have its path intersecting with the path of vehicle V_2. Vehicle V_2 should constrain its position such that the *minimum euclidean distance* criteria with *leader* vehicle V_3 is not fulfilled and *minimum safety distance* with vehicle V_3 is not violated at all. So position variable $P_{V_2}(k)$ is constrained by the position of vehicle V_3 as given in equation 3.30 below

$$\begin{aligned} & P_{V_2,CZ\ Entry} \ < \ P_{V_2}(k) \ < \ P_{V_2,Critical\ Point} \\ \text{if} \quad & P_{V_3,CZ\ Entry} \ \leq \ \hat{P}_{V_3}(k|T) \ \leq \ P_{V_3,End\ of\ evaluation} \end{aligned} \tag{3.30}$$

Critical point $P_{V_2,Critical\ Point}$ of vehicle V_2 and *End of evaluation* point $P_{V_3,End\ of\ evaluation}$ of the vehicle V_3 are positioned as shown in Figure 3.6 for the *Example Scenario 2*.
when the vehicle V_3 has crossed the *End of evaluation* vehicle V_2 becomes unconstrained which is given by equation 3.31

$$\begin{aligned} & P_{V_2,CZ\ Entry} \ < \ P_{V_2}(k) \ < \ P_{V_2,CZ\ Exit} \\ \text{if} \quad & P_{V_3,End\ of\ evaluation} \ \leq \ \hat{P}_{V_3}(k|T) \ \leq \ P_{V_3,CZ\ Exit} \end{aligned} \tag{3.31}$$

3.3.3.1.3 Example Scenario 3

In this section the position constraints for *Example Scenario 3* described in section 2.7.3 are defined. Recalling the scenario, where V_1 is the vehicle with *high* priority

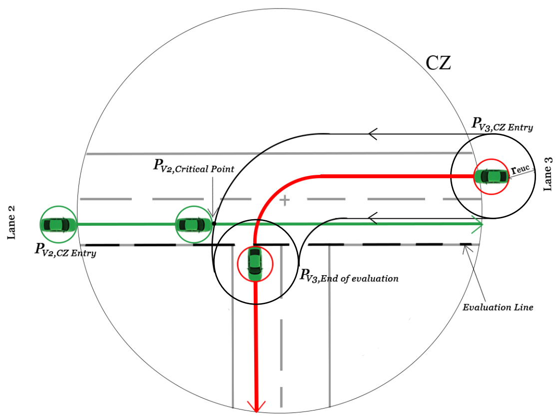


Figure 3.6: V_2 Position Constraints of Example Scenario 2

and is the *leader* vehicle between the pair of vehicles $V_1 - V_3$. In this scenario vehicle V_2 has *medium* priority, since it is entering secondary road from primary road and is the *leader* between the pair of vehicles V_2 and V_3. Vehicle V_3 is a vehicle with *low* priority, has its path intersecting with the path of vehicles V_1, V_2 and is the *follower* vehicle between both the pair of vehicles $V_1 - V_3$ and $V_2 - V_3$.

***leader* - V_1**

The position variable $P_{V_1}(k)$ of vehicle V_1 is unconstrained as in the previous cases. The vehicle $V1$ also has *high* priority, so it traverses through its path with the maximum allowed velocity inside CZ. Such position constraint is given as

$$P_{V_1,CZ\ Entry} \ < \ P_{V_1}(k) \ \leq \ P_{V_1,CZ\ Exit} \tag{3.32}$$

***leader* - V_2**

V_1 is the vehicle with *high* priority and V_2 is the vehicle with *medium*, but the path of these two vehicles never intersect, so there is no *leader* - *follower* assignment between them. V_2 is the vehicle with *medium* priority and has its path intersecting with vehicle V_3 of *low* priority, where V_2 is the *leader* and V_3 is the *follower*. So the position of vehicle V_2 is unconstrained along its path as shown in Figure 3.7

$$P_{V_2,CZ\ Entry} \ < \ P_{V_2}(k) \ \leq \ P_{V_2,CZ\ Exit} \tag{3.33}$$

***follower* - V_3**

Vehicle V_3 has to constrain its position over time instants such that it has fulfilled the *minimum euclidean distance* criteria with *leader* vehicles V_1 and V_2 until vehicles V_1

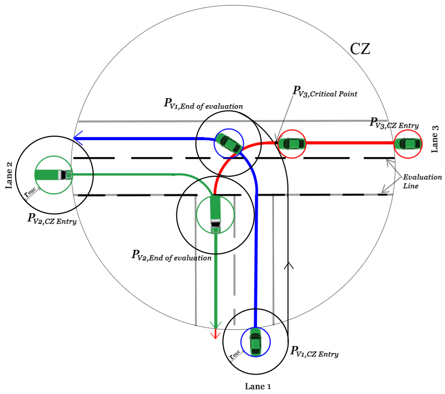


Figure 3.7: V_3 Position Constraints of Example Scenario 3

and V_2 have not crossed their respective *End of evaluation* lines (or not crossed the center of intersection). Once both vehicles V_1 and V_2 have crossed their respective *End of evaluation* line, V_3 and V_2 are on same trajectory, so *follower* vehicle V_3 has to fulfill *desired distance* criteria with vehicle V_2, until both V_3 and V_2 exit CZ. The position constraints between vehicle V_1 and the follower V_3 is same as in *Example Scenario 2* which is given by

$$\begin{aligned} P_{V_3,CZ\ Entry} \ &< \ P_{V_3}(k) \ < \ P_{V_3,Critical\ Point_1} \\ \text{if,} \quad P_{V_1,CZ\ Entry} \ &\leq \ \hat{P}_{V_1}(k|T) \ \leq \ P_{V_1,End\ of\ evaluation} \end{aligned} \tag{3.34}$$

But here the position of vehicle V_3 is not unconstrained, when V_1 has crossed *End of evaluation* line. The vehicle V_3 is constrained by the position of vehicle V_2. The vehicle V_2 is the *leader* and V_3 is the *follower* both have same path after the trajectories intersecting point as in Figure 3.7

$$\begin{aligned} P_{V_3,CZ\ Entry} \ &< \ P_{V_3}(k) \ < \ P_{V_3,Critical\ Point_2} \\ \text{if,} \quad P_{V_2,CZ\ Entry} \ &\leq \ \hat{P}_{V_2}(k|T) \ \leq \ P_{V_2,End\ of\ evaluation} \end{aligned} \tag{3.35}$$

Since there are two critical points on the path of vehicle V_3 due to the trajectory of V_3 crossing with both V_1 and V_3. But the vehicle V_3 can be constrained by one of the most conservative critical point, $P_{V_3,Critical\ Point}$ such that the *minimum euclidean distance* criteria is not violated by vehicle V_3 with both vehicle V_1 and V_2. So equations 3.34 & 3.35 can be condensed to single equation as

$$\begin{aligned} P_{V_3,CZ\ Entry} \ &< \ P_{V_3}(k) \ < \ P_{V_3,Critical\ Point} \\ \text{if,} \quad P_{V_1,CZ\ Entry} \ &\leq \ \hat{P}_{V_1}(k|T) \ \leq \ P_{V_1,End\ of\ evaluation} \\ \text{or if,} \quad P_{V_2,CZ\ Entry} \ &\leq \ \hat{P}_{V_2}(k|T) \ \leq \ P_{V_2,End\ of\ evaluation} \end{aligned} \tag{3.36}$$

The *desired distance* criterion is applied between vehicle V_2 and V_3, once the vehicle V_2 has crossed the $P_{V_2,End\ of\ evaluation}$ point.

$$
\begin{aligned}
(d_{des_{V_3}}(k) \ - \ d_{safe_{V_3}}(k)) \ &\leq \ \hat{P}_{V_2}(k|T) \ - \ P_f(k) \\
&\leq \ (d_{des_{V_3}}(k) \ + \ d_{safe_{V_3}}(k)), \\
if, \quad P_{V_2, End\ of\ evaluation} \ &\leq \ \hat{P}_{V_2}(k|T) \ \leq \ P_{V_2, CZ\ Exit}
\end{aligned} \tag{3.37}
$$

Here, the minimum position constraint $(d_{des_{V_3}}(k) \ - \ d_{safe_{V_3}}(k))$ and maximum position constraint $(d_{des_{V_3}}(k) \ + \ d_{safe_{V_3}}(k))$ are replaced by r_{euc} and $(r_{euc} \ + \ h.v_{V_2}(k))$ same as the position constraint for GCDC scenario. So equation 3.37 becomes

$$
\begin{aligned}
r_{euc} \ &\leq \ \hat{P}_{V_2}(k|T) \ - \ P_f(k) \\
&\leq \ (r_{euc} \ + \ (h.v_{V_2}(k)), \\
if, \quad P_{V_2, End\ of\ evaluation} \ &\leq \ \hat{P}_{V_2}(k|T) \ \leq \ P_{V_2, CZ\ Exit}
\end{aligned} \tag{3.38}
$$

The values of minimum euclidean distance and headway constant being same as in GCDC, such as $r_{euc} \ = \ 15 \ m$, $h \ = \ 1 \ s$

3.3.3.2 Velocity Constraints

The velocity limit applies to all the vehicles inside *CZ*. The constraints for the longitudinal velocity $v_i(k)$ of vehicle i is given as

$$
v_{min} \leq v_i(k) \leq v_{max} \ \forall \ i \tag{3.39}
$$

v_{min} and v_{max} are the minimum and maximum allowed velocities of the vehicles on its path inside the *CZ*. The values for the maximum and minimum velocities are given as

$$
\begin{aligned}
v_{min} \ &= \ 1 \ km/h \text{ (or) } 0.2778 \ m/s \\
v_{max} \ &= \ 30 \ km/h \text{ (or) } 8.33 \ m/s, \text{ on straight path} \\
v_{max} \ &= \ 20 \ km/h \text{ (or) } 5.55 \ m/s, \text{ around curved path}
\end{aligned}
$$

3.3.3.3 Acceleration Constraints

Every vehicle i inside *CZ* is expected to have acceleration state within certain limits which are represented as acceleration constraints.

$$
a_{min} \leq a_i(k) \leq a_{max} \ \forall \ i \tag{3.40}
$$

a_{min} and a_{max} are the minimum and maximum allowed acceleration for all vehicles inside the *CZ*. $a_i(k)$ is the longitudinal acceleration variable for vehicle i at time instant k.
The values of a_{min} and a_{max} adopted in this thesis are:

$$
\begin{aligned}
a_{min} \ &= \ -2 \ m/s^2 \\
a_{max} \ &= \ 1.5 \ m/s^2
\end{aligned}
$$

3.3.3.4 Actuator Constraints

The input acceleration demanded from the actuators should be in the same range as the actual longitudinal acceleration of the vehicle i. The input acceleration $a_i^d(k)$ demanded is also constrained as per the requirements of GCDC.

$$a_{min} \leq a_i^d(k) \leq a_{max} \; \forall \; i \tag{3.41}$$

a_{min} and a_{max} are the minimum and maximum allowed acceleration for all vehicles inside the CZ.

3.4 Summary

Vehicle Control chapter has discussed about the longitudinal control for cooperative autonomous vehicles in the intersection scenario. The discussions shown in this chapter were:

- Control strategy for a vehicle fulfilling cooperative strategies
- Basic *Model Predictive Control (MPC)* problem formulation.
- Longitudinal control problem, which can be solved in MPC framework.
- Application of MPC, with specific variables, to the three intersection scenarios described in this thesis

4
Simulation Environment

The intersection longitudinal controller is implemented and tested in a Simulation Environment that was designed in a project at Mechatronics Research Group, Chalmers University Of Technology, [19]. The main purpose for the development of this simulation environment is to test the active safety systems like collision avoidance, safety distance maintenance (like in intersections) with virtually simulated vehicles. The simulation environment runs on *Matlab* and *Simulink*. The simulation environment simulates virtual vehicles completely with all components like communication protocol, controller, dynamics of vehicles and sensor measurements. This simulation environment is also capable of communicating with a real autonomous test vehicle for Hardware In the Loop (HIL) simulation. An external communication hardware as per GCDC - Interaction Protocol [5] is implemented to communicate the states between the virtual vehicles and real vehicle in case of HIL.

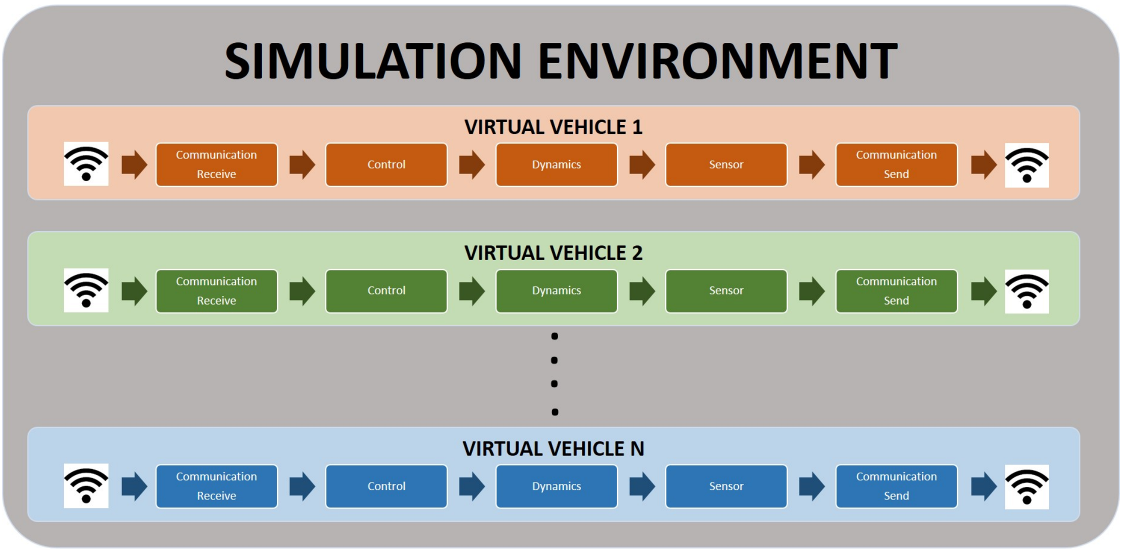


Figure 4.1: Simulation Environment

4.1 Real Time Simulation

Since the virtual vehicles running in simulation should be communicating to the real vehicles in real time, it is mandatory that the simulation should also be in real time, for this purpose, *Simulink Desktop Real Time* was used. A standalone computer runs the virtual vehicles and communicates messages of its simulated states to the real car, wirelessly.

4.2 Vehicle Paths Data

The longitudinal controller defined in Section 3.3 requires the path of the vehicles to be predefined. For convenience in our case, the path of a vehicle is generated with the *Ego Sensor Fusion* of *Volvo S60*. The continuous GPS coordinates in terms of *UTM-Northing* and *UTM-Easting* are recorded by driving the car along the paths of intersection. The GPS coordinates are in *UTM Zero* system. The recorded path is then used by the longitudinal controller. The center lanes of the roads, are also recorded in the same manner. The accuracy of the path and lanes recorded by this way is in the order of few centimeters.

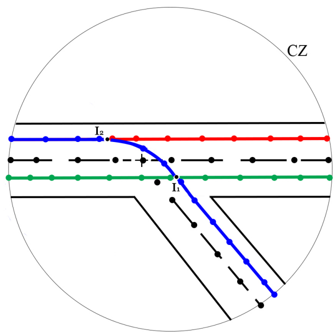


Figure 4.2: Trajectories definition from Recorded GPS data

4.3 MPC Solvers

The longitudinal MPC problem designed in this thesis should be solved by a solver in real time simulation and implementable in real car. The following MPC solvers were tried out during the course of this thesis to arrive at the most suitable solver for our case.

4.3.1 Matlab In-built Solvers

The following *Matlab* built-in solvers were tested for solving the MPC problem in hand.

4.3.1.1 *quadprog*

quadprog is a quadratic programming solver which is available as function in Matlab. The MPC problem in hand was solved with *interior point convex* algorithm. The main disadvantage with this solver was that it was not supported for real time compiling and simulation.

4.3.1.2 *mpcqpsolver*

The *mpcqpsolver* function is also an in-built function in *Matlab*. This solver used *Kwik* method to solve the MPC problem and is the same solver as used in *MPC ToolBox* of *Simulink*. The drawback of this solver is that it did not consider all the inequality constraints for solving the MPC problem.

4.3.2 CVXgen

The CVXgen is a code generation supported real time solver for convex optimization problems [21]. The CVXgen license is available free for academic licensing purposes. The generated C-code was then compiled in S-function builder block of Simulink. This S-function block is capable of accepting constraints that vary for every horizon step of the MPC problem. CVXgen fulfills our requirements of real time solvable and implementable in real car.

5

Simulation Results

This chapter discusses the simulation results of implementing the longitudinal intersection MPC on a virtual vehicle in the T-intersection scenarios considered for study in this thesis. Simulation results are shown in terms of longitudinal position, velocity and acceleration profiles followed by virtual vehicles. The virtual vehicles are communicating among themselves by means of the communication protocol described in 2.6.1, but instead of external communication software the UDP messages are transmitted and received in the host computer itself (from different ports with IP addresses 127.0.0.1). The messages are sent from UDP sent blocks of simulink to geo-networking stack and reflected back from geo-networking stack to UDP receive blocks of simulink. The geo-networking stack is written in java.

5.1 Simulation Results - GCDC Intersection Scenario

The GCDC Intersection scenario was simulated with the virtual vehicles in the *Simulation Environment* described in Section 4. In the simulated GCDC Scenario each virtual vehicle is controlled with the MPC longitudinal controller as discussed in the Section 3.
Since a competitor vehicle in GCDC intersection scenario will be a *follower* vehicle (mostly passenger cars) either $PC1$ or $PC2$, the demanded acceleration and actual acceleration, velocity and position of the follower vehicles are discussed on comparison with the leader vehicle V states. As pointed out earlier in the report the leader vehicle V will be the organisers vehicle in the competition. The whole simulation of GCDC scenario is available as a video on name **Video 1 - GCDC_scenario_Simulation** in the link
`https://drive.google.com/open?id=0B5a5I4V23aoLY0J5RVh1RlZadXM`.
The dotted pink circle in the video around the center of intersection is CZ. Here the radius of CZ is 50 m, which is same as desired by GCDC organisers. The vehicles and their paths are colored as blue for vehicle V, green for vehicle $PC1$ and red for vehicle $PC2$. The small blue circle around vehicle V is circular approximation of vehicle's dimensions. The same approximation is indicated by the green circle around vehicle PC_1 and red circle around vehicle PC_2. The radius of these smaller circles around the vehicles are equal to their lengths (assumed for simplicity in this thesis). The larger blue circle around the vehicle V is is the safety circle which vehicles PC_1 and PC_2 are not supposed to enter, and radius of this circle is equal to the sum of minimum euclidean distance ($r_{euc} = 15\ m$) and the length of the

vehicle. The center of the safety circle is GPS position of the vehicle. The GPS position of the vehicle serves as the center of all the vehicle circles. All the three vehicles are required to enter the CZ at a speed of *30 kmph* and exactly at the same time, this tests effectively the working of intersection longitudinal controller implemented in the vehicles. The big blue circle around the vehicle V is the safety circle which vehicles PC_1 and PC_2 are not supposed to enter, so that the minimum euclidean distance between the vehicles are never violated. The *minimum euclidean distance* criteria is valid until the GPS position of the vehicle V has crossed the point $P_{V,End\ of\ evaluation}$ in cyan colour along the path of vehicle V. If the vehicles PC_1 and PC_2 did not cross the pink constraint lines $P_{PC_1,Critical\ point}$ and $P_{PC_2,Critical\ point}$ respectively along their paths until the vehicle V has crossed $P_{V,End\ of\ evaluation}$. In the simulation it can be seen once the vehicle V GPS position has crossed the $P_{V,End\ of\ evaluation}$ point, PC_1 does not have constraint on its position and the vehicle PC_2 should oblige to the *desired distance* criteria. The two magneta colored circles which appear around vehicle PC_2 are desired distance constraints. The inner circle is the minimum distance limit/constraint equal to r_{euc} and outer circle is the maximum allowed distance limit/constraint equal to $(r_{euc}\ +\ h.v_V)$ from vehicle V.

5.1.1 Longitudinal States of vehicles V & PC_1

The comparison of simulated states - position, velocity and acceleration between vehicles V and PC_1 can be seen as video on name
Video 2 - V_PC1_GCDC_Scen_Simulation in the url link
`https://drive.google.com/open?id=0B5a5I4V23aoLY0J5RVh1RlZadXM`. In the simulation video it can be observed that when the vehicles V and PC_1 are approaching towards the CZ, velocity of vehicles reach the maximum allowed velocity of 30 *kmph* (8.33 m/s). After reaching the CZ the velocity of vehicle PC_1 decreases as the predicted distance of the vehicle starts hitting the constraint point $P_{PC_1,Critical\ point}$, until the vehicle V is not expected to cross its *End of evaluation* point on its path. The input acceleration for vehicle PC_1 is decided accordingly by the MPC controller, so that the required behaviour is seen in the vehicle PC_1.The vehicle PC_1 always predicts outside the MPC problem the time at which the vehicle V is expected to cross the *End of evaluation* line on it's path. Once the vehicle V is expected to cross the *End of evaluation*, the velocity of vehicle PC_1 increases accordingly, and maximum velocity is reached before the vehicle PC_1 leaves the CZ of intersection.

5.1.1.1 Acceleration of V ($a_V(k)$) & PC_1 ($a_{PC_1}(k)$)

Input Acceleration demanded of vehicle $PC1$ can be seen on Figure 5.1 along with the acceleration profile of vehicle V. Acceleration constraints are specified in the graph as black lines which are 1.5 m/s^2 and -2 m/s^2. As there is a constraint for rate of change of input acceleration, the curve of acceleration can be seen to vary in different time instants. The deceleration of vehicle PC_1 can be seen, since the distance constraint is predicted to be violated. Once the leader vehicle V has crossed the end of evaluation point $P_{V,End\ of\ evaluation}$ the demanded input acceleration rises to the maximum allowed limit.

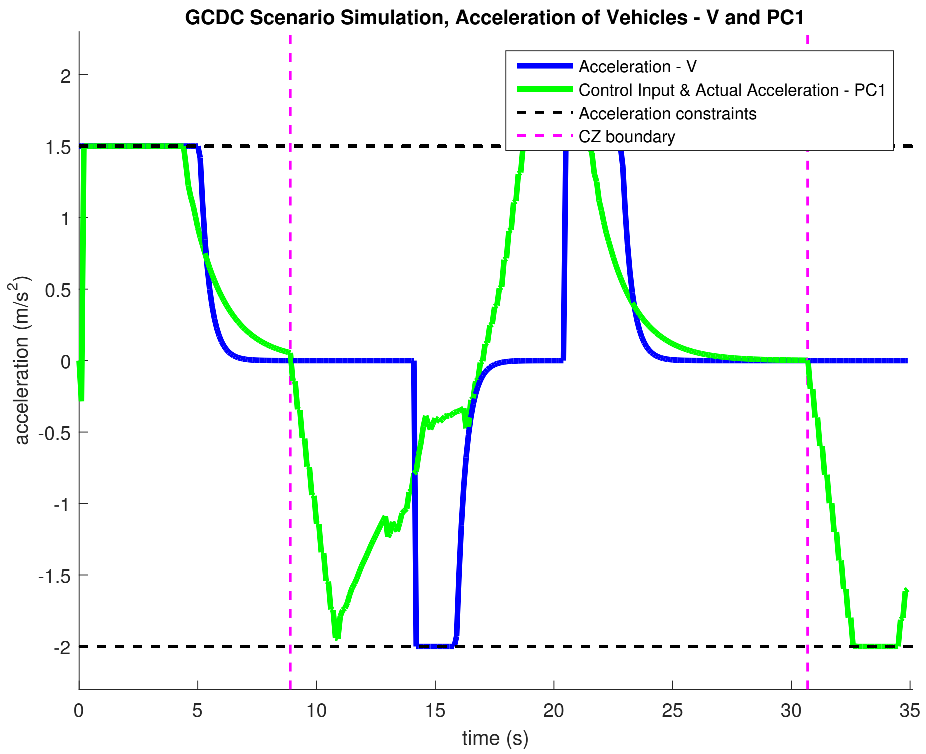


Figure 5.1: Acceleration of $PC1$ ($U_2(k)$) when $\hat{U}_{1,constant} = u_{min}$

5.1.1.2 Velocity of V ($v_V(k)$) & PC_1 ($v_{PC_1}(k)$)

The velocity comparison between the leader vehicle V and follower vehicle $PC1$ is shown in Figure 5.2. The continuous evolution of the velocity of the vehicles can be seen in the video link. The constraint for the velocities of vehicles are shown in black dotted lines which are 8.33 m/s or 30 km/h and 0.2778 m/s or 1 $kmph$. The desired velocity for the follower vehicles are 30$kmph$ since they are travelling only in straight lines. For the leader vehicle the desired velocity is 30 $kmph$ on straight roads and 20 $kmph$ on curved roads, so the dip in velocity can be seen in the velocity line (blue colour) of leader vehicle V. The follower velocity $PC1$ rises back to maximum velocity once the leader has crossed $P_{V,Intersection\ Exit}$.

5.1.1.3 Position of V ($P_V(k)$) & PC_1 ($P_{PC_1}(k)$)

The position of vehicles along the path are compared by means of distance travelled by individual vehicles along their respective path. The distance travelled by *follower* vehicle $P_{PC_1}(k)$ along its path varying across time is shown in the Figure 5.3 in comparison with the distance travelled by the *leader* vehicle V. The position along their paths both in follower vehicle $PC1$ and leader vehicle V are compared for ensuring the performance of intersection longitudinal controller. The position of intersection exit $P_{V,End\ of\ evaluation}$ for leader vehicle V and critical point for follower vehicle PC_1, $P_{PC_1,Critical\ Point}$ are shown in the Figure 5.3. In the figure it can be seen that after the vehicle *leader* V has crossed the position *End of Evaluation*, few seconds later the follower vehicle PC_2 crosses its *Critical point*. The point along

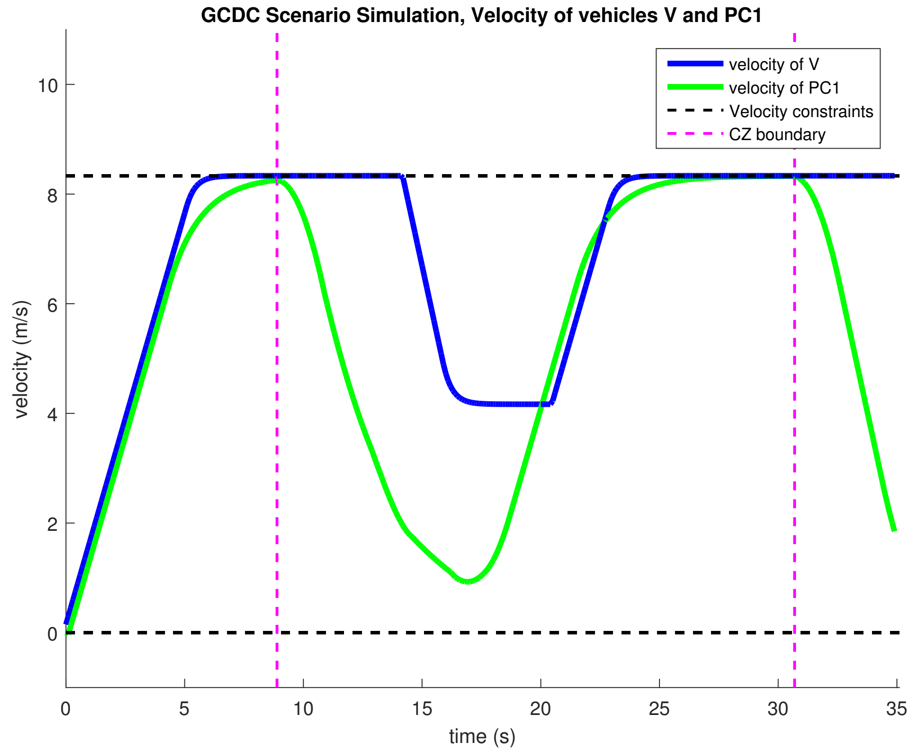


Figure 5.2: Velocity of V and $PC1$ inside CZ

the path of vehicle PC_1 at which it enters the CZ and exits CZ are shown as pink dotted vertical lines. The position $P_{PC_1}(k)$ of follower vehicle PC_1 is assumed to be zero where the vehicle starts from zero velocity. It can be seen from Figure 5.3 that the vehicle PC_1 slows down in reaching it's critical point $P_{PC_1,Critical\ Point}$ so that vehicle V is allowed to exit the end of evaluation point $P_{V,End\ of\ evaluation}$ first, so that the *minimum euclidean distance* criteria is not violated.

5.1.2 Longitudinal States of vehicles V & PC_2

The comparison of states like position, velocity and acceleration between vehicles V and PC_2 is shown as video on name **Video 3- V_PC2_GCDC_Scen_Simulation** in the link `https://drive.google.com/open?id=0B5a5I4V23aoLY0J5RVh1RlZadXM`. The end of evaluation point $P_{V,End\ of\ evaluation}$ for vehicle V is given as cyan coloured dot on the path of vehicle. The vehicle PC_2 is supposed to maintain a distance equal to d_{des} or within minimum and maximum distance limit of r_{euc} and $(r_{euc} + h.v_k(k))$, right from the time instant, the vehicle V has crossed the *End of evaluation*. So the vehicle PC_2 tries to maintain it's position in the space between two magneta coloured circles around vehicle V's - GPS position, thereby fulfilling the *desired distatnce criteria*. Here the vehicle PC_2 is maintaining an adaptive distance from vehicle V like in platooning scenario.

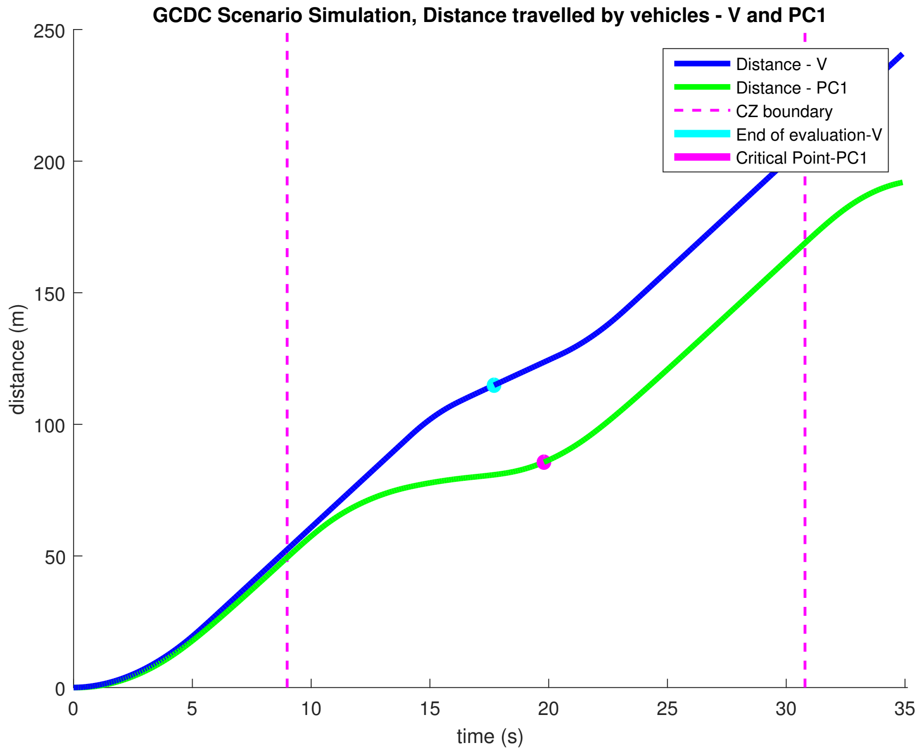


Figure 5.3: Position of V ($P_V(k)$) and $PC1$ ($P_{PC_1}(k)$) inside CZ

5.1.2.1 Acceleration of V ($a_V(k)$) & PC_2 ($a_{PC_2}(k)$)

The acceleration input for the follower vehicle $PC2$ at different time instants as decided by the longitudinal intersection MPC can be seen in the figure 5.4. Here, it is assumed that the input acceleration demanded from the virtual vehicle is obtained without any delay. The controller of vehicle PC_2 accelerates to maximum allowed velocity until reaching the boundary of CZ. After reaching CZ, the vehicle starts decelerating as it predicts that it may not fulfill the *desired distance* criterion as soon as the vehicle V crosses the *End of evaluation*. After the vehicle V has crossed the $P_{V,End\ of\ evaluation}$ the acceleration of vehicle PC_2 oscillates to keep the distance from vehicle V between the maximum allowed distance limits of *desired distance* criterion.

5.1.2.2 Velocity of V ($v_V(k)$) & PC_2 ($v_{PC_2}(k)$)

The velocity profile obtained for vehicle $PC2$ can be seen in Figure 5.5 in comparison with the velocity profile of vehicle V. When the leader vehicle V crosses the $P_{V,End\ of\ evaluation}$ the velocity of follower $PC2$ starts paralleling the velocity of leader V thus exhibiting the characteristics of a platoon follower vehicle.

5.1.2.3 Position of V ($P_V(k)$) & PC_2 ($P_{PC_2}(k)$)

The distance travelled by the leader vehicle V and follower vehicle $PC2$ is as shown in Figure 5.6. In Figure 5.6 the distance travelled by the follower vehicle PC_2 is paralleled to the distance travelled by vehicle V, once the vehicle V has crossed the

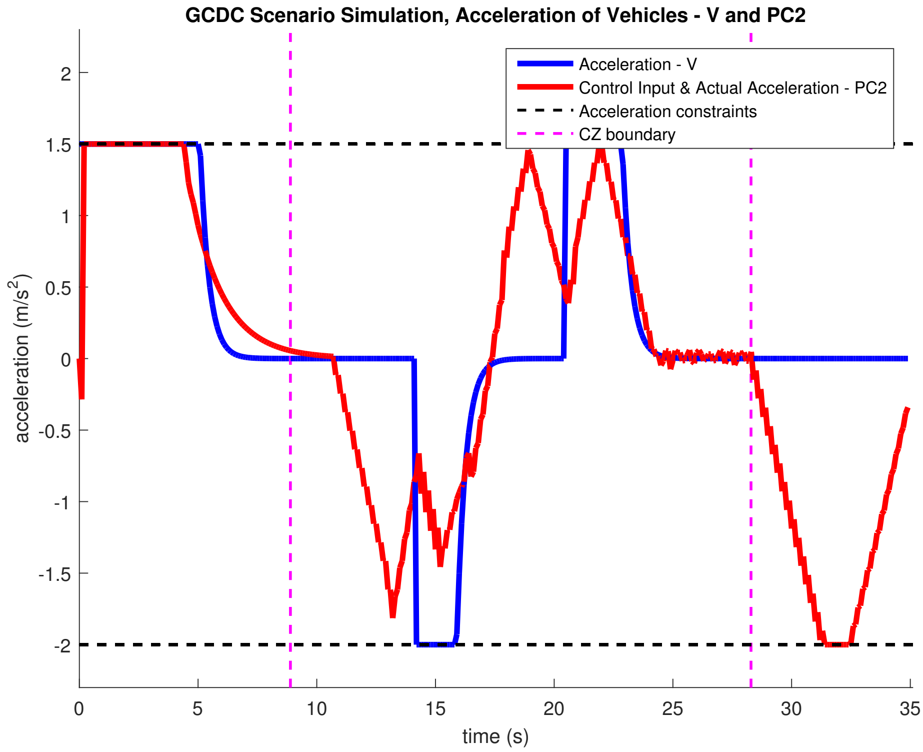


Figure 5.4: Acceleration of $PC2$ inside CZ

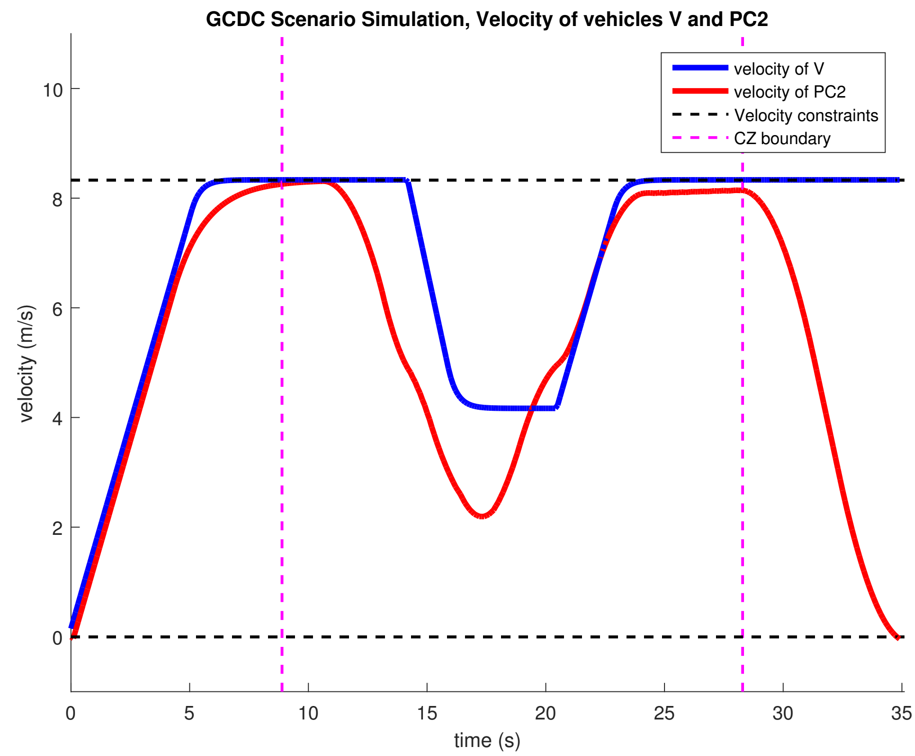


Figure 5.5: Velocity of V and $PC2$ inside CZ

End of evaluation point. This parelleled gap (at it's least) is equal to minimum safety distance r_{safe} which is required to be maintained between the leader and follower vehicles.

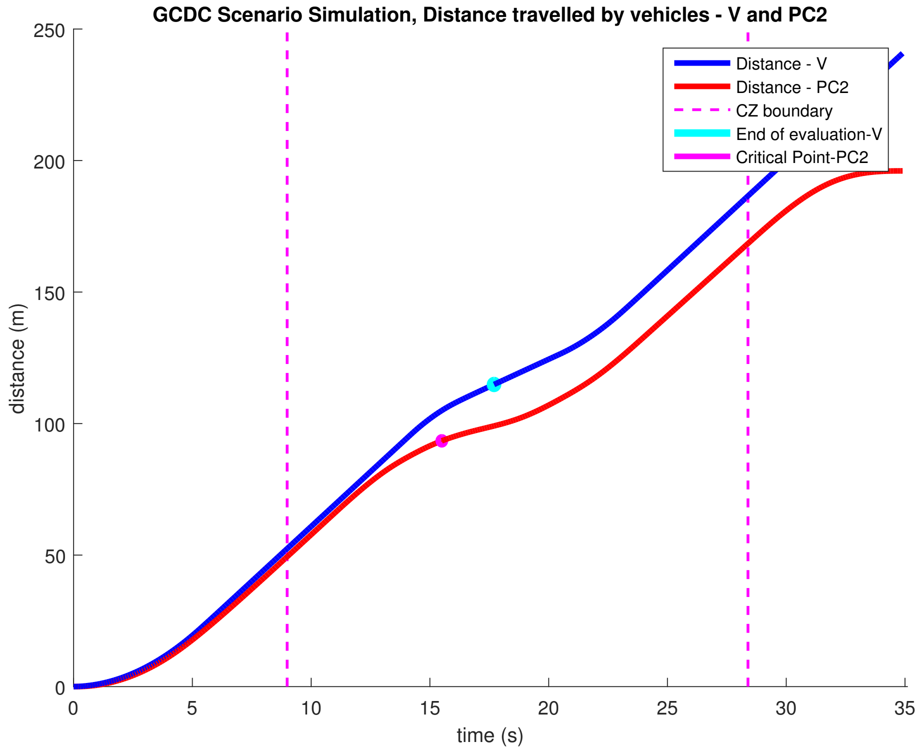


Figure 5.6: Position of V ($P_V(k)$) and $PC2$ ($P_{PC_2}(k)$) inside CZ

5.2 Simulation Results - Example Scenario 2

The simulation of Example Scenario - 2 in an intersection area as depicted in section 2.7.2 is available in video name **Video 4 - Exp__Scenario__2__Simulation** in the link
`https://drive.google.com/open?id=0B5a5I4V23aoLY0J5RVh1RlZadXM`. Here in the simulation the vehicle V_1 has *high* priority and takes *left* turn, vehicle V_2 has *medium* priority, takes *right* turn and vehicle V_3 has *low* priority and goes *straight*. The comparison in longitudinal states position, velocity and acceleration between the *leader* - *follower* vehicles combination are discussed below.

5.2.1 Longitudinal States of vehicles V_1 & V_3

The states position, velocity, acceleration evolution and comparison between vehicles V_1 and V_3 is available under video name **Video 5 - V1__V3__Exp2__Simulation__2** in the link
`https://drive.google.com/open?id=0B5a5I4V23aoLY0J5RVh1RlZadXM`. The vehicle V_3 has *medium* priority and is the *follower* of *leader* vehicle V_1. So the vehicle V_3 has to wait behind its critical point $P_{V_3,Critical\ Point}$ until the vehicle V_1 has crossed the end of evaluation point $P_{V_1,End\ of\ evaluation}$, such that vehicle V_3 fulfilling the *minimum euclidean distance* criteria.

5.2.1.1 Acceleration of V_1 ($a_{V_1}(k)$) & V_3 ($a_{V_3}(k)$)

Acceleration of vehicle V_1 and V_3 from the *Example Scenario - 2* simulation is shown in Figure 5.7. The input acceleration of follower vehicle V_3 is decided by the MPC, so that the desired velocity is reached before entering inside *CZ*. After entering CZ, the vehicle V_3 had to decelerate so that to allow the vehicle V_1 to pass and then accelerate back once there are no constraints.

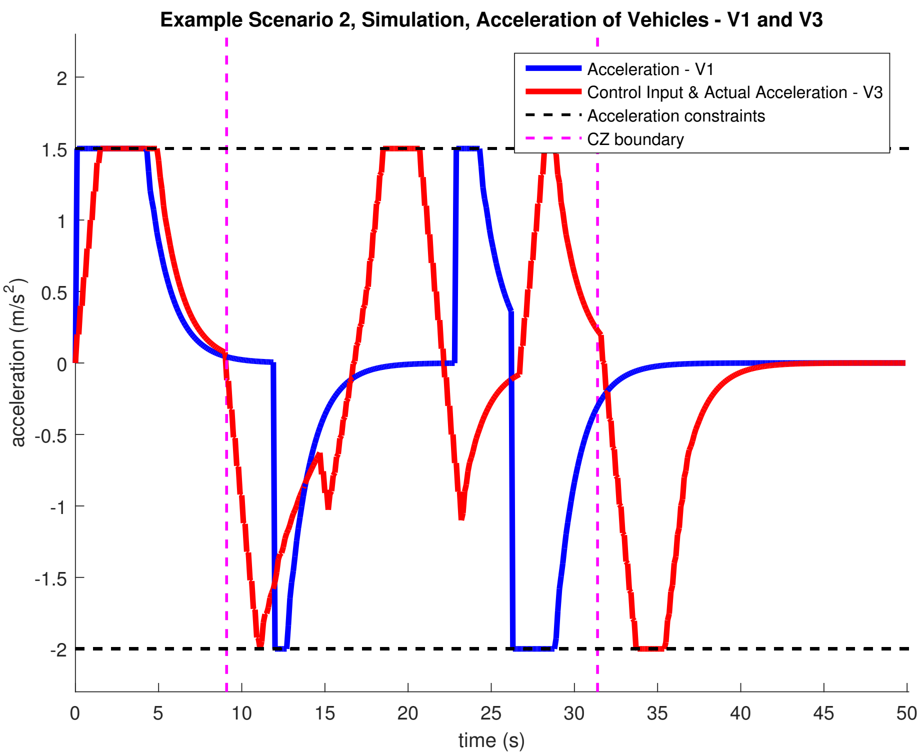


Figure 5.7: Acceleration of V_3 - Example Scenario - 2

5.2.1.2 Velocity of V_1 ($v_{V_1}(k)$) & V_3 ($v_{V_3}(k)$)

The comparison between the velocities of vehicles V_3 and V_1 can be seen in the Figure 5.8. The desired velocity of leader vehicle V_1 is 30 *kmph* and in curved path it is 20 *kmph* which can be seen as drop in velocity as the vehicle V_1 crosses the CZ. So the vehicle V_3 has to wait a bit longer time behind its critical point $P_{V_3,Critical\ Point}$ and can accelerate to desired velocity of 20 *kmph* once the vehicle V_1 has crossed the $P_{V_1,End\ of\ evaluation}$, since path of V_3 is also curved.

5.2.1.3 Position of V_1 ($P_{V_1}(k)$) & V_3 ($P_{V_3}(k)$)

In the figure 5.9 it can be seen that vehicle V_1 crosses the *End of evaluation* position after which the vehicle V_3 crosses it's *Critical point*.

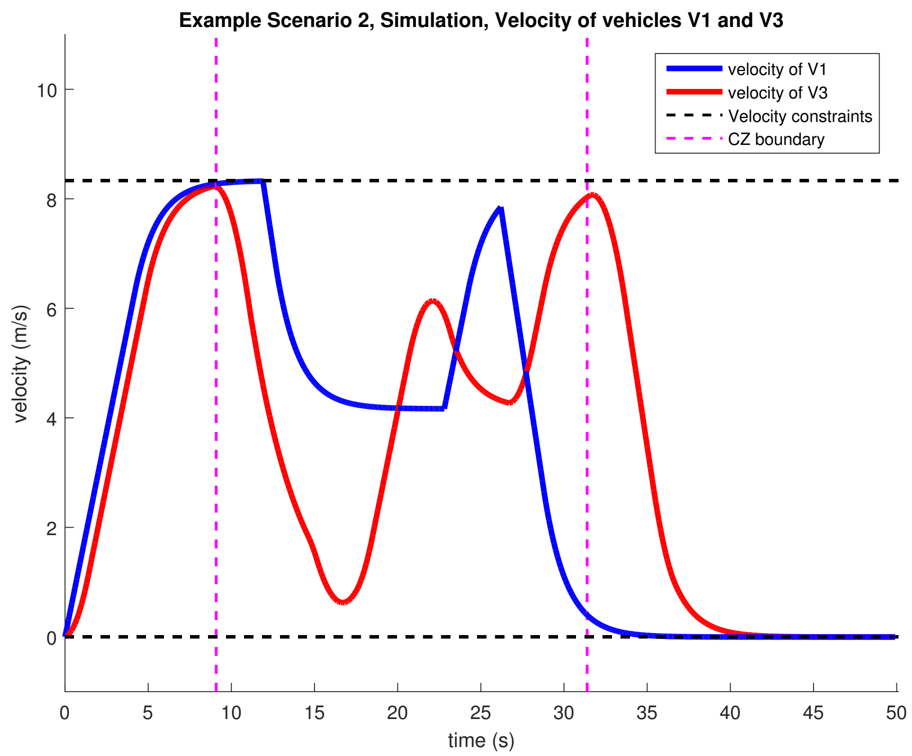


Figure 5.8: Velocity of V_1 and V_3 - Example Scenario - 2

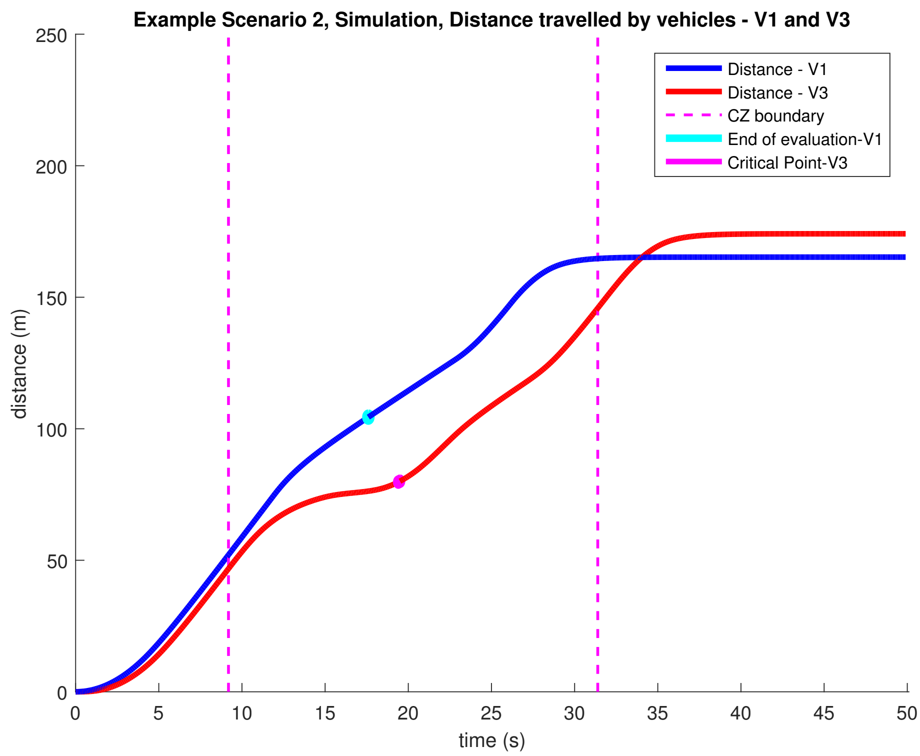


Figure 5.9: Position of V_1 and V_3 - Example Scenario - 2

5.2.2 Longitudinal States of vehicles V_3 & V_2

The states position, velocity, acceleration evolution and comparison between vehicles V_2 and V_3 under video name **Video 6 - V3_V2_Exp2_Simulation** in the

link
`https://drive.google.com/open?id=0B5a5I4V23aoLY0J5RVh1RlZadXM`. The vehicle V_2 has *low* priority and is the *follower* of vehicle V_3, so the vehicle V_2 has to wait behind its critical point $P_{V_2,Critical\ Point}$ until the vehicle V_3 has crossed the end of evaluation point $P_{V_2,End\ of\ evaluation}$, such that fulfilling the *minimum euclidean distance* criteria. This manoeuvre of vehicle V_2 ensures no collision between vehicles V_3 and V_2 while crossing the intersection.

5.2.2.1 Acceleration of V_3 ($a_{V_3}(k)$) & V_2 ($a_{V_2}(k)$)

As shown in Figure 5.10, the acceleration profile of vehicle V_2 deceleration time period (the region of graph below 0 m/s^2) is quite long so that the vehicle V_3 exits the evaluation point $P_{V_3,End\ of\ evaluation}$ at its own phase.

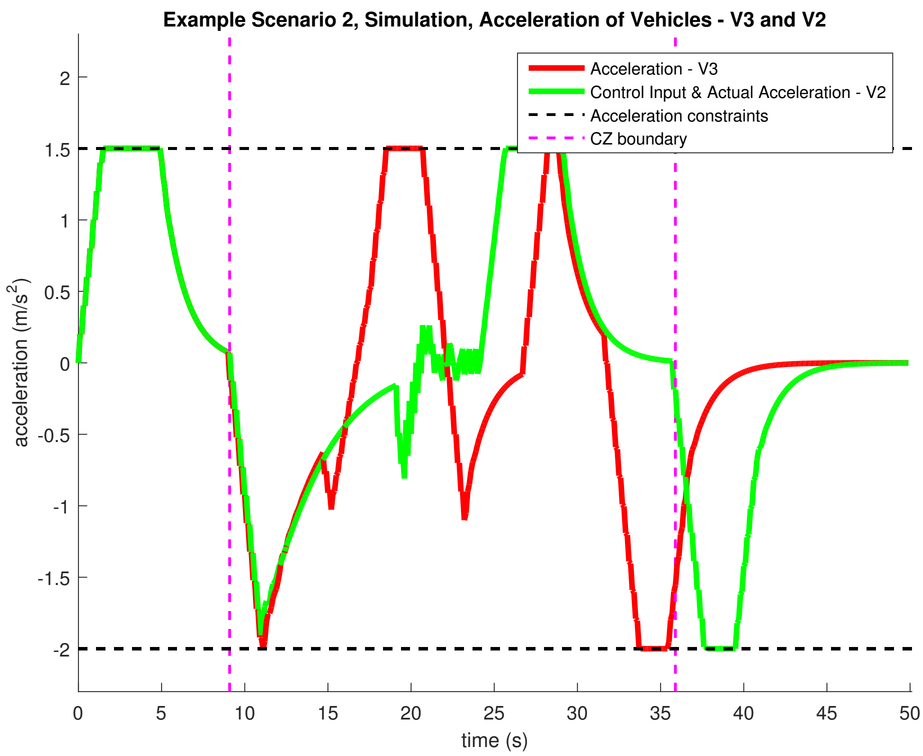


Figure 5.10: Acceleration of V_2 - Example Scenario - 2

5.2.2.2 Velocity of V_2 ($v_{V_2}(k)$) & V_3 ($v_{V_3}(k)$)

The comparison between the velocities of vehicles V_2 and V_3 can be seen in the Figure 5.11. The vehicle V_2 avoids collision with the leader vehicle V_3 by reducing its velocity. The vehicle V_3 velocity is interrupted by it's *leader* vehicle V_1 and is further slowed down by the allowed maximum velocity at the curves which is 20 *kmph*. So the vehicle V_2 slows down for a bit lengthier time until the vehicle V_3 crosses the end of evaluation point $P_{V_3,End\ of\ evaluation}$.

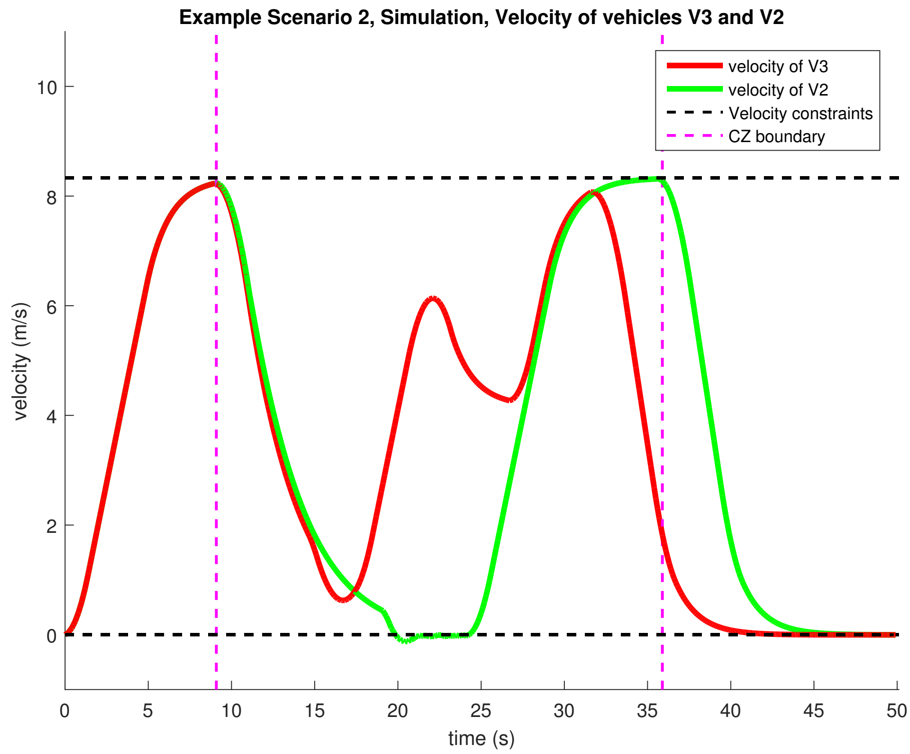


Figure 5.11: Velocity of V_3 and V_2 - Example Scenario - 2

5.2.2.3 Position of V_2 ($P_{V_2}(k)$) & V_3 ($P_{V_3}(k)$)

The vehicle V_2 has *low* priority and is the *follower* of *leader* vehicle V_3. So the vehicle V_2 stays behind its critical point $P_{V_2,Critical\ Point}$ until the vehicle V_3 has crossed the end of evaluation point $P_{V_3,End\ of\ evaluation}$. Figure 5.12 show the relation in position between vehicles V_2 and V_3, the crossing of *End of evaluation* point by V_3 and the time of crossing of *Critical point* by V_2

5.3 Simulation Results - Example Scenario 3

The complete simulation of *Example Scenario - 3* as explained in section 2.7.3 can be seen under the video name **Video 7 - ExpScenario_3_Simulation** in the link
`https://drive.google.com/open?id=0B5a5I4V23aoLY0J5RVh1RlZadXM`. In this scenario, the path of vehicle V_1 has *high* priority and vehicle V_2 has *medium* priority and vehicle V_3 has *low* priority. The path of vehicles V_1 and V_2 are not intersecting, so there is no collision avoidance manoeuvre executed by vehicle V_2. The path of vehicle V_3 intersects with both V_1 and V_2 vehicles' path. The vehicle V_3 is the *follower* vehicle between both the vehicle pairs $V_1 - V_3$ and $V_1 - V_2$, since it has its path intersecting with both the *leader* vehicles V_1 and V_2. There is a single most conservative *Critical point* on the path of the vehicle V_3, so that collision is avoided by vehicle V_3 between V_1 and V_2, such that *minimum euclidean distance* criteria is fulfilled by vehicle V_3. When the vehicles V_1 and V_2 have crossed their *End of evaluation* points, the vehicle V_3 maintains a safety distance from vehicle V_2 fulfilling

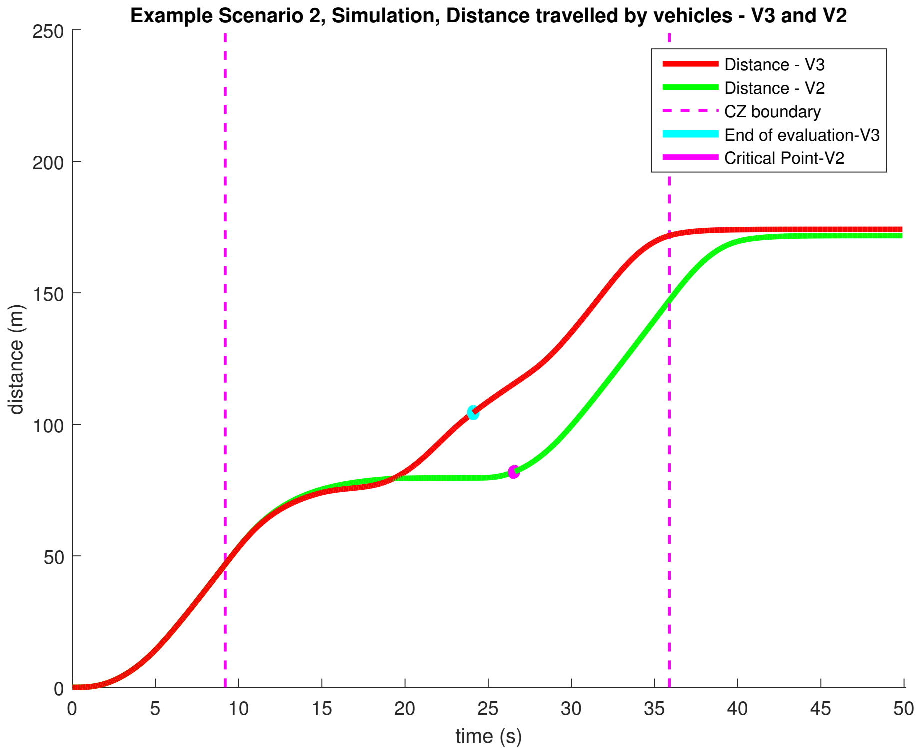


Figure 5.12: Position of V_3 and V_2 - Example Scenario - 2

desired distance criteria. It can be noticed that the vehicle V_2 and V_3 are travelling in the same path after crossing the intersection. The vehicles' speed around the intersection were around maximum speed of 20 *kmph* in the curved path.

5.3.1 Longitudinal States of Vehicles V_2 & V_3

The comparison in longitudinal states position, velocity and acceleration between the *follower* vehicle V_3 and the *leader* vehicle V_2 is available as a video in the name **Video 8 - V2_V3_ExpScen3_simulation** under the link `https://drive.google.com/open?id=0B5a5I4V23aoLY0J5RVh1R1ZadXM`. The vehicle V_1 states are not compared here since including those in graphical representation would be complicated and also the conservative *Critical point* made by vehicle V_1 on the path of vehicle V_3 is of importance for collision avoidance. Vehicle V_2 is most critical in influencing the collision avoidance strategy of vehicle V_3, before, during and after crossing its *End of evaluation*. So the comparison of states between vehicles V_2 and V_3 are considered below.

5.3.1.1 Acceleration of V_2 ($a_{V_2}(k)$) & V_3 ($a_{V_3}(k)$)

The input demanded acceleration for the follower vehicle V_3 is decided by the longitudinal MPC such that the velocity and position profile for the vehicle are attained appropriately. The deceleration profile helps vehicle V_3 to stay behind it's critical point until the vehicle V_1 exits *End of evaluation* point and the oscillating acceleration curve aids in platooning behind the vehicle V_2 till the exit of CZ. The comparison between the acceleration of vehicle V_2 and input acceleration demanded by MPC in vehicle V_3 are shown in Figure 5.13

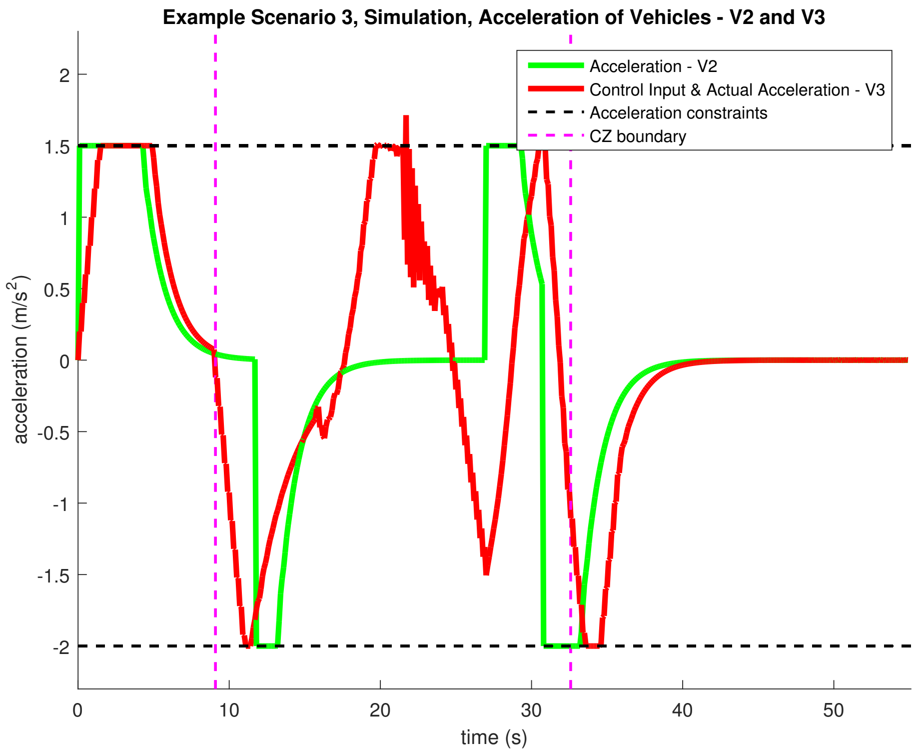


Figure 5.13: Acceleration of V_2 and V_3 - Example Scenario - 3

5.3.1.2 Velocity of V_2 ($v_{V_2}(k)$) & V_3 ($v_{V_3}(k)$)

Velocity comparison between vehicles V_2 and V_3 can be seen in the graph 5.14 The velocity of vehicle V_3 is reduced to the lowest possible by the longitudinal MPC, so that not to violate the *critical point*, $P_{V_3,Critical\ point}$ along the path of vehicle V_3. V_1 and V_2 are not constrained in terms of position are slowing down to 20 *kmph* while taking the turn. After crossing the curve vehicle V_2 switches back to maximum velocity of 30 *kmph* on straight curve while nearing the exit of CZ. Vehicle V_3, parallels the velocity of V_2 while platooning behind vehicle V_2.

5.3.1.3 Position of V_2 ($P_{V_2}(k)$) & V_3 ($P_{V_3}(k)$)

The position comparison of vehicle V_3 and V_2 can be seen in Figure 5.15. The vehicle V_3 gets inside the CZ when it has crossed the dotted pink vertical lines, stays behind the critical point $P_{V_3,Critical\ Point}$ as long as the vehicles V_1, V_2 have not crossed their respective evaluation points $P_{V_1,End\ of\ evaluation}$, $P_{V_2,End\ of\ evaluation}$. The critical point $P_{V3,Critical\ Point}$ was created by the intersection of minimum euclidean distance circle around vehicle V_1 on the path of vehicle V_3, since it is the most conservative point when realised from the view of vehicle V_3. Near the exit of CZ, the distance travelled by V_3 is almost parallel to the the distance travelled by V_2 which are set apart by desired safety distance.

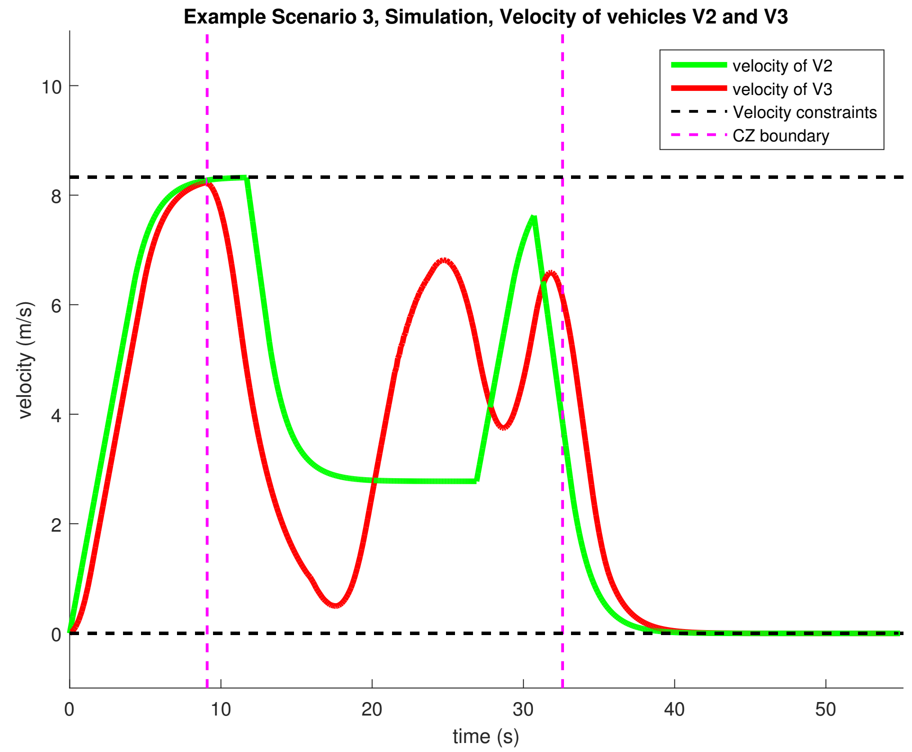


Figure 5.14: Velocity of V_2 and V_3 - Example Scenario - 3

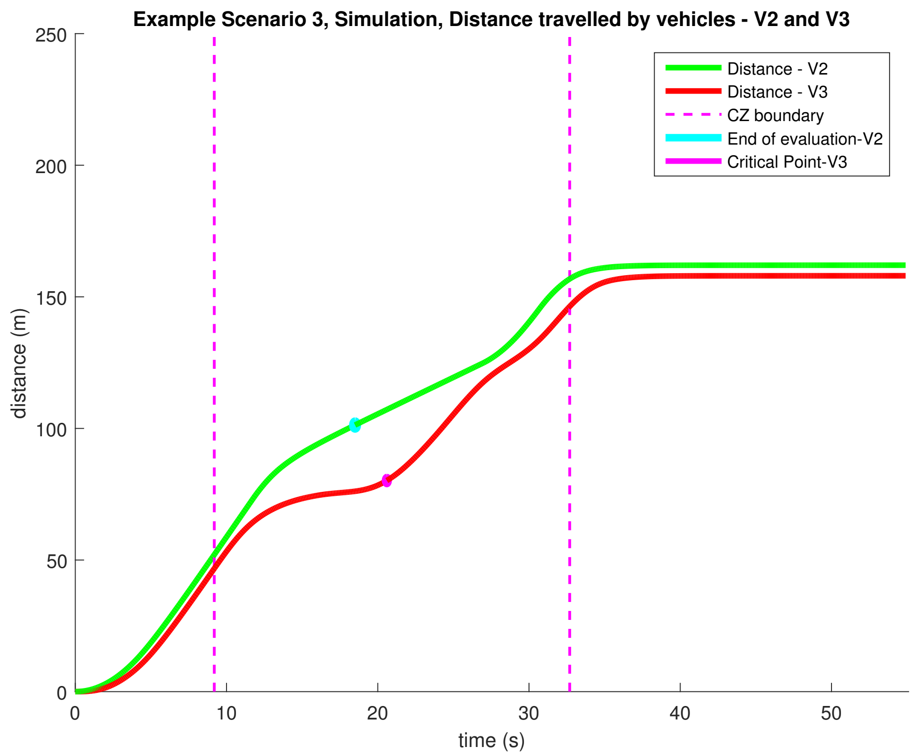


Figure 5.15: Position of V_2 and V_3 - Example Scenario - 3

6

Hardware In Loop (HIL) Testing Results

This chapter describes the Hardware In Loop (HIL) simulation results in the intersection scenarios. The HIL simulation is performed by replacing one of the virtual vehicle by a real autonomous vehicle, a Volvo S60 as experimented in [19]. The virtual vehicles run in Simulation Environment on a Simulink Desktop Real Time software module in a separate standalone laptop computer and communicate to the real car through the wireless communication hardware. Since the virtual vehicles run on a real world GPS coordinates map as the real vehicle, both the virtual and real vehicles are in the same intersection area and conventions of positioning. The real car was placed as one *follower* at a time and the position, velocity, input acceleration and actual acceleration of the real vehicle are graphed in comparison with its respective virtual *leader* vehicle. The experimental setup in the real car which aided the implementation of control algorithms will be discussed in detail in upcoming sections.

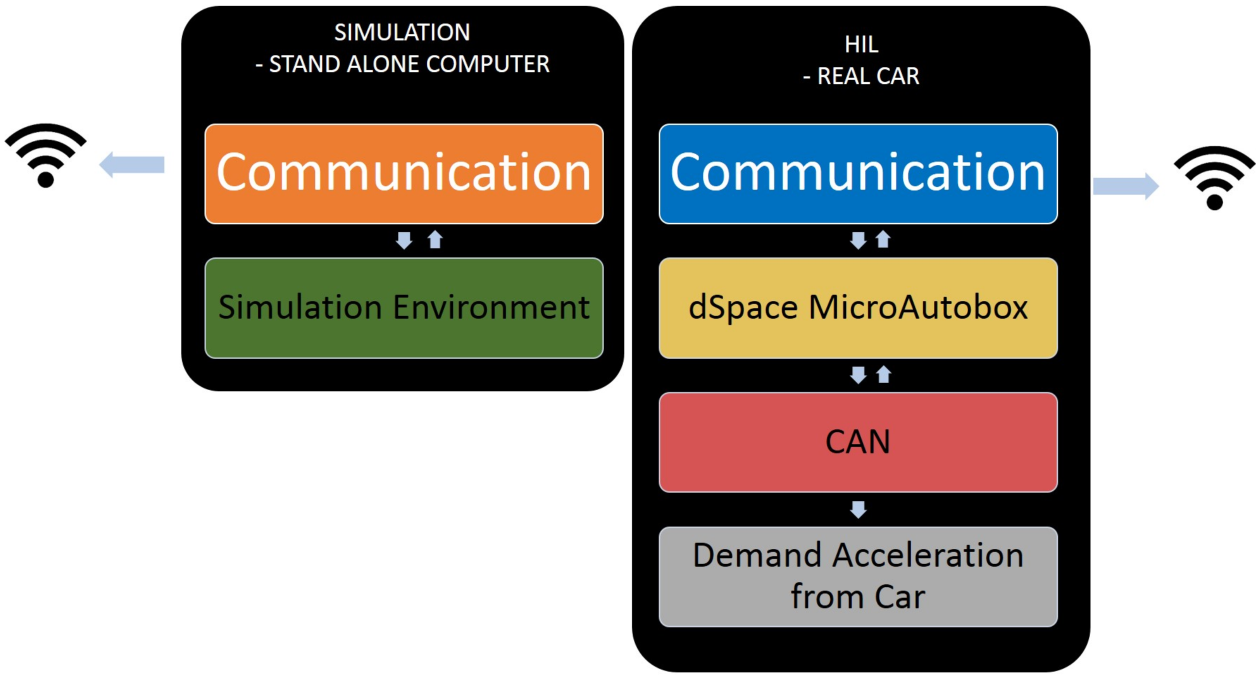


Figure 6.1: Hardware In Loop (HIL) Simulation Scheme

6.1 Experimental Setup

The experimental setup inside the Volvo S60 is as shown in Figure 6.2. The components of set up are

1. dSpace Micro Autobox
2. Trimble GPS device
3. Wifi Modem
4. Alexis Communication hardware
5. Ethernet switches
6. CAN buses

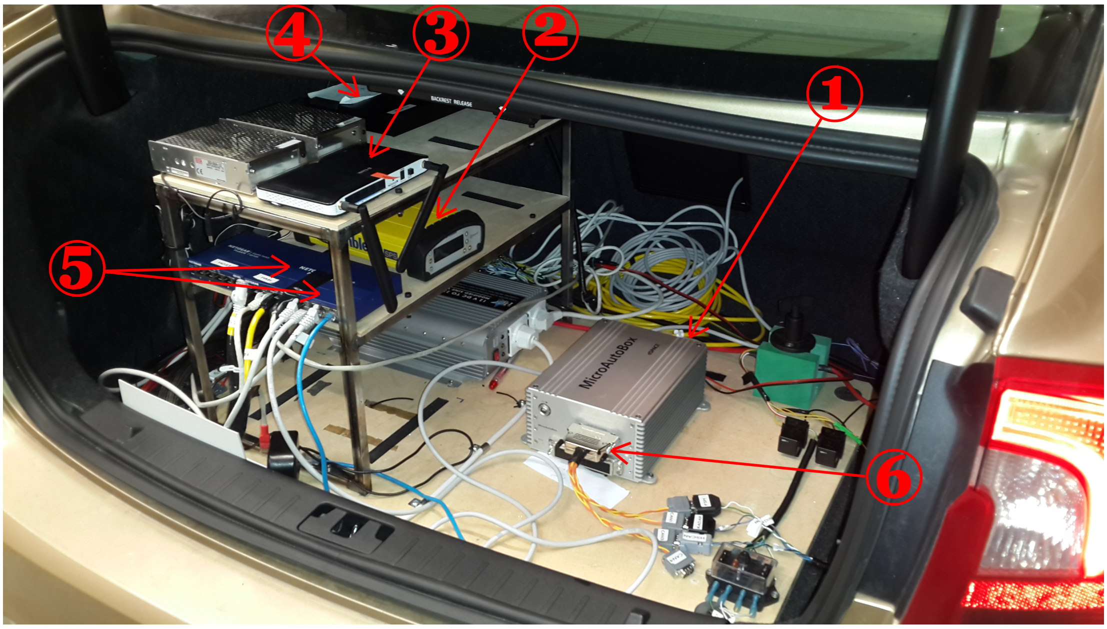


Figure 6.2: Experimental Setup inside Volvo S60

dSpace Micro Autobox is a software rapid prototyping device which communicates with the CAN bus of the car and connect with other devices through ethernet. dSpace Micro Autobox can be interfaced through *Control Desk* software and have the capability to rapid prototype *Matlab/Simulink* programmes in real time. The MPC control algorithm that was tested on virtual vehicles in simulink environment can be run on Micro Autobox and demanded acceleration signal can be sent to the cruise control computer of the car.
CAN Bus help in accessing the important signals that are available from and about the car. The CAN also sends control signals (demanded acceleration) from the MPC control algorithms running in dSpace to the cruise controller of the car. The velocity, actual acceleration signal of the CAN are used by the sensor fusion algorithms[23] to provide the actual physical states as required by MPC control algorithm.
Trimble GPS is the instrument to acquire the current longitude and longitude position of the vehicle. The longitude and latitude measured is the center of the car where the receiver for the device is actually placed. The device can provide accurate GPS data upto to few millimeters when coupled with Real Time Kinematics (RTK) corrections. RTK corrections can be acquired through a base station or through a

web server, where the help of sim card insert-able modem comes to play.
Alexis Communication Hardware is a computer running geo-networking stack as required by the GCDC communication protocol [5]. The hardware and software for running the geo-networking stack was programmed and implemented by Albin Zeverinson as in [10]. The communication hardware transmits packets of Cooperative Awareness Messages(CAM), Decentralised Environment Notification Message (DENM) and iGame Cooperative Lane Change message (iCLCM) at frequencies of 25 Hz, 10 Hz and 25 Hz respectively as required by the GCDC protocol [5]. The packets are sent to the hardware from the simulink programme running in Micro-Autobox. messages packed messages sent to it from the simulink interface. The communication hardware also receives the messages broadcasted by other similar communication hardware. For transmitting messages from the virtual vehicles in HIL simulation the same hardware is used but the hardware gets the CAM, DENM, iCLCM packages from the simulink desktop real time interface running in a stand alone computer/laptop. The vehicular communication is established for HIL simulation is V2V communication and for the GCDC competition - V2V, V2X communication were established between organisers and competitors' hardware. The states that are exchanged in the message sets which are useful for intersection scenario were in accordance with the description as in section 2.6.1.

Ethernet Switches are the connection junction of ethernet/LAN cables connecting different devices like Trimble GPS device, communication hardware, modem and the dSpace MicroAutobox.

6.2 Ego and Participant Vehicles Sensor Fusion

The sensor fusion algorithm applied Extended Kalman Filters (EKF) for fusing the longitudinal states of ego vehicle and participant vehicles in the environment around ego vehicle. More detailed explanation about the sensor fusion algorithms for the ego and participant vehicles in the vicinity can be found in 'Sensor Fusion - Technical documentation' by Robert Hult, Marco di Vaio et al [23].

6.3 HIL Testing Results - GCDC Scenario

Hardware in the Loop testing for the GCDC scenario was performed considering, the *leader* vehicle V as virtual and Volvo S60 replaces *follower* vehicle PC_1 in first set of trial and PC_2 in second set of trials. Replacing the *follower* virtual vehicles with real vehicle would test the collision avoidance manoeuvre of real vehicle with its virtual *leader* vehicle. Later the same behaviour can be expected from *follower* virtual vehicle, even if the virtual *leader* vehicle are replaced by real *leader* vehicle. HIL also tests the behaviour of real vehicle in terms of longitudinal states position, velocity and acceleration when compared to the results obtained in simulation environment. The resultant longitudinal control behaviour of real car is expected to be the same as of virtual vehicle counterpart in the simulation. The states behaviour discussed below are that of the real vehicle *Volvo S60* which has sensor fusion for the longitudinal states. In all the HIL testing the real test vehicle is tuned to highest scores in all the evaluation criteria like *minimum euclidean distance*, *desired distance*, *maximum velocity limit*. The *fastest finish time* criteria was tuned for maximum performance in a way that the test vehicle stays for the lowest time possible inside the CZ.

6.3.1 Longitudinal States of virtual V and real PC_1

In this particular HIL trial the virtual PC_1 vehicle as in simulation from section 5.1.1 was replaced with the real vehicle *Volvo S60* and GCDC scenario was imitated. The real vehicle S60 should enter into CZ at maximum velocity of 30 *kmph* within 20 seconds from the point start signal has been provided. For this purpose for the vehicle reaching exact position at determined time and velocity, the same longitudinal MPC with different set of constraints in position and number of time steps are provided. The change in longitudinal states of position, velocity and acceleration of Volvo S60 as vehicle PC_1 when compared with vehicle V can be seen as plotting video **Video 9 - PC1_GCDC_Scenario_HIL** under the link `https://drive.google.com/folderview?id=0B5a5I4V23aoLY0J5RVh1RlZadXM&usp=sharing` and individual state performances are detailed below.

6.3.1.1 Acceleration of V $(a_V(k))$ & $PC_1,\ S60$ $(a_{PC_1}(k))$

The acceleration comparison graph of HIL testing is a bit different from the acceleration comparison graph of simulation. The ego vehicle considered for study has two different components displayed in the graph a demanded input control signal from the controller and actual acceleration of the test vehicle. The actual acceleration of the test vehicle will be a delayed quantity by nearly 0.5 s from the time an acceleration is demanded by the controller. Recalling the section 6.3, this delay between input and actual acceleration was already modelled in equation 6.3. Before the test vehicle reaches the CZ the controller demands acceleration beyond the constraints because of the distance at which the vehicle is positioned from the CZ and the problem becomes infeasible until the test vehicle is predicted to reach CZ.Inside CZ the MPC problem is feasible and the problem is converged so the input calculated is optimal at all time instants. The input acceleration curve for test vehicle PC_2 in HIL testing is almost identical to the acceleration graph obtained in respective simulation.

6.3.1.2 Velocity of V $(v_V(k))$ and $PC_1,\ S60$ $(v_{PC_1}(k))$

Velocity of the real test vehicle as PC_1, which is the output of ego sensor fusion is compared with the velocity of virtual vehicle V as in Figure 6.4. The maximum velocity, reached by the test vehicle as shown in the graph is always less than the maximum constraint value due to some measurement signal adjustments from the car. The velocities displayed in the dash board of the test vehicle is always higher than the value shown in the graph. It can be seen in the graph that almost maximum velocity is reached by the test vehicle when it enters CZ after which the velocity of vehicle drops to value a little above minimum velocity constraint, in a strategy to

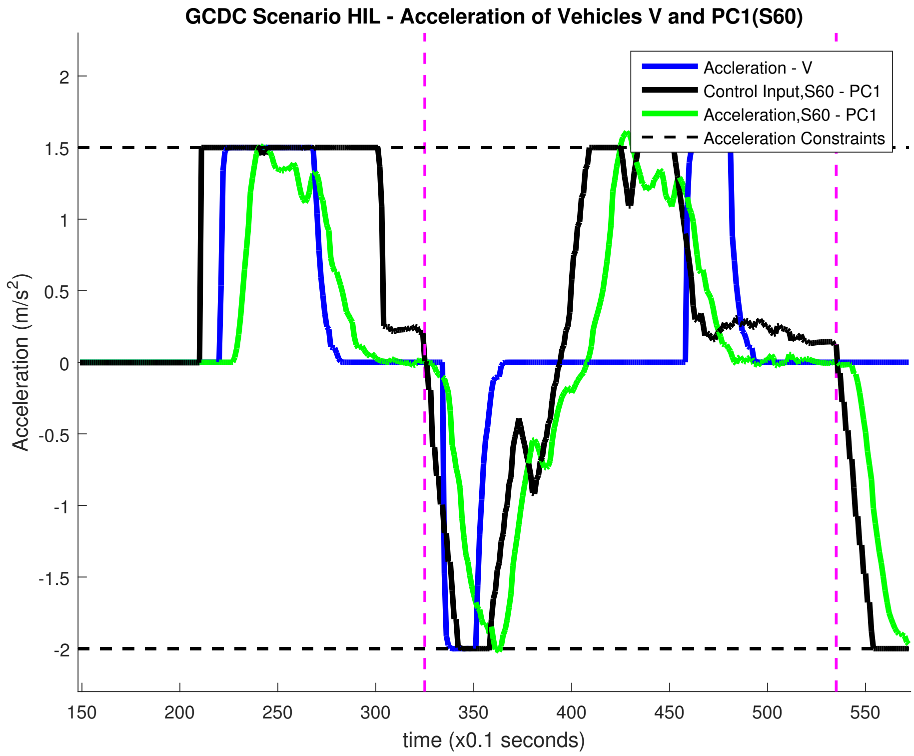


Figure 6.3: Acceleration of V and $PC1$ - GCDC Scenario HIL

avoid violating *Critical point* position constraint. After the vehicle V has crossed it's *End of evaluation* point, the velocity of test vehicle PC_1 rises back to the maximum possible until it reaches the exit of CZ.

6.3.1.3 Position of V ($P_V(k)$) and PC_1, $S60$ ($P_{PC_1}(k)$)

The position covered by the test vehicle over time is represented in terms of distance travelled same as in the case of virtual vehicles. The distance of real vehicle when it starts from the *CZ Entry* point should be known when the scenario start signal is given which plays a role for the vehicle to reach the CZ at exact time and velocity. Figure 6.5 shows the graph of distance covered by PC_1 as a real vehicle, which has similar behaviour as in the virtual vehicle simulation. It can also be noted that virtual vehicle V crosses its *End of evaluation* point first and then the real test vehicle as PC_1 crosses its *Critical point* thereby avoiding collision by creating safety space at the center of intersection. It can also be noted that the distance travelled by virtual vehicle V and real vehicle PC_1 are almost equal until the point where the real vehicle predicts reaching its *Critical point*

6.3.2 Longitudinal States of virtual V and real PC_2

This HIL testing was done in GCDC scenario, by placing the real test vehicle in the position of PC_2. The longitudinal states of the real test vehicle as *follower* PC_2 is studied on comparison with virtual *leader* vehicle V and the simulation is available as video **Video 10 - PC2_GCDC_Scenario_HIL.avi** under the link `https://`

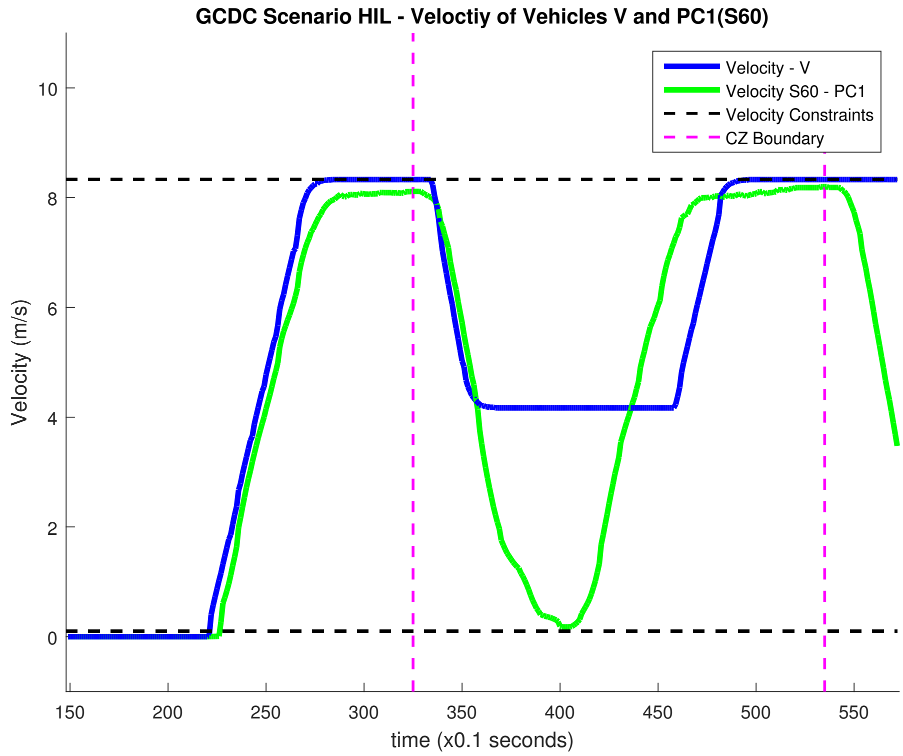


Figure 6.4: Velocity of V and $PC1$ - GCDC Scenario HIL

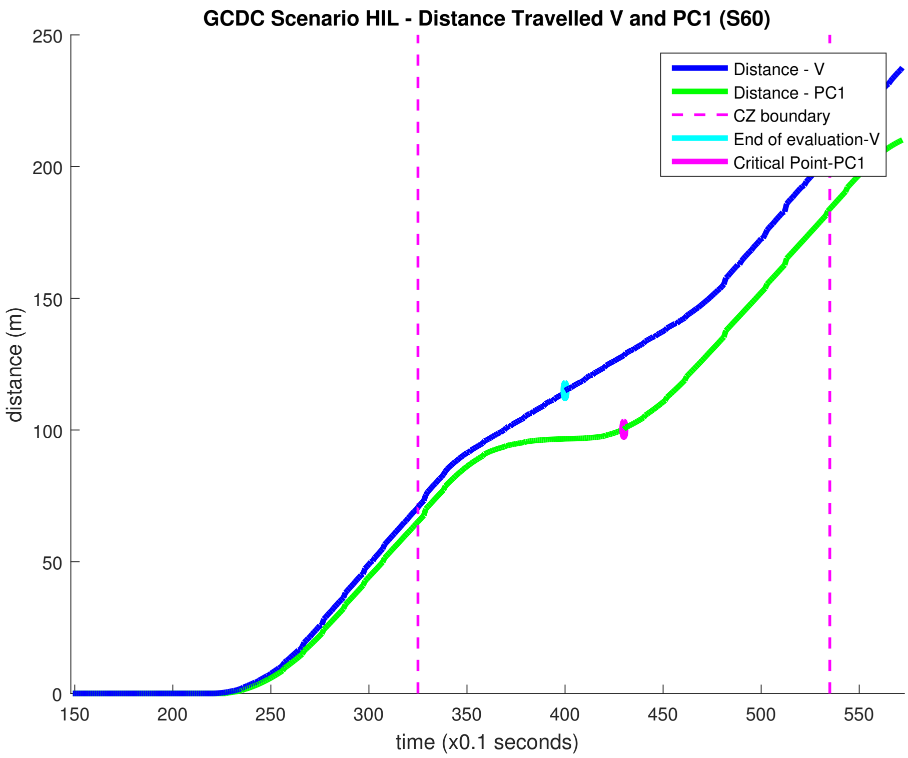


Figure 6.5: Position of V and $PC1$ - GCDC Scenario HIL

drive.google.com/folderview?id=0B5a5I4V23aoLY0J5RVh1RlZadXM&usp=sharing.

6.3.2.1 Acceleration of V $(a_V(k))$ & PC_2,$S60$ $(a_{PC_2}(k))$

The comparison between acceleration curves of vehicle V and test vehicle PC_2 are shown in Figure 6.6. In the graph, actual acceleration on test vehicle PC_2 is delayed from demanded input acceleration by 0.5 s. A sinusoidal wave behaviour can be noticed in input acceleration demanded by MPC which is denoting the intention of test vehicle PC_2 to platoon the *leader* vehicle V.

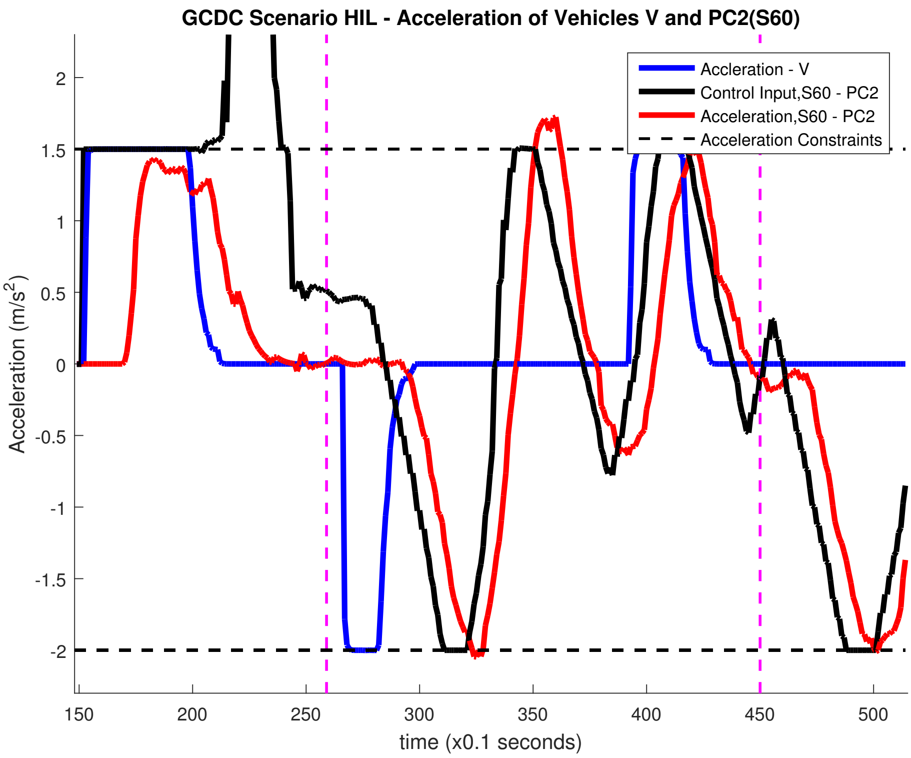


Figure 6.6: Acceleration of V and $PC2$ - GCDC Scenario HIL

6.3.2.2 Velocity of V $(v_V(k))$ and PC_2, $S60$ $(v_{PC_2}(k))$

Figure 6.7 shows the comparison of velocity state between virtual vehicle V and the real test vehicle PC_2. Here the vehicle V is the *leader* and PC_2 is the *follower*. It can be seen from Figure 6.7 that test vehicle PC_2 slows down and stays a little above minimum constraint velocity until the vehicle V crosses *End of evaluation* point. Once the vehicle V crosses its *End of evaluation* point, the test vehicle PC_2 speeds up unless it is not predicting to violate the *desired distance* criteria. When the test vehicle PC_2 starts fulfilling the *desired distance* criteria, the velocity of PC_2 is paralleled with the velocity of V indicating the platooning behaviour between them.

6.3.2.3 Position of V $(P_V(k))$ and PC_2, $S60$ $(P_{PC_2}(k))$

In Figure 6.8, it can be noted that the real test vehicle PC_2 as a *follower* have travelled exactly equal distance as the virtual *leader* vehicle V at a point of time

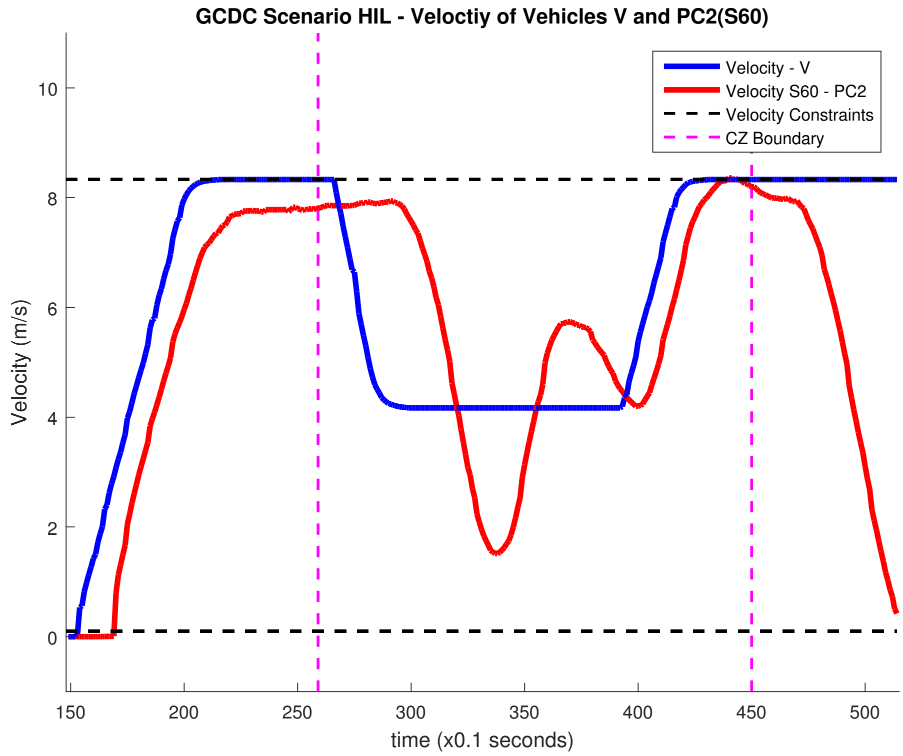


Figure 6.7: Velocity of V and $PC2$ - GCDC Scenario HIL

after entering into CZ. But the once the real test vehicle PC_2 gets closer to the position of virtual vehicle V it starts maintaining an uniform platooning safety distance from the virtual vehicle thereby fulfilling the *desired distance criteria* as required in GCDC.

6.4 HIL Testing Results - Example Scenario 2

In this HIL testing section the real test vehicle replaces *follower* vehicles V_3 and V_2 in Example Scenario 2. The longitudinal states behaviour of test vehicle are studied and results are presented as graphical representations. The expected results from real vehicle in HIL testing is required to be the same as in case of virtual vehicle simulation.

6.4.1 Longitudinal States of virtual V_1 and real V_3

HIL testing for Example Scenario 2, in which the virtual vehicle V_3 is replaced with real test vehicle, Volvo S60 and the longitudinal MPC was tested. Here in this case the virtual vehicle V_1 is the *leader* of vehicle V_3. The simulation representing the position, velocity and acceleration longitudinal states of virtual vehicle V_1 and test vehicle V_3 is as shown in the video **Video 11 - V3_ExpScen2_HIL** under the link `https://drive.google.com/folderview?id=0B5a5I4V23aoLY0J5RVh1RlZadXM&usp=sharing`

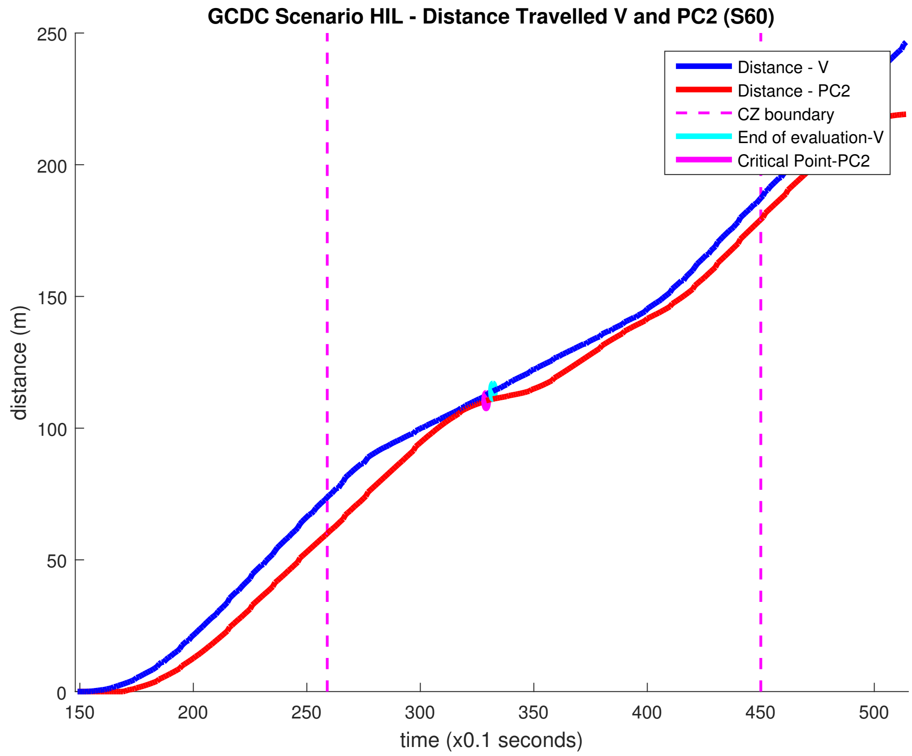


Figure 6.8: Position of V and $PC2$ - GCDC Scenario HIL

6.4.1.1 Acceleration of V_1 ($a_{V_1}(k)$) and V_3, $S60$ ($a_{V_3}(k)$)

Figure 6.9 shows the input acceleration demanded by MPC from test vehicle in place of V_3. It can be seen from the simulation video for this case that the real and virtual vehicles didn't reach the CZ at same time, but the vehicle V_3 entered CZ first and had to wait until behind it's *Critical Point* a little longer before the virtual vehicle V is expected to cross it's *End of evaluation* point which caused the oscillations in demanded control input in the deceleration zone.

6.4.1.2 Velocity of V_1 ($v_{V_1}(k)$) and V_3, $S60$ ($v_{V_3}(k)$)

Figure 6.10 shows the velocity graph comparison between the real test vehicle V_3 and virtual vehicle V. It was intentional in the experiment to setup a test where the vehicle V reaches maximum speed a bit later than real vehicle to check the effectiveness of collision avoidance manoeuvre by the controller in V_3. The almost constant velocity of vehicle V_3 at around 4 m/s^2 is due to the fact that the vehicle V_3 is taking a curved path after crossing it's *Critical Point*.

6.4.1.3 Position of V_1 ($P_{V_1}(k)$) and V_3, $S60$ ($P_{V_3}(k)$)

The test vehicle in place of vehicle V_3 in Example Scenario 2 has the following distance travelled characteristics over time as shown in Figure 6.11. In the graph it can be noticed that the distance travelled by test vehicle V_3 is higher when it crossed it's *Critical Point*, perceived to occur earlier in the time instant before the virtual vehicle V_1 crossed it's *End of evaluation* line. But from the simulation video for

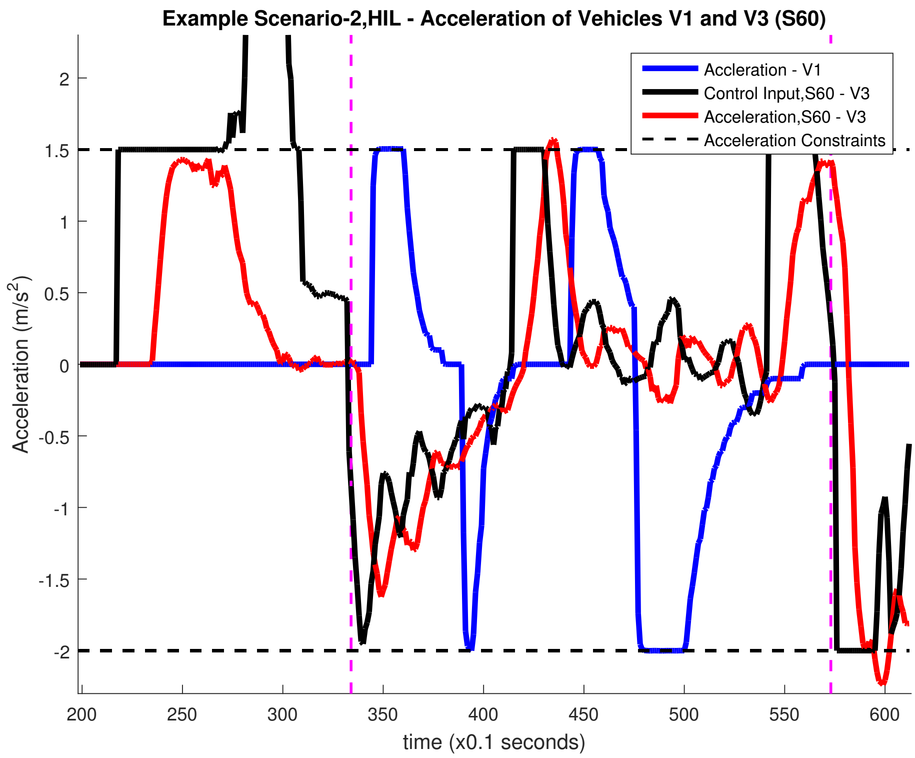


Figure 6.9: Acceleration of $V1$ and $V3$ - Example Scenario 2, HIL

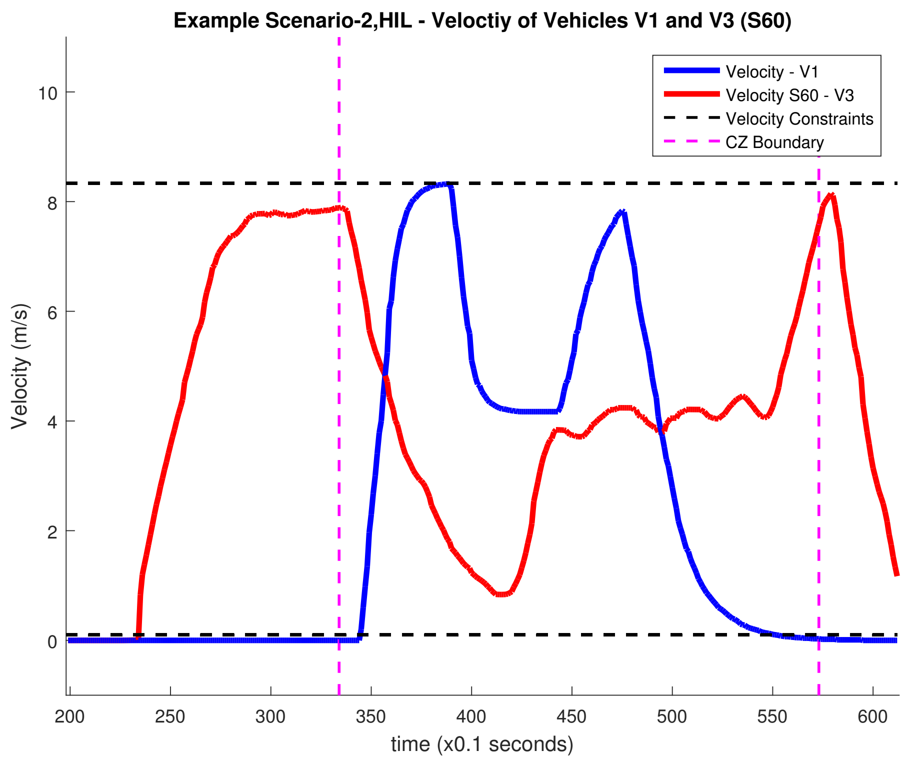


Figure 6.10: Velocity of $V1$ and $V3$ - Example Scenario 2, HIL

this case it can be noticed that this discrepancies does not affect the actual collision avoidance manoeuvre of vehicle V_3. The distance discrepancies are also caused by the GPS position of the vehicle varying in real time as the vehicle V_3 gets into low velocities behind *Critical Point*.

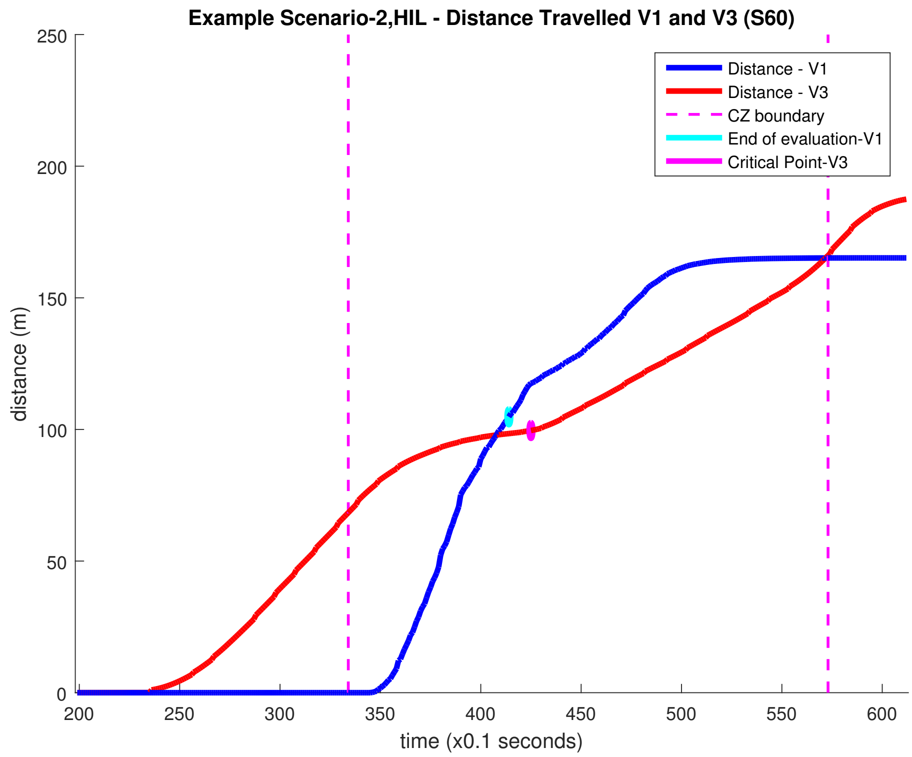


Figure 6.11: Position of $V1$ and $V3$ - Example Scenario 2, HIL

6.4.2 Longitudinal States of virtual V_3 and real V_2

HIL testing for Example Scenario 2, in which the virtual vehicle V_2 is replaced with real test vehicle, Volvo S60, while vehicles V_1 & V_3 are virtual. Here in this case the virtual vehicle V_3 is the *leader* of vehicle V_2, so the vehicle V_1 is not considered in this section for study. The simulation representing the position, velocity and acceleration longitudinal states of virtual vehicle V_3 and test vehicle V_1 are as shown in the video **Video 12 - V2__ExpScen2__HIL** under the link `https://drive.google.com/folderview?id=0B5a5I4V23aoLY0J5RVh1RlZadXM&usp=sharing`. It can be noticed in the top right window of the video that test vehicle V_2 is shifted from it's lane, which was due to the GPS correction offset from the server. It is obvious that the vehicle V_1 is also a virtual vehicle running in the simulation of a stand alone computer, but it's states are not studied in this section.

6.4.2.1 Acceleration of V_3 ($a_{V_3}(k)$) and V_2, $S60$ ($a_{V_2}(k)$)

In Figure 6.12, the acceleration graph of test vehicle V_2 is along with the *leader* vehicle V_2. It can be noticed that the deceleration of test vehicle V_2 is longer than the previous cases since it had to maintain safety distance with *leader* vehicle V_3 which in turn had to maintain safety distance with vehicle V_1.

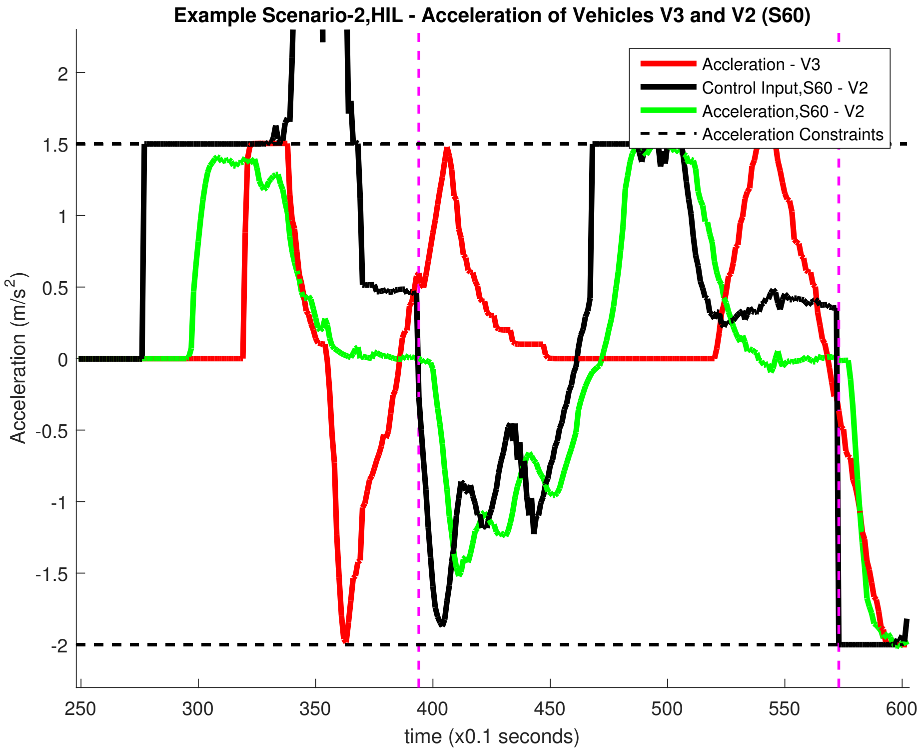


Figure 6.12: Acceleration of $V2$ and $V3$ - Example Scenario 2, HIL

6.4.2.2 Velocity of V_3 ($v_{V_3}(k)$) and V_2, $S60$ ($v_{V_2}(k)$)

In Figure 6.13 the velocity of virtual vehicle V_3 can be noticed to be constant around speed of 4 m/s^2 when in curved path. So the velocity of real test vehicle V_2 gets low for a longer time even after entering the CZ at maximum speed later than its *leader* vehicle V_3.

6.4.2.3 Position of V_3 ($P_{V_3}(k)$) and V_2, $S60$ ($P_{V_2}(k)$)

In Figure 6.14 the aspect of position in terms of distance travelled by the test vehicle V_2 is represented along with the virtual vehicle V_2. It can be noticed that vehicle V_2 after entering into CZ travelled distance almost equal to vehicle V_3 at one point after which slows down and travels lesser distance than V_3 thereby maintaining safety distance and allowed vehicle V_3 to cross it's *End of evaluation* point.

6.5 HIL Testing Results - Example Scenario 3

In this HIL simulation of *Example Scenario 3*, the real vehicle plays the role of vehicle V_3. Since there is no collision avoidance manoeuvre between vehicles such pair is not considered for study. Also the collision avoidance manoeuvre of V_3 as real test vehicle, Volvo S60 depends mostly on the position of vehicle V_2 than V_1, so only the longitudinal states between vehicle V_2 & V_3 is considered for study.

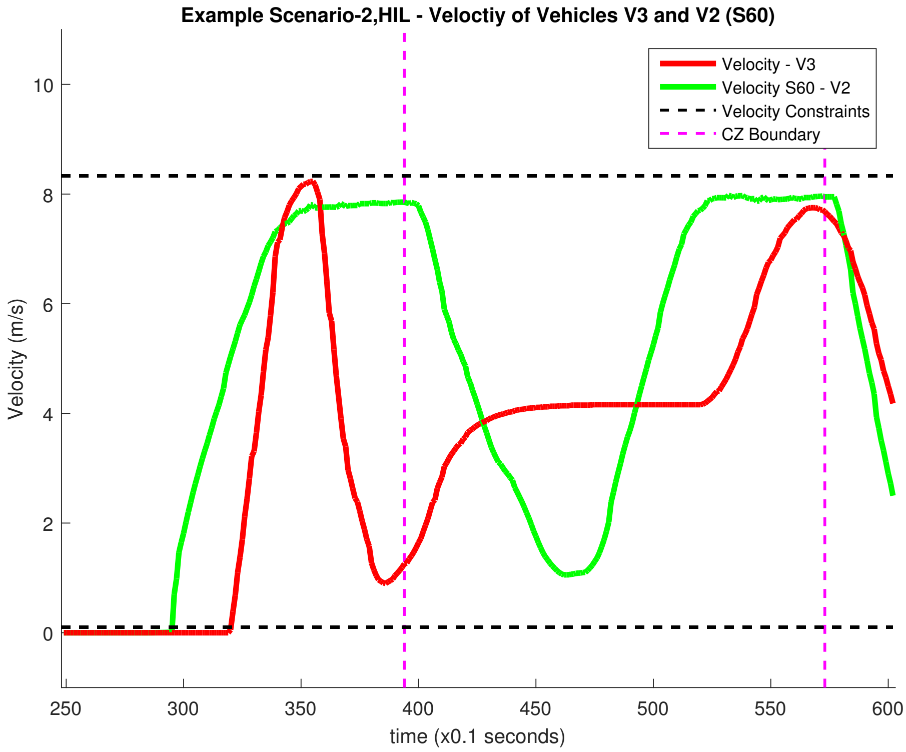


Figure 6.13: Velocity of $V2$ and $V3$ - Example Scenario 2, HIL

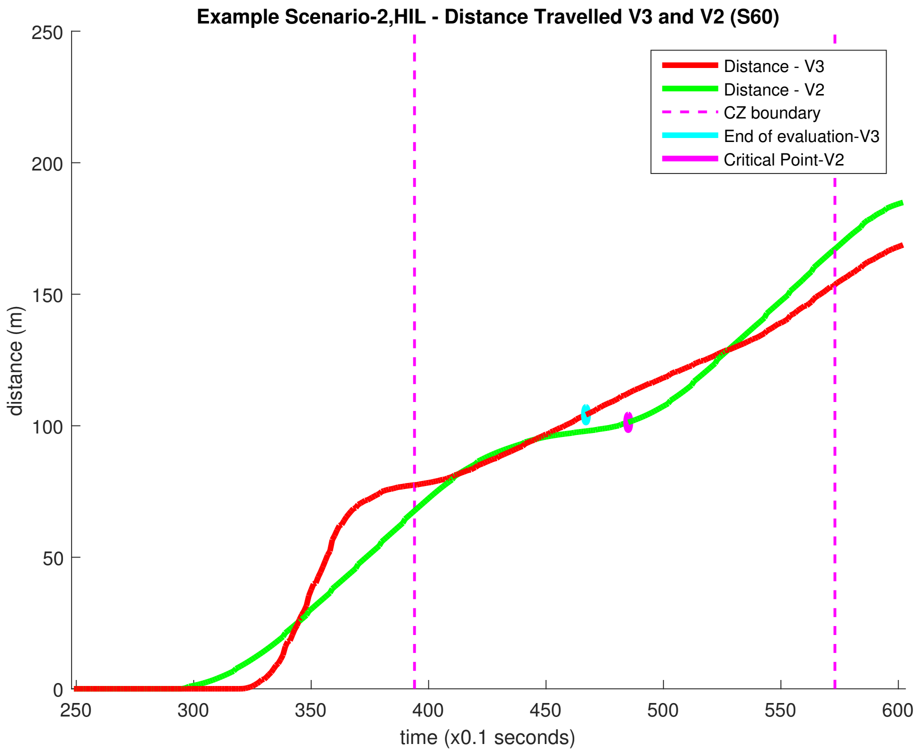


Figure 6.14: Position of $V2$ and $V3$ - Example Scenario 2, HIL

6.5.1 Longitudinal States of virtual V_2 and real V_3

The simulation of *Example Scenario 3* with focus on test vehicle V_3 as *follower* and the virtual test vehicle V_2 is available as a video on the name **Video 12 -**

V2__ExpScen2__HIL in the link
`https://drive.google.com/folderview?id=0B5a5I4V23aoLY0J5RVh1R1ZadXM&usp=sharing`. As discussed earlier in this report, V_3 has a most conservative *Critical Point* so that it can fulfill *minimum euclidean distance* criteria with both it's higher priority vehicles V_1 & V_2 . After which the V_3 has to fulfill *desired distance* criteria with vehicle V_2.

6.5.1.1 Acceleration of V_2 $(a_{V_2}(k))$ and V_3, $S60$ $(a_{V_3}(k))$

Figure 6.15 shows the acceleration curves of both the *leader* V_2 and the *follower* V_3. It can be noticed from the demanded control input of V_3 that the deceleration curve right after reaching CZ is aiding the test vehicle to fulfill the *minimum euclidean criteria* and after which the vehicle is accelerated and oscillations can be noticed depicting the occurrence of platooning behaviour thereby fulfilling the *desired distance* criteria.

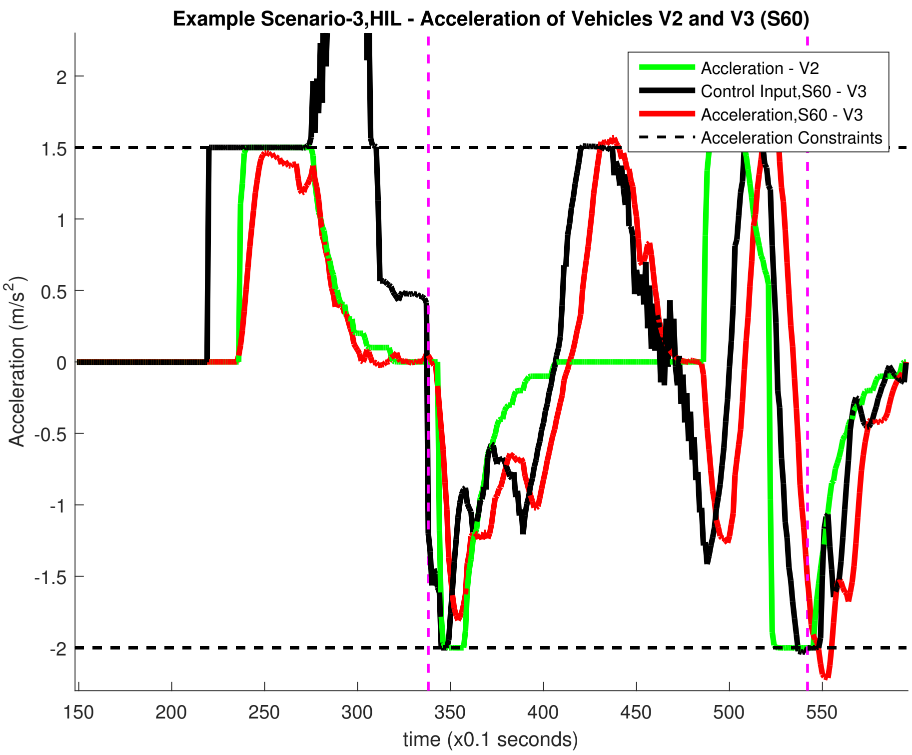


Figure 6.15: Acceleration of $V2$ and $V3$ - Example Scenario 3, HIL

6.5.1.2 Velocity of V_2 $(v_{V_2}(k))$ and V_3, $S60$ $(v_{V_3}(k))$

In Figure 6.16 it can be noticed that the test vehicle V_3 reaches nearly maximum velocity when entering into CZ along with the virtual *leader* vehicle V_2. After entering into CZ the velocity of V_3 drops such that the vehicle V_2 is allowed to cross the *End of evaluation* and then vehicle V_3 maintains constant velocity around 6 m/s^2 while taking the curve and slows down not to violate the *desired distance criteria* and speeds up near exit of CZ, so that to parallel, cope up with the speed of vehicle V_2.

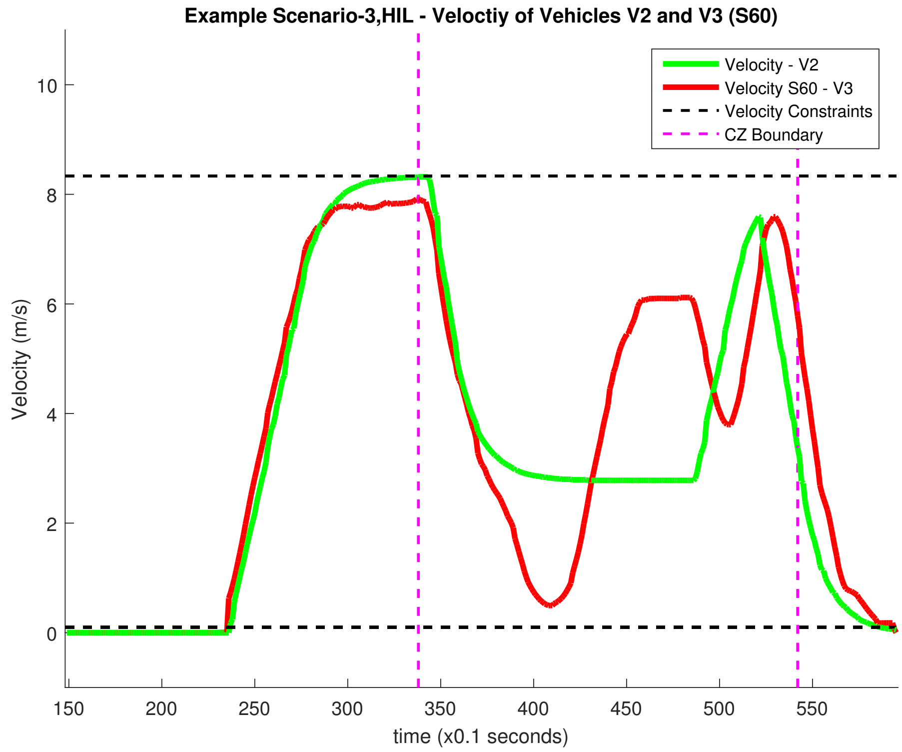


Figure 6.16: Velocity of $V2$ and $V3$ - Example Scenario 3, HIL

6.5.1.3 Position of V_2 ($P_{V_2}(k)$) and V_3, $S60$ ($P_{V_3}(k)$)

In Figure 6.17 it can be noticed that the virtual vehicle V_2 crosses it's *End of evaluation*,at time instant which is later than vehicle V_1 (not shown in graph) crossing it's *End of evaluation* point on it's path. So when the test vehicle V_3 crosses it's *Critical point* on it's path it haven't violated *minimum euclidean distance* criteria with both virtual vehicle V_1 and V_2. Later the vehicle V_3 even though noticeably have travelled longer distance than V_2 tries to parallel the distance travelled by vehicle V_2 in two different speed ranges thereby fulfilling the *desired distance* criteria

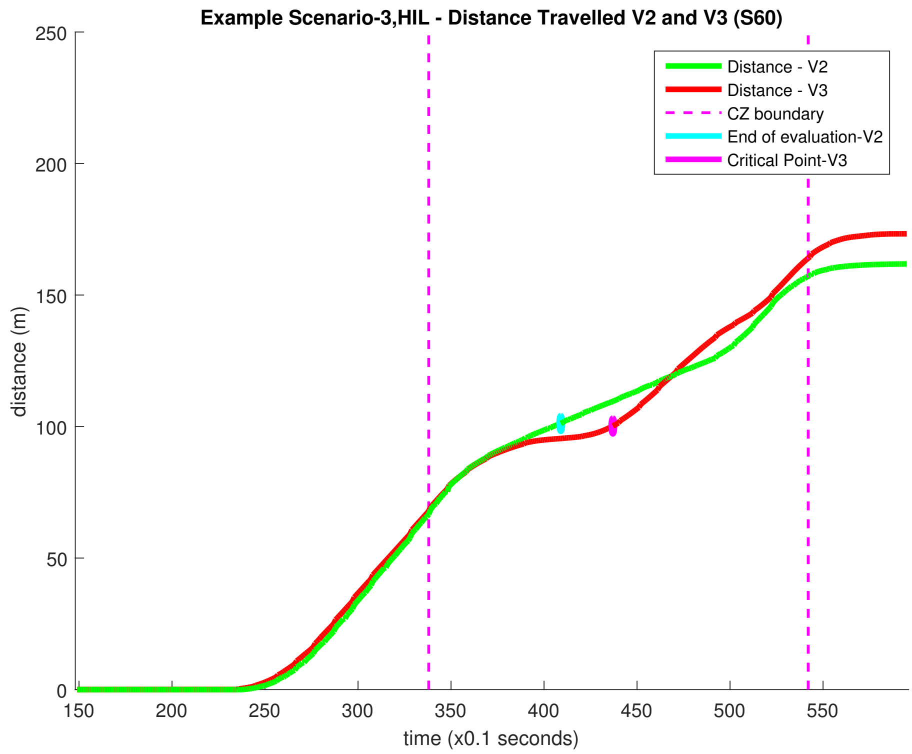


Figure 6.17: Position of $V2$ and $V3$ - Example Scenario 3, HIL

7
Competition Results

7.1 Grand Cooperative Driving Challenge

Grand Cooperative Driving Challenge (GCDC) was held in Helmond, Netherlands on 28-29, May, 2016. There were three different scenarios

1. Platooning Scenario
2. Intersection Scenario
3. Emergency Scenario

The intersection control algorithm described in this thesis was implemented in *Chalmers Car* vehicle. There were 14 competitor vehicles from various research institutes across European Union (EU) that participated in GCDC 2016. The performance of the test vehicle from *Chalmers Car* team in the best heat performance in intersection scenario is discussed in the upcoming section.

7.1.1 Performance of Vehicle as PC_1

This section describes the performance of the test vehicle Volvo S60 in the GCDC intersection scenario while taking place of vehicle PC_1. The *leader* vehicle V was a Organiser's Pacer Car (OPC) which used different control algorithms designed by the organisers. The video **Video 14 - PC1_GCDC_competition_heat** in the link `https://drive.google.com/folderview?id=0B5a5I4V23aoLY0J5RVh1RlZadXM&usp=sharing` shows the performance of longitudinal states of Volvo S60 and OPC. The same heat as captured in actual heat with real cars manoeuvring in the intersection zone can be seen in the video **Video 15 - PC1 Intersection** in the same link. It would be obvious that the longitudinal states of the *leader* OPC described is the output sensor fused states whose raw input is through wireless communication.

7.1.1.1 Acceleration of V, OPC ($a_V(k)$) & $PC_1, S60$ ($a_{PC_1}(k)$)

The graphical representation of OPC vehicle as V shows large oscillations as seen in Figure 7.1. V vehicle also has acceleration and deceleration values which are far beyond the acceptable range as specified in GCDC requirements documents [9]. The acceleration demand for the Volvo S60 vehicle PC_1 from the controller was of similar pattern as was in HIL testing. Here it is obvious that the controller behaved in the similar pattern when receiving wireless communication states both from virtual as well as real vehicle.

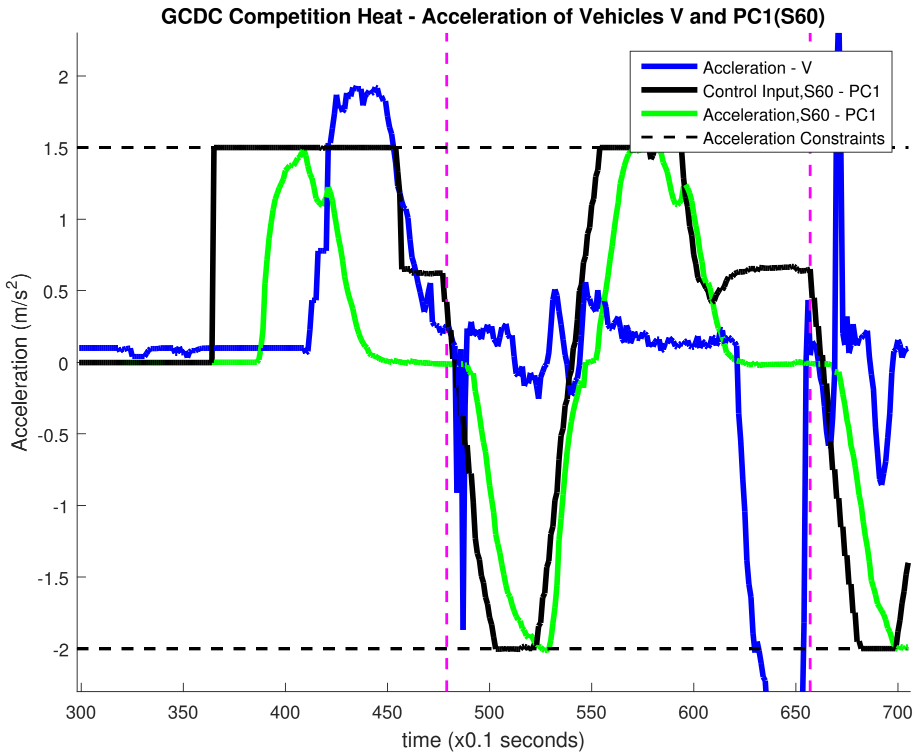


Figure 7.1: Acceleration of V and $PC1$ - GCDC competition heat

7.1.1.2 Velocity of V, OPC ($v_V(k)$) & PC_1, $S60$ ($v_{PC_1}(k)$)

The velocity graphical comparison between OPC as V and Volvo S60 as PC_1 shows that both vehicles have reached almost maximum velocity of 30 $kmph$ or 8.33 m/s as shown in Figure 7.2. The vehicle V has differed from the requirement that it has to slow down to even lesser velocity while taking the turn, which was not the case in any of the heats. Volvo S60 as vehicle PC_1 have slowed down until the point that the vehicle V was allowed to cross the intersection and join the main road while PC_1 was fulfilling the *minimum euclidean distance* criteria.

7.1.1.3 Position of V, OPC ($P_V(k)$) & PC_1, $S60$ ($P_{PC_1}(k)$)

During the GCDC heats it was a fact that the OPC started from a distance close to the intersection than other competitor vehicles, this was also a reason for the OPC to accelerate beyond the acceleration constraints so that to reach the maximum velocity inside the CZ. This had reflected in the distance travelled comparison between the OPC V and *Volvo S60 V* graph, Figure 7.3, where the vehicle PC_1 have travelled more distance when V was passing the *End of evaluation* while PC_1 was staying behind it's *Critical Point*.

7.1.2 Performance of Ego Vehicle as PC_2

In this section the heat when OPC was the *leader* vehicle V and Volvo S60 - Chalmers Car team participated as *follower* vehicle PC_2 is described. The video **Video 16 - PC2_GCDC_competition_Heat** in the link `https://drive.google.com/`

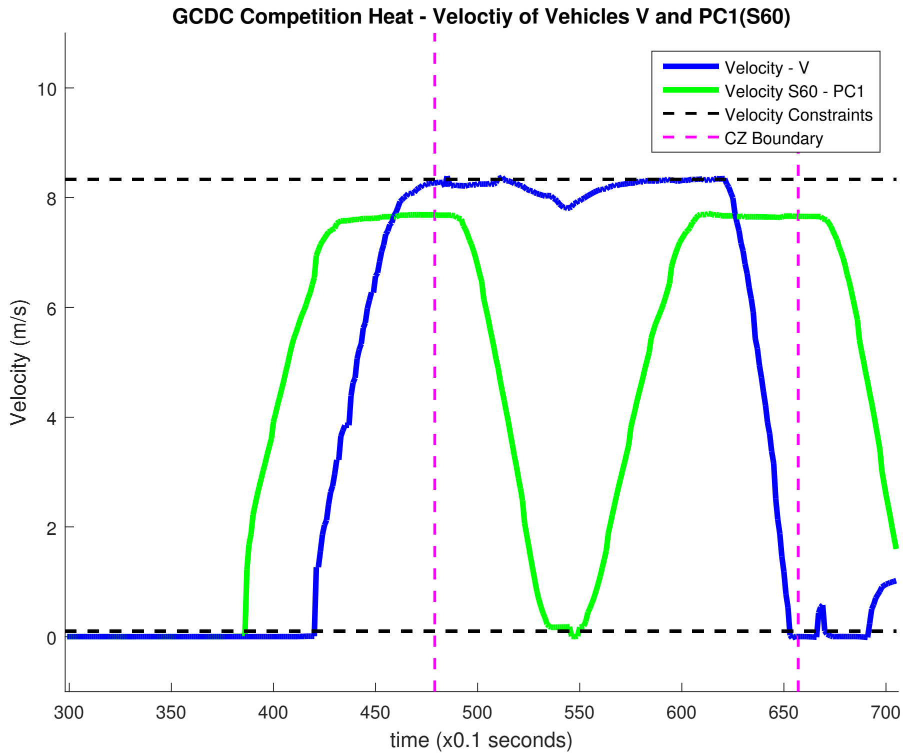


Figure 7.2: Velocity of V and $PC1$ - GCDC competition heat

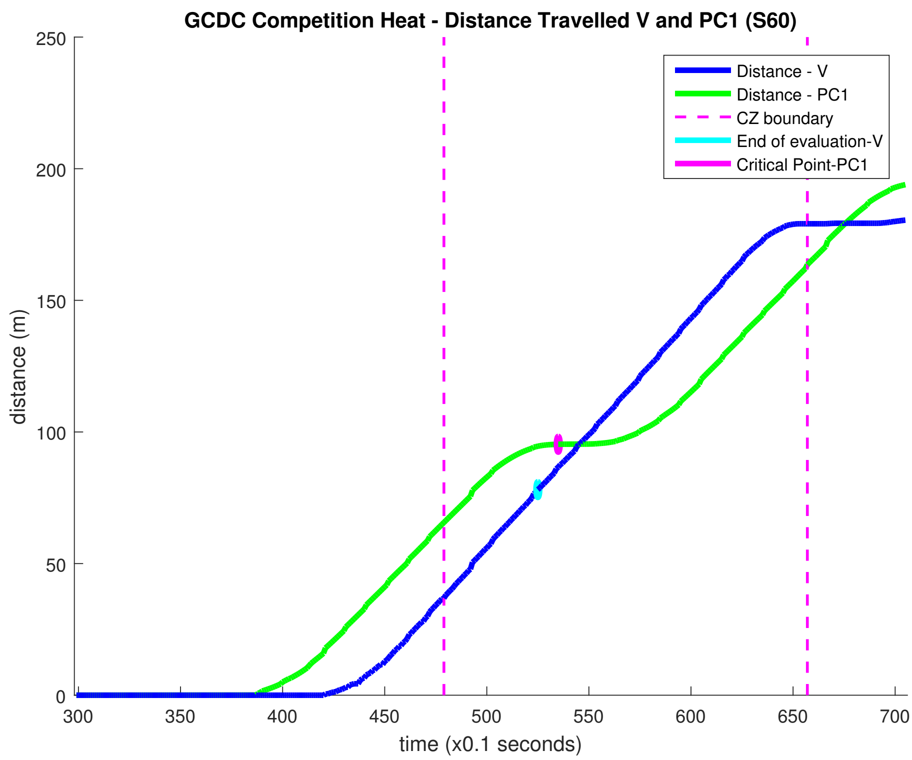


Figure 7.3: Position of V and $PC1$ - GCDC competition heat

`folderview?id=0B5a5I4V23aoLY0J5RVh1RlZadXM&usp=sharing` shows the performance of longitudinal states of Volvo S60 and OPC. The same heat as captured in actual heat with real cars manoeuvring in the intersection zone can be seen in the video **Video 17 - PC2 Intersection** in the same link. The individual longitudinal states of the vehicles V and PC_2 are discussed below.

7.1.2.1 Acceleration of V, OPC $(a_V(k))$ & $PC_2, S60$ $(a_{PC_2}(k))$

Figure 7.4 shows the graphical representation of acceleration profile comparison between OPC vehicle V and Volvo S60 vehicle PC_2. In the graph it can be seen that the vehicle PC_2 is not decelerating for most of the time instant inside CZ. In this heat the vehicle Volvo S60 PC_2 after reaching CZ, have to accelerate and platoon the OPC vehicle V.

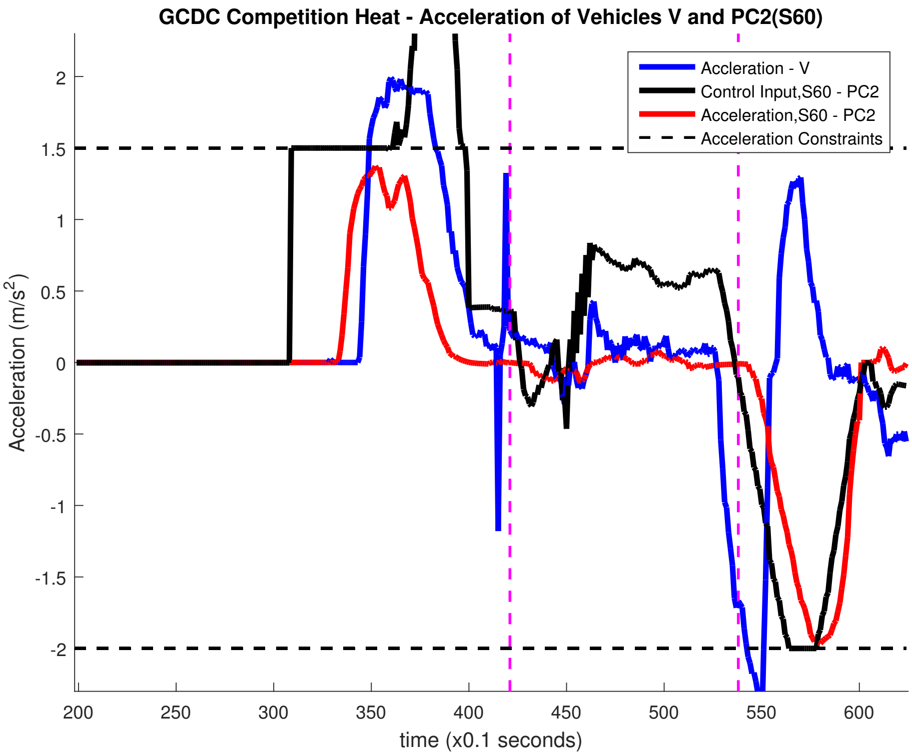


Figure 7.4: Acceleration of V and $PC2$ - GCDC competition heat

7.1.2.2 Velocity of V, OPC $(v_V(k))$ & $PC_2, S60$ $(v_{PC_2}(k))$

Figure 7.5 shows the graphical representation of velocity comparison at various time instants between OPC vehicle V and Volvo S60 vehicle PC_2. In this heat, the OPC V didn't slow down near the curved path of the intersection and the vehicle PC_2 maintains the constant near maximum speed in which it entered CZ. The vehicle PC_2 maintains near maximum velocity so that to fulfill the *desired distance*, by trying to keep up with the position between maximum and minimum distance constraint.

7.1.2.3 Position of V, OPC $(P_V(k))$ & $PC_2, S60$ $(P_{PC_2}(k))$

Figure 7.6 shows the graphical representation of distance travelled comparison between OPC vehicle V and Volvo S60 vehicle PC_2. It can be seen from the video for

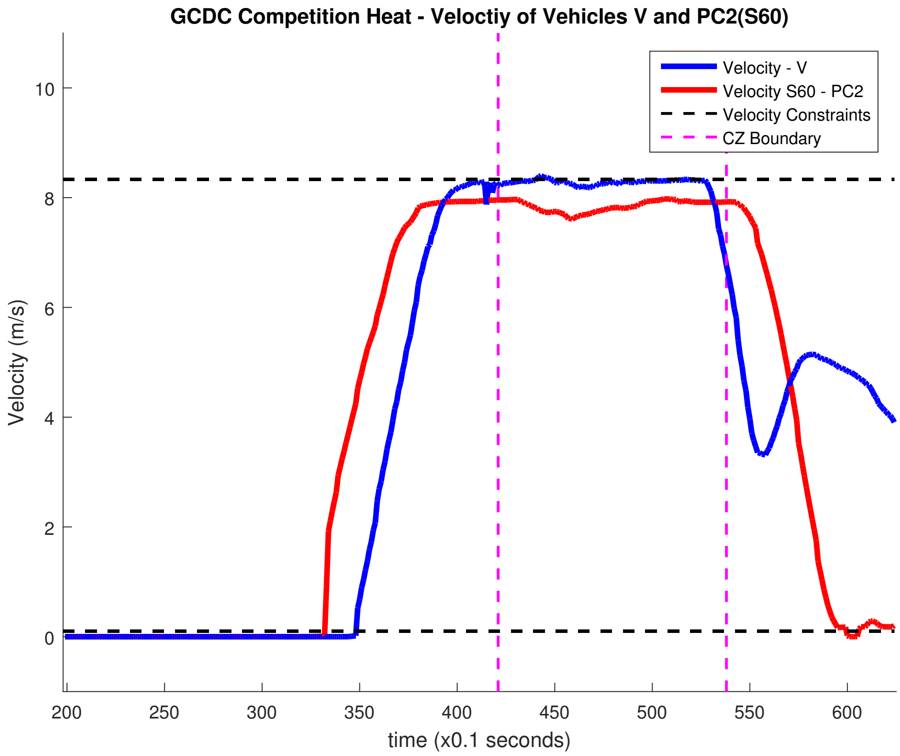


Figure 7.5: Velocity of V and $PC2$ - GCDC competition heat

this heat that the vehicle PC_2 is close to the minimum distance limit within which it should be from the vehicle V. So the vehicle PC_2 travels distance or positions itself parallel to the distance travelled by vehicle V in the graph, parallel distance is the safety distance to be maintained in *desired distance* evaluation criteria.

7.2 GCDC Score and Results

Chalmers Car team came up in **Fourth** place out of 14 competitors participated in GCDC, 2016. The position of a team in the competition is decided based on the collective score earned by a competitor from individual scenario scores. The score of Chalmers team in intersection scenario was **7.5** out of 10. The *Chalmers car* team fulfilled the scoring criteria which are *minimum euclidean distance*, *maximum velocity limit* and *desired distance criteria* which was self checked after the competition based on the logged communication data. But the score reduction was perceived to be on part of *fastest finish time* criteria, as the GCDC organisers decided the criteria scoring analysing all the competitors finishing time performance.

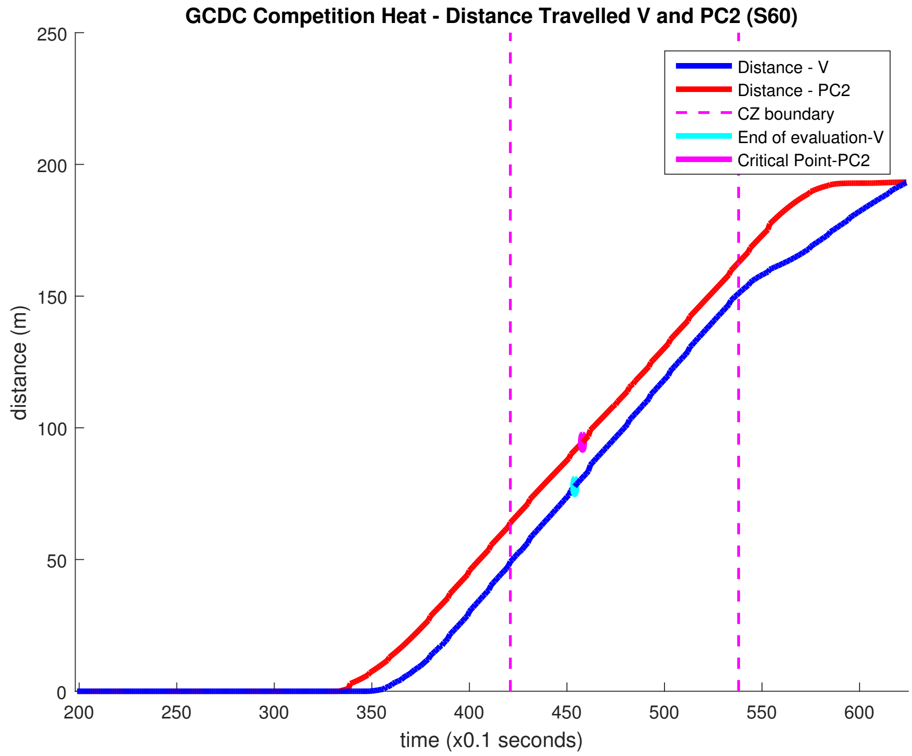


Figure 7.6: Position of V and $PC2$ - GCDC competition heat

8
Conclusion

This thesis was an important aid for the Chalmers Car team in GCDC participation. The longitudinal control of a vehicle in the intersection scenario was implementable as a result of the formulation, simulation and testing carried out in this thesis.
However there are scope for future work on the intersection controller design described in this thesis. Some possible improvements are

1. Analyse the packet loss in wireless communication and the effect of loss in communication on the control of the vehicle
2. Proper tuning of the MPC so that the longitudinal control behaviour of the real vehicle is less conservative in avoiding collision and perform the intersection crossing even faster.
3. Reduce the computational load by switching over to different platforms for real time simulation and rapid prototyping. For example, the work for this thesis was carried out completely using Matlab/Simulink environment, the same kind of work can be experimented in a C++ real time environment.

Bibliography

[1] "Grand Cooperative Driving Challenge" - GCDC 2016 by i-Game www.gcdc.net

[2] "Scenarios (D1.1)", description of scenarios for GCDC 2016. http://gcdc.net/images/doc/D1.1.Scenarios.pdf

[3] "Wireless communication basic spec (D3.1)" - requirements on V2V equipment and software for GCDC 2016 http://gcdc.net/images/doc/D3.1.Wireless.communication.basic.specification.pdf

[4] "Message Set (D3.2)" - concrete V2V messages to implement interaction protocol for GCDC 2016 http://gcdc.net/images/doc/D3.2.Proposal.for.extended.message.set.for.supervised.automated.driving.pdf

[5] "Interaction Protocol", how vehicles should interact with each other in GCDC 2016. http://gcdc.net/images/doc/D2.1.Interaction.Protocol.pdf

[6] "Generic Real Time Control System", technical documentation for GCDC 2016 http://gcdc.net/i-game

[7] "Algorithm Protocol for the transition of control", technical documentation for GCDC 2016. http://gcdc.net/i-game

[8] GCDC "Judging criteria (D7.1)", technical documentation describing the method of judging the winner. http://gcdc.net/images/doc/D7.1.Judging.criteria.draft-2016-03-17.pdf

[9] GCDC "Requirements D(1.4)" document, general requirement for participating Vehicles technical documentation. http://gcdc.net/images/doc/D1.4.Final.requirements.specification.pdf

[10] Geo Networking stack, API and Simulink Communication Interface for GCDC technical documentation by Albin Severinson, Masters Student, Chalmers University of Technology, Sweden. https://github.com/RNDITS/geonetworking/tree/master/vehicle-adapter

[11] "Introduction to Model Predictive Control for Linear and Hybrid Systems" lecture slides by Paolo Falcone, Department of Signals and Systems, Chalmers University of Technology, Göteborg, Sweden

[12] "Model Predictive Control" lecture notes by Bo Egardt, Department of Signals and Systems, Chalmers University of Technology, Göteborg, Sweden

[13] G. R. Campos, P. Falcone, and J. Sjöberg, "Autonomous cooperative driving: a velocity-based negotiation approach for intersection crossing," in IEEE Conference on Intel. Transportation Systems, 2013.

[14] G. R. Campos, P. Falcone, H. Wymeersch, R. Hult, and J. Sjöberg, "A receding horizon control strategy for cooperative conflict resolution at traffic intersections," in Proc. of the IEEE Conference on Decision and Control (CDC), 2014.

[15] R. Hult, G. R. Campos, P. Falcone, and H. Wymeersch, “Technical report: Approximate solution to the optimal coordination problem for autonomous vehicles at intersections,” tech. rep., Department of Signals and Systems, Chalmers University of Technology, avaliable at
http://www.gabrieldecampos.com/Documents/Publications/Technical_Report_ACC15_safe.pdf

[16] Mathias Ernst, Martin Holder, Samuel Scheidegger, "Towards a solution of the cooperative intersection scenario of GCDC 2016," Design Project at Chalmers University of Technology

[17] Master Thesis on "MPC approach to lateral vehicle motion control with safety and comfort requirements" by Nei Li, Systems Control and Mechatronics at Chalmers University Of Technology, Gothenburg, Sweden

[18] 'Vehicle Dynamics and Control' authored by Dr. Rajesh Rajamani, Department of Mechanical Engineering, University of Minnesota, Minneapolis, MN 55455, USA.

[19] Bachelor thesis in, "Testing active safety systems Development of an experimental platform for cooperative driving and collision avoidance strategies" by Albin Dahlin, Andreas Larsen, Edwin Pearson, Jacob Brickner, Martin Hernå, Rasmus Åkerlund at Chalmers University Of Technology, Gothenburg, Sweden

[20] "A Control Matching Model Predictive Control Approach to String Stable Vehicle Platooning", Roozbeh Kianfar, Paolo Falcone, Jonas Fredriksson, *IEEE Journal*

[21] "CVXGEN: Code Generation for Convex Optimization" http://cvxgen.com/docs/index.html

[22] "GPS and Compass sensors for GCDC - 2016" technical report by Jacob Brickner, Lars Sjöberg, Master Students, Chalmers University of Technology, Sweden.

[23] "Ego and Participant Sensor Fusion" technical documentation by Rober Hult, Phd student, Chalmers University of Technology, Sweden and Marco Di Vaio, Master Student, University of Naples Federico II, Italywww.

A

Appendix 1

A.1 MPC Problem in CVXGen Format

```
# Filename: description.cvxgen.
# Description: A description of the CVXGEN problem.

dimensions
  m = 1  # inputs.
  n = 3  # states.
  T = 45 # horizon.
end

parameters
  A (n,n)  # dynamics matrix.
  B (n,m)  # transfer matrix.
  Q (n,n) psd  # state cost.
  P (n,n) psd  # final state cost.
  R (m,m) psd  # input cost.
  x[0] (n)  # initial state.
  u_max (m) nonnegative  # amplitude limit.
  u_min (m) nonpositive
  x_min1 (T+1)
  x_max1 (T+1)
  x_min2 (T+1)
  x_max2 (T+1)
  x_desired (n) nonnegative
  Sva (m) nonnegative  # slew rate limit.
  prevU (m)
end

variables
  x[t] (n), t=1..T+1  # state.
  u[t] (m), t=0..T  # input.
end

minimize
  sum[t=0..T](quad((x[t]-x_desired), Q)
```

```
  + quad(u[t], R))
  + quad((x[T+1]-x_desired), P)
subject to
  x[t+1] == A*x[t] + B*u[t], t=0..T  # dynamics constraints.
  u_min <= u[t] <= u_max, t=0..T  # maximum input box constraint.
  x_min1[t] <= x[t][1] <= x_max1[t] , t=1..T+1
  x_min2[t] <= x[t][2] <= x_max2[t] , t=1..T+1
  u_min <= x[t][3] <= u_max , t=1..T+1
  norminf(prevU - u[0]) <= Sva
  norminf(u[t+1] - u[t]) <= Sva , t=0..T-1  # slew rate constraint.
end
```

A.2 V2V Communication Signals

Message ID	Message description	Scenario
1	Header	1, 2, 3
2	GenerationDeltaTime	1, 2, 3
3	Station ID	1, 2, 3
4	Station Type	1, 2, 3
5	Vehicle Role	1, 2, 3
6	Vehicle length	1,2
7	Vehicle rear axle location	1,2
8	Vehicle width	1, 2, 3
9	Controller type	1, 2, 3
10	Vehicle response time constant	1,2
11	Vehicle response time delay	1,2
12	ReferencePosition (latitude, longitude, confidence)	1, 2
13	Heading (Heading, confidence)	1, 2
14	Speed	1, 2
15	YawRate	1, 2
16	Longitudinal vehicle acceleration	1, 2
17	Desired longitudinal vehicle acceleration	1, 2
18	MIO ID (measured by object vehicle)	1
19	MIO range (measured by object vehicle)	1
20	MIO bearing (measured by object vehicle)	1
21	MIO range rate (measured by object vehicle)	1
22	Time headway	1
23	Cruise speed	1
24	Merge request flag	1
25	Safe-to-merge (STOM) flag	1
26	Merging flag	1
27	ID of fwd pair partner	1
28	ID of bwd pair partner	1
29	Tail vehicle flag	1
30	Head vehicle flag	1
31	Platoon ID	1
32	Travelled distance inside the CZ	2
33	Intention (left, right, or straight)	2
34	Lane on which the vehicle enters the CZ	1,2
35	Intersection vehicle counter	2
36	Pair acknowledge flag	1
37	Reference Time	3
38	EventType (Roadworks, Stationary vehicle, Emergency vehicle approaching, Dangerous Situation (Emergency electronic brake light)	1,3
39	ClosedLanes	1
40	LanePosition	3

Figure A.1: Communicated Signals - 2, indicates intersection scenario

Message ID	Message description	Scenario
41	Participants ready	1,2,3
42	Start scenario	1,2,3
43	EoS (End of Scenario)	1,2,3
44	Reserve/spare/future use	Tbd
45	Reserve/spare/future use	Tbd
46	Reserve/spare/future use	tbd

Figure A.2: Communicated Signals (Contd) - 2, indicates intersection scenario