POLITECNICO DI TORINO

Department of Control and Computer Engineering (DAUIN)

Master of Science in Mechatronic Engineering

Master Degree Thesis

Autonomous Cars & ADAS: Complex Scenario Generation, Simulation and Evaluation of Collision Avoidance Systems

Supervisors

Prof. Stefano Di Carlo (DAUIN, POLITO)

M.Sc. Thomas Ponn (FTM, TUM)

Candidate

Mehrdad Moslemi

A.Y. 2018-19

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

AC	Autonomous Car
AV	Autonomous Vehicle
HAV	Highly Automated Vehicle
AD	Autonomous Driving
ADAS	Advanced Driver Assistance Systems
ACC	Adaptive Cruise Control
LKA	Lane Keeping Assist
LCA	Lane Change Assist
AEB	Automatic Emergency Brake
FCW	Forward Collision Warning
TJA	Traffic Jam Assist
TTC	Time to Collision
HT	Headway Time
ASIL	Automated Safety Integrity Level
V2V	Vehicle to Vehicle
V2X	Vehicle to Grid
R&D	Research and Development
DOF	Degrees of Freedom
Ego-Car	Autonomous Car in Test-Bench
Vego	Longitudinal Velocity of Ego-Car
Lead-Car	Leading Car in front of Ego-Car
V0_Lead	Initial Velocity of Lead-Car
Side-Car	Car in right or left lane of Ego-Car
MIO	Most Important Object
Side-MIO	Most Important Object on the side lane

R-MIO	Most Important Object on the right lane
L-MIO	Most Important Object on the left lane
FCWtime	Forward Collision Warning time
PB1time	Partial Brake 1 time
PB1time	Partial Brake 2 time
FBtime	Full Brake time

1. INTRODUCTION

Automotive stream and technology pervades an enormous area of business in today's world. This is an emerging and continuously evolving field and we have seen constant growth, more so in the past decades. Right from Hybrid Cars to Electric Vehicles, and indubitably Autonomous Cars.

With the increasing demand for Autonomous Vehicles and ADAS equipped Cars, there is a big debate about the reliability of these cars and their performance in various traffic situations, especially in the most critical driving scenarios, in which a very intelligent performance of the car is expected to ensure that the self-driving vehicle is at least as smart as a normal human driver to handle the situation and avoid a collision in the best way to increase the safety on the roads.

Accordingly, as not all the vehicles on the roads are autonomous and equipped with V2V or V2X connections, the criticality grows as the behavior of other vehicles is not predictable and the autonomous car must be able to handle the situation independently if the aim is to reach full autonomy.

Performance levels of Autonomous Car is categorized in six levels by SAE International with regards to technology and legislation mediums in the J3016 standard [1], which defines the SAE Zero level as no automation, up to SAE Level 5 for full vehicle autonomy.

An overview of these autonomy levels is represented in figure 1 as the most updated chart of autonomy levels stated in in the latest version of June 2018 for J3016 standard. [2]

In level 0, driver support features are limited to providing warnings and momentary assistance; i.e. lane departure warning or blind spot warning are in this category. In level 1, features like brake, steering or acceleration are provided to driver. Level 2, provides features like lane centering and adaptive cruise control at the same time. Up to this level, driver is the only responsible person for the behavior of the car, as the ADAS functions are just considered as an assistant to the driver, but cannot substitute the driver in any case. However, in level 3 responsibility is share between driver and car, but the driver should take action whenever system requests. Traffic Jam Assist (TJA) is an example for level 3 assistance. In level 4 there is more independency for the cars and system performs the lateral and longitudinal dynamic driving tasks in all situations in a defined use case and driver is not required during defined use cases. There is a possibility that pedals or steering would not be installed in these cars. However, in level 5, system performs the aforementioned tasks in all situations encountered during the entire journey and no diver is required at all. [2]

This standard is one of the most cited references for the capabilities of autonomous cars. Regarding the emergence of numerous autonomous cars on the roads in recent years, features of each level are more clearly explained in this recent update of this chart to show how are they increasing the consumers' safety and convenience. [2]

In this version of the chart, inputs from the insurance companies, the American Automobile Association, and the Transportation Research Board has been included during its development phase. Moreover, the SAE's marketing group and committee have worked on the graphics and texts in the chart to ensure its technically faithfulness to J3016 standard. Obviously, this chart would be also updated with regards to the future developments in the Autonomous Driving industry and changes or updates in the J3016 standard. [2]

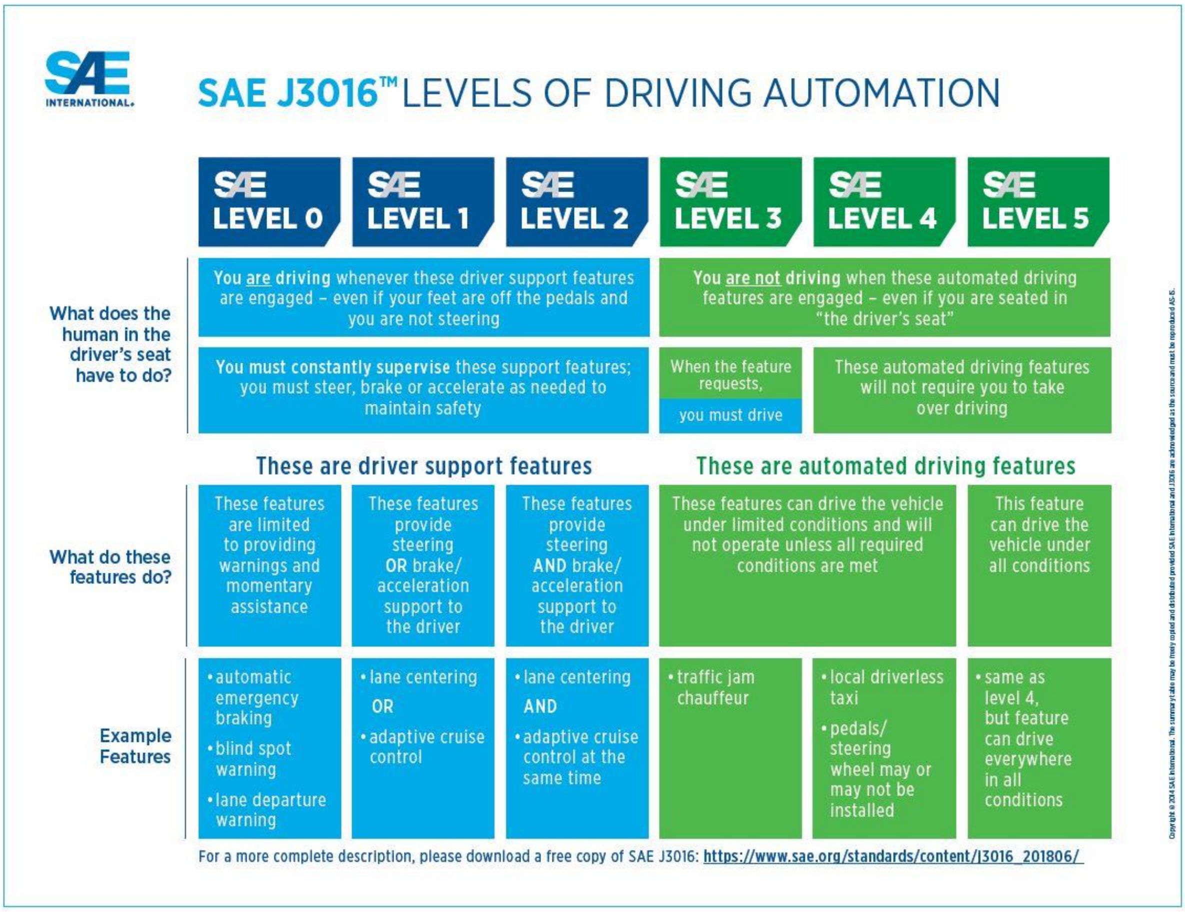


Figure 1. Levels of Driving Automation according to SAE J3016 [2]

The approach in this Thesis will be considering level 3 and 4 of autonomy for defined cases. In order to evaluate the safety of an autonomous car in defined critical driving scenarios, a Test-Bench will be implemented consisting of four major ADAS functions as Adaptive Cruise Control (ACC) [24], Lane Keeping Assist (LKA) [25], Lane Change Assist (LCA), and Automatic Emergency Brake (AEB) [26].

As the evaluation of safety of autonomous vehicles is extremely time consuming and complex due to the infinite number of possible traffic scenarios and system performances; for this reason, based on the available research on complex scenario generation, the focus during this safety assessment can be placed on some specific situations and thus the number of necessary test cases can be reduced without endangering sufficient test room coverage.

In this thesis, the aforementioned issues will be considered for the evaluation of an Autonomous Car's behavior.

Accordingly, the research is structured in five main parts as following:

- **State of the Art:**
  An overview of Autonomous Driving technology is introduced and also automotive safety with regards to the functional safety within ISO 26262 standard [9] and safety of the intended functionality (SOTIF) [10] is described. Furthermore, previous research on complex driving scenarios generation is discussed, and then, the concept of Advanced Driver Assistance Systems (ADAS) is presented.

- **Methodology and Approach**
  In order to evaluate the performance of an AC model, a Test-Bench is implemented in Matlab & Simulink, constituting of four major ADAS functions (ACC, LKA, LCA, AEB). This system is developed with regards to the criticality issues considered in the generation of driving scenarios.
  The process of implementing this Test-Bench, regarding the priority of ADAS functions in the performance of Autonomous Car will be discussed in this chapter.

- **Simulation and Results**
  The implemented Test-Bench will be used to simulate some selected driving scenarios and the performance of system will be represented using various plots.
  There will be also the discussion of specific cases for each use case of this test-bench to give an overview of the simulation capabilities in this system.

- **Discussion and Evaluation of Results**
  With regards to the plots obtained from simulation, performance of the system will be evaluated to prove how reliable the system would be. Accordingly, safe zones of relative distance and speed for cars in the scenarios will be distinguished. Also cases with critical and dangerous situation are stated to show the weak points of this system.

- **Summary and Outlook**
  A summary of the research will be presented and an outlook of the future possible research extension with this system is discussed.

2. STATE OF THE ART

2.1. OVERVIEW OF AUTONOMOUS DRIVING

2.1.1. DEFINITIOIN OF AUTONOMOUS CAR

"An autonomous car can drive itself from Point A to Point B with no manual input from the driver. The vehicle uses a combination of cameras, radar systems, sensors, and global positioning system (GPS) receivers to determine its surroundings and uses artificial intelligence to determine the quickest and safest path to its destination. Mechatronic units and actuators allow the "brain" of the car to accelerate, brake, and steer as necessary." (Morgan Stanley Research Global) [3, p.14]

2.1.2. HISTORY OF AUTONOMOUS DRIVING

The word "automobile" has roots of two Greek and Latin words of "autos" and "mobilis", which was stressing the mobility of driver without the need for horses when the first pioneers of car making discovered the possibility of transition from horses and carriages to another type of vehicles call automobile. However, in this transition, certain autonomy and obstacle-avoidance skills of horses which was gained through repetition of ride within the boundaries of human rules, was lost. For instance, when using horses and carriages, people had been transferred with by the horse even in cases that they were not able to control the horse. This is the concept of autonomy that transportation has lost since then, and autonomous cars are the means to reach this autonomy again, and of course go far beyond the historic form. [4, p.2]

First step to build up an autonomous vehicle was the radio controlled car called Linrican Wonder, which was demonstrated by Houdina Radio Control in New York City in 1926. A modified form of this car was named "Phantom Auto" and demonstrated by Achen Motors in December 1926 in Milwaukee. Later in 1939, General Motors supported Norman Bel Gedde's exhibit Futurama at the World's Fair, which was introducing embedded-circuit powered electric cars. In the 1940s through the 1980s, some individual attempts were done towards launching the first test prototypes of autonomous cars in the US, Japan, and Europe. For instance, RCA Labs produced a miniature car is 1953 which was controlled by wires laid on the grounds of the lab. Later in 1958, this project was expanded to do and experiment on Highways. Then General Motors promoted this project, building up models which were able to simulate automatic steering, acceleration and brake control. Some advanced models were then presented by General Motors in 1959 and 1960 which led to another project launch by Ohio State University in 1966 to develop driverless cars. Another test of driverless cars was done by Transport and Road Research Laboratory of United Kingdom in 1960s. In the 1980s, a vision-guided driverless Mercedes-Benz was designed at the Bundeswehr University of Munich which was able to reach 63 km/h on the streets without traffic. A project called

Prometheus was conducted by EUREKA through the years of 1987 to 1995 with a funding of over 1 billion US dollar. To name other important projects, Autonomous Land Vehicle (ALV) project in united states was the first road-following demonstration which was using computer vision, LIDAR and autonomous control, being able to reach the speed of 31 km/h. In 1995, the NHOA (No Hand Across America) project developed by the Carnegie Mellon University's NavLab, achieved 98.2% of autonomous driving on 5,000 km cross country journey. ARGO project in 1996, travelling at an average speed of 90 km/h with 94% autonomy was another attempt towards autonomy of driving before the 21 century. In the early 2000s, an automatic public transport system called ParkShuttle started operating in Netherlands. Also attempts for military usage were started by US government, funding the projects Demo I, Demo II, and Demo III were done in those years. [5]

However, the dream has been coming true in the recent years, due to the required achievements in technology. The "DARPA Grand Challenge" held in 2004, 2005 and 2007; and "Google's Driverless Car" started in 2010, could be mentioned as some important events in this industry. [3, pp. 13,14]

Moreover, in the last few years, the concept has been believed and accepted increasingly, and there is a real demand for autonomous cars on the roads. Emergence of cars like Tesla, equipped with autopilot which are already self-driving on the roads is one of the most recent breakthroughs in autonomous driving concept. Additionally, autonomous shuttles and public transportation systems are being implemented widely these days and they are being accepted and adopted by the citizens.

This is the start of a new era in the world, which is already going to develop ideas of smart cities and intelligent transportation systems.

2.1.3. USE CASES OF AUTONOMOUS CARS

1) Interstate Pilot Using Driver for Extended Availability [4, pp. 12,13]

In this mode, the autonomous car takes over the driving task on interstate or interstate-like expressways. Furthermore, the driver can seat like a passenger in the car and enjoy the comfort of journey, while doing other activities like reading a book, watching TV or eating,

2) Autonomous Valet Parking [4, pp. 14,15]

This capability enables the driver to leave the car on the street, and the car drives itself remotely to a parking slot. Plus, call can collect the driver after coming back to use the car. This saves a lot of time and also improves the efficiency of parking slots use.

3) Full Automation using driver for Extended Availability [4, pp. 16-18]

In case of driver's desire to hand over the driving task to the self-driving car in permitted areas, this will be possible with full self-driving capability of autonomous cars. Obviously, this is possible just on the legalized areas approved for these vehicles; and newly built areas or road without the legitimation must be driven manually by the driver. The driver can take the control of vehicle whenever it is required by law, or when he/she has the intention to drive manually.

4) Vehicle on Demand [4, pp. 18-20]

This is a use case which provides the autonomous driving in all scenarios, with or without passengers and cargo. Consequently, the driving robot moves the vehicle to any requested location by the passengers and transports them to the desired destination. The act of transportation is completely automated and passengers can enjoy the whole journey, doing their desired activities. This task can be done 24 hours a day as long as there is enough energy supply for the drive.

2.1.4. ADVANTAGES OF AUTONOMOUS CARS

1- Less road accidents and deaths [3, p. 14]

Due to the fact that the major part of the road accidents is caused by human error or mechanical failure, an autonomous car with computer-controlled ability could be less faulty in hardware, and more obeying traffic laws. Therefore, the higher ratio of autonomous cars on the roads will lead to less road accidents and deaths.

2- Less fuel consumption [3, p. 15]

Since the self-driving car is equipped with good predictive algorithms to measure the load conditions, less fuel will be consumed comparing to manually operated cars.

3- Better traffic patterns [3, pp. 15,16]

Cars with the capability of V2V and V2X, will have a better chance to define their surroundings and their position in the environment; so cars would reach to higher speeds and closer distances in a traffic, since all the vehicles are interacting with each-other and can predict the next move of a vehicle next to them.

4- Customer efficiency and Time saving [3, p. 16]

The occupants of an autonomous car will spend less time in the traffic due to the smoother traffic flow. In addition, they will not have to grab the steering or look at the road; Thus, they can spend the traveling time, pursuing their other activities such as reading, sleeping, watching TV, etc.

5- Enhancements in the economy [3, p. 16]

While people will have more time to read, watch TV, and surf the internet while commuting, there will be more opportunities for the businesses to advertise and encourage people to buy their products. For instance, much more billboards would be seen by the side of the roads when autonomous cars pervade the roads.

6- Applications in Military [3, p. 16]

Considering aerial defense and ground warfare equipped with autonomous vehicles, less humans will be needed to act as soldiers and this will keep troops out of injury.

2.1.5. AUTONOMOUS CARS' PARADIGM

When studying the Autonomous Car concept, we notice that several elements constitute this specific paradigm. Two main sides (Supply-side and Demand-side) are interacting to build this paradigm. The main role-players are to be discussed as the following:

1- Customer approval [3, p. 17], [6, p. 16]

The most important actor in the demand-side is the people who will use these cars and the overall opinion of the public should be positive in order to allow self-driving cars enter the roads. People need to be persuaded that it is safe to put their lives or their loved ones' lives in a car which acts like a robot. It is likely to take a lot of time for people to adapt this new arrival.

2- Cost [3, p. 17], [6, p. 16]

Since the aim is to make autonomous cars, *"the prevailing vehicle"* on the roads, the cost of this additional technology should be rational in the public's opinion. For instance, addition of $1,000-2,000 in semi-autonomous cars or $3,000-5,000 in a full-autonomous car would be quite applicable and satisfying for the consumers.

3- Technology and Levels of Autonomy [3, pp. 17,59]

In order to make the dream come practical, there are lots of technological challenges to be solved to take the path of putting these cars on the road. Furthermore, the main industries involved in autonomous cars should have an extensive cooperation with third-party companies such as R&D Centers and product development enterprises. It is vital to have this collaboration even with actors of miscellaneous technologies to survive the contest and accelerate the emergence of automated cars. For the sake of illustration, joining with "*Telecom Services*" to develop *"Wireless Networks"* could be a normal combination.

4- Liability [3, p. 18], [6, p. 18]

This element is one of the most critical factors. Considering a case of accident for an autonomous car, it must be clarified who is the responsible for it; whether the person sitting behind the wheel and doing nothing, or the fully-automated car? Accordingly, liability should be considered in a more extensive concept, also including the role of insurance industry and their regulations for autonomous cars on the roads. *Insurance costs benefit from a structural decline in auto accident frequency that should continue with the autonomous cars*. However, the higher costs of autonomous cars in accordance to normal cars would increase the insurance tariffs for self-driving cars.

5- Legislation [6, p. 17]

Since, it is not actually necessary for the people who *"get behind the wheel" of autonomous cars* to have a typical driving license, the governments must release laws specific for the autonomous cars on the road and also their occupants. Recently in the US, California and Nevada, this has been done through issuing laws for the application of self-driving cars on roads and granting a specific license to the people getting on these cars.

6- Timeline of Penetration [3, pp. 18, 37-44]

The real outcome of autonomous cars will be noticed when the major part of the cars on the road are equipped to this technology. However, regarding the current statistics of 250 million cars on the US roads, and 1 billion worldwide, it will take decades for this technology to fully diffuse in the market. However, the process could be paced via *"government or industry aided funding and/or mandates."*

7- Infrastructure [3, p. 18]

Although, in the recent prototypes, the dependence on infrastructure has been decreased; there is still a need for some plain infrastructures as well as road markings and signage, GPS mapping, strong telecom networks and ideally some level of vehicle-to-grid (V2X) communication.

8- Security [3, p. 19]

Perceiving the autonomous cars from their computer-controlled nature, unavoidable security concerns raise. E.g. these systems could be hacked by experts in programming; Although it is so hard to hack them wirelessly and the hacker would require to get inside the car to connect some wires, the future technological development might make it easy to even attack the car's control system wirelessly.
This is why the *"AUTOSAR automotive software development standards"* have been stimulated to inhibit potential felonies to keep all the ECUs in the car, safe from hackers.

9- Regional Differences in Autonomous Car Development [6, p. 20]

Although the autonomous car concept and legislations is being developed in some countries as Japan, China, Europe, and Canada, so far, the main progress has been done in US.

2.2. OVERVIEW OF AUTOMOTIVE SAFETY

With the recent advancements in Automotive industry, we have entered a new era of enhancements in safety and comfort, plus optimization of fuel consumption and emission. Vehicle safety is a crucial topic for the long-term dream of integrated safe system approach. In the last few years there has been a large contribution of collision avoidance and primary safety technologies to the automotive industry. In the meantime, there are a lot of in-vehicle technologies being developed which are capable of increasing safety and reducing crash injuries. However, these systems have also the risky behaviors that might cause other problems to arise, while solving the previous issues. This is why a lot of research and test is needed to prove the safe functioning of these systems. [7, p.3]

As autonomous vehicles use sensing, planning, reasoning, and acting, there is a huge use of sensory data and V2V or V2X communications to obtain the information about situation, and then very sophisticated algorithms are used to interpret, process and convert these data to commands for the actuators. During this process, different types of failures might occur which are needed to be avoided by doing the following steps: [8, pp. 10-11]

- Evaluation of failure and modes and their impact
- Investigation of the enhanced capabilities to predict failures in a traffic scenario
- Exploring additional requirements for fail-operational driving
- Determination of required safety levels
- Development of methodologies for testing to demonstrate safety and reliability
- Providing standardized and certified test procedures and environments for fail-operational vehicles in any ambient condition

According to the European Commission's Car 21 Strategy, automotive industry should be leading in technology to drive clean, fuel-efficient, safe, and connected. Also, vehicle safety should be improved for both drivers, passengers and unprotected road users. Since year 2015, the European New Car Assessment Program (EuroNCAP) has developed a new role in assessing the safety quality of the e-Safety systems through Advanced EuroNCAP and a new road map. [7, p.3]

On the other hand, US Department of Transportation (DOT) with collaboration to National Highway Traffic Safety Administration (NHTSA) has released a guideline, considering the safety assessments for design, development, test and production of Highly Automated Vehicles (HAV). [8, pp. 3-16]

Subsequently, safety concepts can be investigated through several concepts and requirements such as surrounding conditions that may affect he performance of an autonomous vehicle, or the internal parameters that build up the control system of a self-driving car to replace the human driver. Sections 2.2.1 and 2.2.2 are devoted to introduce and give an overview of these concepts. [8, pp. 17-40]

2.2.1. FUNCTIONAL SAFETY WITHIN ISO 26262 [9]

Nowadays, one of the most important issues in developing new vehicles, is safety. As the new technologies come into automotive industry, in representation of every new functionality, it is essential to ensure that this new feature meets the functional safety requirements.

With regards to the increasing complexity in technologies, and software and mechatronic systems, the risk of systematic and hardware failures is increasing; which is needed to be studied under the standards of functional safety. Accordingly, ISO 26262 has been proposed as a standard series, including instructions to avoid or diminish these risks by suggesting related requirements and processes.

The ISO 26262 series of standards interacts with functional safety of Electronic or Electrical systems that is achieved through safety measures including safety mechanisms. It also delivers a framework within which safety-related systems based on other technologies (e.g. mechanical, hydraulic and pneumatic) can be considered.

Figure 2 shows the overall structure of the ISO 26262 series of standards, which is based upon a V-model as a reference process model for the different phases of product development.

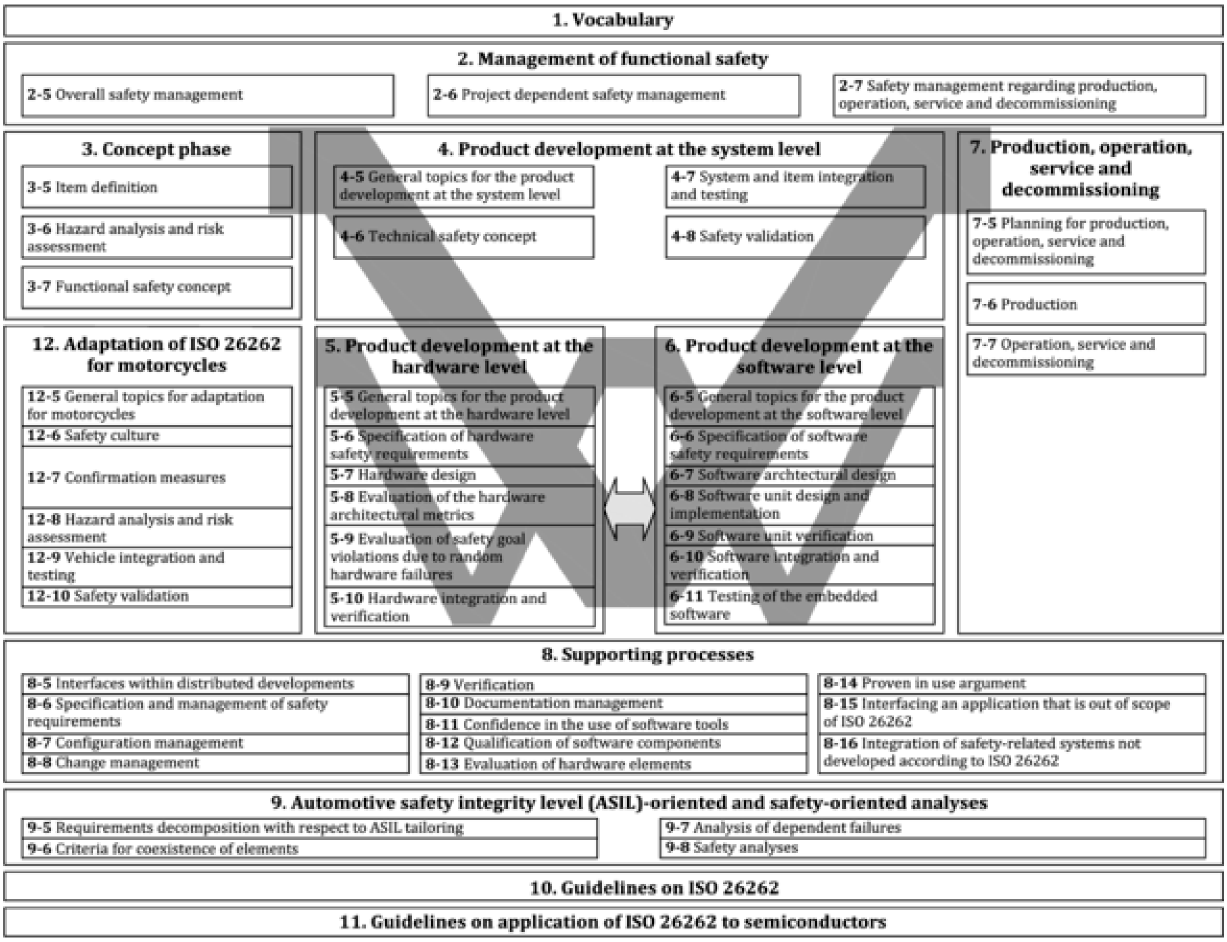


Figure 2. Overview of the ISO 26262 series of standards [9, p. viii]

2.2.2. SAFETY OF THE INTENDED FUNCTIONALITY (SOTIF)

As the technology of autonomous driving is reaching the roads, it raises many possibilities and questions about the safety of this new transportation system. Main concerns are regarding the substitution of human drivers with intelligent systems, which will impact concepts such as the ethical judgements, privacy and security on the roads. With regards to these issues, US Department of Transportation (DOT) provides a framework to address the actions that are needed in the autonomous driving world. A part of this framework is considering the Vehicle Performance for Automated Vehicles, which provides the practices for the safe pre-deployment design, development and testing of highly automated vehicles before their commercial production and their appearance on the roads. [10, pp. 7-15]

Safety assessments cover a wide range of areas in this reference. However, regarding the topic of this thesis which is focused on Collision Avoidance systems, it is worthy to consider the most relevant area as Object and Event Detection and Response (OEDR) which is a crucial filed to study before development of this system. [10, pp. 7-15]

OEDR addresses detection of any situation that is related to the real-time driving tasks and implementation of the automatic system or driver model, to cope with these situations. These cases include detection and responding to other vehicles on every lane of the road, pedestrians, cyclists, animals, and any other object that may interfere with the safe operation of autonomous car. Moreover, this capability should be operational in various conditions for any usual or unusual and emergency case that may affect the performance of autonomous car. Consequently, the aforementioned performance of AC can be divided in two main parts as “Normal Driving” and “Crash Avoidance Capability” which are described as below: [10, pp. 27-31]

Normal Driving is considered as the general cases that every AC would encounter during a regular traffic. There are a numerous set of functions which are used during this performance, from which, the main ones to mention would be: Car Following, Lane keeping, Making Logical Lane Changes, Detecting and Responding to Speed Limits and Changes, Obeying Traffic Rules, Responding to other Vehicles and Environment, Detection and Response to Emergency Vehicles and Police, etc. [10, pp. 28-30]

- **Crash Avoidance Capability** is considered as the ability to address pre-crash situations and handling them. This could be related to cases such as loss of control, crashes during lane change or lane merge, head-on and opposite direction or rear-end collisions, road departure or low-speed situations like performing a parking maneuver or driving backwards. [10, pp. 30-31]

All of these situations and similar cases should be considered and defined in the development and test phases of autonomous vehicles to ensure the safe performance of these cars on real roads and traffic conditions. In chapter 3, an AC model will be developed with regards to these concepts to simulate some selected scenarios in chapter 4. [10, pp. 27-31]

2.3. LITERATURE RESEARCH FOR COMPLEX SCENARIO GENERATION

The concept of AVs proposes the need for testing these vehicles under several conditions and scenarios to evaluate their performance. Obviously, there would be infinite number of scenarios defined on the roads which may vary in major or minor details. Therefore, in order to handle the limited room for testing, it is required to define a methodology to choose the most relevant scenarios which can include the most complex and critical cases. Furthermore, the results of these tests would be applicable to simpler scenarios since they would have been already considered in the generation of complex scenarios.

Regarding the scenario complexity, several important research has been already done by some researchers and they have used various technics for the automation of methodology in order to diminish the number of required test cases. In the following, some of these researches are considered to clarify the concept:

Alnaser [11] compares the AV verification to Hardware Design and introduces the AVVF framework and the scenario level of abstraction as the center of this framework. This framework connects functional verification, sensor verification, diagnostics and industry/regulatory communication with the scenario abstraction level. [11]

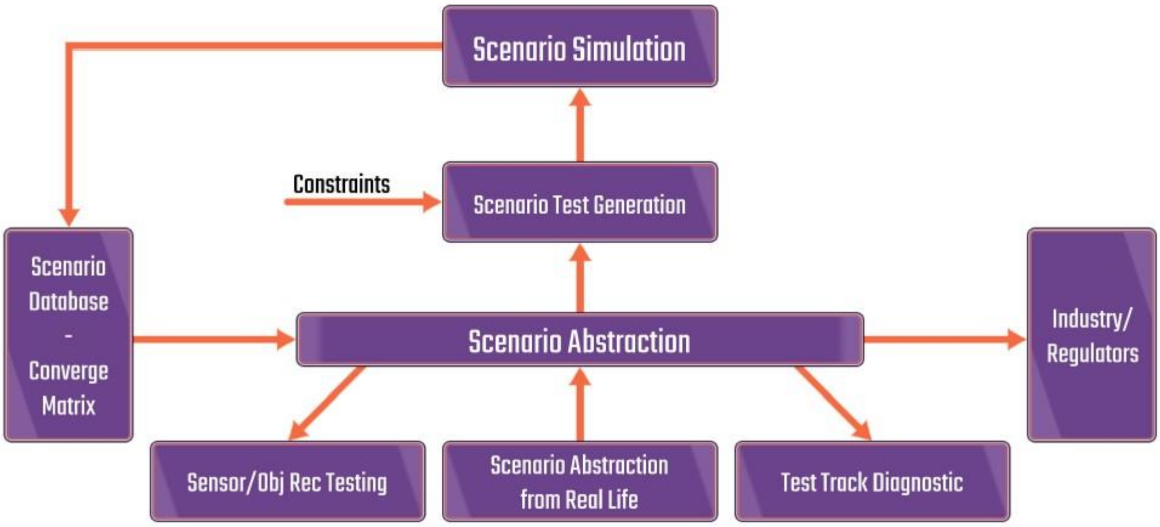


Figure 3. Scenario Abstraction and AVVF Framework [11, p.4]

Ambersbach [12] uses functional decomposition to reduce the required size of test suites for scenario-based testing. Later [13], he combines the scenario-based approach with a functional decomposition of the HAD function to be proved, to specify particular test cases based on the FTA. Plus*, a generic six-layer decomposition for HAD function is proposed based on a requirement definition. Furthermore, the potential to reduce the approval effort is outlined and a methodology to create test cases and to define corresponding fail criteria based on this decomposition and relevant scenarios is shown and applied to one exemplary scenario.*

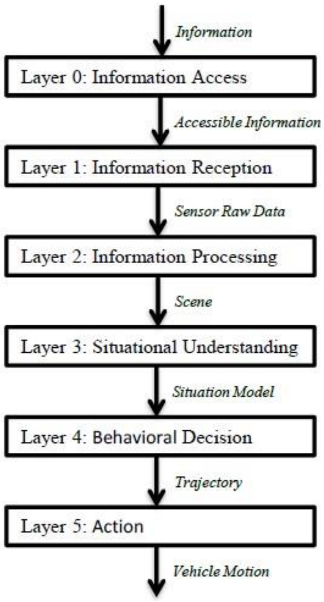


Figure 4. ***six-layer decomposition for HAD function [12, p. 4]***

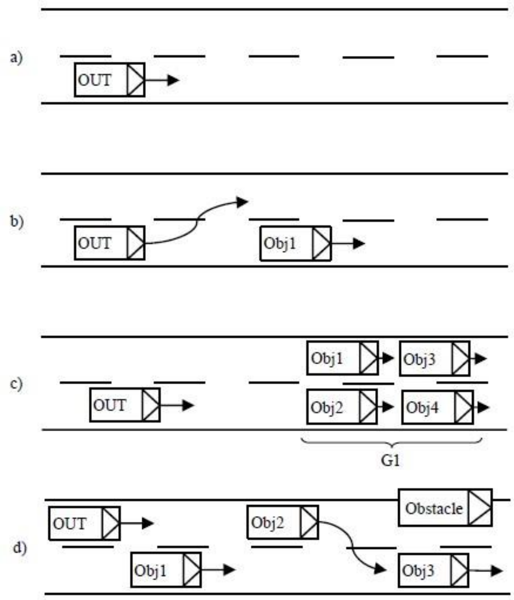


Figure 5. Exemplary Scenarios ***[12, p. 5]***

Huang's research [14] on test scenarios is considering eight relative positions of surrounding vehicles and possible moves of the Ego-Car and surrounding vehicles. Then he describes some test scenarios with one and two obstacle vehicles. They consider also the curved road scenario with two obstacles and introduce the important parameters in this type of scenarios.

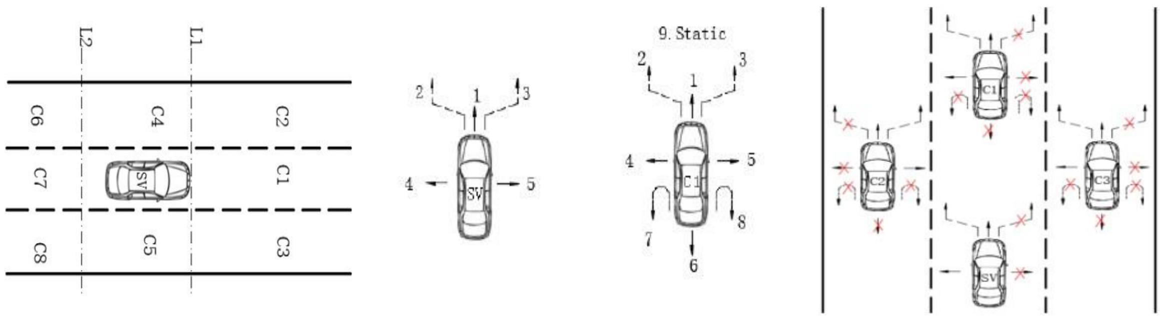


Figure 6. Possible moves of Ego-Car and surrounding vehicles [14, pp. 2,3,5]

Rocklage [15] introduces automated scenario generation for regression testing of autonomous vehicles, which is defined as a black-box system in a virtual simulation environment. This is done by combining the combinatorial interaction testing approach with a simple trajectory planner as a possibility checker to generate efficient test cases with variable coverage. Finally, the underlying constraint satisfaction problem is solved with a simple backtracking algorithm.

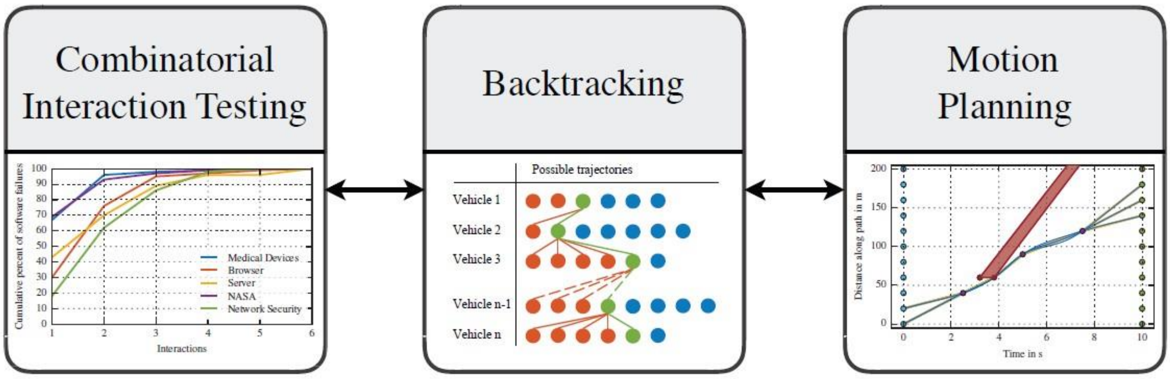


Figure 7. Main components of proposed algorithm by Rocklage [15]

Wang [16] has gone through the problem of massive data requirement for testing and performance evaluation of automated vehicles and states the difficulty for cognitive algorithms dealing with typical datasets that usually compromise of types of roadways, scenes and specific characteristics. They propose a traffic sensory data classification paradigm by quantifying scenario complexity for each roadway segment which is also based on road semantic complexity and traffic element complexity.

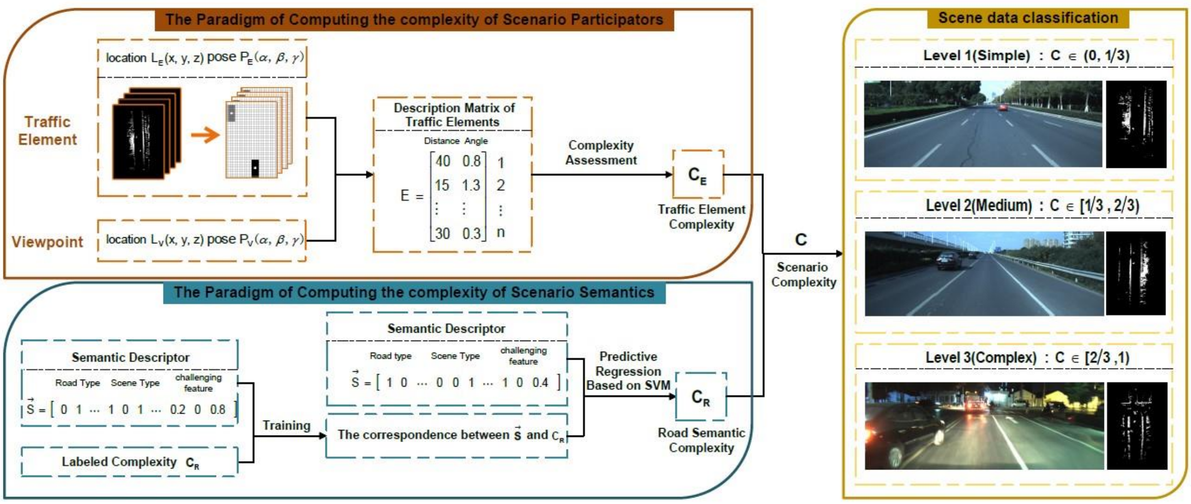


Figure 8. The detailed illustration of the proposed traffic sensory data classification via quantifying scenario complexity [16, p. 2]

Xia [17] has developed an automatic method using analytic hierarchy process (AHP) to generate test cases that is ensuring both coverage and effectiveness; Moreover, an improved test case generation algorithm is proposed based on the pairwise independent combinatorial testing tool (PICT), which ensures both combinatorial coverage and complexity of test cases. As a result, test scenarios are generated by clustering these discrete test cases, considering similarity and complexity. Then the cases with higher complexity are merged together to increase the test efficiency. For the validation of this method, they use a lane departure warning system (LDW).

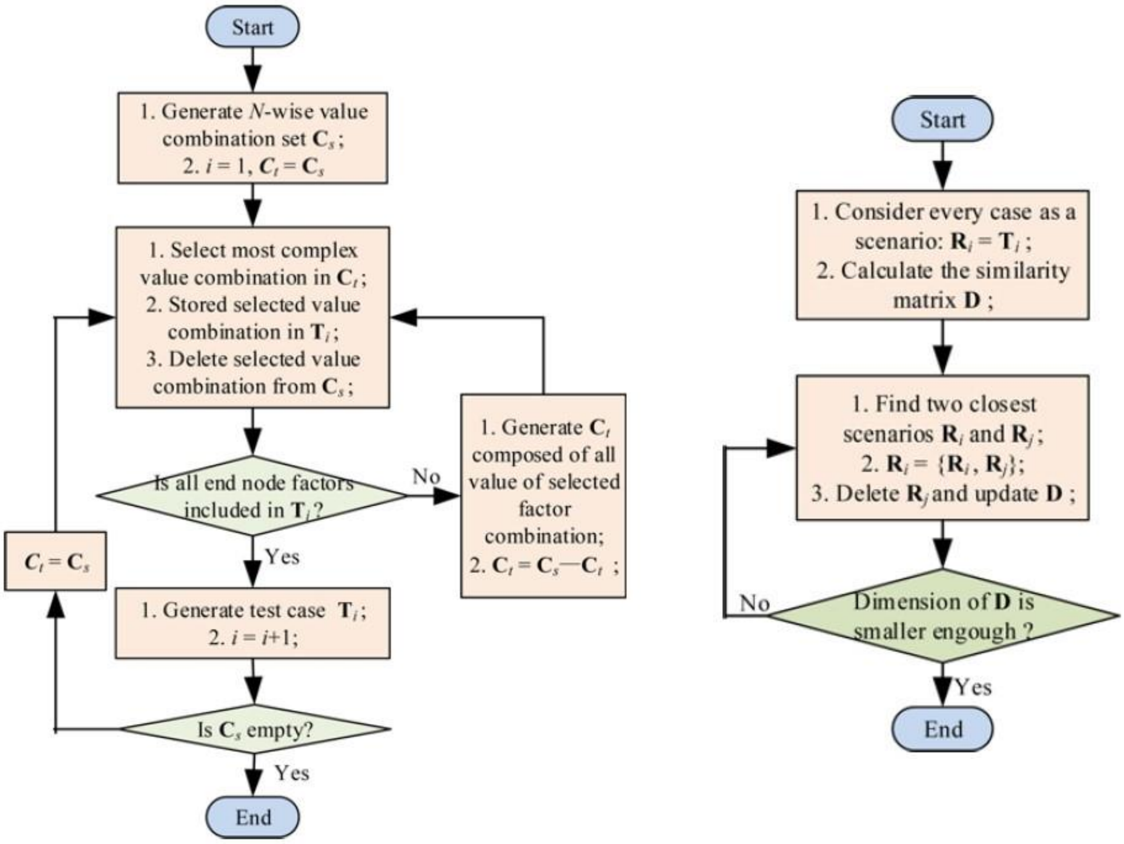


Figure 9. Test case generation and Scenario clustering procedures [17, p. 3]

The mutual points in all these researches are considering different elements which are affecting a scenario to be critical as the following:

- Road Conditions
- Environmental Conditions
- Surrounding Traffic (Vehicles, Humans, Animals)
- Vehicle Dynamics
- Disturbances

All these elements can have their effect on every single scenario in the real-world. However, in the phase of simulation and test, there are a lot of limitations in the simulation platforms, such as user interface, capability of including environmental conditions, and disturbances while simulating the performance of vehicle dynamics and controller. However, by a conclusion of different research, each one including an aspect of these parameters, and doing ground vehicle tests, it is possible to get a good level of reliable results for the test of different scenarios.

2.4. ADVANCED DRIVER ASSISTANCE SYSTEMS (ADAS)

With regards to the alarming statistics of over 1 million deaths on roads globally per year, plus 50 million injuries, from which, about 30% are caused by high speed and about 21% caused by driver distraction, it is essential for new technology to take action. Advanced Driver Assistance Systems are the required tools to avoid collisions and accidents whether by alerting the driver or taking over control of the vehicle. Nowadays, there are numerous number of ADAS functions such as ACC, LKA, LCA, AEB, FCW, TJA, etc. which are evolving rapidly. Obviously, a complete Autonomous Car is equipped with all these functions in order to perform the full required driver behavior in a car involved in various traffic situations. However, there are already cars on the market which are not fully automated, but use a number of these facilities to help the driver's performance on roads. [18, p. 1-3]

ADAS compromises of three concepts working together as "Sense, Plan, Act". These three concepts are introduced in the following thoroughly:

2.4.1 Sense

2.4.1.1. Sensing Technologies

Main systems used to implement ADAS are "Radar, LiDAR, Cameras, Ultrasonic Sensors, and GPS". With these systems, the car gets the angular field of view from about 20° to 360°, plus the distance in front, back and sides of the car, with regards to the systems used. [18, p. 4-5]

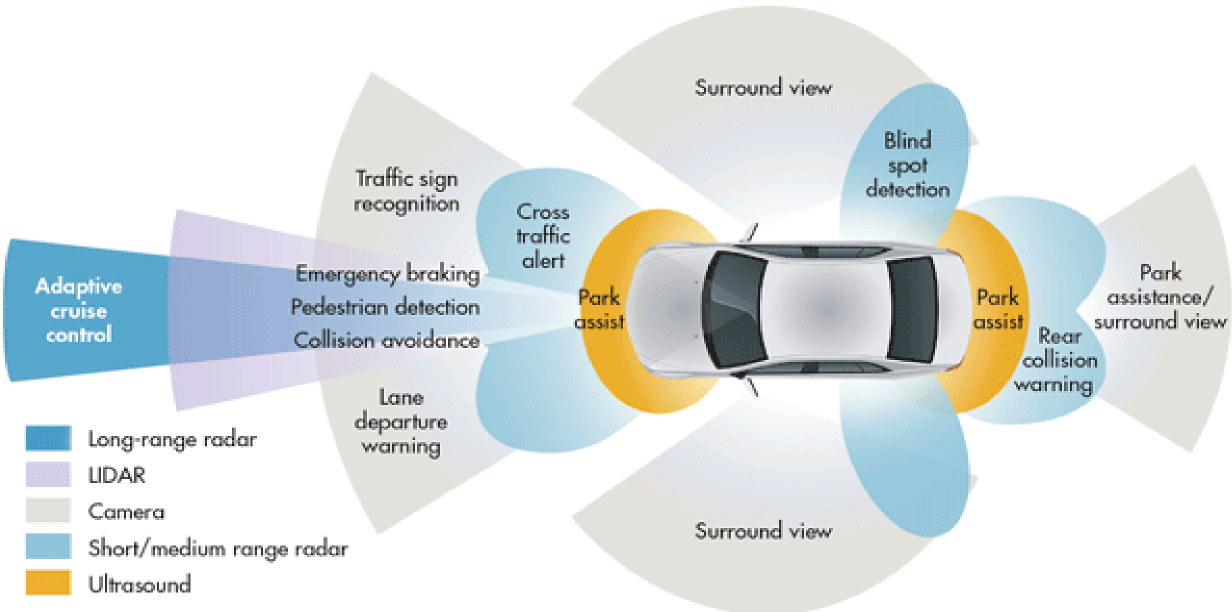


Figure 10. Sensing Technologies in cars [19]

Each of these technologies are described briefly in the following:

- **Radar [20, pp. 4-6]**
  A Radar system consists of a transmitter producing electromagnetic waves, a radio receiver and a data processing device/system. Depending on the system, radio waves are pulsed or continuously transmitted from the transmitter, which reflect off the object and return to the receiver, giving information about the object's location. Radio waves are reflected if they meet an object, so if the reflected waves are received again, it means that there is an obstacle detected in the range of radar. Accordingly, vehicle systems use short, medium and long range for different ADAS goals.
  Radar has several advantages compared to visual observation. For instance:
    - Ability to operate day or night, in light or darkness over a long range;
    - Operating in different weather conditions such as rain, fog and snow;
    - Detection and Tracking of moving objects with the possibility of high resolution imaging, which results in object recognition
    - Automatic operating with no need for human intervention
- **LIDAR [20, pp. 7-9]**
  LIDAR is an acronym for Light Detection and Ranging, which was first developed in early 1960s, is a blend of light and radar. It bounces light off objects to detect their position, in the same way that radar uses radio waves. The system works transmitting pulses of light and counting the time until it's return and calculating the distance from objects, accordingly. Large field of view in LIDAR in comparison to narrower field of view in radars is a great advantage, so that LIDAR can create 3D data imaginary and so, detecting the shape of objects. It is also capable of detecting obstacles on the curves of a road.
  Working principle of LIDAR is based on firing rapid pulses of laser light at a surface and a sensor measures the time for the return of each pulse. As the light travels in constant speed, the calculation of distance in this instrument is very accurate. The repetition of this principle, leads to a complex map of the target object and its distance.
  LIDAR usually compromises of the following units:
    - Laser: emitting laser light (safe for eyes)
    - Scanner and Optics: to determine resolution and range
    - Photodetector and Receiver: to read and record the signals
    - Navigation and Positioning: just used for systems creating live mapping

  Advantage of LIDAR versus Visual Observation:
    - Identifying structure of the obstacle to distinguish objects based on size and mass
    - Better range and filed of view than other sensors, to detect obstacles on curves
    - Manipulability to enable the system to take evasive actions
    - Providing 360° sensing around a vehicle to combine with GPS and create a constantly changing "live map".

- **Cameras [20, pp. 10-12]**
  Cameras are optical instruments that can act as the eyes of a vehicle. Their functioning is very similar to the functioning of the human eye and it can record or capture images that might be stored locally or transmitted to another location. As cameras sense objects without any contact, they are considered as remote sensing devices.
  As today's cameras process digital data, this is done by using a sensor which converts the light energy into electrical energy, and each point of light is known as a pixel which becomes the digital data and can be seen as an image instantly, or can be stored for later use as an image. The quality of the image is very dependent on the lens, sensor and its processing capability.
  Cameras used in ADAS usually do not create images, however, they use the data in real time to generate a 3D data of the obstacle and send it to various control units to process it for the vehicle control purposes or alarming the driver.

- **Ultrasonic Sensors [20, p. 13]**
  Ultrasonic use a form of Sonar (Sound Navigation and Radar) which is used to detect the distance and direction of an object by calculating the time for a wave to travel to the target and back. An Ultrasonic Sensor is a set of a speaker and a microphone to emit and receive the ultrasound which is a very high frequency acoustic wave beyond human hearing range. Ultrasonic Sensors used in vehicles are for the purpose of detecting obstacles while parking manually or automatically. These sensors are one of the cost-effective technologies, widely used in today's cars.

- **Global Positioning System (GPS) [20, p. 14]**
  GPS receives the satellite and signal information transmitted from satellites that circle the earth, and uses this data to calculate the user's location. This technology has been in use for many years as the mean of navigational aid for drivers to lead them through routes to find the most accessible roads and reach a specific house number accurately.
  The great functionality of GPS used in ADAS is to identify the vehicles position and provide predictive information such as sharp turns or hazardous situations ahead of the car; and obviously when used in an autonomous car, it provides the full road information in front of the car.

2.4.1.2. Sensor Fusion [20, pp. 15-16]

As no individual sensing system is capable of providing 100% correct functionability in all weather conditions and due to several variable conditions which occur during driving a car, by merging different sensing systems together, it is more feasible to reach a more consistent system to act securely in all different situations.

Sensor fusion is the result of achievements in data processing systems and gives the opportunity to merge data from sensors and also prioritize the most accurate signal to be used as the driving signal for ADAS actuators. With this approach, the weakness of a typical kind of sensor in a specific weather condition is compensated by another sensor. For instance, in a snowy weather, the radar signal takes the priority over others as it is more accurate in snow in comparison to other kind of sensors.

Another advantage of sensor fusion is to use different sensors for different levels of functions within an ADAS function. For example, AEB can use radar for long and short ranges, while using LIDAR for mid-range, speed and 3D object detection. Moreover, cameras would be used for getting the detailed information about road and pedestrian position.

To give a view of sensor capabilities, radars are very good in detecting nearly every object, however, they cannot recognize the type of the object. While vision systems can read street signs and lane markings, but their performance is not good when there is a direct sun light or when the lenses of cameras are dirty. However, LIDAR bridges these two deficiencies together and covers variables and details to provide a more complete system.

Amalgamation of different sensory systems gives a higher efficiency than a single system and widens view from about 20 degrees to nearly 360 degrees in the best cases.

Simulating synthetic radar and vision detections provides the ability to create rare and potentially dangerous events and test the vehicle algorithms with them. In an autonomous vehicle, there are several sensory systems which work simultaneously to detect both road and lane boundaries, plus other vehicles and any other object or environmental condition.

The signal from these sensors and cameras are transmitted to the central computer of the car and translated using sensor fusion methods to be used for actuators. Accordingly, actuators perform the required task and enable or disable ADAS functions to accomplish the control task of a full or semi-autonomous vehicle.

One important factor in sensor fusion is the implementation of sensors and cameras in correct places so that they can cover the most optimal range needed for the car. Although there would be some blind spots in the sides, with this set of sensors and cameras, a very high percentage of vision in front, back and sides of the car is covered.

Consequently, when there is enough data from sensors, and translated in the central computer of the car, it is just the controller tasks which enable, disable, or merge ADAS functions to keep the car in the best performing condition in critical driving scenarios.

In the next sections, four major ADAS functions which will be used for designing the Test-Bench in this thesis, are introduced.

2.4.2. PLAN

2.4.2.1. OVERVIEW OF AUTOMOTIVE CONTROL SYSTEMS

With regards to the increasing demand for autonomous driving in the last two decades, the focus of many research groups has been changed to control design for automated vehicles. The aim of these research is to increase safety and reduce collisions, besides traffic utilization and energy efficiency. Briefly, the main control task of an autonomous car is to drive from the origin to destination, safely and efficiently, while obstacles and impossible maneuvers are avoided. [8, pp. 43-44]

This performance is usually reached in three main steps as:

1) Computing the shortest or best path from A to B
2) Processing environment data, computing the possible lanes and goal points
3) Computing the best trajectory based on previous steps

To reduce the complexity of the control tasks during highway driving, the dynamics of a HAV is usually divided into longitudinal and lateral motion. [8, pp. 44-46]

Usually the longitudinal behavior is modeled by a simple first order system, but the lateral behavior is more complex and is modeled with kinematic or dynamic bicycle models.

A vehicle's motions control is very crucial for the purpose of Advanced Driver Assistance Systems and Autonomous Driving; as the vehicle should define and follow a trajectory, while maintaining the dynamic capabilities of the system and not exceeding maximum stability limits. For this reason, a controller must have some important features to meet the requirements of autonomous cars' control. These characteristic are as below: [8, pp. 48-49]

- Real-time Capability: Execution of the control law on an embedded control unit within a defined and guaranteed calculation time
- Parametrization: Easily tunable parameters
- Structure: Ability of controller to work on different vehicles
- Robustness: Due to unknown conditions and disturbances, robust performance is required
- Nonlinearities / Dependence of Vehicle Speeds: Controller must work from zero speed to at least 130 km/h. [8, pp. 48-49]

These characteristics can be reached by a 2DOF controller including a feed-forward term based on the reference trajectory, and a feedback controller for disturbance rejection. Moreover, in some controllers, the Look-Ahead Distance (LAD) capability is introduced to improve the controller performance.

There are various control systems that can be used to obtain different goals of autonomy. In the following section, a short introduction to each system is provided:

2.4.2.2. INTRODUCTION TO VARIOUS CONTROL SYSTEMS

- **PID Control [8, pp. 49-50]**
  The PID controller has a simple control law, taking into account the error variable P as "proportional", I as "Integral ", and D as "Derivative" of the error variable.
  This controller's advantage is its generic applicability; however, usually this controller is outperformed by other control approaches.

- **Fuzzy Control [8, p. 50]**
  Similar to a PID controller, also Fuzzy Control uses the error with its integral and derivative. Its application is usually in cases that there is no mathematical model or cases that it is difficult to obtain models. This makes it possible to use this controller for nonlinear dynamics and systems with multiple inputs and outputs.

- **Neural Networks [8, p. 50]**
  A system of interconnected neurons, where each connection has a weight, tuned by training data or online. This builds an adaptive net capable of learning and a controller can be designed based on a model, using this approach.

- **Linear Quadratic Regulator (LQR) [8, p. 51]**
  This controller uses a linear plant model and optimal control theory to obtain an optimal state feedback controller. This approach needs the information of a plant model in advance and actual signals of all states during operation, thus requires a state observer.

- **Feedback Linearization [8, p. 51]**
  This is a common technic which renders the closed-loop system linear with the help on nonlinear compensation.

- **Sliding Mode Control (SMC) [8, p. 51]**
  SMC relies on a variable structure controller and is robust with respect to a specific class of modeling uncertainties and external disturbances. Two main consisting parts of SMC are "Definition of a desired dynamic variable by the sliding variable" and "Defining a controller to obtain the desired dynamic".

- **H_Infinity Control [8, p. 51]**
  This is a robust approach which controls a plant affect by modeling uncertainties and parameter variations

- **Model Predictive Control (MPC) [8, pp. 51-52]**
  The principle of MPC is using a model at each time step to predict the behavior of system over a predefined horizon. Advantage of MPC is the possibility of consideration of different

types of constraints for states and inputs, while its high computational complexity is a disadvantage that yields to scarce use of it in real-time applications.

As in this thesis, the focus is on MPC control, this control approach is described in section 2.4.2.3.

2.4.2.3. MODEL PREDICTIVE CONTROL (MPC)

This strategy is widely adopted in industry as an effective means to deal with multivariable constrained control problems. The main idea of MPC is to choose the control action by repeatedly solving on-line an optimal control problem. This aims at minimizing a performance criterion over a future horizon and yields an optimal control sequence, possibly subject to constraints on the manipulated inputs and outputs, where the future behavior over a specified time horizon, is computed according to a model of the plant. This future behavior is usually called the prediction horizon. At each discrete-time instant k, the measured variables and the process model (linear, nonlinear or hybrid) are used to (predict) calculate the future behavior of the controlled plant. This is achieved by considering a future control scenario, which is usually called control horizon, as the input sequence applied to the process model, which must be calculated such that certain desired constraints and objectives are fulfilled. The first control in this sequence is applied to the plant. At the next time step the computation of the optimization is repeated starting from the new state and over a shifted horizon, leading to a moving horizon policy. This is the main difference from conventional control which uses a pre-computed control law. An important advantage of this type of control is its ability to cope with hard constraints on controls and states. Nearly every application imposes constraints; actuators are naturally limited in the force (or equivalent) they can apply, safety limits states such as temperature, pressure and velocity and efficiency often dictates steady-state operation close to the boundary of the set of permissible states. The prevalence of hard constraints is accompanied by a dearth of control methods for handling them, despite a continuous demand from industry that has had, in their absence, to resort often to ad hoc methods. Model predictive control is one of few suitable methods, and this fact makes it an important tool for the control engineer. [34]

The main advantage of MPC is possibility of explicitly handling constraints. Plus, designing a controller for non-linear systems is easier in MPC. [8, p. 57]

2.4.3. ACT

2.4.3.1. OVERVIEW OF VEHICLE MODELING [20, PP. 301-302]

Considering the scientific and technical competition between car manufacturers, it is a crucial issue for these manufacturers to provide their designed brand new car to the market as soon as possible. Furthermore, as much as the gap of transition from concept to product would be diminished, better the market for that car with newest features would be in the market. [20, pp. 301-302]

These new generation of cars are equipped with a numerous number of electrical and electronic systems. From which an important fraction is used for safety-relevant purposes. Accordingly, to have the best results in the production, it is necessary to do computer simulations before the start of producing a car. The goal of these computer models is to monitor and reveal the possible failures in the dynamic behavior of the vehicle when operating with other sub-systems in the car. [20, pp. 301-302]

However, as each individual car would have a different vehicle dynamics model than others, in this phase, it is required to have specific know-hows for each car to reach the best simulation results. [20, pp. 301-302]

Figure 11 represents the standard vehicle-driver-road control loop:

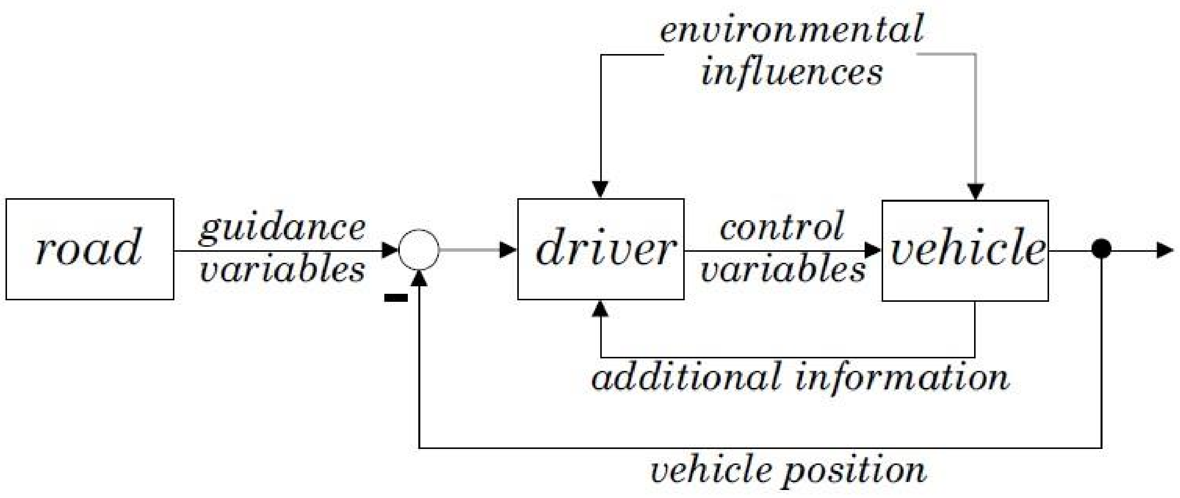


Figure 11. The Standard Vehicle-Driver-Road Control Loop [20, p. 302]

In modeling of vehicle dynamics, it is required to consider some important issues: [20, pp. 301-302]

- Reducing the complexity of model to a sufficient level for vehicle dynamics
- Implementation in a widely used programming language for widely spread use
- Interaction of the sub-models considering design and simulation time
- Accuracy, only up to necessary level in order to reduce time consuming tests

2.4.3.2. COORDINATE SYSTEMS [20, PP. 302-304]

In order to analyze the theoretical aspects of vehicle dynamics and design a controller, it is required to identify the equations of motion and write the interaction between different subsystems in the form of mathematical equations. While the most exact models are produced with the methods of theoretical physics such as Lagrange and Euler, however, these equations lose their reference to physical quantities as the calculations are based on general coordinate systems. Alternatively, modeling a vehicle in the simplest way and with the least computing time possible, is a better approach. Considering the simplified models, usually a variation is made between models, for the purposes of drive dynamics and vertical dynamics analysis without linking them together. However, some methods merge the vertical dynamics and drive dynamics while including all important non-linearites. Consequently, calculations are also limited to four coordinate systems as: [20, pp. 302-304]

- **COG:** Chassis (Center of Gravity) coordinate system
- **Un:** Undercarriage system
- **W:** Wheel coordinate system
- **In:** Fixed Inertial system

Excluding fixed inertial system, all other coordinate systems move with the driving vehicle.

Figure 12 Represents the COG coordinate system for a car with 6 degrees of freedom

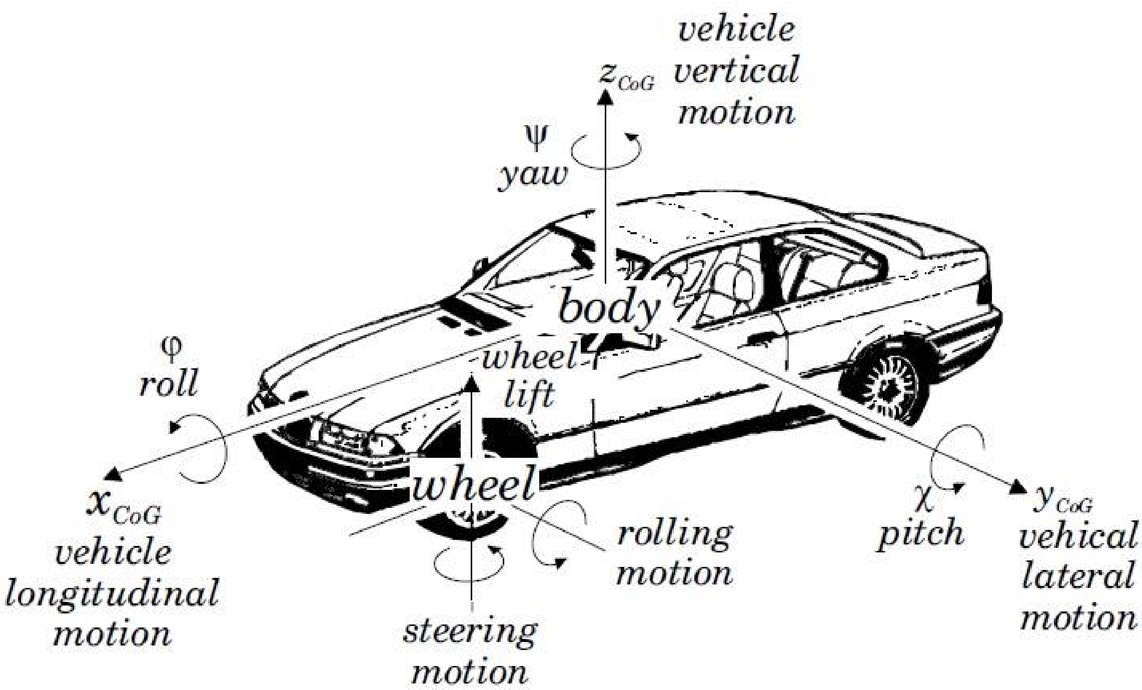


Figure 12. COG coordinate system for a car with 6 degrees of freedom [20, p. 303]

2.4.3.3. VEHICLE DYNAMICS MODELS

Vehicle dynamics models are distinguished with regards to degrees of freedom. Most simplified model is a 2DOF model, which provides the lateral and yaw motions. The use case for this model is when there is no need to consider the longitudinal behavior of vehicle. [21. pp. 13,14]

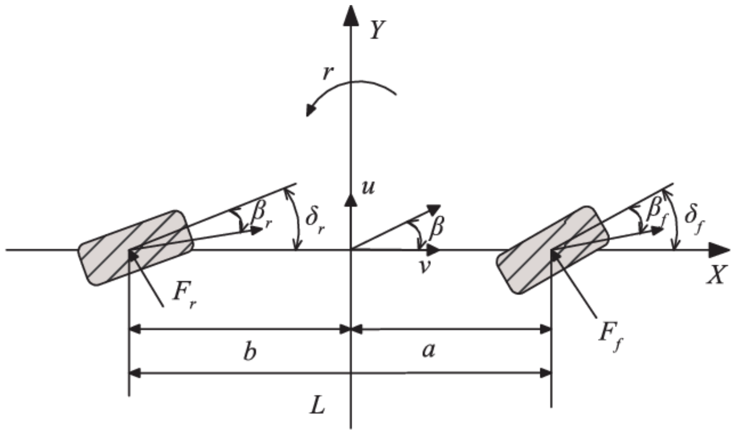


Figure 13. 2DOF vehicle model [22, p. 3]

On the other hand, there are 3DOF models which consider also the longitudinal behavior of vehicle. These models are widely used for simulation purposes in which several behaviors of a vehicle such as velocity, acceleration, braking, and steering are being studied. 3DOF models are considered in two configurations as “single track” and “double track”. This is described with regards to the number of vehicles and axles considered in the model. Single track model is also called bicycle model as it is considering one wheel in front and on wheel in back, which are connected together with one axle; while, a double track model has 2 wheels connected together in front and 2 wheels connected together in back, while the two axles are also connected together in the center. [21. pp. 14,15]

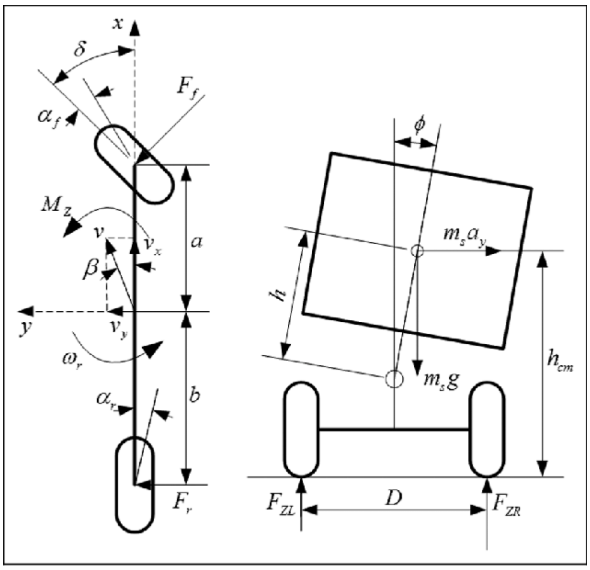


Figure 14. 3DOF vehicle model [23, p. 4]

Higher degree of freedom models is not considered in this thesis and thus not represented.

2.4.4. ADAS FUNCTIONS IN THIS THESIS

2.4.4.1. ADAPTIVE CRUISE CONTROL (ACC) [24]

As one of the core systems in ADAS, Adaptive Cruise Control (ACC) is widely used in the cars nowadays and indisputably, this system is one the most crucial systems for autonomous driving concepts. Having the word adaptive in its name, this technology is the most up to date and upgraded version of normal cruise control that was able to keep a constant speed. However, thanks to sensor fusion technology and a controller, ACC is capable of keeping a safe distance from the car travelling in front of it, while maintaining the desired velocity. Obviously, as far as the driver choses to keep a larger safe distance, ACC will first reduce speed to increase the relative distance, and then according to the velocity of the lead car, if it would be higher than driver's desired speed, it will keep the set velocity; However, in case that the proceeding car has a lower speed of driver set velocity, it will keep the distance and thus, the velocity would be lower than set value. ACC sets the priority of distance or velocity, regarding to the traffic situation and capability of the system.

2.4.4.2. LANE KEEPING ASSIST (LKA) [25]

This system is one of the essentials of autonomous driving, as in order to perform other functions of ADAS, it is essential to first detect the lane in which the car is located and keep in lane, and then perform the following tasks of driving.

There has been various research in this field before, using different methods such as fuzzy logic and model predictive control.

Several methods are capable to be implemented in a system. However, the most feasible system for this thesis, with regards to simulation facilities will be chosen in chapter 3.

2.4.4.3. LANE CHANGE ASSIST (LCA)

Lane Change Assist (LCA) is a system that performs the maneuver of changing lane in an Autonomous Vehicle. There are various forms of LCA with regards to the task they perform. Full LCA is capable of doing a lane change in both normal and emergency situations and decides like a human driver to perform a complete task of overtaking. However, there are also simplified models of LCA which perform the primary lane change but do not return the car to previous car and thus, the task of overtaking is not possible with these systems. Some other systems are also able to do a lane change to left and continue until there is a new situation to do another lane change to the other left lane.

Several research has been done and a lot are still being developed in this field to improve the controllability of cars in critical situations and doing more smooth turns while avoiding a collision. Accordingly, the most appropriate method to meet this thesis requirements will be taken into account to implement in the system.

2.4.4.4. AUTOMATIC EMERGENCY BRAKE (AEB) [26]

Automatic Emergency Brake (AEB) is an advanced active safety system that helps drivers avoid or mitigate collisions with other vehicles or vulnerable road users. AEB systems improve safety by:

1. Avoiding accidents by recognizing critical conditions and braking with great power
2. Mitigating the severity of inevitable collisions by reducing the speed of the car

2.5. RESEARCH GAP

Regarding the aforementioned issues about legislation, the emergence of fully automated cars on the roads will take a while. In the meantime, research is being extended on improving every individual ADAS function and adding them to normal cars to prepare the steps for the arrival of ACs on the roads. However, despite the numerous research in the recent years, studying several ADAS functions and also sensor fusion and vehicle dynamics, there is not so much material with regards to the combination of these functions and prioritizing them to test the very crucial cases in which normal functioning of ACC or LKA would not be enough to avoid collision.

Furthermore, this thesis is considering the implementation of four main systems as mentioned, to evaluate the cooperation of these control systems together.

Next chapters will be devoted to the implementation, simulation, and evaluation of these systems when working simultaneously.

3. METHODOLOGY

3.1 IMPLEMENTATION OF AN AUTONOMOUS CAR TEST-BENCH IN MATLAB & SIMULINK

Following the definition of Automotive Safety and Control, ADAS and Driving Scenario concepts, a Test-Bench is needed in order to simulate some selected scenarios which will be discussed in section 3.2, and check if the metrics of criticality for scenarios are also consistent in simulation and if the Ego-Car can use ADAS functions appropriately to do the safe trip and avoid collision or not.

The main aim here is to avoid any collision. So, as the surrounding conditions and disturbances could not be controlled, the only option to avoid collision is the intelligent functioning of Ego-Car and controlling the velocity, acceleration and steering angle of the Ego-Car by the use of an adaptive controller. For this purpose, and Adaptive MPC controller and some PID controllers are recruited to accomplish this task.

Ego-Car is modeled using a Bicycle Model and classical formulas of vehicle dynamics. Plus, it is equipped with 8 sensors as described in section 3.1.2.3 to create data for sensor fusion and tracking and act accordingly. This Ego-Car model has been implemented in the Test-Bench by merging and configuring four ADAS functions (ACC [24], LKA [25], AEB [26], LCA) in Matlab & Simulink.

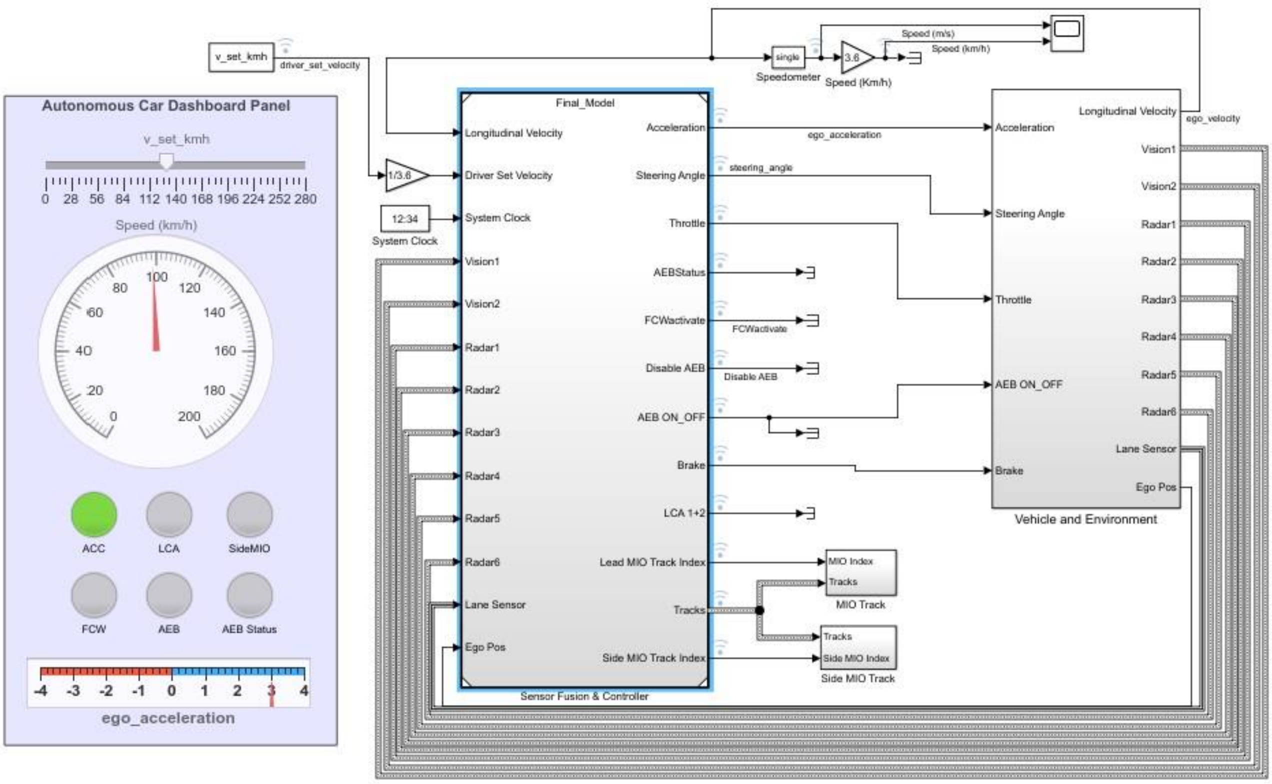


Figure 15. Test-Bench User Interface

This Test-Bench contains two main Blocks as described in table 1:

Table 1. Main Blocks of Test-Bench and their corresponding input/outputs

Block	inputs	outputs
Sensor Fusion and Controller	- Longitudinal Velocity - Driver Set Velocity - System Clock - Radar and Camera Detections - Position of Ego-Car	- Acceleration - Steering Angle - Throttle - Brake - FCW Activation - AEB ON_OFF - AEB Status - Tracking Results
Vehicle and Environment	- Acceleration - Steering Angle - Throttle - Brake - AEB ON_OFF - AEB Status	- Longitudinal Velocity - Radar and Camera Detections - Position of Ego-Car

- **Sensor Fusion & Controller** includes the sub-blocks for tracking and sensor fusion, plus controller blocks for ADAS functions. This block is described in section 3.1.1.
- **Vehicle and Environment**, consists of Vehicle Dynamics, plus Actors and Sensor Simulation sub-blocks, as described in section 3.1.2.

Also, the Dashboard Panel displays the following information about Ego-Car:

- **V_set:** defined as the desired speed of Ego-Car to reach when possible in the scenario
- **Speed:** Longitudinal Velocity of Ego-Car (km/h)
- **Ego_Acceleration:** Acceleration of Ego-Car while using ACC
- **ACC:** Represents the Activation or Deactivation of Adaptive Cruise Control
- **LCA:** Represents the Activation or Deactivation of Lane Change Assist
- **Side-MIO:** Represents the presence of Side-Car in right or left lane and in relative $\Delta x = \pm 5m$
- **FCW:** Represents the Forward Collision Warning as described in section 3.1.1.5
- **AEB:** Represents the Activation or Deactivation of Automatic Emergency Break
- **AEB Status:** Shows the different steps for emergency braking described in section 3.1.1.5

To maintain the control goals, the system includes combined longitudinal and lateral control of the Ego-Car as following:

- **ACC (Longitudinal control)**: Maintains a driver-set velocity and keeps a safe distance from the preceding car in the lane by adjusting the acceleration of the Ego-Car.
- **LKA (Lateral control)**: Keeps the Ego-Car travelling along the centerline of its lane by adjusting the steering of it

- **LCA (Lane Change Assist):** Enabled when there is enough space in one or both of the side lanes of the Ego-Car and the ACC system cannot maintain the safe distance from the lead car if it wants to reach the set velocity higher than the velocity of lead car
- **AEB (Automatic Emergency Braking):** Enabled when ACC cannot maintain the safe distance from the lead car and LCA cannot perform because the side lane is not free or the relative distance is too small that lane changing is not safe and might yield to collision

This system can adjust the priority of the goals to avoid collision, when they cannot be met simultaneously.

As already stated, the priority of the system is set to use LCA in the second stage of collision avoidance. So, if the system would have been able to avoid collision and meet ACC set velocity without lane change, LCA will not be used. However, if LCA is performed due to logic explained previously, AEB might be needed in the new lane to avoid collision. In the other case, AEB would be used as the first priority if there would not be enough time for avoiding collision with ACC or LCA.

3.1.1 SENSOR FUSION AND CONTROLLER BLOCKS

This block consists of subsystems which do the sensor fusion and control the Ego-Car's behavior on a highway / road with the aid of four ADAS functions (ACC, LKA, LCA, AEB). Detailed inputs and outputs of these subsystems are represented in table 2.

Table 2. Description of inputs/outputs for Sensor Fusion and Controller Blocks in Simulink Model

Block	inputs	outputs
Tracking and Sensor Fusion	- Radar and Camera Detections - Lane Detections	- Relative X and Y Distances - Relative X and Y Velocities - Tracks and Track Indexes - Ego-Lane
Estimate Lane Center	- Lane Detections - Longitudinal Velocity	- Road Curvature - Lateral Deviation - Relative Yaw Angle
MPC Controller	- Driver Set Velocity - Road Curvature - Lateral Deviation - Relative Yaw Angle - Longitudinal Velocity - Relative X and Y Distances - Relative X and Y Velocities	- Acceleration - Steering Angle
LCA Enabler	- Longitudinal Velocity - Relative X - Ego Position	- Headway Time (HT)* - LCA ON_OFF

LCA System	- LKA Steering Angle - R, L, Side-MIO Indexes - Ego Position - Ego-Lane	- LCA Steering Angle
AEB Enabler	- Headway Time - LCA to LKA Switch signal - Relative X - Relative Vx	- Time to Collision (TTC)** - AEB ON_OFF
AEB System	- TTC - Longitudinal Velocity - Acceleration	- Throttle - Brake - FCW - AEB Status - AEB Disabler

*Headway Time = $\frac{Relative\ Distance}{V_{ego}}$ ** TTC = $\frac{Relative\ Distance}{Relative\ Velocity}$

In the following sections, each block for the implementation of ADAS functions in Test-Bench will be described:

3.1.1.1 TRACKING AND SENSOR FUSION

This block receives the signals coming from sub-block Actors and Sensor Simulation inside the Vehicle and Environment main block as input and then transmits these signals, plus the prediction time (system clock) to the “Multi Object Tracker” [27] block which has a pre-built function in Matlab and creates confirmed tracks of the road and surrounding cars. Then these confirmed tracks are sent to three different functions described in the following, for defining the Lead-Car in front (MIO) and the cars in the side lanes of the Ego-Car (Side-MIO) or the car which innervates the Ego-Lane from a side lane and changes role from Side-MIO to MIO and appears as the new Lead-Car. The input signals are updated continuously, giving the opportunity to Ego-Car for performing the tasks on time.

In order to get the tracking information with regards to Ego-Car, it is required to first define the position of Ego-Car and measure all other tracking with regards to the position of Ego-Car. For this reason, at first, a function defines the Ego-Lane according to the initial position of the Ego-Car in the scenario. This approach is used due to limitations in the user interface of scenario generator in Matlab. The function to define this is using an algorithm which is able to find the Ego-Lane in all roads up to 4 lanes. However, it is needed that in defining the scenario, Ego-Car is initially placed exactly in one of the exact values stated in table 3. This will put the Ego-Car in the middle of corresponding lane.

This algorithm can be improved to define each lane with a range between two boundaries of the lane. However, as it requires some changes in the function, and it this thesis it is enough that Ego-

Car starts from the center of the lane, this function is satisfying the simulation requirements; but it can be extended for later need.

Table 3. Ego-Lane Definition Rule

Road Type	Initial Y position of Ego-Car*	Ego-Lane
2 Lane Road	2	1
	-2	2
3 Lane Road	4	1
	0	2
	-4	3
4 Lane Road	6	1
	2	2
	-2	3
	-6	4

Scenarios that will be tested in this thesis, will be considering only cases of 4 lane highways, so the initial Y position can get values of 6, 2, -2, -6 and Ego-Lane is defined accordingly.

As previously mentioned, the algorithms used in functions to find MIO and Side-MIO are also represented in the following:

MIO: A horizon of 100 m in front of the Ego-Car and in the same Ego-Lane is checked for tracks of an object. This is done by first finding the position of boundaries (right and left lines) on the Ego-Lane. Then using the position selector, the longitudinal and lateral positions of the track relative to Ego-Car is computed. Similarly, the longitudinal and lateral velocities of the tracks are calculated. So, using these data, the nearest object to the Ego-Car in both longitudinal and lateral positions is considered as the Most Important Object (MIO) and the MIO index is set to the number of tracks received from that specific object.

Side-MIO: This tracking is done to find objects in the side lanes of the Ego-Car in order to define the possibility of lane change. The procedure to find Side-MIO is somehow similar to MIO finding, while the criteria is changed accordingly. In this function, the horizon to check for objects is 5 meters in-front and also 5 meters behind, and in the right or left lane of the Ego-Car. These side boundaries are described by defining the two nearest lane markings in the right and also left side of the Ego-Car. So, the boundary is defined as inside the lane on the right or left of the Ego-Lane. Consequently, objects that are detected in the right lane are considered as R-MIO and objects detected on the left lane are called L-MIO. The index for R-MIO and L-MIO are defined accordingly, when a track is available. Furthermore, another general index as Side-MIO index is defined as the maximum index between R-MIO index and L-MIO index. So, when there is only one object in the right or left side inside the horizon and defined boundaries, just one of the R-MIO or L-MIO is defined and Side-MIO gets the same index; However, in case of available tracks for both right and left side, the object which is closer to Ego-Car and has the most number of tracks, is defining the Side-MIO index. As these indexes are later used in LCA system to define the direction of lane change, the algorithm there will be checking all these three indexes at the same time, in order not to have conflict in decision making. The process for this decision making will be described in section 3.1.5.

3.1.1.2 LANE CENTER ESTIMATION

This is a pre-built block in Matlab that is consisted of two parts. The first part gets lane detections from sensors and based on the position of lane markings on right and left, and calculates the center of the lane. Then, provides "Curvature, Curvature Derivative, Relative Yaw Angle, and Lateral deviation" as output. In the second part, same curvature, curvature derivate, and longitudinal velocity are inputted in a calculation as figure 16 to find the curvature of the road which is later used by MPC controller. Gain "K" in this figure is equal to Prediction Horizon of MPC Controller, multiplied by simulation sample time. Which both will be defined in section 3.1.1.3.

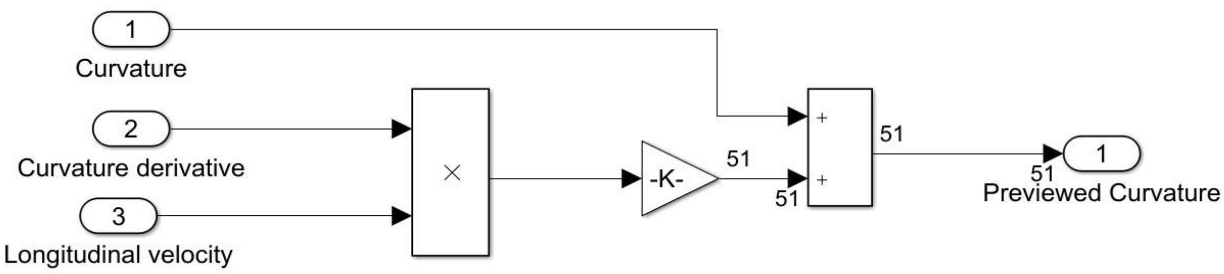


Figure 16. Calculation of Previewed Curvature

3.1.1.3 MPC CONTROLLER FOR ACC AND LKA [33]

MPC controller is the core of this control system, which receives the vehicle dynamics and road information, to adjust the Acceleration and Steering Angle accordingly and keep the Ego-Car in the best situation to maintain a balance of keeping desired speed and distance, while staying in the center of the lane. [33]

This is done by an Adaptive MPC which receives the signals from sensor fusion block, containing positioning and velocity data of the MIO. Moreover, it gets the signals as "Curvature, Lateral Deviation, and Relative Yaw Angle" from "Estimate Lane Center" Block. It also uses the pre-defined parameters as "time-gap, minimum and maximum steering" to perform the calculation of "Acceleration" for ACC and "Steering Angle" for LKA systems. For this purpose, a model for adaptive MPC is defined with the use of another function which uses the sample time (T_S), longitudinal velocity, time-gap, and initial velocity of Ego-Car as input parameters. [33]

Table 4. Parameters for MPC controller

* Simulation Sample Time (T_s) = 0.1 s	* ACC Time Gap = 1.5 s
ACC Default Spacing = 3.7 m	* ACC velocity error gain = 0.5
* ACC spacing error gain = 0.5	* ACC relative velocity gain = 0.4
* ACC maximum acceleration = 3 m/s^2	* ACC minimum acceleration = -3 m/s^2
LKA maximum steering angle = 0.08 rad	LKA minimum steering angle = -0.08 rad
* Prediction Horizon = 50	

From the values above, those marked with * are the same as in pre-built MPC controller in Matlab, however, other values are found during the implementation of this system, with trial and error to find the pest performance of the system.

By default, ACC system decides to switch between two modes, regarding the sensor fusion data. Hence, if the lead car is in a close distance, the ACC system switches from speed control to spacing control. Similarly, if the lead car is far away, the ACC system switches from spacing control to speed control. In other words, the ACC system makes the Ego-Car travel at a driver-set speed as long as it maintains a safe distance. As ACC system is using a configured version of pre-built ACC model in Matlab, in this model, safe distance between two cars is calculated by this formula: "$D_{safe} = D_{default} + T_{gap} \times V_{ego}$"; Where, $D_{default}$ is the ACC default spacing and T_{gap} is the time gap between the vehicles. [24]

ACC default spacing has a value of 15 meters for the default adaptive cruise controller system. However, as in this system, it is combined with lane change and emergency braking, safe distance is kept with regards to the time gap of 1.5 seconds, and just a default spacing of 3.7 m is applied in this formula in order to measure the relative distance from the very front of Ego-Car. As by default, the measurements in Automated driving toolbox are done from the center of mass of the vehicle, which for the simulations in this thesis is placed at 3.7 m from the front and 1.3 m from the back for a car that is 5 meters long. The Schematics for default ACC spacing and control as represented in figure 17.

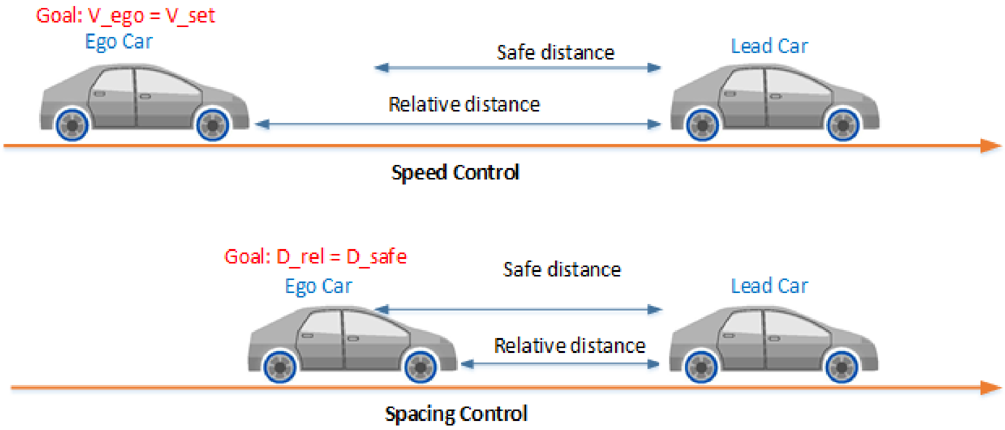


Figure 17. Safe Distance and Relative Distance definition in ACC [24]

LKA system also uses Adaptive MPC and the “Steering Angle” output of MPC is for lane keeping purpose. This system is also available as a pre-built system in Matlab, and by default, it gets the measurements of the lateral deviation and relative yaw angle between the centerline of a lane and the Ego-Car and depending on the curve length that the sensor can view, the curvature in front of the ego car can be calculated from the current curvature and curvature derivative. Using these measurements, LKA system keeps the ego car travelling along the centerline of the lanes on the road by adjusting the front steering angle of the ego car. The goal for lane keeping control is to drive both lateral deviation and relative yaw angle close to zero. [25]

Again in this system, the controller updates itself continuously to get the most updated data about the road and act accordingly. Obviously, this task gets more difficult on curved roads, with regards to the curve angle. [25]

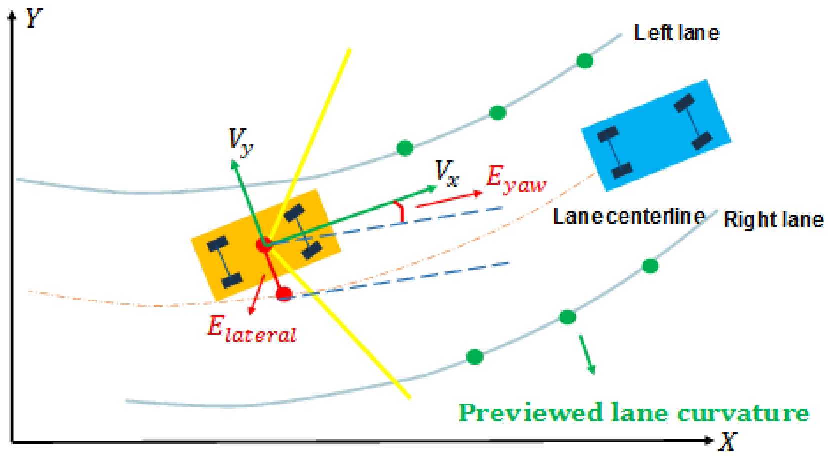


Figure 18. Schematic of Lane Curvature estimation for LKA [25]

3.1.1.4 IMPLEMENTATION OF LANE CHANGE ASSIST (LCA)

LCA system in this Test-Bench is designed as a one-way lane change (not overtaking) to perform a partial lane change to reach desired ACC speed or avoid collision in case of emergency. Accordingly, as default, this system is enabled when the Ego-Car is trying to increase speed and reach the ACC desired set velocity, but the Lead-Car is driving at a lower speed and so, the ACC system keeps the velocity equal to Lead-Car velocity and also maintains a safe distance with regards to the time gap of 1.5 seconds. However, when a critical situation arises and ACC is not capable of avoiding collision due to immediate reduction of relative distance or relative velocity between Ego-Car and Lead-Car, LCA is enabled to move to a left or right side lane, which would be free, to avoid collision and also reach desired velocity.

The algorithm to define the lane change direction is defined as in table 5:

Table 5. Lane Change Logic

Both side lanes free	Just left lane free	Just right lane free
Change lane to left	Change lane to left	Change lane to right

LCA is set to be activated when these three conditions are met simultaneously:

- Headway time is less than 1.5 seconds
- Set velocity for ACC is higher than the velocity of the lead-car
- There is enough space in one or both of the side lanes.

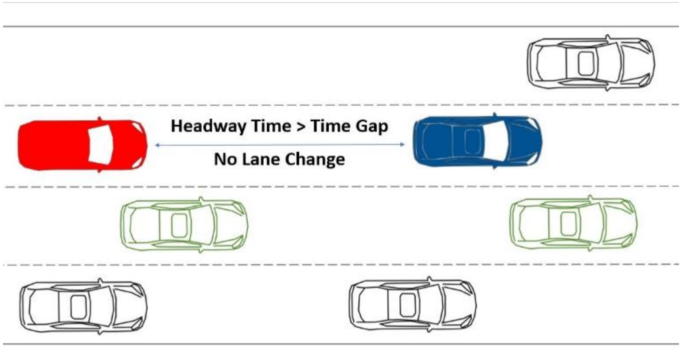
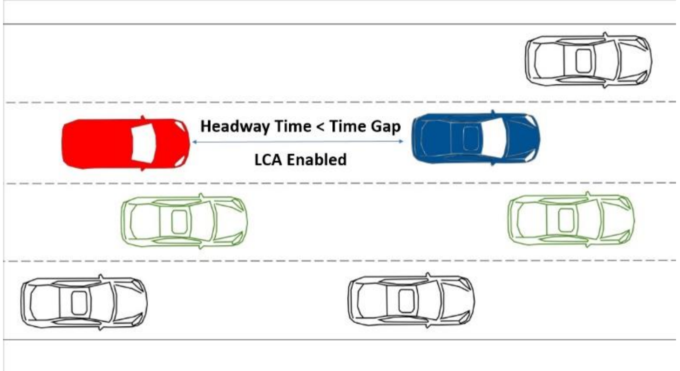
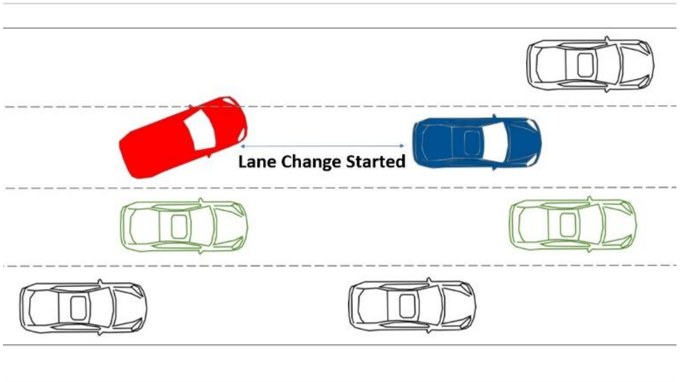
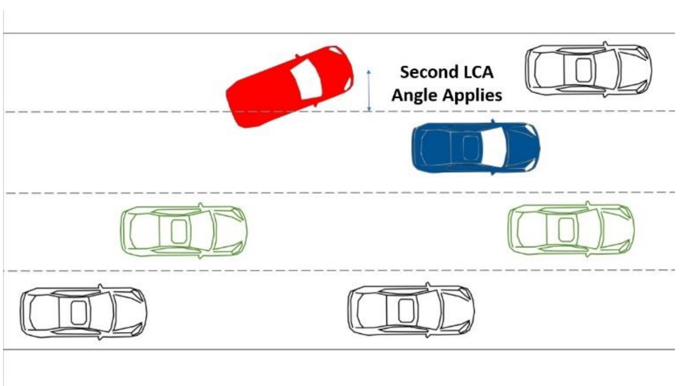
The priority is set so that if both of the side lanes are free, left lane will be selected for lane change, and right lane is used just in case the left lane is occupied. So, LCA performs the maneuver of changing the lane to meet ACC goal of set speed and also avoid collision. After the completion of LCA, it is disabled and the LKA system is enabled again to keep the Ego-Car in the new Lane. During LCA steering, acceleration is set to maximum (3 m/s^2) and after it is disabled, ACC takes the action for acceleration and controls it to meet the ACC and LKA goals.

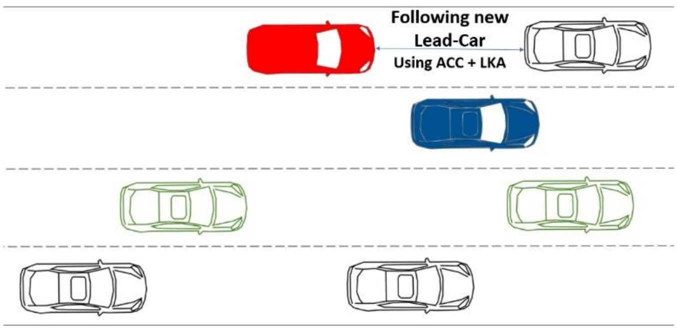
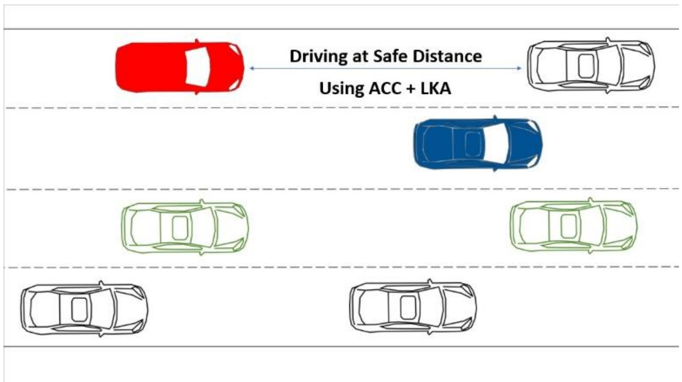
In order to accomplish the task of LCA, a very complicated algorithm is implemented defining every possible condition that may arise during a driving scenario. So, all the possible cases are described for roads from 1 to 4 lanes and according to the initial value of Ego-Lane where the Ego-Car is located at the beginning of the scenario. The functioning of LCA is an event-based process which is enabled when the conditions for a lane change as mentioned before, are met. Accordingly, when LCA is enabled, the steering angle switches from LKA to LCA, until the completion of lane change. Lane change starts with application of a very small steering angle (0.02 rad) as the first lane change steering angle. Obviously, in each time step, this value is added to the heading angle of Ego-Car, and when the conditions are met to confirm that Ego-Car is in the new lane, another steering angle with the value of heading angle at that moment and in the opposite direction to the first direction comes into account to decrease the heading angle to zero, so that Ego-Car can place itself in the new lane. When the lane change maneuver is done successfully and Ego-Car is moved to the new lane, LCA is disabled and steering switches to use LKA again to keep on the center of the new lane.

Clearly, all of the actions applied in this system are event-based and not time-based. For instance, the condition of being in the new lane is defined by checking the center of mass of the Ego-Car passing the markings of initial Ego-Lane. So, when this condition is met, the resulting action appears in the next time step. Even though this causes a potential delay in the control system, however, with the limitations in the user interface of Matlab & Simulink and Automated Driving Toolbox, this is one of the most feasible approaches to meet the goals of this Thesis. As mentioned before, in this system, lane change is completed at this stage and the rest of the scenario is considered to be handled with ACC or AEB which will be discussed in the simulation and results section.

In order to clarify the steps of decision making and performing in LCA system, Table 6 includes schematic representation of each step with some comments about the same step.

Table 6. Schematic behavior of Ego-Car for Performing a Lane Change

	- Ego-Car is initially in Lane 3 - There is enough safe space between two cars - Both cars travel at the same speed - ACC is keeping both speed and distance - LKA is keeping the car in lane - Headway Time is greater than Time Gap - No need to change lane in this step
	- Lead-Car brakes suddenly - Relative distance decreases immediately - Headway Time drops below Time Gap - LCA is enabled but no lane change started yet - Side-MIO is checked - Since left lane is free, lane change to left is activated for next step
	- Lane change to left is started with an initial steering angle - Car continues with this angle until it passes the left lane
	- Left lane is passed and the car is placed in new lane - 2nd Lca angle is applied to compensate Heading angle - LCA is deactivated - LKA is enabled againg to keep the car in lane

	- Heading angle is set back to zero and car travels in middle of the lane - New Lead-Car is detected - Relative distance is smaller than safe distance - Headway Time is smaller than Time Gap - ACC is activated to keep the safe distance and speed
	- ACC has performed its task of distance and velocity control - If the Lead-Car has velocity equal to or larger than V_set, Ego-Car will reach the V_set and continue at this speed - If the Lead-Car travels at a lower speed than V_set, Ego-Car will also drive at the same speed

3.1.1.5 IMPLEMENTATION OF AUTOMATIC EMERGENCY BRAKE (AEB)

AEB system is considered as the last priority in this Test-Bench to avoid collision, when ACC and LCA are not capable of avoiding collision. In this case, AEB [26] is activated and does the braking in three stages, while increasing the braking force in each stage to increase the distance to lead car or stop the car completely. Accordingly, it might happen that due to limited time, AEB would not be able to avoid collision too, but it will reduce the speed as much as possible, at least to mitigate injuries if collision occurs.

Two conditions are checked for enabling AEB:

1- Headway Time should be smaller than 0.5 s
2- Ego-Car should be using LKA as steering and LCA should be disabled

After Enabling the AEB, conditions for the TTC and Braking Time are calculated as below, and then checked as in table 7, to activate different levels of emergency brake.

$$\text{Headway Time} = \frac{Relative\ Distance}{V_{ego}} \qquad \text{Time to Collision (TTC)} = \frac{Relative\ Distance}{Relative\ Velocity}$$

$$\text{Forward Collision Warning time (FCWtime)} = \frac{V_{ego}}{acceleration}$$

$$\text{Partial Braking 1 time (PB1time)} = \frac{V_{ego}}{3.8} \qquad \text{Partial Braking 2 time (PB2time)} = \frac{V_{ego}}{5.3}$$

Full Braking time (FBtime) = $\frac{V_{ego}}{9.8}$

Table 7. AEB Steps Logic

Condition	AEB Status
[(abs(TTC)<FCWtime) && TTC<0]	0 = Ready for Emergency Braking
[(abs(TTC) < PB1time) && TTC<0]	1 = Partial Braking with 3.8 (m/s^2)
[(abs(TTC) < PB2time) && TTC<0]	2 = Partial Braking with 5.3 (m/s^2)
[(abs(TTC) < FBtime) && TTC<0]	3 = Full Braking with 9.8 (m/s^2)

These stages are also represented in figures 19 and 20.

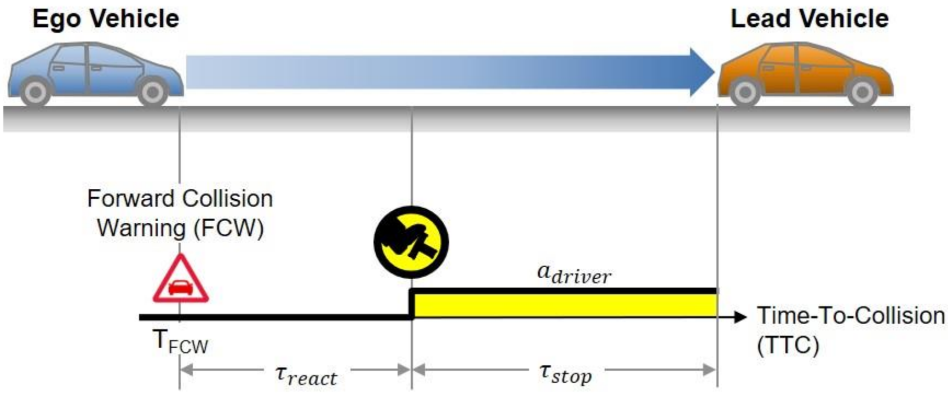


Figure 19. AEB Activation Logic [26]

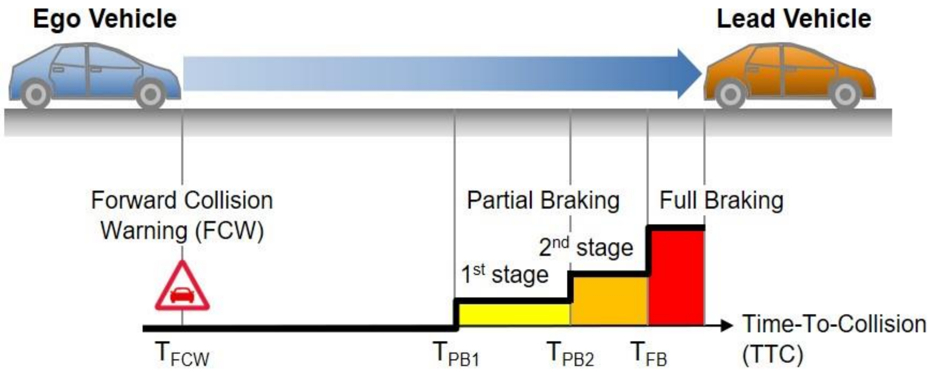


Figure 20. AEB Braking Steps [26]

3.1.2 VEHICLE AND ENVIRONMENT

Vehicle and Environment subsystem is used for closed-loop simulation of the controller. It consists of three main blocks as below, which will be defined with details in sections 3.1.2.1 to 3.1.2.3:

- **Vehicle Dynamics [28]:** includes the bicycle model and lower level dynamics to model Ego-Car
- **SAE J670E to ISO 8855:** converts the coordinates from Vehicle Dynamics, which uses SAE J670E [29], to Scenario Reader, which uses ISO 8855 [30].

- **Actors and Sensor Simulation:** includes the sensor blocks implemented on Ego-Car

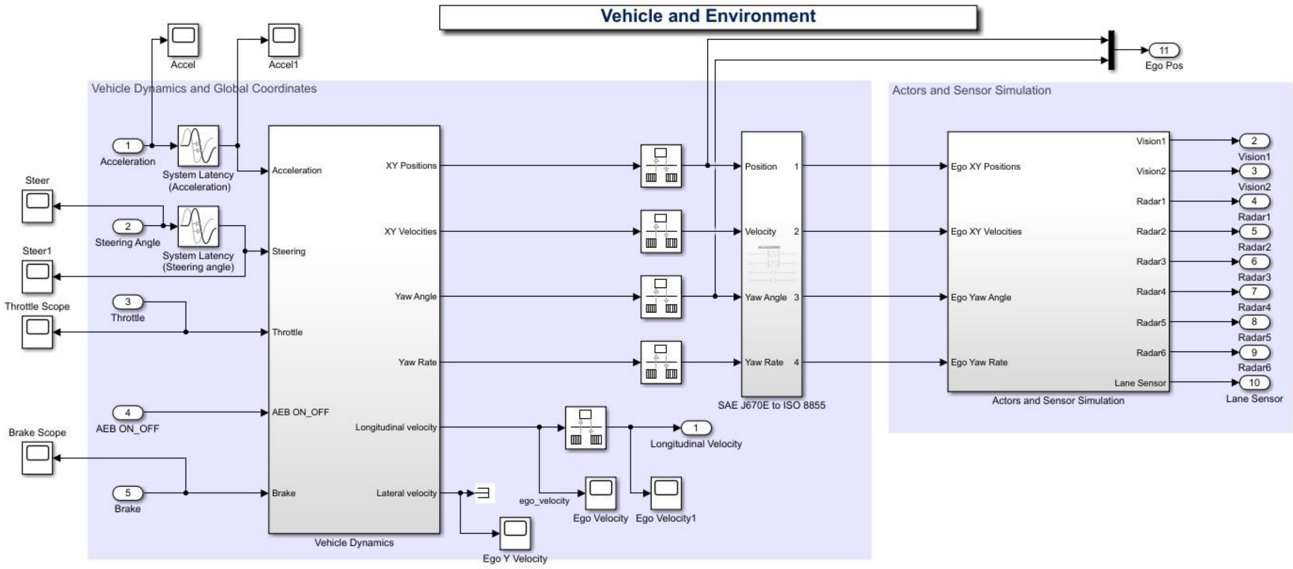


Figure 21. Inside Vehicle and Environment Block

3.1.2.1 VEHICLE DYNAMICS

This system's vehicle dynamic is modeled using a Bicycle Model - Force Input block from the Vehicle Dynamics Blockset™. This Blockset uses "Vehicle Body 3DOF" [31], which implements a rigid two-axle vehicle body model to calculate longitudinal, lateral and yaw motion. In this model, pitch, roll, and vertical motions of vehicle are not important. In order to simplify the model, single track configuration of Vehicle Body 3DOF is selected which has the following characteristics:

- Forces act along the center line at the front and rear axis
- No lateral load is transferred

Parameters for this Bicycle Model are configured below: [25]

- m = Total vehicle mass equal to 1575 (kg)
- Iz = Yaw moment of inertia of the vehicle equal to 2875 (mNs^2).
- lf = Longitudinal distance from the center of gravity to the front tires equal to 1.2 (m).
- lr = Longitudinal distance from center of gravity to the rear tires equal to 1.6 (m).
- Cf = Cornering stiffness of the front tires equal to 19000 (N/rad).
- Cr = Cornering stiffness of the rear tires equal to 33000 (N/rad).

3.1.2.2 SAE J670E AND ISO 8855 COORDINATE SYSTEMS

As the definition of relative placement of road and vehicle is necessary for the simulation purposes, it is important to define the coordinate systems in which the road and vehicle are placed. For this purpose, firstly one general coordinate system is needed, where the road and actors are defined with regards to it. Then, a local coordinate system for Ego-Car is needed, so that all the measurements are transformed to the Ego-Car coordinate system to have all the data relative to Ego-Car. With this definition, it is possible to translate all the road and other actors in the coordinate system of Ego-Car, so the control logics would be more feasible. On the other hand, this process in the opposite direction is also required, as the position and outputs from Vehicle Dynamics block are put into the Scenario Reader block inside Actors and Sensor Simulation block which will be introduced in section 3.1.2.3. This is done by the aid of a pre-built block in Matlab called **SAE J670E to ISO 8855**. This block converts the coordinates from Vehicle Dynamics to Scenario Reader.

3.1.2.3 ACTORS AND SENSOR SIMULATION

Inside this block, the pre-built Scenario Reader block and sensors are defined. Positioning of the sensors is very important to reach the maximum possible field of view, while keeping the functionality. As represented in figure 22, two vision detectors (Blue) are put in front and back. Also, two long range radars (Red) are placed in front and back of the car, plus four short range radars (Red) on the sides, plus a lane detector sensor, which altogether make a set of 9 sensors and cover a wide range around the car as below:

- Long-Range radars have the field of view for 174 meters long with opening angle of 20 degrees
- Short-Range radars have the field of view for 30 meters long with opening angle of 90 degrees
- Vision Detectors have a maximum range of 150 meters and detect objects up to 50 m/s
- Lane sensor detects the lane markings up to 150 meters in front

Figure 22 shows the placement of these sensors:

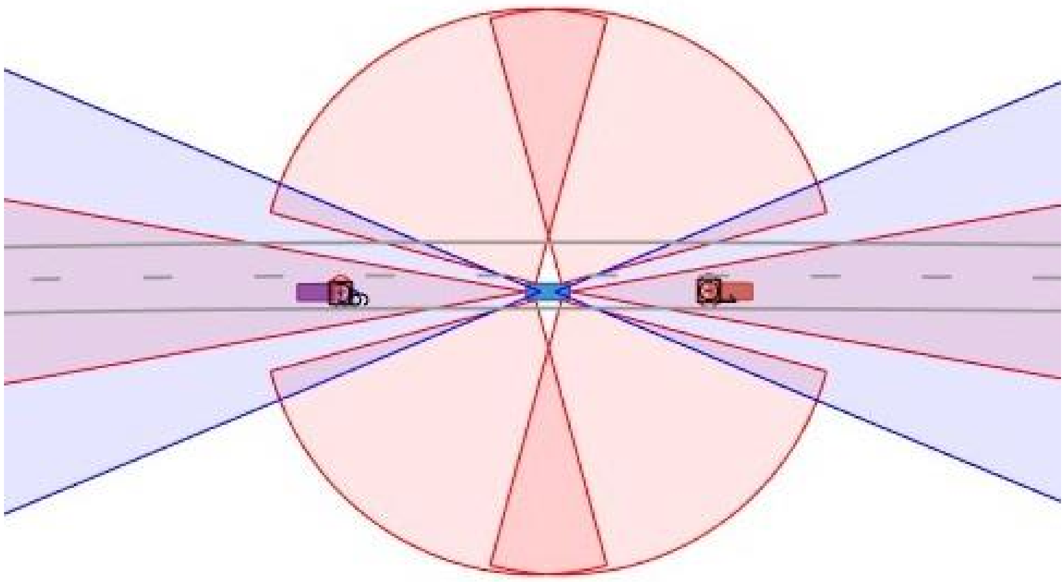


Figure 22. Sensor Placements of Ego-Car [29]

3.2 SELECTION OF DRIVING SCENARIOS

In order to generate Complex Scenarios, a methodology should be defined to choose the most critical cases in the potential infinite number of scenarios and the simulation tests would be done just for these scenarios. Subsequently, the results will be covering even the simpler scenarios not tested, as they have already been considered in the first steps of complex scenario generation and those selected test cases are inclusive of all the conditions that may arise in less critical situations.

Taking into account the aforementioned information in chapter 2, the focus in this thesis is just on Highway roads and some critical scenarios are being selected to test the performance of the designed Test-Bench. These scenarios are consisting of the variations in the following parameters, to be tested with different velocities and relative distances for Ego-Car, Lead-Car, and Side Cars.

- Number of Lanes
- Ego-Lane (The lane in which Ego-Car is placed initially)
- Traffic (Relative Position, Velocity and TTC of surrounding vehicles)

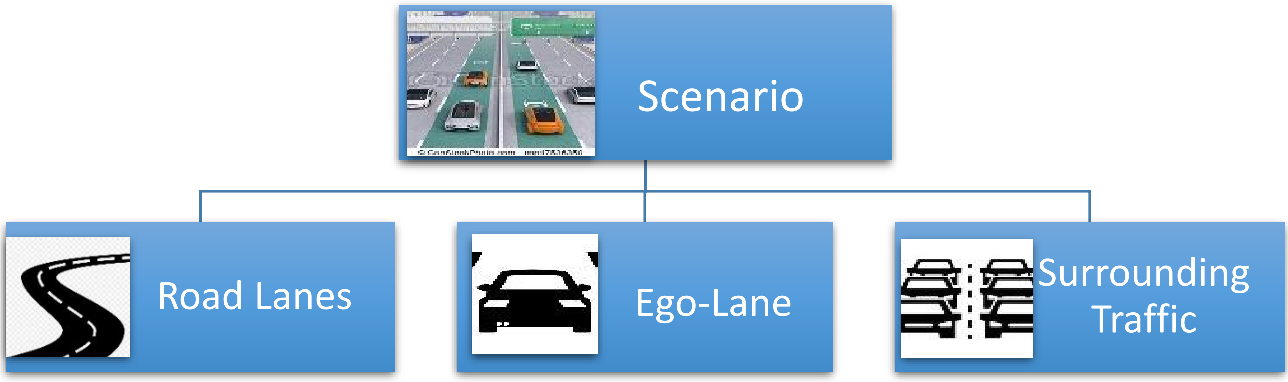


Figure 23. Constituting Elements of a Complex Scenario

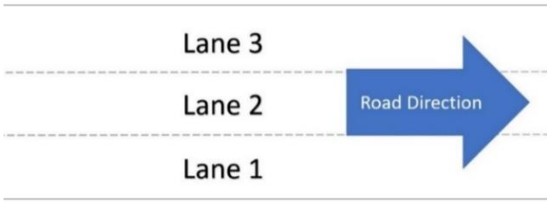
- **Number of Lanes**

Considering the Highways, there would be 2, 3 or 4 lanes as general. However, as the scenarios will be generated according to the position of Ego-Car and the surrounding traffic, it is derived that the right and left lanes in a two-lane road, will have the same characteristics for the Ego-Car performance as first and third lanes in a three-lane road and similarly, the first and fourth lanes in a four-lane Road. Moreover, the middle lane in a three-lane road is similar to the condition of second or third lane of a four-lane Road.

Consequently, to reduce the test cases, only four-lane roads will be considered in this research and obviously, the results will be applicable also to two and three-lane roads or some higher-lane roads similar in the characteristics.

Table 8 represent the numbering of lanes on 2, 3 and 4 lane highways in this thesis.

Table 8. Lane Numbering in Scenario Generation

	**2-Lane highway** - Right Lane = Lane 1 - Left Lane = Lane 2
	**3-Lane highway** - Right Lane = Lane 1 - Middle Lane = Lane 2 - Left Lane = Lane 3
	**4-Lane highway** - Right Lane = Lane 1 - 2nd Lane from Right = Lane 2 - 3rd Lane from Right = Lane 3 - Left Lane = Lane 4

- **Ego-Lane**

As all the scenarios will be tested according to the Ego-Car performance, it is essential to define the initial lane of Ego-Car in each scenario. So, all the other conditions such as relative position and velocity of other actors in the scenario are defined with regards to the Ego-Car. The initial lane of the Ego-Car will be called Ego-Lane.

- **Surrounding Traffic**

After the definition of road and lanes, the surrounding traffic known as other actors play an important role in the performance of the Ego-Car. The designed Test-Bench will evaluate the performance of Ego-Car in different complex scenarios, considering the Relative Position (RP), Relative Velocity (RV) and Time to Collision (TTC) to enable or disable collision avoidance systems to perform accordingly.

Considering all the aforementioned data, some complex scenarios could be derived as represented in the table 9, plus considering different Relative Distances (Δx) and Relative Longitudinal Velocities (Δv) between the Ego-Car and Lead-Car.

Scenarios are generated using the “Scenario Designer” app of Matlab’s Autonomous Driving Toolbox. First step is to define the road geometry and markings. Then the actors other than Ego-Car are created in this UI with their corresponding initial position, trajectory, and speed during the scenario runtime.

In the selection of scenarios, the most complexity is being considered to save test room space. So, when there would be similar scenarios, the most comprehensive case is going to be tested.

Test cases considered for this Thesis are just Highways; however, the Test-Bench provided would be still be functional to test some other specific traffic conditions like intersections inner cities.

According to the aforementioed information, in the following some scenarios are represented that include some of the most critical cases in highways.

These schematics are generated with regards to the capability of scenario generator in matlab. Since there are some limitations in this app such as setting the acceleration and exact positiong of other vehicles with regards to the Ego-Car.

So, the simulation of scenarios in Simulink will be done with regards to these limitations which are described under each set of figures.

The following scenarios include cases for two main criticalities. The first cases are when the lead-car decelerates suddenly, or the ACC of Ego-Car fails to keep the required distance, so the LCA is enabled and it should act properly with regards to the Ego-Lane and surrounding traffic conditions as below:

The lane numbers are in order of 1 to 4 from right to left regardingly.

- For Ego-Lane = 1
  - If the left lane is free, Ego-Car will change to left lane (2); otherwise, AEB will be enabled with no lane change
- For Ego-Lane = 4
  - If the right lane is free, Ego-Car will change to right lane (3); otherwise, AEB will be enabled with no lane change
- For Ego-Lane = 2 and 3
  - If just the left lane or both of the side lanes are free, Ego-Car will change to left lane
  - If only the right lane is free, Ego-Car will change to right
  - If both of the side lanes are occupied, AEB will be enabled with no lane change

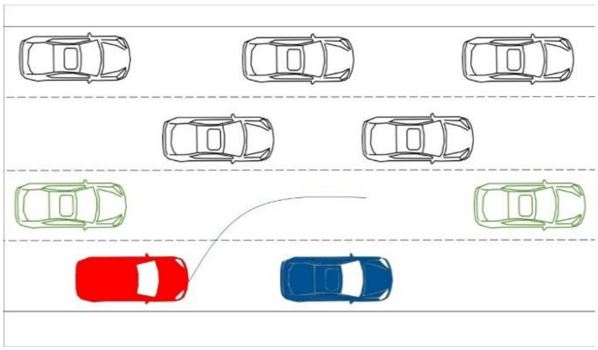
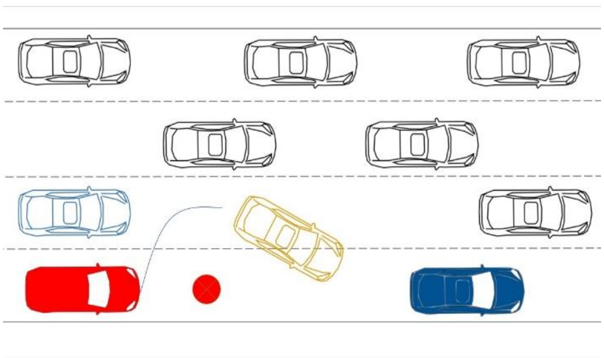
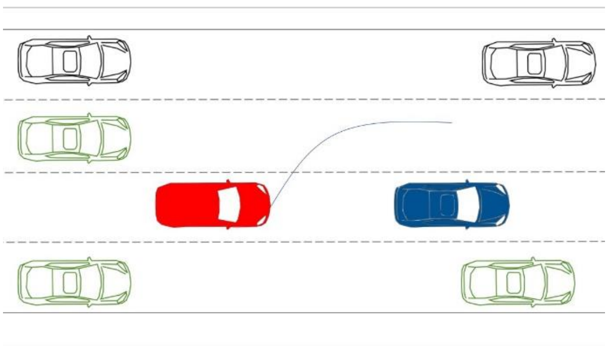
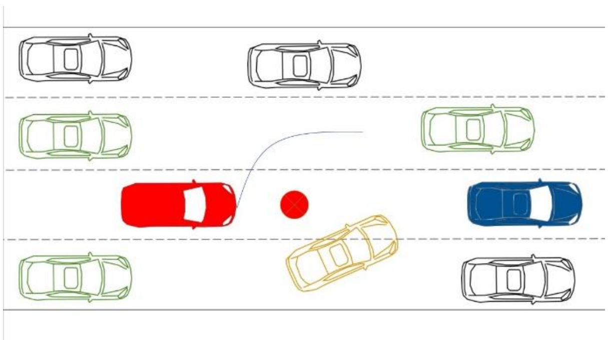
In the following, some most possible placements of Ego-Car and surrounding vehicles are represented. However, as there are scenarios that are very similar to each other, in the simulation section, just the scenarios of initial Ego-Lane=3 will be simulated with different conditions as will be represented in chapter 4.

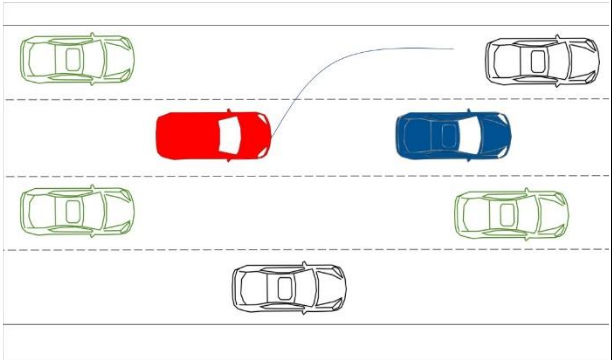
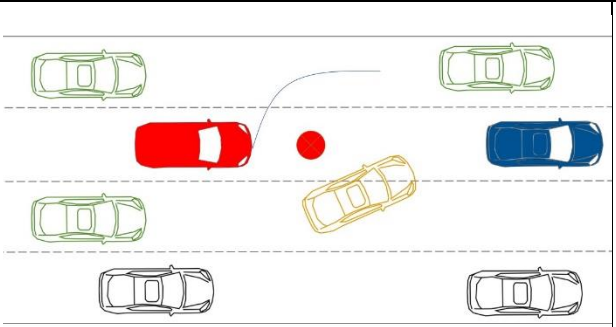
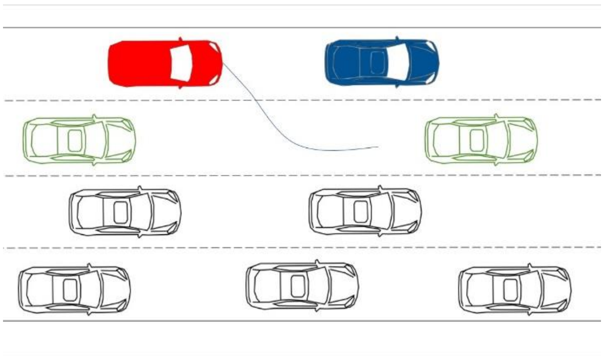
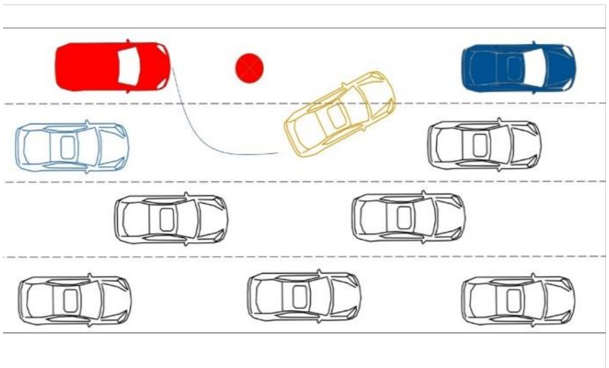
Also considering cases for initial Ego-Lane of 1 and 4, in these cases, there is no other lane in one side of the Ego-Vehicle. Thus, the LCA system should recognize being on these lanes in order not to do a lane change towards the no-road direction. So, if there is a free lane on the other side, LCA is enabled and performed. Otherwise, the situations should be handled just with ACC or AEB.

This task as achieved with the lane definition for Ego-Car. So, when the Ego-Car is recognized to be placed on these lanes, signals sent to LCA system define that in case of a lane change activation and possibility of a lane change, Ego-Car should just change to the side in which a valid lane is detected.

The aforementioned conditions are represented in the table 9.

Table 9. Description of Some Possible Scenarios on 4 Lane Highway Roads

Possible Scenarios for Ego-Lane = 1	
	- Ego-Car is initially in Lane 1 - Lead-Car brakes suddenly and the relative distance decreases; so LCA is enabled - If LCA system recognizes the possibility of lane change in the free spot in left, lane change is performed and scenario continues in lane 2 - If lane change is not possible, AEB is activated to perform the emergency braking and avoid collision
	- Ego-Car is initially in Lane 1 - Lead-Car is in distance; so LCA is not enabled - A side car intervenes into Ego-Lane and becomes the new Lead-Car in very close distance - If the blue side car would not be there, and there is enough time to change lane, LCA takes the action and changes to left lane - If lane change is not possible, AEB is activated to perform the emergency braking and avoid collision
Possible Scenarios for Ego-Lane = 2	
	- Ego-Car is initially in Lane 2 - Lead-Car brakes suddenly and the relative distance decreases; so LCA is enabled - If LCA system recognizes the possibility of lane change in the free spot in left, lane change is performed and scenario continues in lane 3 - If lane change is not possible, AEB is activated to perform the emergency braking and avoid collision
	- Ego-Car is initially in Lane 2 - Lead-Car is in distance; so LCA is not enabled - A side car intervenes into Ego-Lane and becomes the new Lead-Car in very close distance - If there is enough space and time to change lane, LCA takes the action and changes to left lane - If lane change is not possible, AEB is activated to perform the emergency braking and avoid collision

<table>
<tr><th colspan="2">Possible Scenarios for Ego-Lane = 3</th></tr>
<tr><td></td><td>- Ego-Car is initially in Lane 3
- Lead-Car brakes suddenly and the relative distance decreases; so LCA is enabled
- If LCA system recognizes the possibility of lane change in the free spot in left, lane change is performed and scenario continues in lane 4
- If lane change is not possible, AEB is activated to perform the emergency braking and avoid collision</td></tr>
<tr><td></td><td>- Ego-Car is initially in Lane 3
- Lead-Car is in distance; so LCA is not enabled
- A side car intervenes into Ego-Lane and becomes the new Lead-Car in very close distance
- If the blue side car would not be there, and there is enough time to change lane, LCA takes the action and changes to left lane
- If lane change is not possible, AEB is activated to perform the emergency braking and avoid collision</td></tr>
<tr><th colspan="2">Possible Scenarios for Ego-Lane = 4</th></tr>
<tr><td></td><td>- Ego-Car is initially in Lane 4
- Lead-Car brakes suddenly and the relative distance decreases; so LCA is enabled
- If LCA system recognizes the possibility of lane change in the free spot in right*, lane change is performed and scenario continues in lane 3
- If lane change is not possible, AEB is activated to perform the emergency braking and avoid collision</td></tr>
<tr><td></td><td>- Ego-Car is initially in Lane 4
- Lead-Car is in distance; so LCA is not enabled
- A side car intervenes into Ego-Lane and becomes the new Lead-Car in very close distance
- If the blue side car would not be there, and there is enough time to change lane, LCA takes the action and changes to right lane*
- If lane change is not possible, AEB is activated to perform the emergency braking and avoid collision</td></tr>
</table>

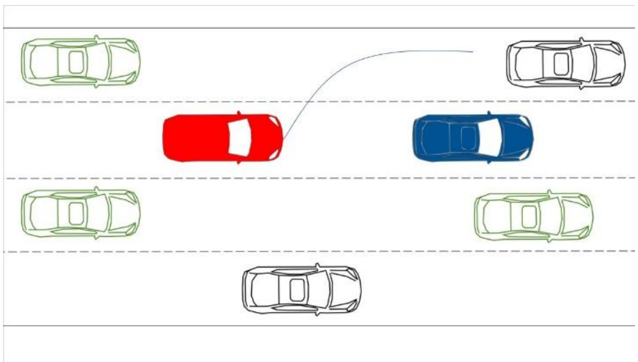
*Lane change to right will not be considered in this Thesis and for scenarios starting on Lane 4, just the AEB system will be enabled in case of emergency.

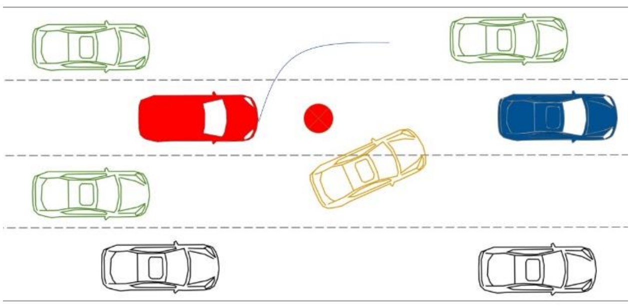
4. SIMULATION AND RESULTS

In this chapter, considering the capabilities of implemented Test-Bench a selected complex scenario with initial Ego-Lane=3 will be simulated as the most comprehensive case, to evaluate the performance of Ego-Car and defining safe zones for "Δv" and "Δx" between Ego-Car to Lead-Car in which collision is avoided. The system is designed to perform one of the collision avoidance tasks according to scenario situations:

4.1 OVERVIEW OF SELECTED DRIVING SCENARIO FOR SIMULATION

Table 10. Selected Scenarios for Simulation with initial Ego-Lane=3





Scenario description:

- Ego-Car is initially in lane 3

Lead-Car brakes (Or Similiarly, a car from side lane intervenes to Ego-Lane) suddenly and so, the relative distance between Ego-Car and Lead Car decreases suddenly

Desired control tasks by priority:

1. ACC tries to decelarate until it reaches the velocity of Lead-Car to keep safe distance again
2. If ACC succeeds to keep distance and set the speed, collision is avoided, and then, the system checks the possibility of lane change with regards to the free space on the left and headway time of Ego-Car
3. If there is a possibility of lane change and headway time is in the range of LCA threshold (Table 11), lane change is performed
4. After the lane change is done and Ego-Car is on the 4^{th} lane, LCA is disabled
5. Ego-Car uses ACC and LKA to continue in the middle of new lane and also avoid collision in new lane
6. In case that there would be another car in the new lane, ACC will do the task of setting speed and distance again. However, lane change will not be done again in the 4^{th} lane
7. If the relative distance is too small that ACC can not handle it and Headway Time is less than 0.5 second, AEB will be activated to avoid collision

This Scenario will be simulated with the initial speed of 100 km/h for Ego-Car and initial speeds of 80, 90, 100, 110 and 120 km/h for Lead-Car in each simulation. From the side cars, the ones in the 4th lane are important as for lane change, their positioning will be checked and if lane change is done, the faster car in the 4th lane will be in front the Ego-Car after lane change is complete. Accordingly, the speed of the fast driving car in the 4th lane is set to 130 km/h, and it is located 10 meters in front of the Ego-Car and in the left lane. So, for a few seconds at the beginning of the scenario, the left lane is seen as occupied for a few seconds. Then as the initial speed of Ego-Car is 100 km/h (30 km/h slower than the Side-Car), the Side-Car passes forward and the left lane becomes empty for lane change. In this step, depending on the initial **Δx** to Lead-Car and Headway Time, Ego-Car should decide to whether handle the situation with ACC and LKA, or activate LCA or alternatively switch to AEB. Desired V_set for Ego-Car is 130 km/h in all the test cases. So, the AC will try to increase speed from 100 to 130 km/h, if there would be a possibility of it according to scenario situation.

Due to limitations in the UI of Scenario Designer, it is not possible to modify the acceleration and velocity change for Lead-Car and Side-Car. So, the velocity of actors other than Ego-Car are all constant during a scenario. Therefore, in order to be able to simulate the real situations, the variations in the Lead-Car initial velocity for each scenario give the opportunity to have different **Δv**s and simulate situations that would be considered as the Lead-Car is decelerating suddenly and so, the **Δv** and **Δx** between it and Ego-Car decreases. Accordingly, the scenario is tested with different initial relative distances (**Δx**) to Lead-Car, equal to 50, 40, 30, 20, and 10 meters, to check in which Δx the criticality grows to its maximum.

Considering the above-mentioned scenarios, criticality increases when the lead-car has a slower speed than Ego-Car and the side cars behind the Ego-Car accelerate. So, if the side cars on the left lane would have enough space in between and the velocity and acceleration of Ego-Car is enough to locate itself in the free space, it will be the best performance of Ego-Car. Otherwise if the Ego-Car starts to change lane but the Side-Car coming from behind accelerates suddenly or the Side-Car in front decelerates suddenly, Ego-Car will cancel the lane change at that point and change to ACC+LKA again to get back in the previous lane. Obviously if there would not be enough distance to the Lead-Car again, AEB should be activated immediately to avoid the collision.

In order to have a better understanding of the system performance, these scenarios have been tested with two feasible configurations as below:

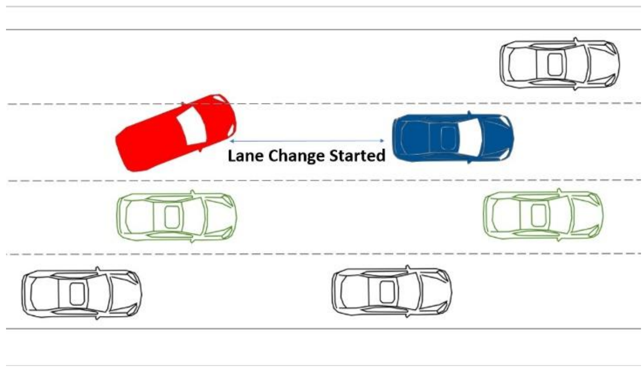
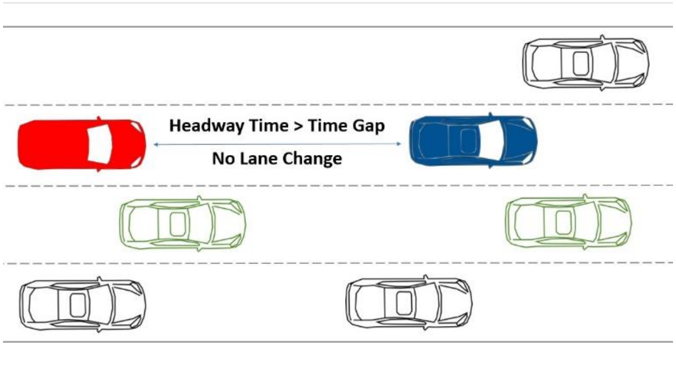
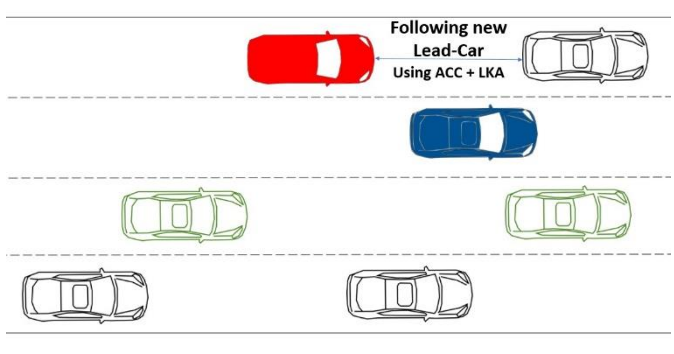
Table 11. Configurations of thresholds for testing system performance

	Time Gap	LCA Activation Range	AEB Activation Range
Configuration 1	1.2 s	0.5 s < HT < 1.2 s	HT < 0.5 s
Configuration 2	1.5 s	0.5 s < HT < 1.5 s	HT < 0.5 s

4.2 SIMULATION RESULTS

As the desired goal of each scenario is to avoid any collision, and there is possibility of at least 2 main collisions to avoid in each scenario (one in lane 3 and one in lane 4); Accordingly, two phrases as “Collision 1 Avoided” and “Collision 2 Avoided” are used in tables 13 and 14 for result presentation. Schematics in table 12 represent the cases for each collision avoidance concept.

Table 12. Collision Avoidance Concepts in this Thesis

<table>
<tr><td colspan="2">Initially, Ego-Car is in lane 3 and the Lead-Car in lane 3 is considered as Lead-Car 1
So, if the collision with this Lead-Car is avoided, it will be called “Collision 1 Avoided”.
Obviously there are three possibilities to avoid this collision, by using ACC, LCA, or AEB. And whichever of the functions would be able to avoid the collision, it will be considered as a successful case for Collision Avoidance 1.</td></tr>
<tr><td>Collision 1 avoided
by LCA in lane 3</td><td>
</td></tr>
<tr><td>Collision 1 avoided
by ACC or AEB in lane 3</td><td>
</td></tr>
<tr><td colspan="2">In Cases that lane change is performed and the first collision is avoided, Ego-Car finds it behind another Lead-Car in lane 4 which is called Lead-Car 2. For this case, collision should be avoided by ACC or AEB, according to the headway time.</td></tr>
<tr><td>Collision 2 avoided
by ACC or AEB in lane 4</td><td>
</td></tr>
</table>

Tables 13 and 14 represent the simulation results for configuration 1 and 2:

Table 13. Simulation Results for Configuration 1

Configuration 1											
Time Gap = 1.2 s					LCA active for 0.5 < HT < 1.2			AEB active for HT < 0.5			
Initial Δx to = 50 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	√	√	---	---	34	1.37	Yes	--	--	--
	90	√	√	X	---	30	0.88	Yes	--	--	No
	100	√	√	X	---	32	0.86	Yes	--	--	No
	110	√	√	√	---	39	1	Yes	27.5	0.6	Yes
	120	√	√	---	---	50	1.32	Yes	--	--	--
Initial Δx to = 40 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	√	√	√	---	19	0.5	Yes	39.8	1	Yes
	90	√	√	√	---	26.2	0.72	Yes	29.8	0.78	Yes
	100	√	√	√	---	34.4	1.12	Yes	32.4	0.84	Yes
	110	√	√	√	---	37.4	1.03	Yes	23.4	0.57	Yes
	120	√	√	√	---	44	1.2	Yes	21.7	0.49	Yes
Initial Δx to = 30 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	√	√	√	---	8.7	0.16	Yes	29.7	0.78	Yes
	90	√	√	√	---	16	0.4	Yes	26	0.68	Yes
	100	√	√	√	---	19	0.45	Yes	29.5	0.77	Yes
	110	√	√	√	---	25.6	0.7	Yes	28.4	0.74	Yes
	120	√	√	√	---	25.2	0.69	Yes	28.1	0.72	Yes
Initial Δx to = 20 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	√	√	---	√	9.5	0.29	Yes	---	---	---
	90	√	√	---	√	12.5	0.36	Yes	---	---	---
	100	√	√	X	---	14.3	0.38	Yes	---	---	No
	110	√	√	X	---	14.6	0.39	Yes	---	---	No
	120	√	√	√	---	14.5	0.38	Yes	43.7	1.1	Yes
Initial Δx to = 10 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	---	---	---	X	---	---	No	---	---	---
	90	---	---	---	X	---	---	No	---	---	---
	100	√	√	---	√	4.5	0.03	Yes	---	---	---
	110	√	√	---	√	4.7	0.03	Yes	---	---	---
	120	√	√	---	√	5	0.05	Yes	---	---	---

√: Acceptable Performance   X: Not Acceptable Performance   ---: Not Used / Not Applicable

Table 14. Simulation Results for Configuration 2

Configuration 2											
Time Gap = 1.5 s					LCA active for 0.5 < HT < 1.5				AEB active for HT < 0.5		
Initial Δx to = 50 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	✓	✓	✓	---	32.8	1	Yes	26.2	0.72	Yes
	90	✓	✓	✓	---	39.4	1.2	Yes	26	0.72	Yes
	100	✓	✓	✓	---	42	1.21	Yes	25.4	0.64	Yes
	110	✓	✓	✓	---	48.4	1.4	Yes	20.9	0.47	Yes
	120	✓	✓	✓	---	47.5	1.5	Yes	21.6	0.51	Yes
Initial Δx to = 40 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	✓	✓	✓	---	18.9	0.5	Yes	26	0.72	Yes
	90	✓	✓	✓	---	26	0.72	Yes	29	0.75	Yes
	100	✓	✓	✓	---	33.8	0.98	Yes	28.4	0.74	Yes
	110	✓	✓	✓	---	35	1.12	Yes	27.4	0.72	Yes
	120	✓	✓	✓	---	47	1.38	Yes	27.5	0.7	Yes
Initial Δx to = 30 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	✓	✓	✓	---	8.6	0.16	Yes	25.7	0.7	Yes
	90	✓	✓	✓	---	16.4	0.41	Yes	27	0.72	Yes
	100	✓	✓	✓	---	16.3	0.41	Yes	27	0.72	Yes
	110	✓	✓	✓	---	25.9	0.71	Yes	29.3	0.77	Yes
	120	✓	✓	✓	---	25.8	0.71	Yes	27.2	0.72	Yes
Initial Δx to = 20 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	✓	✓	---	✓	9.5	0.29	Yes	---	---	No
	90	✓	✓	---	✓	12.5	0.36	Yes	---	---	No
	100	✓	✓	✗	---	14.6	0.39	Yes	---	---	---
	110	✓	✓	✓	---	14.4	0.38	Yes	42	1.3	Yes
	120	✓	✓	✓	---	14.4	0.38	Yes	45.4	1.3	Yes
Initial Δx to = 10 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	---	---	---	✗	---	---	No	---	---	---
	90	---	---	---	✗	---	---	No	---	---	---
	100	✓	✓	---	✓	4.4	0.027	Yes	---	---	---
	110	✓	✓	---	✓	4.8	0.048	Yes	---	---	---
	120	✓	✓	---	✓	5	0.051	Yes	---	---	---

✓: Acceptable Performance ✗: Not Acceptable Performance ---: Not Used / Not Applicable

4.3 DIFFERENT CASES OF SIMULATION RESULTS

Considering the simulations with configuration 1 and 2, there are different set of results with regards to the scenario elements. Each case is represented with the corresponding plots in the following:

4.3.1 CASE 1) NO LANE CHANGE (CONFIGURATION 1)

As represented in tables 13 and 14 in cases that there is a large initial distance of 50 meters between Ego-Car and Lead-Car at the beginning of scenario, then it symbolizes sudden braking of Lead-Car and Δx decreases suddenly; so ACC handles the situation and increases Δx. As the Headway Time is larger than threshold, LCA is not enabled and Ego-Car continues the scenario, following the Lead-Car.

Figure 24 represents the moment that ACC is decelerating to set the Ego-Car speed equal to Lead-Car and keep distance.

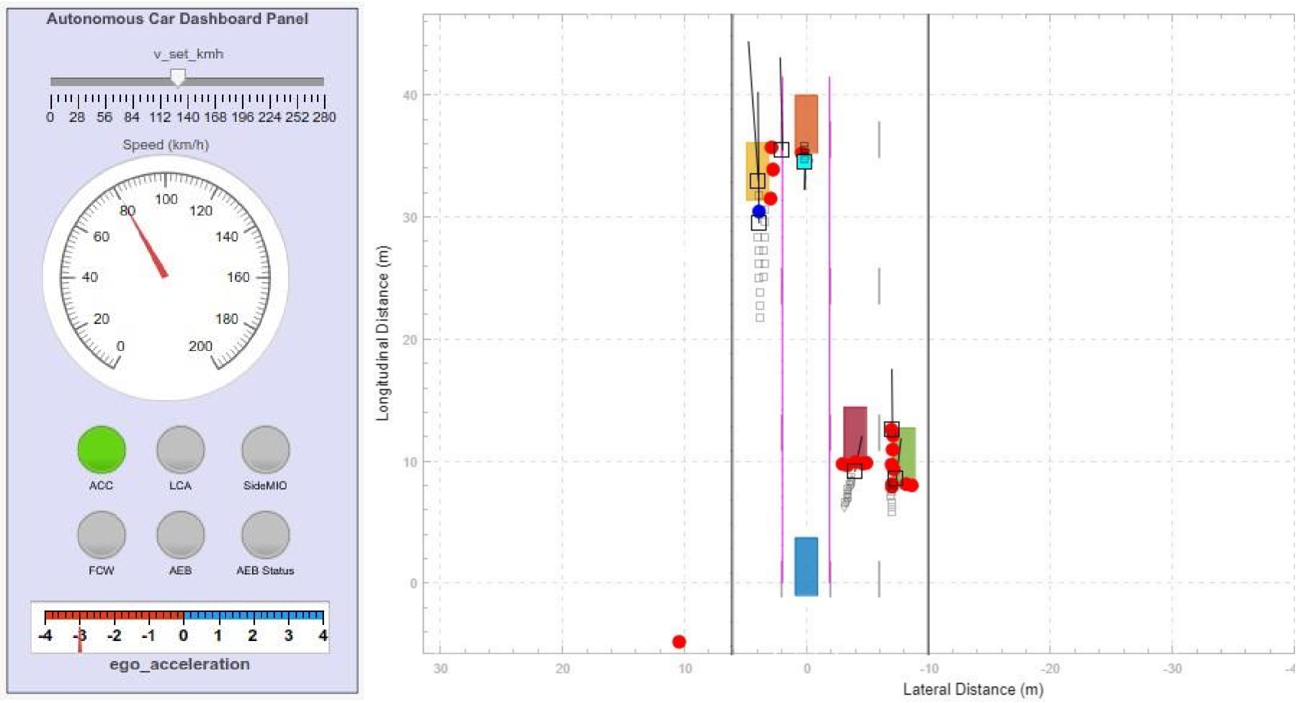


Figure 24. ACC system while maintaining safe distance and desired speed

Regarding the cars represented in figue 24, the blue car is Ego-Car, the orange on is Lead-Car, while the yellow one on the 4th lane is the fast driving car which starts at the beginning of the simulation with a Δx of 10 meters relative to Ego-Car and as its initial velocity is 130 km/h, it passes the Ego-Car a few seconds later. The importance of this behavior for this car will be represented in the following sections as in cases that Ego-Car aims to change lane, this car plays an important role in timing of the lane change, and also plays the role of second Lead-Car when the lane change is complete and Ego-Car finds itself behind this car.

As represented, this task is done very well and the dashboard on the left side represents the speed of 80 km/h for Ego-Car while ACC is ON and the acceleration is equal to -3 m/s^2. After this moment, the acceleration fluctuates a bit and then finalizes around zero as the goal of controller is met.

For these scenarios, useful plots to look into would be the Δx, Headway Time, TTC, Speed, Acceleration, and Steering. As the most critical case is when initial speed of Lead-Car is the lowest, plots in the following are for the case of "V0_Lead = 80km/h ", and "initial Δx = 50 m".

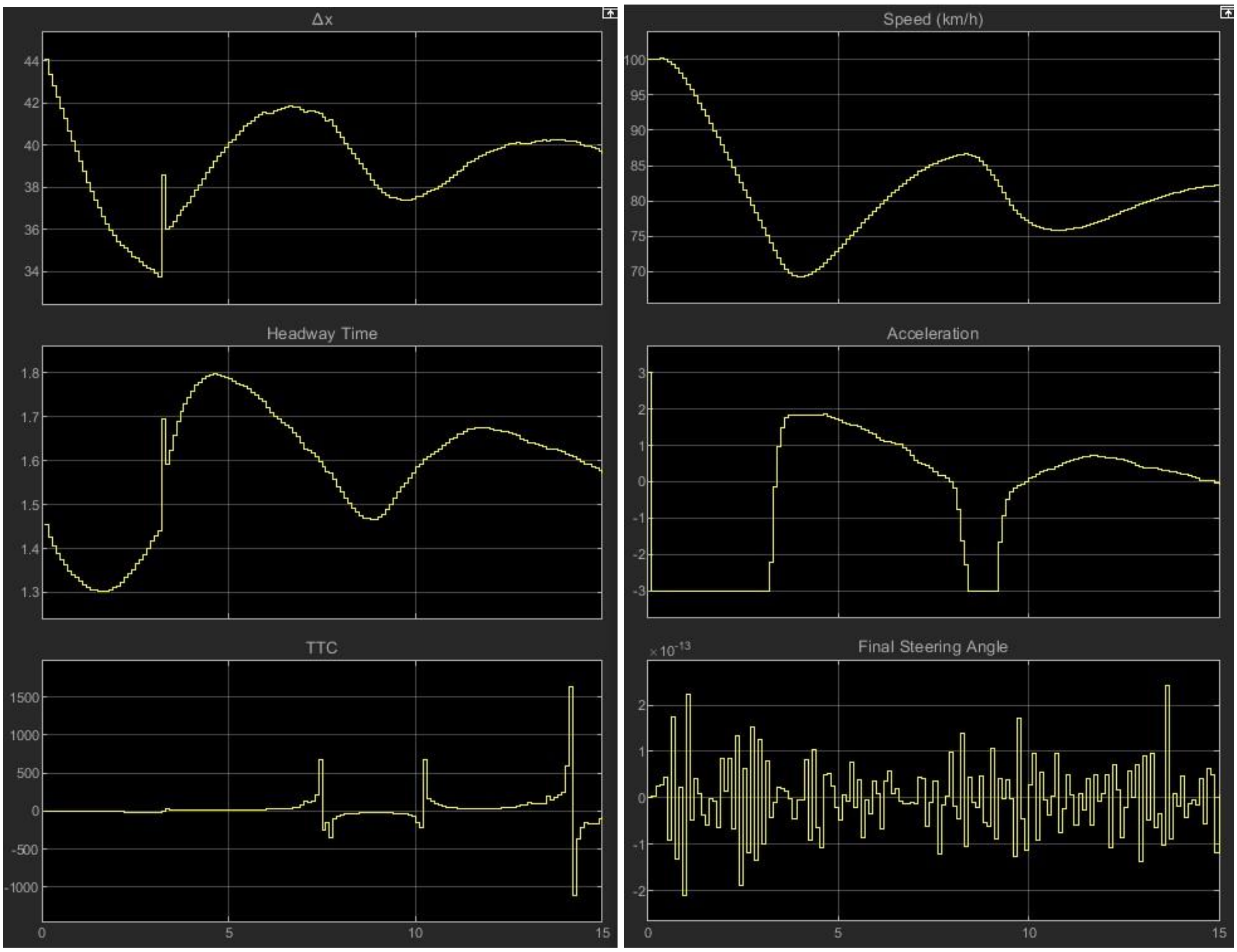


Figure 25. Simulation Plots for Case 1: No Lane Change (configuration 1)

As seen in figure 25, Δx decreases until 34 meters, and the ACC manages to keep distance and increases it to around 42 meters, then reaching a final level around 40 meters. Meanwhile, Headway Time has a similar characteristic as Δx and decreases to the level of 1.3 seconds, but then increases with regards to the increasing Δx and finalizes around 1.6 seconds. Obviously LCA is not enabled as its trigger is set to 1.2 seconds in this configuration. Moreover, TTC represents a relevant characteristic with regards to Δx and Headway Time.

On the other hand, speed of Ego-Car represents a slope in decreasing up to 70 km/h as the Lead-Car is in nearer distance than supposed for ACC. However, after maintaining the safe distance, velocity increases up to 85 km/h and then normalizes itself with the 80 km/h speed of Lead-Car. Clearly, the task of changing speed is done with the aid of acceleration control done by adaptive MPC. This is why the acceleration fluctuates from 3 to -3 and then normalizes around 0 as the desired speed and distance is set.

In the meantime, the heading angle of Ego-Car is kept very close to zero, with decimal fluctuations to keep the lane. This is represented in the last section of the charts as “Final Steering Angle”. As shown, the range of change in steering angle is very small ($\pm 2 \times 10^{-13}\ rad$), not felt during the drive and considered as zero steering angle.

Two peaks in the plots of Δx and Headway time are due to simulation bugs in Simulink, in which at the time of moving from initial lane to new lane, there is a moment that no Lead-Car is tracked due to Ego-Car direction, and this causes a peak to appear on the plots. However, this has no negative effect on the real performance and results of simulation.

4.3.2 CASE 2) SUCCESSFUL LANE CHANGE (CONFIGURATION 2)

This case meets the most desired performance of Ego-Car when a critical situation arises; as it uses the maximum capability of the implemented system. According to the simulations and above-mentioned tables, this usually happens in the best way, when the initial Δx is equal to 30 or 40 meters; although some limited cases with Δx=20 meters are also representing a good performance of the Ego-Car. This case happens in the most critical situation, when using configuration 2 with initial Δx=30 meters and initial Lead-Car speed of 80 km/h. This criticality is measured with the least relative Δx to the Lead-Car while doing the lane change. Accordingly, in this case, Ego-Car Starts the lane change at Δx=…., but as doing the LCA, since it is also accelerating, Δx decreases to the minimum value of 8.6 meters which is a very small distance at this speed. However, since ACC and AEB are disabled when doing a lane change, and as Ego-Car recognizes this lane change to be done safely, it is performed and the task is completed.

Related plots to this simulation are represented and discussed in the following:

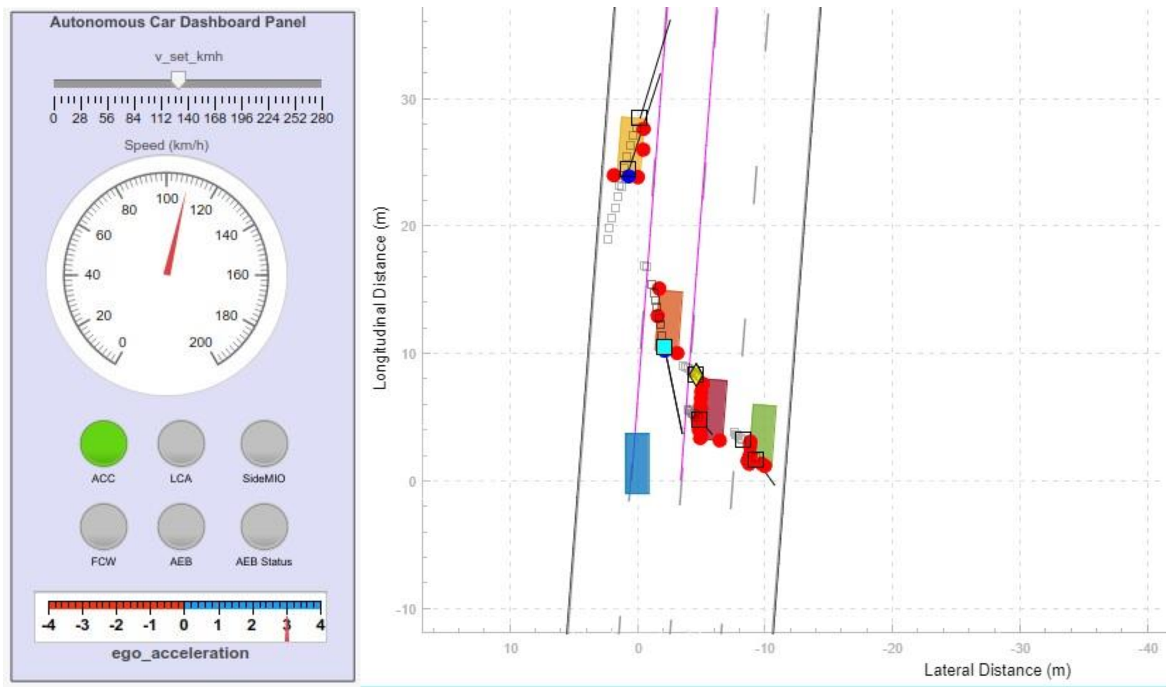


Figure 26. Case 2 - Scene 1: Ego-Car performing the lane change

Figure 26 represents the moment that lane change is being performed and relative Δx is the minimum. As shown on the dashboard, acceleration of Ego-Car at this point is equal to 3 m/s^2 and it tries to reach the set velocity of 130 km/h while completing the lane change action.

In the next scene, lane change is completed and Ego-Car is placed in the new lane. At this step, as the relative distance is smaller than default for ACC and accordingly, Headway Time is smaller than time gap of 1.5 seconds, acceleration is set to -3 m/s^2 to decrease the speed until required distance is satisfied. This causes fluctuations in the speed of Ego-Car, which will be represented in the following plots.

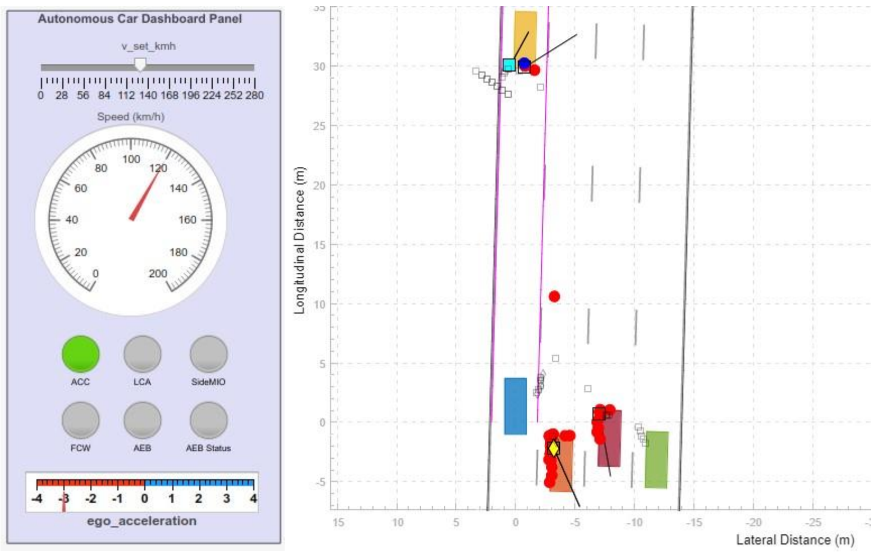


Figure 27. Case2 - Scene 2: Ego-Car right after performing the lane change and finding the new Lead-Car

Finally, as the required time gap is reached and also the velocity of Ego-Car is equal to the 130 km/h speed of new Lead-Car on the 4th lane, acceleration is set to zero, so that the Ego-Car keeps its speed as long as the Lead-Car is also keeping this speed without braking.

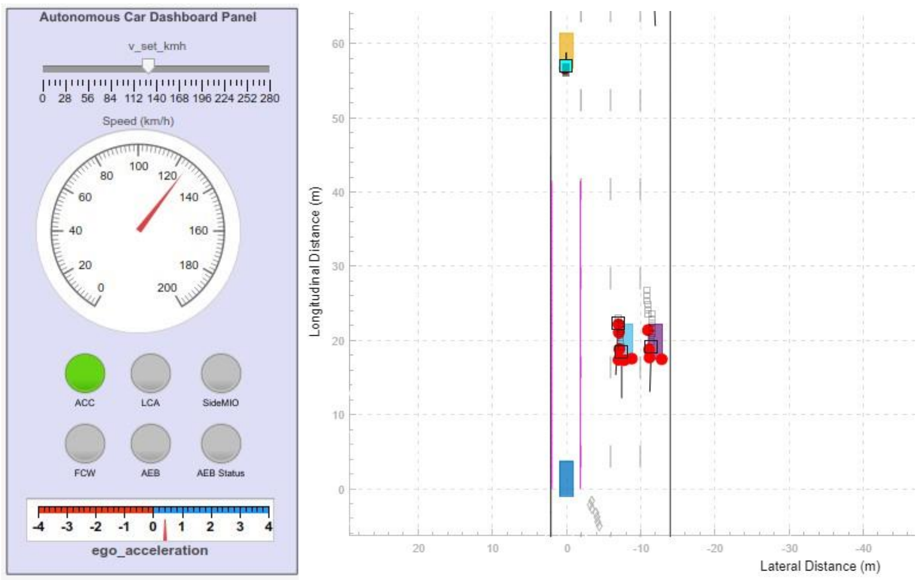


Figure 28. Case 3-Scene 3: Final positioning of Ego-Car with regards to new Lead-Car on the 4th lane

Plots in Figure 29 also represent the behavior of Ego-Car in this scenario:

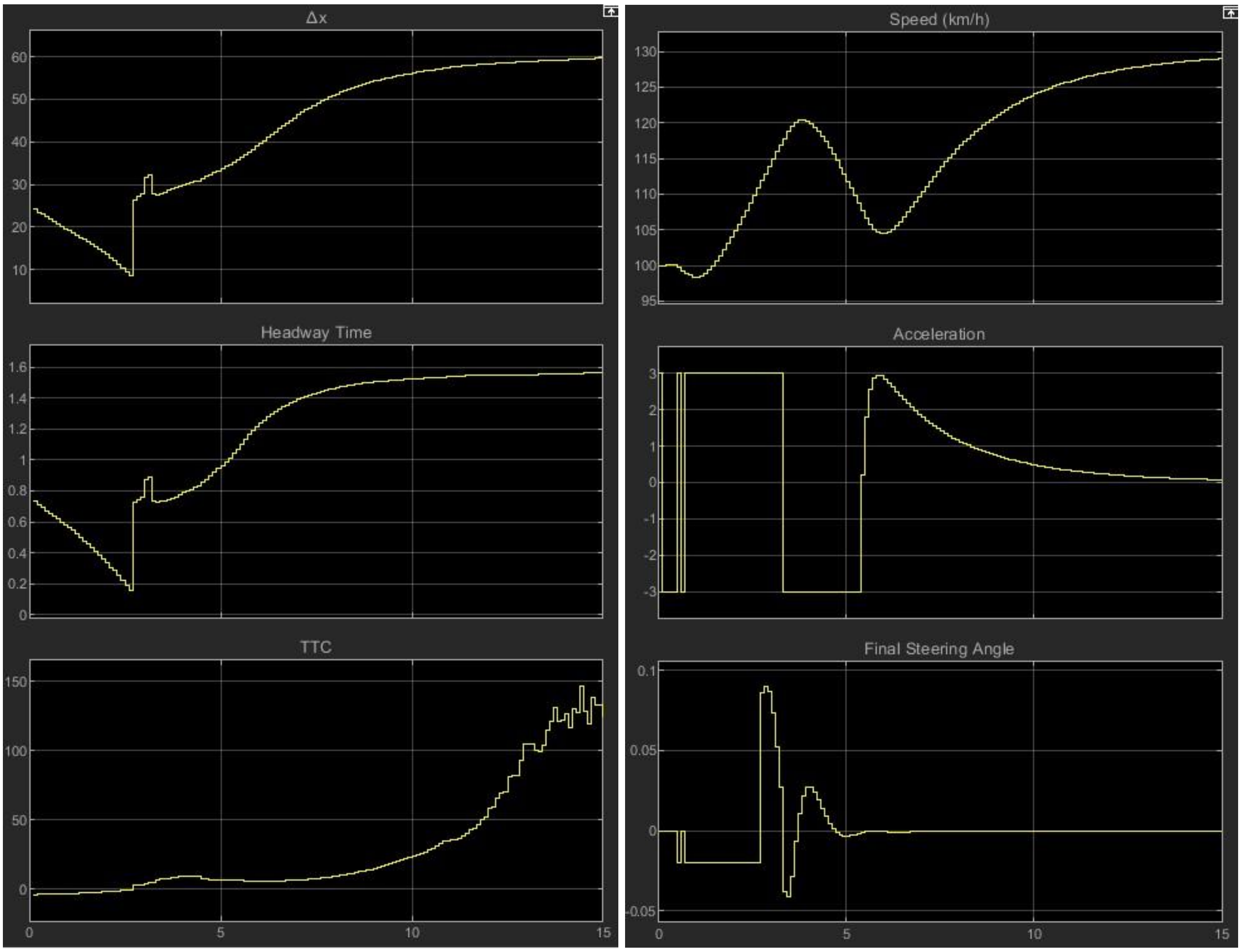


Figure 29. Simulation Plots for Case 2: Successful Lane Change (Configuration 2)

At the beginning of scenario, Headway Time is around 0.8 seconds and thus LCA is activated. Consequently, steering angle is applied to start the lane change. While performing the lane change, there is a decrease in relative distance until 8.6 meters with the corresponding Headway Time of 0.16 seconds. Obviously this is a really critical case and in case of another braking of Lead-Car, there is a real danger of collision. However, as due to limitations in simulation tools, it is not considered in this thesis that the Lead-Car may do another brake while Ego-Car is doing lane change, this performance is satisfying as far as no collision happens between the cars. Moreover, this performance shows a good capability of the system that handles situations even when there would be doubt about its performance.

Considering the plots above again, we see some major differences in comparison to case 1. In this case, TTC has less fluctuations and as the system avoids collision while doing lane change and increasing relative distance; so TTC and Headway Time are at their minimum when the lane change is performed with close distance, and then increases continuously. On the other side, velocity changes in the range of 98 to 130 km/h with regards to the acceleration that changes between -3 to +3 m/s^2 and finally reaches its stability at zero.

Since in this case a lane change is also performed, final steering angle is a combination of LCA and LKA outputs. So as seen in the plot, it changes in the range of -0.02 and 0.09, while using the values of -0.041 to 0.027 rad when using lane keeping assist to keep Ego-Car inside the lane. The larger range for LKA in this case due to the heading angle after completing lane change, in order to compensate the steering added to Ego-Car’s heading.

4.3.3 CASE 3) SUCCESSFUL AEB (CONFIGURATION 1 OR 2)

Third important case to study is when relative Δx and Headway Time are too small that if the Ego-Car performs lane change, collision would happen during this maneuver. This is why the system is designed in a way that if Headway Time is smaller than 0.5 seconds, priority of AEB is set higher than LCA and Ego-Car tries to reduce speed immediately. This has been tested again for the major possible scenarios and as expected, the most critical situations arise when initial Δx and V0_lead are the lowest.

As represented in the tables 13 and 14, among successful cases of this scenario, initial Δx=10 meters and V0_lead=100 km/h make the most critical case that is capable of avoiding collision.

Plots in the figure 30 show the performance of system in this case:

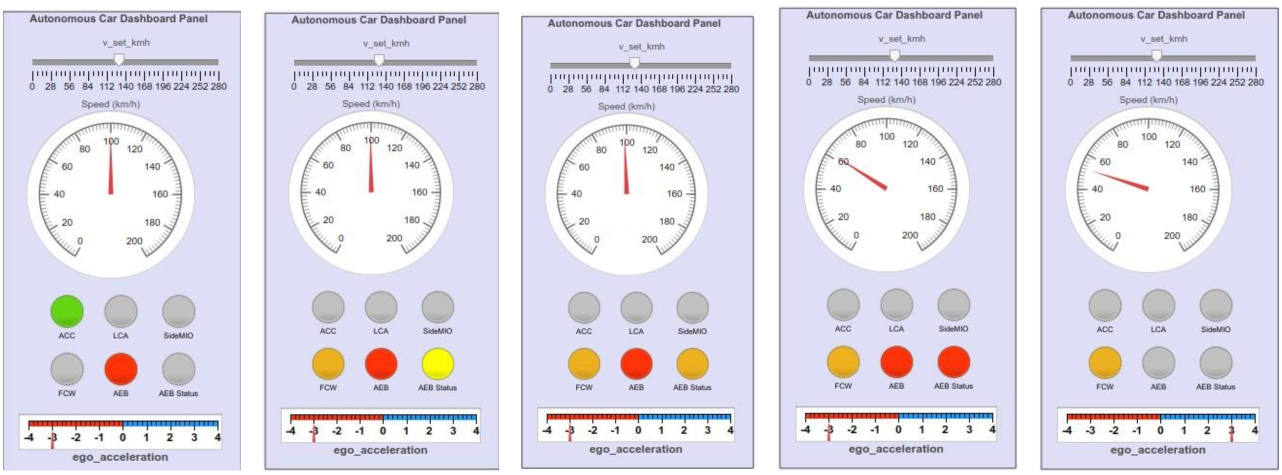


Figure 30. Steps of Automated Emergency Braking (AEB) represented on Dashboard

The above represented dashboard screen shots show the steps of emergency braking in a few seconds. As seen from left to right, AEB is first activated and then regarding the comparison of TTC to Braking Time, each step of AEB is activated as previously described in the methodology section. Each step of AEB is recognizable with the color of light on the dashboard. In the first figure from left, AEB is activated but not used yet, as the initial priority is set to ACC and it is still active. In the second figure, FCW represents that ACC is not able to stop the car, so AEB is used with the first level, represented by yellow color in the AEB status. Similarly, level 2 of AEB is enabled immediately as first partial braking would not be enough. The procedure continues as the 3rd level comes into account with full braking and reduces the speed until a safe distance is satisfied and headway time is higher than 0.5 seconds. This is met when the speed of Ego-Car has decreased to 60 km/h. Then the system releases braking and changes again to ACC to accelerate again and reach the desired speed while keeping distance. Alternatively, LCA can get the control if the conditions for lane change are met. This is also another complete functioning of system, as all of the four ADAS functions implemented would be used.

However, due to limitations in scenario generation app of Matlab, this case is not done in the simulations in spite of being a part of system's capabilities. The limitation mentioned is the fact that introducing actors other than Ego-Car in this app is event based and not time based. So, as we want to have an adaptive system that decides according the scenario situation, it is not possible to define another car in the 4th lane follows the first car on lane 4 with a varying distance and velocity. So, as the Ego-Car brakes very suddenly and decreases speed, the other car goes forward a lot and Ego-Car loses the chance to do a lane change after first collision is avoided with AEB; as there is not a car in lane 4 to check the performance of Ego-Car, considering the ACC and AEB in new lane.

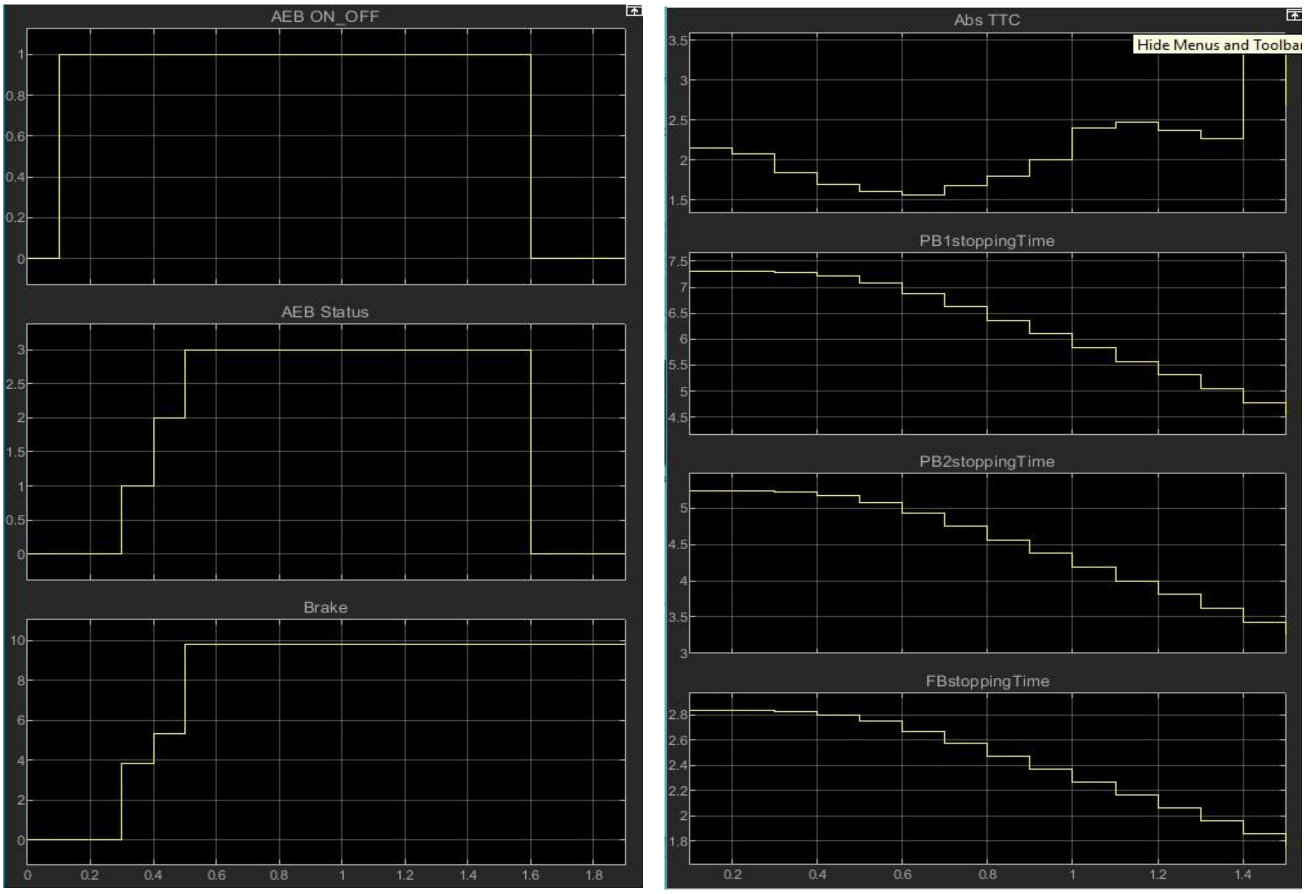


Figure 31. Simulation Plots for Case 3: Successful AEB (Configuration 1 or 2)

As seen in figure 31, after AEB is enabled with regards to headway time, AEB Status is zero at the beginning. This means that AEB has been turned on but waiting for other conditions to apply braking forces. Later, when conditions are met, braking steps 1, 2, and 3 are applied with their corresponding forces, as seen on the left charts. In this example, as the distance between two cars is very small, three stages of emergency braking are applied very fastly after eachother, however in larger relative distance, these steps would be applied partially and if one stage of braking would be able to avoid collision, the other stages will not be applied and consequently, the emergency brake is released and disabled, giving the opportunitiy to continue trip with ACC acceleration and velocity.

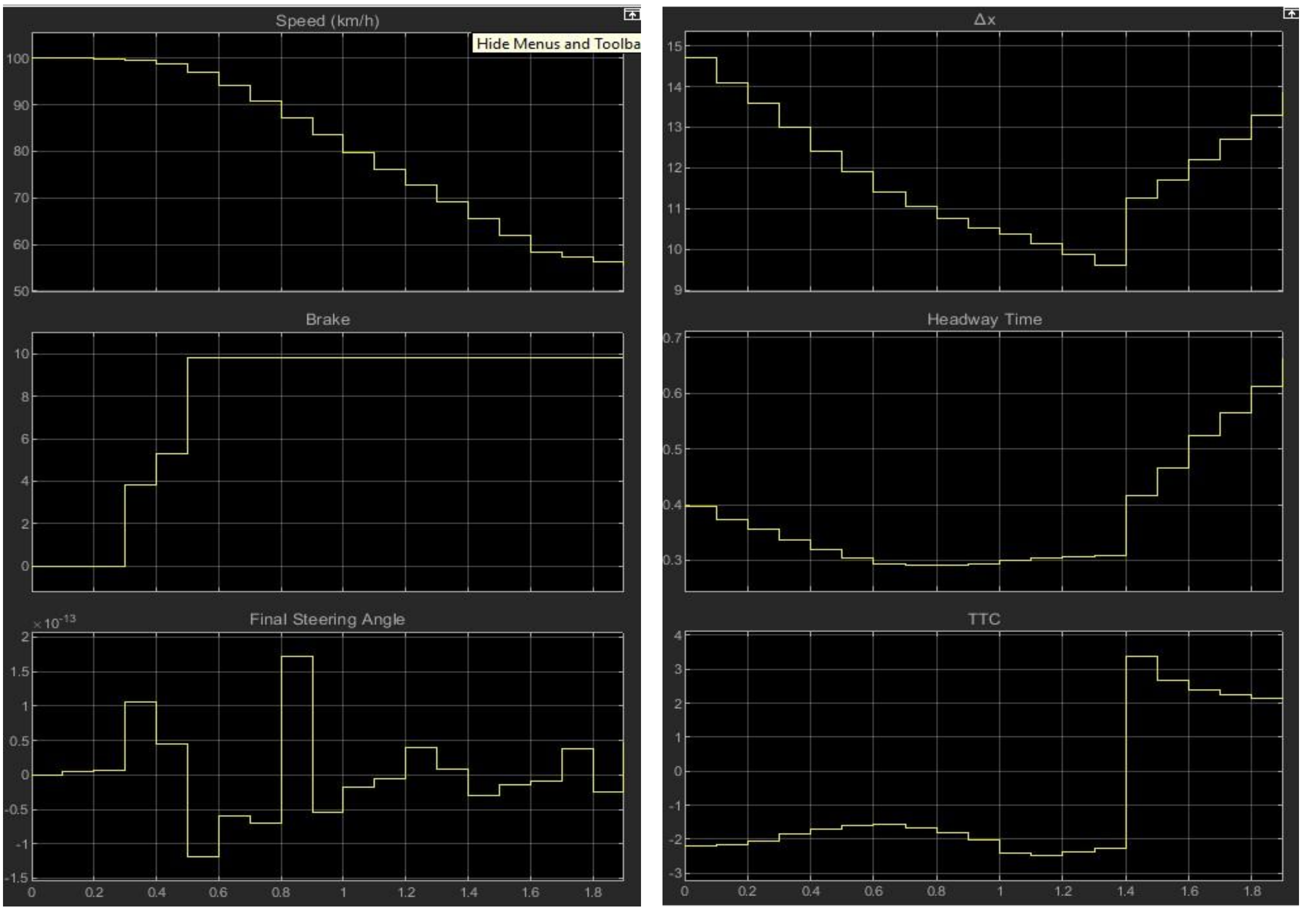


Figure 32. Simulation Plots for Case 3: Successful AEB (Configuration 1 or 2)

Considering the plots in figure 32 reveals that after braking is applied, speed is decreased with a sharp slope, thus HT and TTC also have a sharp increase after collision is avoided. Lane keeping system has also operated well keeping the car in lane with a very small derivation near to zero from center of the lane.

4.3.4 CASE 4) UNSUCCESSFUL LANE CHANGE (CONFIGURATION 1 OR 2)

This case happens in a couple of scenarios, due to a similar fault in the simulation; which could also be a potential failure in the system and algorithm; however, limitations in the scenario generation are the main reason for this. Since there is not another interface to test the system; it is not recognized surely which of these reason cause the problem. Therefore, these cases are reported as unsuccessful situations for doing a lane change.

Two situations that this problem arises, are described with their corresponding plots in the following, to clarify the concept of this problem

Situation 1) Lane Change after performing a successful AEB

As seen in the plots of figure 33, emergency braking is performed and distance to Lead-Car increased. Then LCA is enabled and Lane change starts. However, the second steering angle in the LCA is not applied by the system and the car loses control of steering, continuing with the first LCA angle and so gets out of the road without finishing the lane change task.

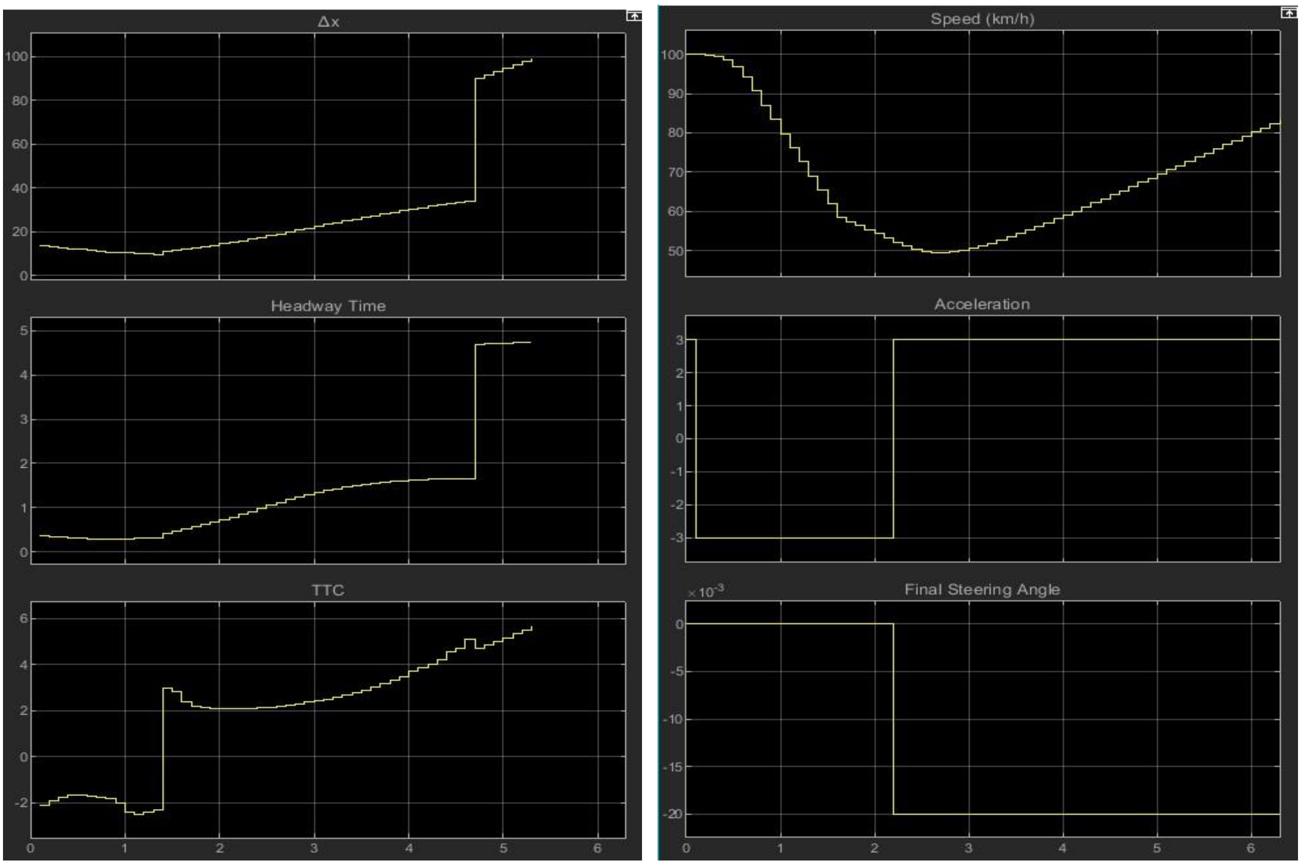


Figure 33. Simulation Plots for Case 4 - Unsuccessful Lane Change - Situation 1

Situation 2) Lane Change for some specific initial V0_lead and Δx as below

- Configuration 1
  - Δx = 50 and initial V0_lead = 90 and 100
  - Δx = 20 and initial V0_lead = 100 and 110
- Configuration 2
  - Δx = 50 and initial V0_lead = 120
  - Δx = 20 and initial V0_lead = 100

Also in this cases, the Ego-Car loses control while doing lane change and exits the road. Again it is not clear if the problem is from the control system or it is a bug in the simulation interface. As this test was done several times and every time, there was a minor difference between results, it is probable that this case would be a system failure of Automated Driving Toolbox in handling these kind of simulations. However, it is just an assumption and would be later investigated.

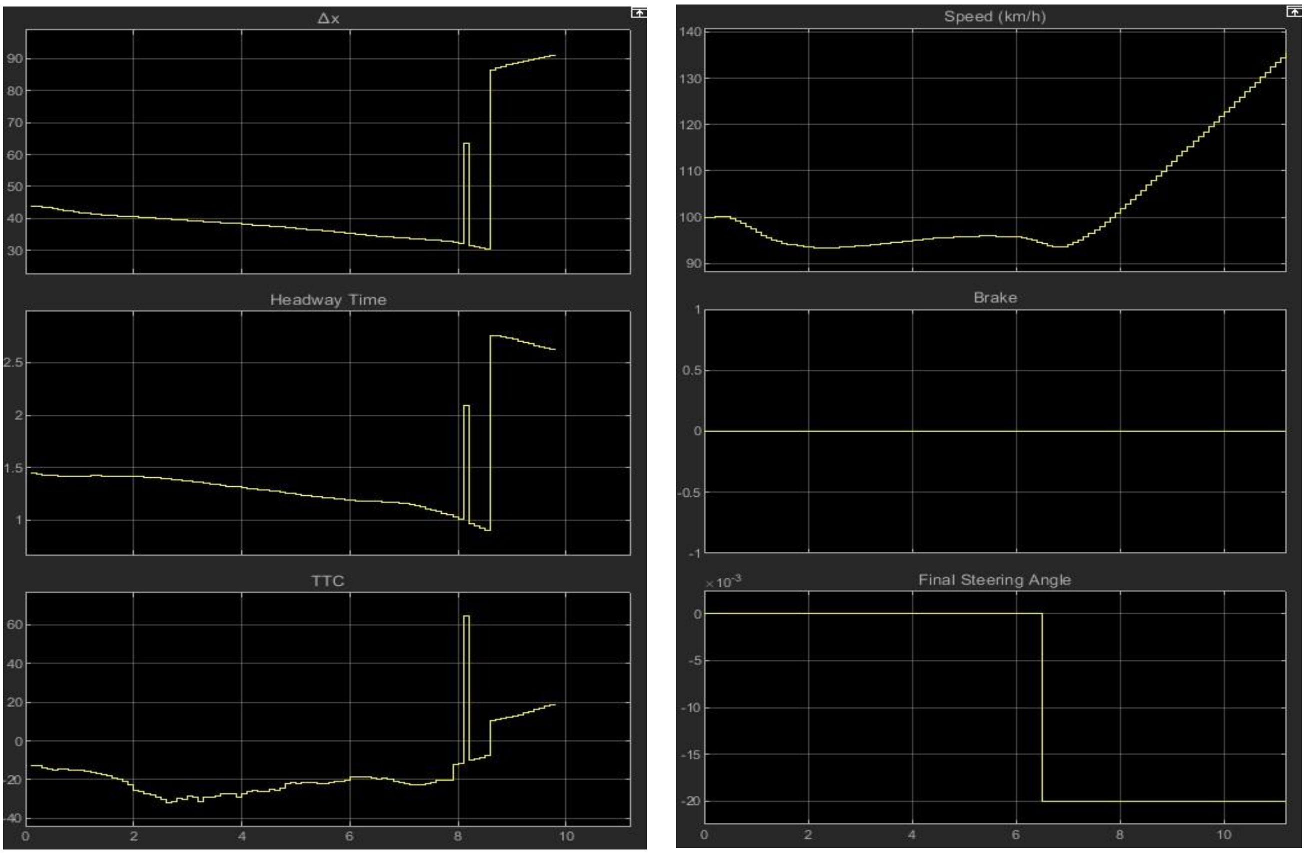


Figure 34. Simulation Plots for Case 4 - Unsuccessful Lane Change - Situation 2

Figure 35 shows the situation of Ego-Car (Blue) after losing control and moving out of the road.

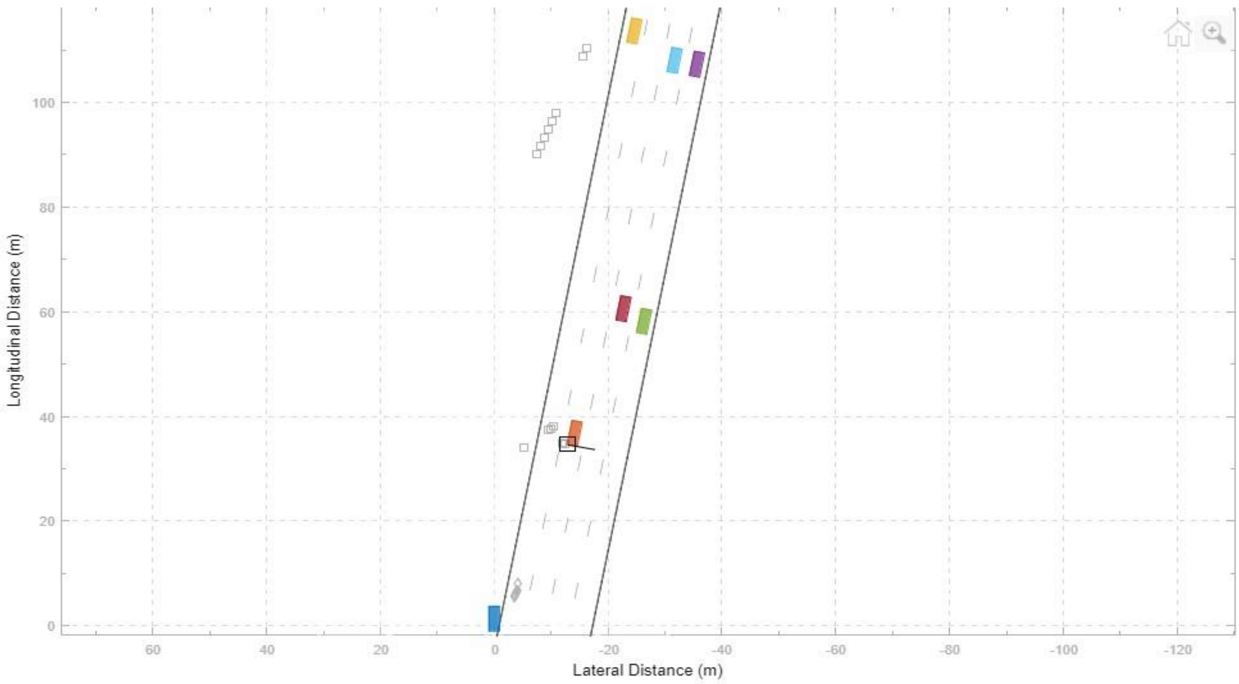


Figure 35. Situation of Ego-Car after losing steering control

4.3.5 CASE 5) COLLISION OCCURRENCE (CONFIGURATION 1 OR 2)

This case usually happens in very extreme scenarios, like when Δx is 10 meters and relative velocity between Ego-Car and Lead-Car is very small. For instance, while driving on highways (considering the minimum velocity of 80 km/h on highways) 10 meters is really a very small distance and very rare real systems would be able to avoid collision in this case. However, as the goal of this thesis is to test unpredictable situations that may arise, these extreme cases are also considered, so the performance of system shows its maximum capabilities.

As already mentioned in the tables 13 and 14, this system is even successful in some Δx = 10 m situations, in which V0_lead is higher than 100 km/h.

However, with initial Δx of 10 meters, and V0_lead equal to 80 or 90 km/h, system is not able to avoid collision at all and the collision happens.

Figure 36 shows how the simulation results come out for these cases.

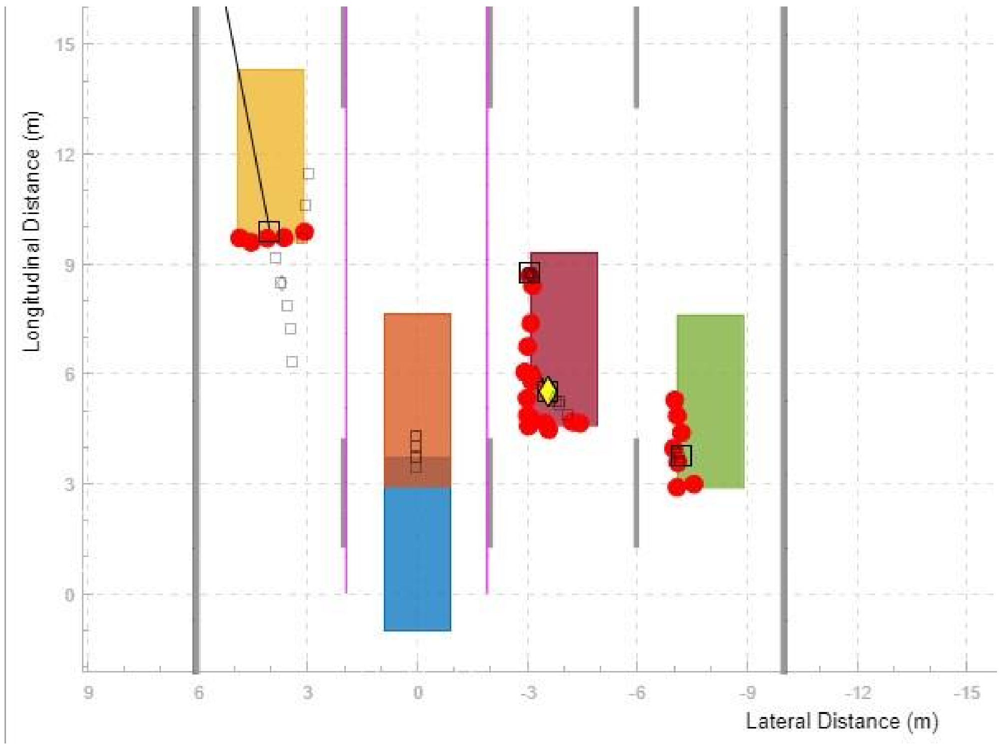


Figure 36. Collision Occurrence in Case 5

As seen above, in these cases the crash happens just after the start of simulation and thus simulation is stopped.

5. DISCUSSION AND EVALUATION OF THE SIMULATION RESULTS

Considering the critical cases of simulations in chapter 4, these criticalities are summarized in the tables 15 to 18, to make an evaluation of the system. It is obvious that the criticality arises with reducing the initial Δx and V0_lead. However, there are also cases with larger Δx and V0_lead that the system works partially well, but yields to failed functioning at the end as discussed in chapter 4. Thus, they are considered as faulty behavior of the system.

Table 15. Critical Scenario Results for Configuration 1

Configuration 1		Time Gap = 1.2 s				LCA active for 0.5 < HT < 1.2			AEB active for HT < 0.5		
Initial Δx = 50 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	90	✓	✓	✗	---	30	0.88	Yes	--	--	No
	100	✓	✓	✗	---	32	0.86	Yes	--	--	No
Initial Δx = 20 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	✓	✓	---	✓	9.5	0.29	Yes	---	---	---
	90	✓	✓	---	✓	12.5	0.36	Yes	---	---	---
	100	✓	✓	✗	---	14.3	0.38	Yes	---	---	No
	110	✓	✓	✗	---	14.6	0.39	Yes	---	---	No
Initial Δx = 10 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	---	---	---	✗	---	---	No	---	---	---
	90	---	---	---	✗	---	---	No	---	---	---
	100	✓	✓	---	✓	4.5	0.03	Yes	---	---	---
	110	✓	✓	---	✓	4.7	0.03	Yes	---	---	---
	120	✓	✓	---	✓	5	0.05	Yes	---	---	---

✓: Acceptable Performance ✗: Not Acceptable Performance ---: Not Used / Not Applicable

Table 16. Safe, Critical and Dangerous zones for Configuration 1

Configuration 1	Δx = 10 m		Δx = 20 m		Δx = 30 m		Δx = 40 m		Δx = 50 m	
V0_lead = 80 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB
V0_lead = 90 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB
V0_lead = 100 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB
V0_lead = 110 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB
V0_lead = 120 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB

Not Used Safe Critical Dangerous Collision

Table 17. Critical Scenario Results for Configuration 2

Configuration 2		Time Gap = 1.5 s				LCA active for 0.5 < HT < 1.5			AEB active for HT < 0.5		
Initial Δx = 20 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	✓	✓	---	✓	9.5	0.29	Yes	---	---	No
	90	✓	✓	---	✓	12.5	0.36	Yes	---	---	No
	100	✓	✓	✗	---	14.6	0.39	Yes	---	---	---
Initial Δx = 10 m	V0_Lead (Km / h)	ACC	LKA	LCA	AEB	Min Δx1 (m)	Min HT1 (s)	Collision 1 Avoided?	Min Δx2 (m)	Min HT2 (s)	Collision 2 Avoided?
	80	---	---	---	✗	---	---	No	---	---	---
	90	---	---	---	✗	---	---	No	---	---	---
	100	✓	✓	---	✓	4.4	0.027	Yes	---	---	---
	110	✓	✓	---	✓	4.8	0.048	Yes	---	---	---
	120	✓	✓	---	✓	5	0.051	Yes	---	---	---

✓: Acceptable Performance ✗: Not Acceptable Performance ---: Not Used / Not Applicable

Table 18. Safe, Critical and Dangerous zones for Configuration 2

Configuration 2	Δx = 10 m		Δx = 20 m		Δx = 30 m		Δx = 40 m		Δx = 50 m	
V0_lead = 80 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB
V0_lead = 90 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB
V0_lead = 100 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB
V0_lead = 110 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB
V0_lead = 120 km/h	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA	ACC	LKA
	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB	LCA	AEB

Not Used Safe Critical Dangerous Collision

Comparing these tables reveals that unsuccessful cases are less than successful cases and as the configuration 2 is selected to be used generally, it is obvious that this configuration is capable of handling situations with Δx greater than 30 meters, for every speed tested in this thesis. It is also able to avoid collision in more than half of the cases for Δx = 20 and 10. However, as these results are for very specific cases, further simulations with different criteria for scenario complexity would be needed to prove the applicability of this system as a safe system.

As already mentioned, the main problem for doing more simulations is the limitations in user interface of Automated Driving Toolbox and Scenario Designer app in Matlab and Simulink.

Cases marked as “Critical” with yellow color in tables 16 and 18, show the cases that first collision is avoided with the aid of ACC or AEB, but in the next step of simulation, Ego-Car has lost control of steering and keeping in lane. So, this cases are critical as the system should be improved to perform a better decision making in these cases; which would be either done by canceling any kind of lane change after emergency braking, or improving the performance of adaptive MPC by configuring the values of its parameters such as prediction horizon.

Changing the value for prediction horizon of adaptive MPC was tested as a possible solution after studying the lane change failures; since it was supposed that due to limited prediction horizon of 50 and sample time of T_s =0.1, Ego-Car would be able to observe lead-car in the new lane just when it is in the range of 5 seconds in front, however, increasing prediction horizon to 100 which will increase the view of system to 10 seconds in front, did not also solve the issue. As in the case of $\Delta x = 20$ and V0_lead = 100 for configuration 2, the second Lead-Car in lane 4 goes up to 80 meters forward, while Ego-Car brakes for emergency and then performs lane change. However, despite it is detected as the new Lead-Car after lane change, the second angle of lane change is not applied to compensate the heading angle and car travels with the first angle, adding continuously to the heading angle and exiting the road. This is also another confirming error for the problems of event based scenario generation.

Taking into account all the simulation results and also in-depth consideration of the 5 major cases already discussed in chapter 4, that may arise in this system, it is discovered that the overall functioning of this system in the specific scenarios that went under test, is acceptable in simulation level.

As it was expected, this system also has some failures like every other system that might be designed. However, as the topic of this thesis the evaluation of collision avoidance systems, these results are good enough for this purpose; while, failed cases are considered as the points that would be considered in later research by other enthusiastic researchers.

Considering the simulation results, it is clear that this implemented system is capable of meeting the level 3~4 autonomy for specified cases. All tested scenarios have been handled by the designed Autonomous Car Test-Bench independently as it is required in this level. Consequently, the Ego-Car is able to meet the expectations and avoid collision in specific cases as the system is designed and algorithms allow.

Obviously every system has its own disadvantages beside its capabilities. For this system, the main disadvantage would be mentioned as not performing the complete overtaking when a lane change is done. It is due to limitations in time and also the topic of the thesis which is the evaluation for collision avoidance systems. So, in this concept, it was considered enough that the Ego-Car passes first car using LCA and then switches to ACC and LKA to continue its journey on the new lane; or when needed, switching to AEB to avoid the second collision.

Simulations in this thesis are done with an exemplary Test-Bench which is not connected to a real system and just provides a simplified model of an autonomous car for simulation purposes. In order to check the validity of simulations, it is required to perform the same scenarios with real vehicles and check consistency of the results. However, as ground vehicle tests are not a part of this thesis, it is not considered here. It can be later checked in another research to prove the validity of these simulations.

6. SUMMARY AND OUTLOOK

6.1 SUMMARY

As one of the most recent technological upgrades in new world, autonomous driving is an exciting topic to discuss for every aspect of transportation, from passenger cars to public transportation, commercial vehicles and military vehicles.

Although automated driving is considered to solve a lot of traffic issues and of course it will come to the aid of drivers and passengers to enjoy a more secure journey, it will take a lot of time to reach the desired level of safety and independence. This is due to various unforeseen behaviors that may be arisen from other vehicles. Consequently, despite the automated driving systems are designed with several complex algorithms, there is always a possibility of hazardous actions from surrounding traffic, including vehicles, bikes, humans, and even animals that might intervene the road. A part of these issues might be fixed when the majority of vehicles on the road would be equipped with V2V and V2X communication. Accordingly, AVs will have more data about the traffic situations and will be able to communicate with each other to clarify the traffic situation while giving each other the possibility of better decision making in mutually shared road environment.

This thesis provides an overview of Autonomous Driving and ADAS, through the view point of automotive control systems, by implementing a test-bench to evaluate the performance of an autonomous car model in some critical driving scenarios and evaluating the performance of this system.

First two chapters of this thesis provide an overview and some useful information about the concepts and technologies that are needed to develop these ideas. Consequently, the route that autonomous driving concept has already taken and what is expected to reach in the future, with regards to the standards, legislation and limits is discussed and presented.

In the third chapter, concepts and technologies that are already mentioned, are taken into account to build up a model that would correspond to some specific simulations to test and evaluate the performance of an automated car. For implementation of this model, four main ADAS functions (ACC, LKA, LCA, AEB) based on Matlab pre-built models are merged into a test-bench and configured to obtain the best possible performance. Obviously, each of these individual systems are already available vastly in the market, being tested or used in several brand new cars. However, in the limited simulation systems such as Matlab-Simulink, each individual function is provided as a sample system to introduce the capabilities of Automated Driving toolbox; but when merging them together, there are a lot of issues that have been taken into account for this thesis, so that the overall system meets every single requirement of each system and also the whole system operates smoothly without losing the functionability of its single components. This has been done with in-depth study of each system and also using the available literature about these systems.

The core of this system is an Adaptive Model Predictive Controller (MPC) which controls the acceleration and steering of the car. Furthermore, additional algorithms are used to enable the system for switching between four ADAS functions with regards to the traffic situation and priority of functions described in the system. Finally, by testing the system while implementing and passing lots of trials and errors, an optimized system has been reached to test the desired driving scenarios.

Chapter 4 includes the simulations of the proposed scenarios with two configurations for the system which are used to differentiate between two cases of driving style, from which, configuration 2 is more defensive but shows a better overall performance than the other less conservative configuration 1.

The aforementioned configurations are defined considering the system capabilities and requirements for each sub-system of the test-bench, as in table 19:

Table 19. Configurations of thresholds for testing system performance

	Time Gap	LCA Activation Range	AEB Activation Range
Configuration 1	1.2 s	0.5 s < HT* < 1.2 s	HT < 0.5 s
Configuration 2	1.5 s	0.5 s < HT < 1.5 s	HT < 0.5 s

* Headway Time = $\frac{Relative\ Distance}{V_{ego}}$

Results obtained through simulation of specific scenarios are concluded in two tables for both of the configurations, indicating safe zones for the use of implemented system. Again it is observed that configuration 2 is a more logical setting for this system and performs lane change and emergency braking in a more logical way. Studying the results shows that the overall performance of this system in the specified conditions is satisfying.

Obviously, performance of this system is evaluated according to some specific and limited considerations that were also considered in the design phase of this system. However, there would be a large number of cases that this system could not be able to handle situations. This is the usual characteristic of every system and this is why there is never an end to research in this area.

To put in the nutshell, considering the limited time and simulation tools in this research, the obtained results are satisfying the expectations of the thesis topic. However, as this field of study is really vast, there would be further research in this area that would complete this system to work in more complex scenarios.

6.2 OUTLOOK

In the following, some main streams that would be investigated more in future research are described:

- One consideration which is already included in the algorithm but not used for the tests, is the capability of lane change to right lane in case of emergency. This would be done just in case that there is only a free spot on the right side and it is not possible to avoid collision with AEB. However, as discussed in the previous chapter, in cases that AEB is not able to avoid collision, it is always because of very small distance between two cars; So, if lane change is performed instead of AEB, there will be again a collision and probably it will be more dangerous since the Ego-Car accelerates during the lane change and this also reduces the relative distance to Lead-Car.

- Another possibility is considering the complete overtaking possibility from both left and right sides as in US it is allowed to perform an overtake from the right side. This is a worthy case to study if right side overtaking would be considered for normal and non-emergency lane changes; Even though, for Europe cases, just the left side overtaking will be possible, as the right side overtaking is prohibited by low.

- In this system, ACC, LCA and AEB work as a substitution to each other. In the methodology chapter, effort was put firstly to implement them to act simultaneously in some cases to improve functionability, but again due to some simulation and algorithm limitations, it was not possible to go further. Later research would be done to amalgamate these systems so that in very critical cases they can act simultaneously or switch to each other faster than in this system. In this case, the problem of increasing distance more than needed after AEB would be solved, and also after lane change, lane keeping assist would take the control of steering earlier.

- Considering one important failure of the system in losing control of steering after the lane change which is due to partial malfunctioning of LCA that does not provide the second steering angle due to unknown problems in this user interface, same scenarios would be tested in another system and simulation interface to discover the reason of this problem and to find whether it is a potential risky situation at those speeds and relative distances or it is a failure in the controller of this system.

- Although this Thesis is considering just the straight highway scenarios, all tested scenarios can be also tested for curved roads of different Radius (R) and Angles (α) to check the consistency of system performance, as the estimation of lane center is more difficult in curved roads. Moreover, it is also applicable to simulate some specific cases inside the cities and junctions. As the principle of detecting objects and keeping lane, doing the lane change or emergency braking is quite the same, but the only case is that the control algorithm should be developed more for two-way roads in cities.

- In order to get the most of this system, it is possible to test some more extreme scenarios with smaller thresholds for Time Gap and Headway Time regarding the activation of LCA and AEB, so that performance of the system would be also tested for the most critical conditions.

- Finally, if possible, real-time simulations and ground vehicle tests would be done for the same scenarios to prove reliability of simulation results.

LIST OF FIGURES

LIST OF TABLES

BIBLIOGRAPHY

[1] SAE J3016: Surface Vehicle Recommended Practice, 2018.

[2] SAE J3016: Levels of Driving Automation, May 2018
URL:https://www.sae.org/news/2019/01/sae-updates-j3016-automated-driving-graphic

[3] Morgan Stanley Research Global; (2013); Autonomous Cars: Self-Driving the New Auto Industry Paradigm

[4] Maurer M., Gerdes, J. C., Lenz, B., Winner, H.; (2016); Autonomous Driving. Berlin, Heidelberg: Springer Berlin Heidelberg.

[5] Bimbraw K.,Autonomous Cars: Past, Present and Future; Thapar University

[6] DiClemene J., Mogos S., Wang R.; (2014); Autonomous Car Policy Report; Carnegie Mellon University

[7] European Road Safety Observatory; (2016); Advanced Driver Assistance Systems

[8] Watzenig D., Horn M.; Automated Driving., Springer International Publisher AG Switzerland

[9] ISO 26262, Road Vehicles – Functional Safety – Part 5: Product development at the hardware level

[10] NHTSA; US DOT; (2016); Federal Automated Vehicle Policy: Accelerating the Next Revolution in Roadway Safety

[11] Alnaser A. J., Akbas M. I., Sargolzaei A., Razdan R.; Autonomous Vehicles Scenario Testing Framework and Model of Computation; Florida Polytechnic University

[12] Amersbach C., Winner H.; Functional Decomposition – a Contribution to Overcome the Parameter Space Explosion during Validation of Highly Automated Driving; Technical University of Darmstadt

[13] Amersbach C., Winner H.; Functional Decomposition – An approach to Reduce the Approval Effort for Highly Automated Driving; Technical University of Darmstadt

[14] Huang L., Xia Q., Xie F., Xiu H., Shu H.; Study on the Test Scenarios of Level 2 Automated Vehicles; 2018 IEE Intelligent Systems Symposium; Changshu, Suzhou, China

[15] Rocklage E., Kraft H., Karatas A., Seewig J.; Automated Scenario Generation for Regression Testing of Autonomous Vehicles

[16] Wang J., Zhang C., Yuehu L., Zhang Q.; Traffic Sensory Data Classification by Quantifying Scenario Complexity

[17] Xia Q., Duan J., Gao F., Hu Q., He Y.; Test Scenario Design for Intelligent Driving System Ensuring Coverage and Effectiveness; International Journal of Automotive Technology

[18] IAEA Oral Manual. 5 – Advanced Driver Assistance Systems. November 2018

[19] Online URL:http://roboticsandautomationnews.com/2016/09/22/advanced-driver-assistance-systems-trump-driverless-cars-by-stealth/7304/

[20] Kienke U.; Nielsen L.; Automotive Control Systems; 2nd Edition; Springer ISBN 3-540-23139-0 Springer Berlin Heidelberg New York

[21] Vehicle Dynamics Modeling – VtechWorks; URL:https://vtechworks.lib.vt.edu/bitstream/handle/10919/36615/Chapter2a.pdf

[22] Yu S., Wang J., Wang Y., Chen H.; Disturbance Observer Based Control for Four Wheel Steering Vehicles with Model Reference

[23] Zhu B., Piao Q., Zhao J., Guo L.; Integrated Chassis Control for Vehicle Rollover Prevention with Neural Network time-to-rollover warning metrics

[24] Adaptive Cruise Control Using Model Predictive Control URL:https://www.mathworks.com/help/releases/R2018b/mpc/ref/adaptivecruisecontrolsystem.html

[25] Lane Keeping Assist with Lane Detection URL:https://www.mathworks.com/help/releases/R2018b/mpc/ug/lane-keeping-assist-with-lane-detection.html

[26] Automatic Emergency Braking with Sensor Fusion URL:https://www.mathworks.com/help/releases/R2018b/driving/examples/autonomous-emergency-braking-with-sensor-fusion.html

[27] Multi Object Tracker Block in Matlab and Simulink URL:https://www.mathworks.com/help/releases/R2018b/driving/ref/multiobjecttracker.html

[28] Vehicle Dynamics Blockset in Matlab and Simulink URL:https://www.mathworks.com/help/releases/R2018b/vdynblks/index.html

[29] Vehicle Dynamics Terminology J670E-2008

[30] ISO 8855, Road Vehicles – Vehicle Dynamics and road-holding ability - Vocabulary

[31] Vehicle Body 3DOF in Matlab and Simulink URL:https://www.mathworks.com/help/releases/R2018b/vdynblks/ref/vehiclebody3dof.html

[32] Sensor Fusion Using Synthetic Radar and Vision Data
URL:https://www.mathworks.com/help/releases/R2018b/driving/examples/sensor-fusion-using-synthetic-radar-and-vision-data.html

[33] Lane Following Controller with Sensor Fusion and Lane Detection
URL:https://www.mathworks.com/help/releases/R2018b/driving/examples/lane-following-control-with-sensor-fusion-and-lane-detection.html

[34] Zheng T.; Model Predictive Control; SCYIO; ISBN 978-953-307-102-2