POLITECNICO DI TORINO

Master's degree in Mechatronic Engineering

FINAL THESIS PROJECT

ARTIFICIAL INTELLIGENCE ON BOARD: ELECTRIC VEHICLE AUTONOMOUS DRIVING

Tutor:

Prof. Carlo Novara

Student:

Cristiana Perlongo

April 2019

CONTENTS

ABSTRACT

In a period, in which technological development dominated the socio-economic scene, with an aim of evolving and adapting to human needs, the Artificial Intelligence sector could only become the focus of scientific research. When the keyword becomes "comfort", the automation world is highly considered: machine learning, from this point of view, is of great interest for the human assistance process.

The focus of the following thesis research is, in particular, to recreate a case study in which the concept of machine autonomy embraces the theme of mobility. The automotive industry, in fact, has concentrated for many years a large part of its resources in the implementation of assisted driving systems and, more recently, autonomous driving. The new vehicles are equipped with an increasing number of optional and advanced security systems.

In detail, the project carried out during this research, which took place at the company bylogix srl of Grugliasco, concerns the problem of the Path Following, i.e. the ability of the vehicle to follow a desired trajectory in relation to factors that depend from the intrinsic characteristics of the car and the surrounding environment.

The Path Following topic raises many questions related to the ethical, bureaucratic and penal responsibilities of any high-risk situations, as well as the definition of the priority levels to be attributed to the entities involved. Moreover, as the urban reality is varied and dynamic, it is necessary that such systems can predict the most common situations and react very quickly to external stimuli.

The present case concerns the implementation of a controller which, given a predetermined trajectory, can process the optimum steering angle for maintaining the path.

In particular, the LQR control has been chosen, starting from a reference signal involving the desired values of the lateral position, the lateral velocity, the yaw angle and the yaw rate, to minimize a cost function with the aim of reducing the deviations between the mentioned values and the real ones. The dynamics of the vehicle, which in the real case is a new generation electric model, has been approximated for this purpose using the Bicycle Model.

Through the tests performed on the MATLAB / Simulink platform, it was possible to compare the results obtained by the different choices of the weight coefficients of the matrices involved. Finally, the same study was revised for the use of the LQI control to highlight any aspects that can be deduced from the comparison between the various results.

CHAPTER 1 – INTRODUCTION

1.1 Thesis Purpose

The following study is the result of an experimental thesis research, which has been carried out in collaboration with a company. It is based on the design of a controller for the autonomous driving of an electric vehicle.

In particular, it was decided to apply the Linear Quadratic Regulator (LQR), and to compare different solutions obtained varying certain weight indices, for the Path Following problem.

The company hosting the project is bylogix srl, which provided its advanced equipment and the help of its qualified staff.

Figure 1 – bylogix srl

Bylogix srl is an engineering company active since 2007 in various sectors. It deals with the design and development of SW and HW systems for integrated electronic components, the development of test and validation systems, as well as the verification of safety requirements according to today's standards.

The company is mostly concerned about the automotive sector: considering the market needs, it focused on developing the technologies inherent to autonomous driving.

The activities span the entire V-model of software development.

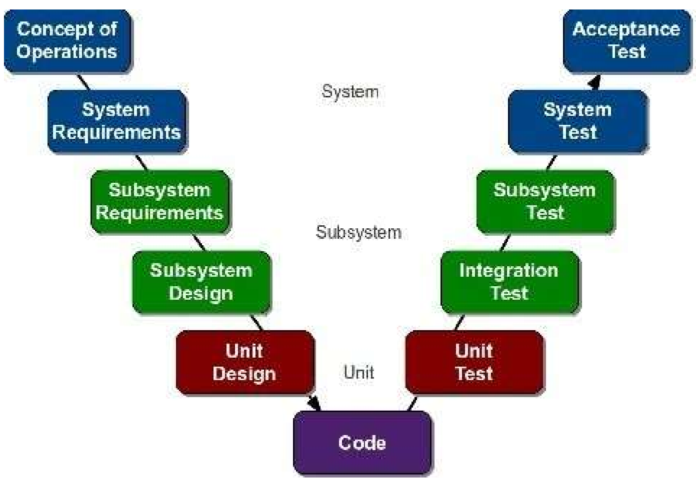

Figure 2 – V-Model of the SW Development

1.2 Artificial Intelligence

Artificial Intelligence is the discipline that deals with designing and creating machines capable of acting independently. It studies the theoretical foundations and the technical methodologies useful for the design of hardware and software systems. Those systems are capable of giving to the electronic computer performances which are considered relevant to human intelligence.

Figure 3 – A.I

The first steps of this branch of computer science were moved in the 50s, when the first model of artificial neuron was presented: since that moment, the concept and the study of neural networks and intelligent machines have initiated. In particular, the intelligent system implements behavioural processes that tend to imitate the human being ones, with operations pushing it to act and think humanly, or rationally. Artificial Intelligence is currently classified in two major types:

- Weak Artificial Intelligence: it identifies systems capable of simulating some human cognitive functions, without reaching their real intellectual abilities;
- Strong Artificial Intelligence: it identifies the so-called "wise systems", able to develop its own intelligence without having to emulate the cognitive abilities of the human being.

At the base of this subdivision, there are the concepts of machine learning and deep learning, which deal with the learning model that characterizes the machine.

Machine learning is a set of methodologies through which the software is allowed to learn, so that it is able to perform a task without being programmed: in other words, it learns how to manage situations and correct its errors.
Deep learning, on the other hand, needs neural networks so that the system can emulate the mind of man: a much more powerful computational capacity is needed, which can support different layers of calculation and analysis.
To sum up, we can say that the functioning of an Artificial Intelligence machine is characterized by four functional levels:

- Comprehension;
- Reasoning;
- Learning;
- Interaction.

Currently, there are many application sectors: from automotive to healthcare, from marketing to cybercrime, from management to public security.
Many ethical problems still exist. Above all, it is still problematic to predict whether a machine can think, or if it could become a danger to humanity.
Certainly, progress must be motivated, possibly through the collaboration between man and machine.

Figure 4 – Sophia Robot

1.3 The Autonomous Driving

According to the modern needs asked by fast technological progress, the world of Artificial Intelligence could not fail to achieve successes in a sector experiencing an extreme growth such as the Automotive one. In fact, the automotive companies are in continuous competition to offer their customers the best avant-gardes. Regarding that, most of the energy is used in the development of one of the industry's current pillars: assisted and autonomous driving. An autonomous vehicle is able to study the surrounding environment, without the human intervention, through equipment such as sensors and navigation devices. In order to perform this, cameras, GPS, motion and speed sensors and artificial vision systems are mounted.

An advanced control system, that interprets the information received to identify appropriate routes, obstacles and road signs, allows the car to choose and follow the right path by making the right driving maneuvers.

Obviously, the realization of this technology isn’t that simple, since the problem of prediction and the one of real time control coexist: driving in a complex and unpredictable environment, like the urban one, requires a great reactivity and the ability to respond to variations and to surrounding stimuli.

The concept of Artificial Intelligence, in this contest, comes into play when a vehicle, after collecting and memorizing a set of experiences, acquires the ability to make decisions faster: it begins to recognize scenarios and to react, relying on events already verified, according to an optimization perspective.

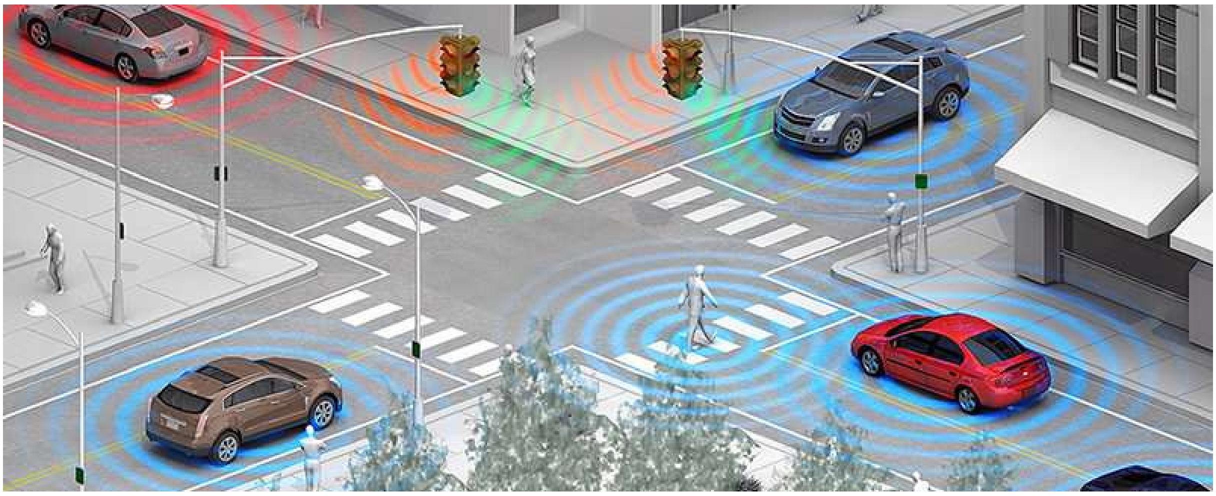

Figure 5 – Urban Environment

In 2014, the most recent international standard, referred as the J3016, was published. It defined six different levels for automatic driving, based on how much the driver should intervene.
In particular, the defined levels are:

- Level 0 - No autonomy from the car: man takes care of any aspect of the driving;
- Level 1 - Driving assistance: man takes care of every aspect of the driving, but he is supported by electronic devices capable of warning him about dangerous situations;
- Level 2 - Partial Automation: man manages the driving, but integrated safety systems intervene on the acceleration, the braking and, partially, on the steering wheel;
- Level 3 - Conditional Automation: the car is able to manage the driving in ordinary situations, acting on braking, acceleration and direction, but man intervenes in case of problematic situations;
- Level 4 - High Automation: the automatic system can handle any situation, except when driving in extreme conditions;
- Level 5 – Complete Automation: the car can manage any situation in complete autonomy, no human intervention is required.

Nowadays, autonomous vehicles are still under test. In some countries, those vehicles are being tested on the road. The product sale has not yet started due to the fact that some tests have caused fatal accidents.
Despite this, once the control technique has been perfected, autonomous driving would reduce road accidents by 90%, facilitate the mobility of the elderly and the disabled people. It would also reduce significantly the traffic in the city.
Many diatribes related to ethical interests still exist, such as the civil and criminal responsibility of the driver and the software developers when an accident occurs. In addition, there is the problem of how the system will choose whom to preserve the most: on one hand the passengers and pedestrian protection, on the other one the minimization of the overall damage.
Through all these considerations, it can be said that the autonomous driving will revolutionize the urban reality and the role of the man within it.

1.4 ADAS Systems

1.4.1 ESC – Electronic Speed Control

The ESC is an electronic active safety control unit for motor vehicles: during the heeling phase, this device regulates the engine power and brakes the interested wheels with different intensity, in order to stabilize the set-up of the car. The system corrects the oversteering and understeering situations, as well as the trajectory loss.

Figure 6 – ESC

The ESC uses some information that comes from the car itself:

- 4 speed sensors (one per wheel) that communicate to the control unit the velocity of each wheel;
- 1 steering angle sensor, which communicates the driver's intentions to the control unit, through information about the position and movement of the steering wheel;
- 3 accelerometers (one per space axis), positioned at the center of the vehicle, which indicate to the control unit the forces acting on the car;
- Some sensors already present.

The control unit acts both on the power supply of the engine, reducing the torque, and on the individual brake calipers, by correcting the dynamics of the vehicle. In the case of understeer, the brakes intervene by braking the rear wheel inside the curve, and generating a mechanical moment opposite to the wrong direction, while in the event of oversteer the same operation is performed by braking the front wheel outside the curve. This system is usually joined by the traction control systems (TCS) and the anti-locking wheels system (ABS), being practically complementary in the control of the stability of the vehicle in different driving conditions.
The EU has decided to make the system mandatory for new car models since the 1st November 2011.

1.4.2 AEB – Autonomous Emergency Braking

The AEB is a device used to generate the maximum braking power available, if the applied one is not enough. In emergency braking situations, if the ABS is not activated, the system increases the pressure on the pedal even if the driver decreases the pressure. Through this action, the range of the ABS will be reached, which will result in the prevention of the wheels locking.

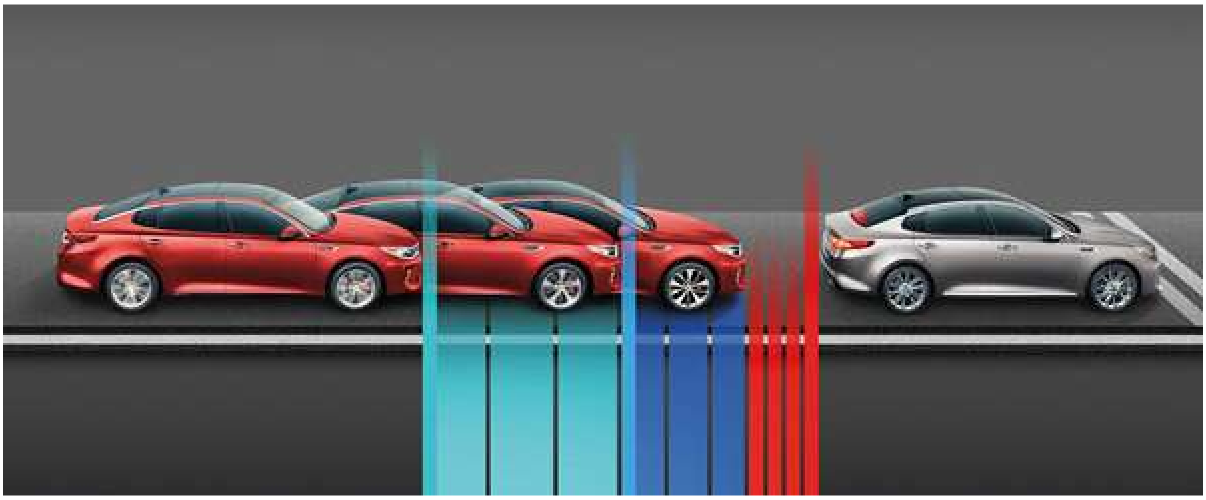

Figure 7 – AEB

The AEB produced by Bosch, is even more advanced: an infrared video camera system detects the moment when the car is getting too close to the ahead one: over 30 km/h, it warns the driver, and provides higher braking force if this activates the brake pedal, or it performs an emergency braking with maximum power in order to avoid collision. In the future, assisted braking will also be combined with the recognition of cyclists and pedestrians. These systems already exist in some car models.

1.4.3 TCS – Traction Control System

The TCS is an electronic control unit that prevents the slippage of the driving wheels of the car during acceleration. Currently, all manufacturers must enter it by law, as standard or as an accessory.

The system identifies the sliding of the wheels thanks to the sensors installed at each wheel and to an electronic computer that processes the acquired data and acts on some variables.

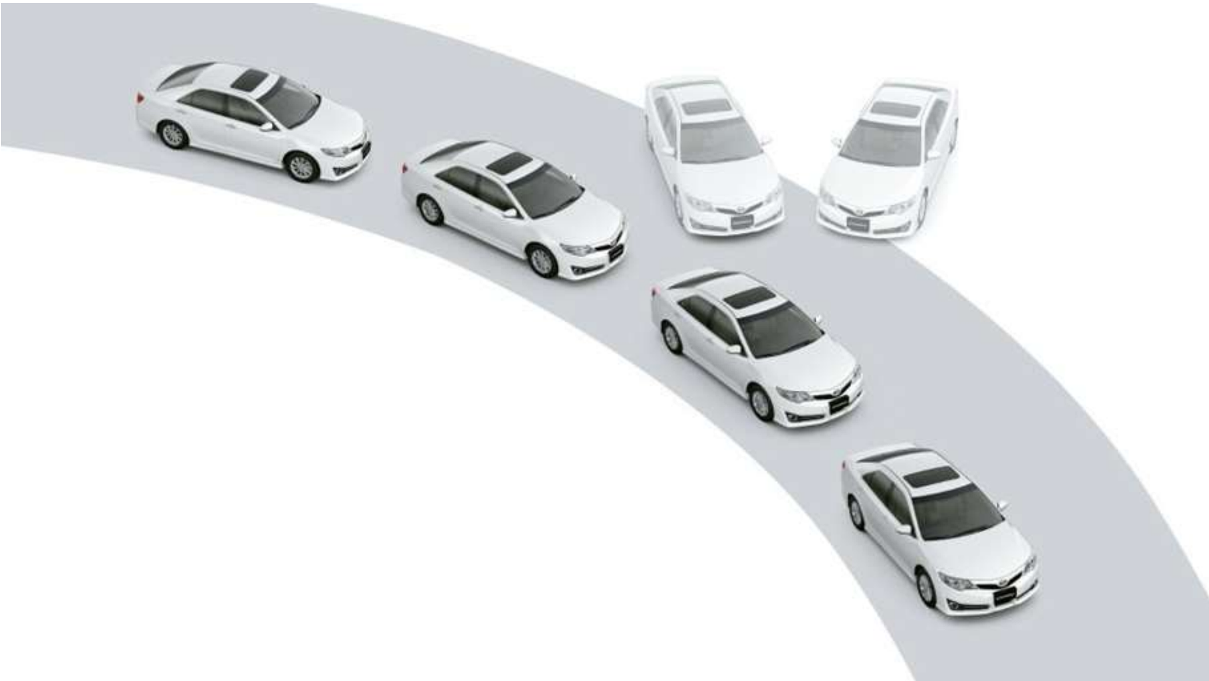

Figure 8 – TCS

These systems can intervene on:

- Brakes: When a wheel loses grip, it dissipates all the power supplied in torque by the engine, leaving the other wheels with greater grip without power and stopping the car. To overcome this problem, the speed sensors send data to the control unit, which activates the brake at the wheel involved and allows the redistribution of the power supplied by the engine.
- Motor Power: In the presence of slipping of the driving wheels, the system activates the brakes on the wheels involved in order to eliminate the applied torque excess and allow the same wheels to use the grip offered by the road.

The system is useful in case of rain, icy road or loss of traction by a wheel, but it is disadvantageous in the presence of irregular surfaces (sand, snow, etc.), since slippage occurs since the beginning and the locking of the wheels prevents the movement of the vehicle.

1.4.4 ABS – Anti-Lock Braking System

The ABS is an electronic control unit that prevents the locking of the wheels. On each wheel there is an Encoder (angular position transducer), composed of a transducer and a phonic wheel. This wheel consists of a gear wheel that turns with the vehicle wheel, and a fixed inductive proximity sensor that detects the passage of the teeth of the wheel mentioned before. The electronic control unit counts the number of teeth passing in a time unit, it calculates the rotation speed of the wheel and, if it detects the locked wheels during braking, it activates the hydraulic pump to decrease the braking force: practically, it performs the same action that the driver would accomplish by releasing the brake pedal.

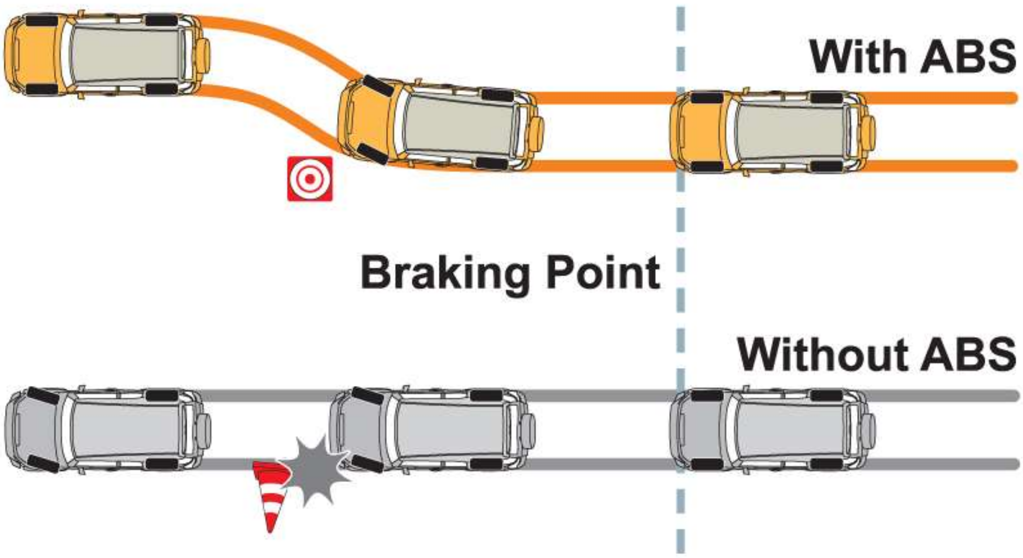

Figure 9 – ABS

ABS is a one-way system, i.e. it only performs a brake release action: the closing force of these must be provided by the driver through the brake pedal.
In an emergency braking, the driver can press the pedal as hard as possible without worrying to lock the wheels, since the control unit will decrease this force to the limit of vehicle holding. The presence of EBD (Electronic Brake Distribution) is fundamental for the optimization of

this operation. In fact, with it, the braking force can be transferred between one axis and the other, thus being able to exploit all the adhesion that the wheels are able to provide, and which is normally different between front and rear axle.
When the ABS system starts operating, the driver senses vibrations coming from the brake control, caused by the variation of the oil pressure in the braking circuit.
Many accidents are caused by drivers who, frightened by the vibration of the pedal, release the brake without completing the braking. For this reason, some cars are equipped with a braking system that includes the emergency braking assistant.
During the braking, part of the load is transferred to the front axle by lowering it, and the vertical force on the wheels increases considerably: during the braking with ABS, the steering wheel vibrates, and it is advisable to hold it firmly with two hands.
The system is disadvantageous not only in two-wheeled vehicles, but also in the presence of irregular terrain.

1.4.5 ACC – Adaptive Cruise Control

The ACC is a cruise control system: it helps the driver in keeping the safety distance from the vehicles ahead, warning him if manual intervention is necessary. The ACC uses a radar sensor, which detects moving objects that precede the vehicle on the same track.

Figure 10 – ACC

This system maintains the speed of the vehicle, which has been set before, on constant value until the presence of other vehicles is detected. If a vehicle moving at low speed is detected, the ACC will reduce the engine power and, if necessary, it will operate the braking maneuver to maintain the preset safety distance. An alarm is generated if the driver intervention is required to maintain a predefined distance.

The anti-collision warning function warns the driver to brake quickly, in order to avoid any collisions.

The ACC system exonerates the driver from continuous adjustments to the cruise speed set, reducing his efforts and making driving more comfortable.

1.4.6 BSD – Blind Spot Detection

The system detects the presence of other vehicles in blind spots. In order to perform this, it uses some sensors, cameras and radar positioned under the rear bumper of the vehicle.

Figure 11 – BSD

If a vehicle is detected, the system alerts the driver with a bright or acoustic warning.

1.4.7 Park Assist

The INRIA (National Institute for Information Technology and Automation) has defined five different types of Park Assist systems, classified in a progressive scale of automation:

- Driver Assistance
- Partial Automation
- Conditional Automation
- High Automation
- Full Automation.

The Park Assist helps the driver in searching for adequate longitudinal and transverse parking spaces, and supports him during maneuvering phases, automatically intervening on the steering wheel in the most suitable way to park the car in the available area. The first phase of Park Assist is to measure the parking space. After the driver has chosen the "seat", the system takes the car to the optimal position to start the maneuver and, in some systems, it also proceeds to automatically moving the steering wheel to access it: the driver only has to accelerate or brake.

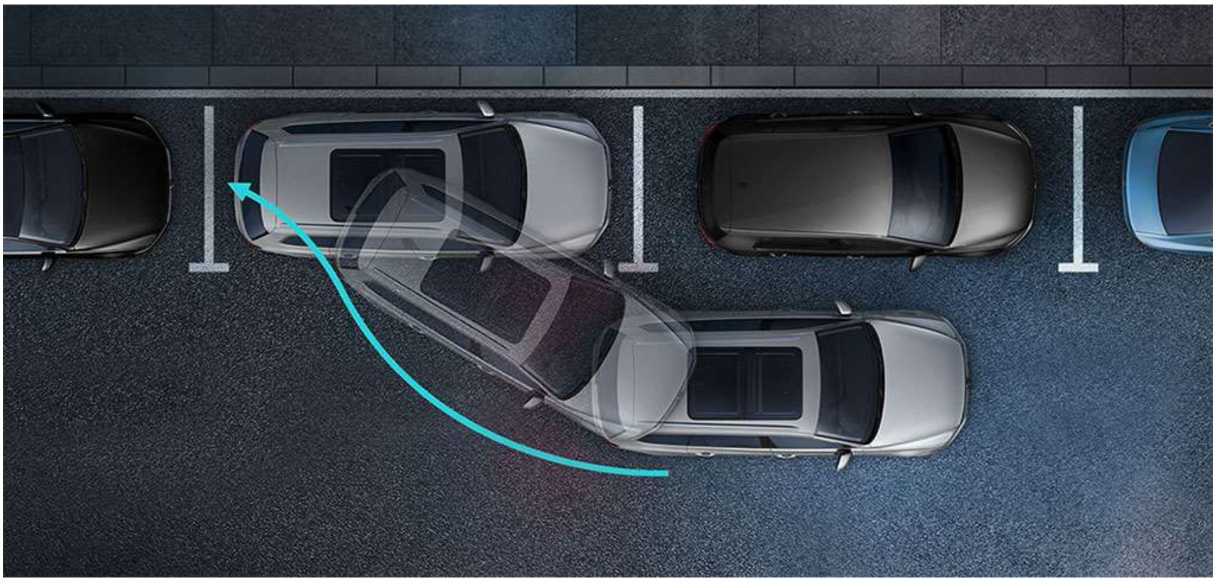

Figure 12 – PAM

The park assist systems of the most modern cars are equipped with ultrasonic sensors (high frequency mechanical waves) distributed along the outer perimeter of the car. They are able to detect the exact dimensions of the parking space and the presence of any obstacles. The software installed in the control unit will automatically calculate the available space, creating a virtual image of the environment surrounding the vehicle, and determining the manoeuvres necessary for parking. Moreover, thanks to a series of electrical impulses that are transformed into mechanical impulses, they are able to move the steering wheel by itself to carry out the ideal manoeuvres and to park the car horizontally, vertically, etc.

1.4.8 LKA – Lane Keep Assist

The LKA system has been designed to ensure that the roadway is maintained by the vehicle.
In fact, a video camera recognizes the demarcation lines of the lane placed in front of the vehicle: if the car is approaching too much the white line, it intervenes on the steering wheel by turning it gently, in order to restore the right direction.
In particular, the lane maintenance assistance system applies a braking force identical to each wheel.

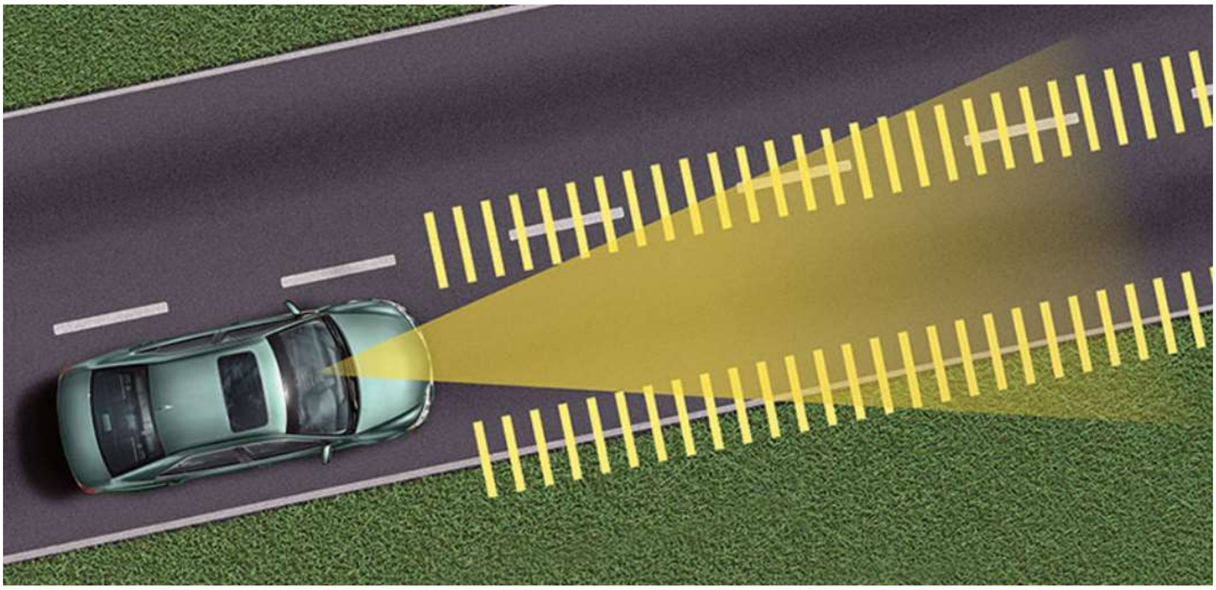

Figure 13 – LKA

CHAPTER 2 – SPERIMENTAL SETTING

2.1 The Vehicle

2.1.1 Electric Vehicles

The birth of the electric vehicle dates to the nineteenth century, with the development of the first "advanced" prototypes with electric propulsion.

At that time, this method of feeding was more convenient and reliable than combustion, until the technological advances of the automotive industry confined electric vehicles in very few sectors.

Figure 14 – Electric Vehicle

The electric car is a car whit an engine that uses, as primary energy source, the chemical energy stored in rechargeable batteries, and made available to the engine as electricity. Overall, this vehicle has greater energy efficiency than internal combustion engines.

Among the positive factors, there is the very low maintenance demand, especially for brakes and engines, both subject to less wear. In addition, there are parts that are not present, and that in a conventional car need to be replaced. From a fiscal point of view, these vehicles enjoy facilitated taxation and special benefits. Finally, the possibility to recover energy is interesting, thanks to technologies such as regenerative braking. The biggest disadvantage, however, concerns the installed batteries autonomy, especially their charging times, which vary according to the available power supplied by the appropriate current distributors. We should not forget the high costs of buying cars, probably due to the low competition on the automotive market.

2.1.2 Citroen E-Mehari: Features and Performances

The car used to implement this study is an electric vehicle of the Citroen company, and the model is E-Mehari.

Figure 15 – Citroen E-Mehari

It was equipped with a system of cameras, speed sensors and GPS devices, as well as an intelligent and open HW and SW processing platform that acts as the vehicle's brain, giving it the perception of the surrounding environment.

The Citroen E-Mehari, like all the electric cars, has the advantages and disadvantages discussed above. In particular, considering the main problem, the battery of the Citroen E-Mehari requires a charging time of 13 hours through a 230 V power outlet, or equal to 8 hours through a 400 V special distributor.

Figure 16 – Vehicle Parameters

The figure above shows the physical dimensions of the car, reported in such a way as to intercept the position of the center of gravity. This detail will be very important for the equations of motion of the model, in order to be controlled, and for the positioning of the body in the considered reference system.
Another highly useful fixed parameter is the mass, which in our case is about 1404 kg (704 kg for the front axle, 700 kg for the rear axle).
Finally, considering the fixed parameters, the cornering stiffness of the wheels and the yaw inertia moment of the car are very important: the values of the cornering stiffness are about 25000 N for the front wheel and 33000 N for the rear one, while the value of the yaw inertia moment has been set at 26000 kgm^2.
As for the car's performance, the battery provides a range of almost 200 km, it can develop 68 hp, and the maximum speed is around 110 km/h. Finally, the maximum torque is about 166 Nm, and it ensures an acceleration from 0 to 50 km/h in 6.4 seconds.
To sum up, the Citroen E-Mehari will not be a product intended for large-scale sale, but it is certainly an interesting innovative object, which opens the way to new concepts of mobility, and is a good subject for experimental activities.

2.2 The On-Board Technology

2.2.1 Nvidia PX2

The car's supercomputer is a very powerful system designed for autonomous driving: Nvidia PX2, in fact, is a system that reaches 8 teraflops and 24 tera operations of deep learning per second. This high computing power is guaranteed by the union of two Tegra processors, and by two separate GPUs.

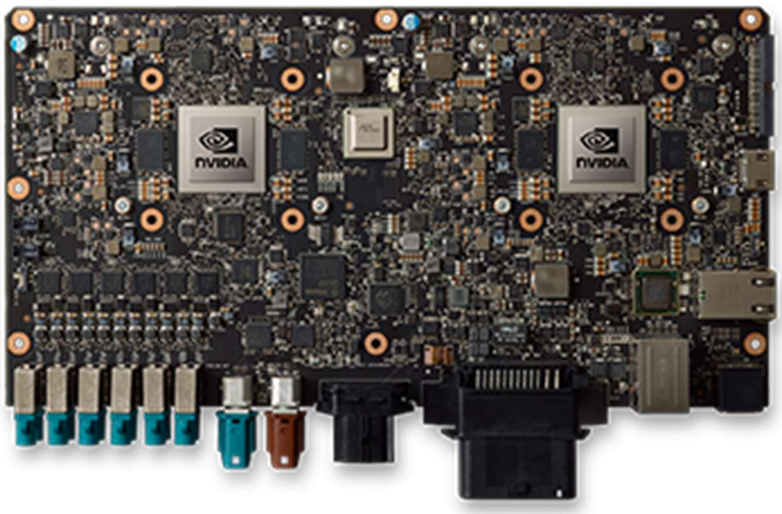

Figure 17 – NVIDIA Board

The system is called upon to collect and process the inputs of tens of sensors, and to decide in real time how to respond. High performance is necessary for this decision to ensure timely reactions and margins of error to be inferior than the ones of a human driver.

The Nvidia DRIVE PX2 platform combines deep learning, sensor fusion and surround vision to change the driving experience. It can understand in real time what is happening around the vehicle, precisely locating it within a high-definition map acquired with end-to-end mapping technology. Moreover, it can plan a safe route forward, basing its choice on the study of the available free space.

What we obtain is a robust representation of the surrounding reality, and the ability to memorize the collected experiences: the system, recognizing already occurred events and already traveled roads, speeds up reaction times and gives the right priority to the passenger safety and the minimization of the damage.

It is important to specify that the platform is designed to support the ASIL-D index (Automotive Safety Integrity Level D), i.e. the highest initial risk classification, and the highest level of safety, according to ISO 26262 protocol.
Thanks to the use of DNN neural networks, the system can recognize the objects that characterize the environment and can classify them according to their type. In fact, it is possible to see, through the acquisition of images on monitors, that each object is delimited by a different colored box depending on whether it is another vehicle, a pedestrian, an object or a road sign.

Figure 18 – NVIDIA Display Visualization

Deep learning is also able to solve issues related to the environment, and therefore to know how to manage critical situations such as rain, snow, fog or poor visibility. From the market point of view, the Nvidia DRIVE PX2 platform has already been adopted by at least 50 production companies in the automotive sector, which have not resisted the advanced performance and simplicity of use of this avant-garde. Obviously, autonomous driving systems are still under development, therefore they are subjected to continuous updates and necessary revisions with the aim of improving their performance. It is not difficult to deduce that the platform considered in this study will soon be replaced by new and more intelligent versions.

2.2.2 GPS Device

The co-protagonist, in this autonomous vehicle reality, is undoubtedly the GPS (Global Positioning System). This device, in fact, uses a civil satellite positioning and navigation system able to provide the mobile terminal with information on its geographical position, as well as its time in any meteorological condition and in every point of the earth.

This localization takes place through a series of artificial satellites in orbit: each satellite transmits a radio signal, and the various signals are processed by the receiver.

Figure 19 – Satellite

The operating principle of the GPS system is based on the spherical positioning method, which starts from the measurement of the time taken by a radio signal to travel the satellite-receiver distance. The receiver accurately calculates the satellite's propagation distance starting from the difference between the time received and the one of its clock, synchronized with the atomic clock of the satellite.

Today, GPS devices are used in various sectors: in the context of daily mobility, they can be considered substitutes for traditional paper maps.

Figure 20 – GPS Mapping

2.3 Data Collection

With reference to our system, data acquisition and processing take place thanks to the close collaboration between GPS and the Nvidia platform. The first, in fact, maps the route in terms of terrestrial coordinates: latitude and longitude. Therefore, the vehicle must follow the route established by the geolocation device. The Nvidia platform, on the other hand, is useful in recognizing obstacles and possibly recalculating the pre-established route according to their size and position of the obstacles.

In detail, the study is based on the calculation of the optimal steering angle so that the vehicle traverses the trajectory date. In order to perform this, it was decided to make, in each iteration, a comparison between the bearing angle of the vehicle and the bearing angle of the path section near the vehicle.

The bearing angle is the angle formed between the direction of the line that joins the north and south terrestrial and the direction of the considered entity.

Starting from the terrestrial coordinate, the system has to calculate the (X,Y) coordinates with the respect to an absolute reference system. After that, it shall make the right calculations to describe the proper desired trajectory.

The aim of the thesis, as already anticipated, is to design a controller for the problem of Path Following. This controller might be able to follow the ideal path established according to the modalities previously treated, and to minimize the lateral position error with the respect to the trajectory and the relative yaw angle.

Figure 21 – Data Collecting

CHAPTER 3 – THE MODEL

3.1 System Modeling

The process of modeling a physical system allows us to study a problem by representing the essential and particular aspects of the reality considered. In other words, it is the cognitive process that transforms a physical entity into a theoretical model, using the language of logic and constructing the mathematical relationships that link the behavior of the objects involved. This abstraction of reality through a language, allows to realize simulations of processes and forecasts, as well as to explain concepts.

In fact, the models give us the chance to reason about the system, and to make predictions about the future behavior of the system itself.

Modeling a system is a dynamic and evolutionary process, which is essentially developed in three phases:

- Interpretation: it depends on the knowledge of the observer;
- Representation: it consists in the passage from the mental image of the system to the mental image of the model;
- Outsourcing: it is the phase in which the image becomes shareable thanks to the use of a language and a representation system.

The model we choose to represent a dynamic system depends on which are the questions we have to answer: there are more models corresponding to a single physical system, whose levels of accuracy depend on the phenomenon of interest.

It is possible to classify the models in three different types:

- "Black-Box" Models: they derive only from experiments;
- "Grey-Box" Models: they are based on the model, and the experiments are necessary only for the identification of the parameters and for the validation;
- "White-Box" Models: they do not provide any experiment.

The choice also depends on the study case: it is obvious that, where the real system was not yet available, or the experiments are dangerous, it is useful to move towards the "model-based" approach, in order to extrapolate the characteristics of the system in the considered operating conditions. In the case of dynamic systems, i.e. systems whose characteristics vary during the time, the external description of the black-box model, i.e. the input/output model, is preferred.

The purpose of block diagrams is to emphasize the flow of information and to hide the details of the system: different processes are represented as boxes.

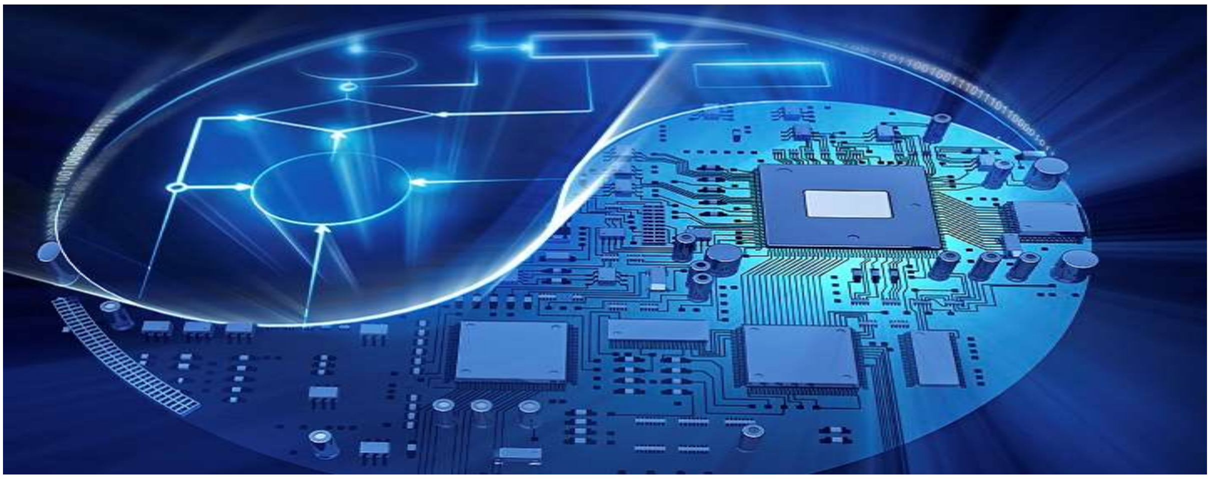

Figure 22 – Block Diagram Concept

A common class of mathematical models used to describe dynamic systems is the one of Ordinary Differential Equations (ODE), represented mathematically in the following way:

$$\frac{dx}{dt} = f(x)$$

where $x = (x_1, x_2, \dots, x_n) \in R^n$ it is the vector that describes the current state of the system, while the equation describes the rate of variation of the state as a function of the state itself.
The equation we have just seen is defined as an autonomous system, since there are no external influences. In usual applications, however, it is useful to consider the effect of external disturbances or controlled forces within our model. In this way, the equation is rewritten as follows:

$$\frac{dx}{dt} = f(x, u)$$

where u represents the effect of external influences and, as can be seen from the equation itself, it affects the rate of variation of the state.
The model described above is called Controlled Differential Equation.

The input/output concept of the system is very useful in engineering systems, as it allows the system to be decomposed into individual components by connecting them through their inputs and outputs. In particular, it is highly important in the study of the special class of linear invariant time systems.

3.2 The Control View

More classic implementations of dynamic systems focus on autonomous systems. In the control, it is necessary to consider the presence of external influences, as well as to know the state of the system by exploiting the information made available by the sensors. The disturbances and uncertainties of the model become critical elements, as they are the main reason why feedback chains are used.

Since the control systems enjoy cooperation with sensorial devices, it is possible to obtain the models experimentally.

The representation of the model, in the context of the control, is given by the following equations:

$$\dot{x} = f(x, u)$$
$$y = g(x, u)$$

where $\dot{x} = dx/dt$, with $x \in R^n$ state vector, $u \in R^q$ is the control signal vector, while $y \in R^p$ is the measurement vector.

The model described is called "state-space model", whose degree corresponds to the size of the state vector. Furthermore, it is possible to define it as a time-invariant system, since the functions f e g do not directly depend on time.

A system is called linear when the functions f e g are linear with the respect to x and u, whereby a linear state-space model can be represented as follows:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

where A, B, C and D are constant matrices. In particular, A is the dynamic matrix, B is the control matrix, C is the sensor matrix and D is the direct term.

Models of this type help us to predict the evolution of the state of a system starting from initial conditions, and to study the stability of an equilibrium point and the frequency response of input/output.

3.3 Vehicle Dynamics Modeling: The Bicycle Model

In order to simplify it, the vehicle is also recurrently approximated to a simplified model of its dynamics.
The so-called Bicycle Model, in fact, has been used in this study, as it lends itself well to an essential description of the system.

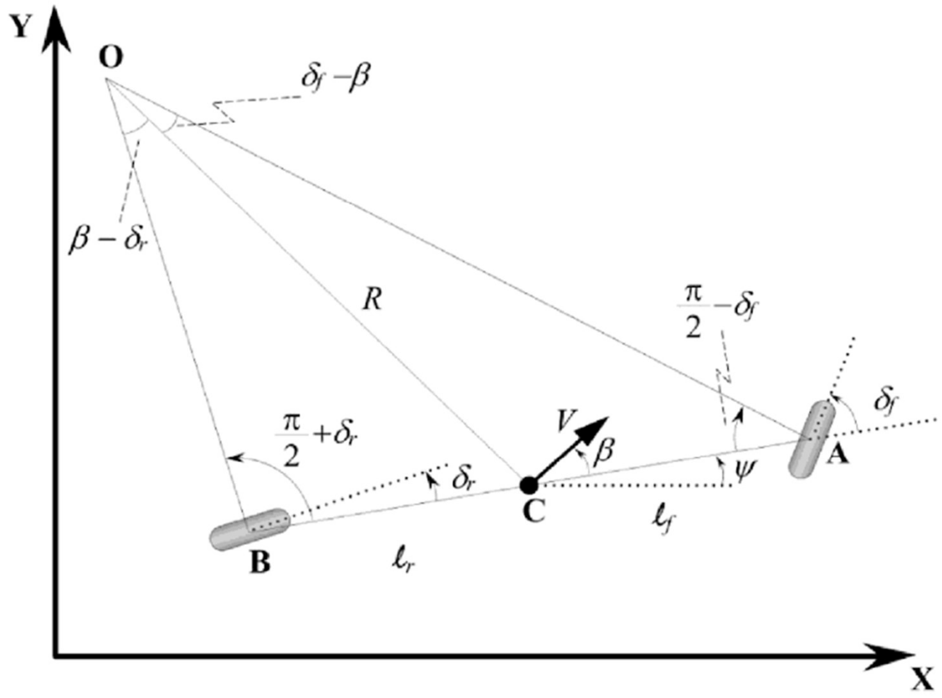

Figure 23 – Bicycle Model

The vehicle is represented as a system composed of a front wheel and a rear wheel, connected to each other by an axle.
The quantities involved are defined as follows:

- m: Vehicle Mass;
- δ_f: Steering Angle of the Front Wheel;
- δ_r: Steering Angle of the Rear Wheel;
- β: Slip Angle;

- Ψ: Yaw Angle;
- C_f: Cornering Stiffness of the Front Wheel;
- C_r: Cornering Stiffness of the Rear Wheel;
- V_x: Longitudinal Velocity of the Vehicle;
- l_f: Distance from the Center of Mass to the Front Axle;
- l_r: Distance from the Center of Mass to the Rear Axle;
- a_y: Lateral Acceleration of the Vehicle;
- F_{yf}: Lateral Force at the Front Wheel;
- F_{yr}: Lateral Force at the Rear Wheel;
- I_z: Yaw Inertia Moment of the Vehicle;
- R: Curvature Radius of the Road;
- α_f: Slip Angle of the Front Wheel;
- α_r: Slip Angle of the Rear Wheel.

The equations of motion, for this model, derive from the equilibrium equations of lateral forces and moments. They are respectively shown below:

$$ma_y = F_{yf} + F_{yr}$$
$$I_z\ddot{\Psi} = l_f F_{yf} - l_r F_{yr}$$

where the lateral acceleration a_y, the lateral frontal force F_{yf} and the lateral rear force F_{yr}, are expressed as follows:

$$a_y = \ddot{y} + V_x\dot{\Psi}$$
$$F_{yr} = C_r\alpha_r cos\delta_r$$
$$F_{yf} = C_f\alpha_f cos\delta_f$$

In turn, the slip angles of the front and rear wheels are represented by the following expressions:

$$\alpha_f = \delta_f - \theta_{vf}$$
$$\alpha_r = \delta_r - \theta_{vr}$$

where θ_{vr} and θ_{vf} are the angles of the velocity vector at the two wheels:

$$\theta_{vr} = \tan^{-1}\left(\frac{\dot{y} - L_r\dot{\Psi}}{V_x}\right)$$
$$\theta_{vf} = \tan^{-1}\left(\frac{\dot{y} - L_f\dot{\Psi}}{V_x}\right)$$

By making appropriate substitutions, the following is obtained:

$$\begin{aligned} ma_y &= F_{yr} + F_{yf} \\ &= C_r\alpha_r cos\delta_r + C_f\alpha_f cos\delta_f \\ &= C_r(\delta_r - \theta_{vr})cos\delta_r + C_f(\delta_f - \theta_{vf})cos\delta_f \end{aligned}$$

More precisely:

$$m(\ddot{y} + V_x\dot{\Psi}) = C_r\left(\delta_r - \tan^{-1}\left(\frac{\dot{y} - L_r\dot{\Psi}}{V_x}\right)\right)cos\delta_r + C_f\left(\delta_f - \tan^{-1}\left(\frac{\dot{y} - L_f\dot{\Psi}}{V_x}\right)\right)cos\delta_f$$

Making an approximation, and considering the angle δ_r to be null:

$$\begin{aligned} m(\ddot{y} + V_x\dot{\Psi}) &\cong C_r\left(\delta_r - \frac{\dot{y} - L_r\dot{\Psi}}{V_x}\right) + C_f\left(\delta_f - \frac{\dot{y} - L_f\dot{\Psi}}{V_x}\right) \\ &= \left(C_r\delta_r - C_r\frac{\dot{y}}{V_x} - C_r\frac{-L_r\dot{\Psi}}{V_x}\right) + \left(C_f\delta_f - C_f\frac{\dot{y}}{V_x} - C_f\frac{-L_f\dot{\Psi}}{V_x}\right) \\ &= -C_r\frac{\dot{y}}{V_x} - C_f\frac{\dot{y}}{V_x} - C_r\frac{-L_r\dot{\Psi}}{V_x} - C_f\frac{-L_f\dot{\Psi}}{V_x} + C_r0 + C_f\delta_f \\ &= \left(-C_r\frac{1}{V_x} - C_f\frac{1}{V_x}\right)\dot{y} + \left(-C_r\frac{-L_r}{V_x} - C_f\frac{-L_f}{V_x}\right)\dot{\Psi} + C_f\delta_f \end{aligned}$$

$$m(\ddot{y} + V_x\dot{\Psi}) = \left(-\frac{C_r + C_f}{V_x}\right)\dot{y} + \left(\frac{C_rL_r - C_fL_f}{V_x}\right)\dot{\Psi} + C_f\delta_f$$

Finally, we obtain the equation describing the behaviour of $\ddot{y}$, which is also the first derivative of the first element of the state vector:

$$\ddot{y} = \left(-\frac{C_r + C_f}{mV_x}\right)\dot{y} + \left(\frac{C_r L_r - C_f L_f}{mV_x} - V_x\right)\dot{\Psi} + \frac{C_f}{m}\delta_f$$

Going back to the equilibrium equation of the moments, and making the appropriate substitutions:

$$I_z\ddot{\Psi} \cong -C_r\left(\delta_r - \frac{\dot{y} - L_r\dot{\Psi}}{V_x}\right)L_r + C_f\left(\delta_f - \frac{\dot{y} - L_f\dot{\Psi}}{V_x}\right)L_f$$

$$I_z\ddot{\Psi} \cong -C_r\left(-\frac{\dot{y} - L_r\dot{\Psi}}{V_x}\right)L_r + C_f\left(\delta_f - \frac{\dot{y} - L_f\dot{\Psi}}{V_x}\right)L_f$$

We finally obtain the equation that describes the behaviour of $\ddot{\Psi}$, which is instead the derivative of the second element of the state vector:

$$\ddot{\Psi} = \left(\frac{C_r L_r - C_f L_f}{I_z V_x}\right)y - \left(\frac{C_r {L_r}^2 + C_f \dot{L_f}^2}{I_z V_x}\right)\dot{\Psi} + \frac{C_f L_f}{I_z}\delta_f$$

The final system, taking into consideration the structure of the type:

$$\begin{cases}\dot{x} = Ax + Bu \\ y = Cx + Du\end{cases}$$

will be described by the matrices A, B, C and D.

As can be seen, the state is given by the vector $x = (y \quad \dot{y} \quad \Psi \quad \dot{\Psi})'$, while the input is $u = \delta_f = \delta$.

The matrices A, B, C and D, according to the previous demonstrations and other simplifications, are therefore represented as follows:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\dfrac{2*C_r + 2*C_f}{mV_x} & \dfrac{2*C_r + 2*C_f}{m} & \dfrac{2*C_rL_r - 2*C_fL_f}{mV_x} \\ 0 & 0 & 0 & 1 \\ 0 & \dfrac{2*C_rL_r - 2*C_fL_f}{I_zV_x} & \dfrac{-2*C_rL_r + 2*C_fL_f}{I_z} & -\dfrac{2*C_r{L_r}^2 + 2*C_f{L_f}^2}{I_zV_x} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 2*\dfrac{C_f}{m} \\ 0 \\ 2*\dfrac{C_fL_f}{I_z} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D = 0$$

CHAPTER 4 – LQR/LQI CONTROL

4.1 Control Theory

The Control Theory is a branch of engineering that aims to study the trend of a dynamic system, whose parameters are therefore variable over time.

In practice, it takes care of modifying the behavior of this system by manipulating appropriate input quantities: it may be required that the output remains constant at a preset value as the input varies (regulation problem), or that it follows the dynamics of the input itself (command problem).

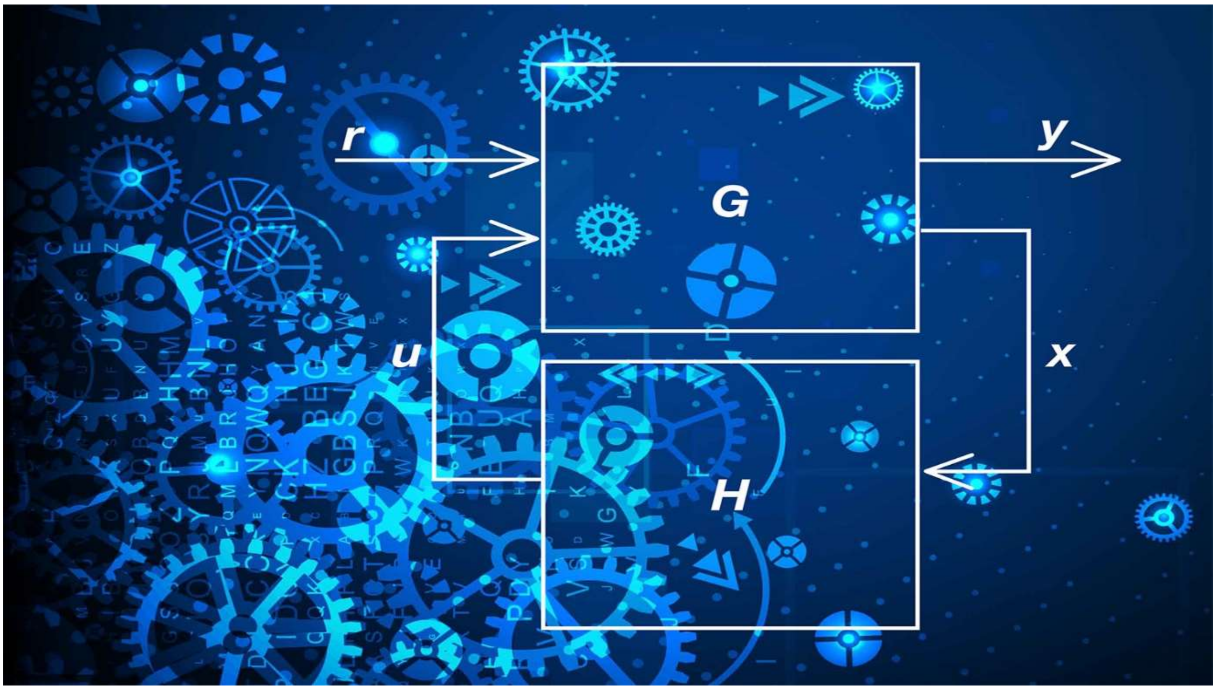

Figure 24 – Control Theory

The control system is the result of an in-depth study of the definition of the mathematical model that most faithfully describes the trend of the real model.

Depending on the number of inputs and outputs, a system can be defined as SISO if it has only one input and only one output, while it is of the MIMO type when it is characterized by multiple inputs and outputs.

At the same time, we can classify the input variables as follows:

- Control Variables: they can always be measured;
- Disturbances: sometimes they cannot be measured, and they negatively affect control.

As for the output variables, however, we have:

- Performance Variables: they are the controlled quantities, they can be directly or indirectly measured;
- Intermediate Variables: these are the physical variables that can be used to indirectly calculate the performance variables.

An automatic control system, depending on the type of problem studied, can be designed in different ways: the most common structures are open-loop control and closed-loop control.
Open-loop control is a predictive type of control, which can be implemented in the presence of a mathematical model so precise that it does not require knowledge of the output value.
The closed-loop control, instead, is a feedback control, which has a greater complexity but is able to stabilize a system that is not. In general, the output value is subtracted from the reference signal, in order to give the so-called error signal to the control system.

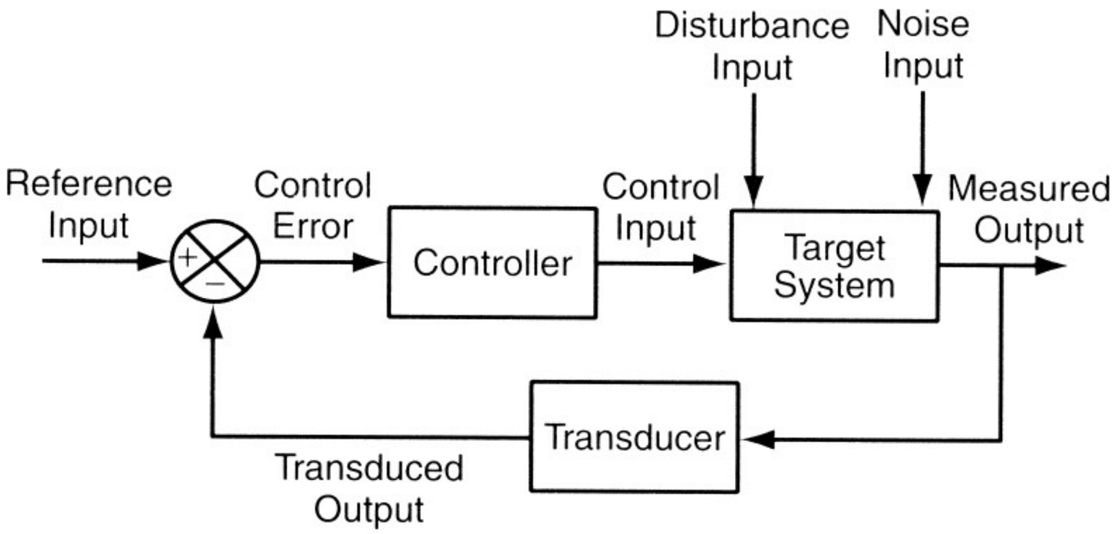

Figure 25 – Closed-Loop Control System

The input signals can be of various types, but the most common are:

- Dirac Delta Function;
- Ramp Function;
- Step Function;
- Sinusoidal Function.

Once the characteristics of the problem under examination are identified, the most appropriate control solution is adopted. To guarantee reliability and stability to the system, we tend to choose feedback control, whose possible types are the followings:

- PID Control (Proportional Integrative Derivative);
- Sliding Mode Control;
- Adaptive Control;
- Optimal Control;
- Robust Control;
- Dead-Beat Control.

For this study, the optimal control was selected, which is a set of control algorithms that stabilize a dynamic system thanks to the minimization of a cost function indicated with $J(x, u)$. With x we indicate the state of the system, while with u we indicate the input generated by the controller following the same minimization.

In particular, the control solution adopted in the present project is the one called LQR (Linear Quadratic Regulator), and it has been compared to the LQI (Linear Quadratic with Integral) variant.

4.2 LQR: Linear Quadratic Regulator

As part of the optimal control, as already mentioned, the LQR controller is a dynamic compensator able to minimize the quadratic cost function $J(x, u)$ with carefully chosen weighting factors. This means that the error tends to zero under steady conditions, through the accurate selection of the weighted matrices Q and R.

Indicating with x the state, with u the input to be generated, and with y the obtained output, the linear system is described by the following equations:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

where $x \in R^n$, $u \in R^q$ and $y \in R^p$, while A, B, C and D are matrices that do not depend on time. In particular, A is an $n \times n$ matrix, B is an $n \times q$ matrix, C is a $p \times n$ matrix and D is a $p \times q$ matrix.

The cost function to be minimized is shown as follows:

$$J = \int_0^{\infty} (x^T Q x + u^T R u + 2x^T N u) dt$$

The resulting controller depends on the solution of a predetermined Equation of Riccati, that is an ordinary differential equation which is quadratic in the unknown function.
The control law in feedback, which minimizes the mentioned cost, is:

$$u = -Kx$$

where K is given by:

$$K = R^{-1} B^T P$$

and P is found thanks to the Equation of Riccati chosen as follows:

$$A^T P + PA - PBR^{-1} B^T P + Q = 0$$

According to the theorem of the solution existence, for each positive semidefinite Q matrix, and for every positive definite R matrix, there is always a solution u that is optimal for the minimization of the cost function.
In the case in question, which is not limited to a finite time horizon, Q and R must also be constant to ensure that the cost function is always positive. Furthermore, to ensure that the same function is also limited, A and B must be controllable.
About the stability problem, which is fundamental for the correct functioning of the controller, the LQR regulator guarantees this property without requiring design changes: if the system is stabilizable and detectable, limiting the cost index also stabilizes the system.

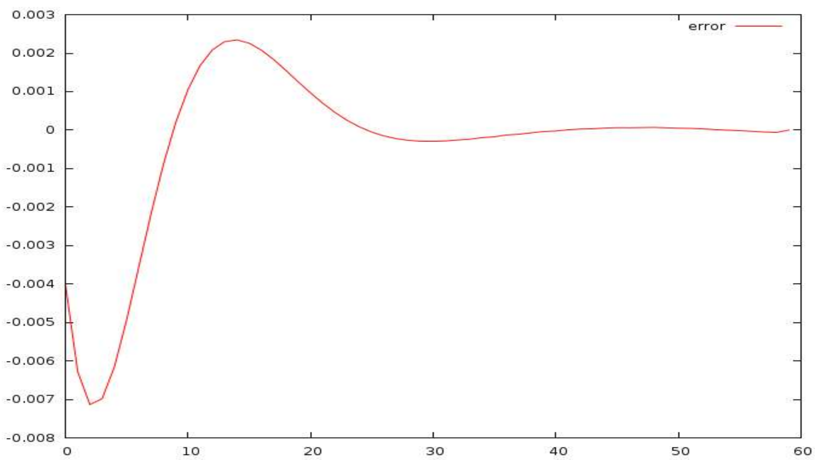

Figure 26 – Error Signal

4.3 LQI: Linear Quadratic with Integral Regulator

As an alternative to the LQR controller, here we also study the LQI variant to our controller. The LQI controllers (Linear Quadratic with Integral) are the static feedback controllers based not only on the system status, but also on the integral of the tracking error. They stabilize external circuits and minimize the quadratic cost functions associated with the step exogenous inputs, which is defined as follows:

$$J = \int_0^\infty (z^T Q z + u^T R u + 2z^T S u) dt$$

where $z = [x; x_i]$, x_i is the integrator state and Q and R dimensions must be adapted according to the matrix computation.

The control law in feedback, which minimizes the mentioned cost and ensures that the output follows the reference signal, is:

$$u = -K[x; x_i]$$

with K optimal gain.

The LQI version solves the LQR problem for the augmented system:

$$\begin{bmatrix} \dot{x} \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix} + \begin{bmatrix} B \\ -G \end{bmatrix} u$$

where G is the plant.

LQI control systems have been applied to many fields: motion and vibration control of the three-dimensional flexible vibrating table, man-machine robotic systems, electrodynamic suspension control, dynamic positioning of the floating platform and ocean shuttles, and direction control of unmanned submarine vehicles and navigation of tunnelling robots.

However, what we know is that the problem of synthesising the LQI controllers which stabilize the internal loops remains unsolved. With LQI control systems, external circuits are based on unit feedback and their stability is guaranteed by the LQI design. Furthermore, their open-loop stability is equivalent to the one of the internal loop. Therefore, in order to remove the fundamental constraints on the performance of external circuits caused by the open circuit in stability, it is always better if the LQI controllers stabilize the inner rings.

4.4 The Vehicle Controller

In the current case, considering the first treated Bicycle Model, it was convenient to study the behaviour of the errors and to base the problem on their minimization. Here, we apply the demonstration to the LQR case, but similar reasoning can be made about the LQI variant. Considering the structure of the linear system:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

and remembering the expression of the input for the LQR control:

$$u = -Kx$$

we can define the vectors involved in the studied problem:

$$x = \begin{bmatrix} y \\ \dot{y} \\ \Psi \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} y \\ V_y \\ \Psi \\ \dot{\Psi} \end{bmatrix} \qquad , \qquad u = \delta_f = \delta \qquad , \qquad y = x$$

In order to revising the model, aimed at basing the study on error minimization, here we define the error vectors:

$$e = x - x_d$$

$$v = u - u_d$$

It is demonstrable that it is possible to rewrite the first equation of the system by using these vectors, without altering the matrices described above, and obtaining the following law:

$$\dot{e} = Ae + Bv$$

where, according to the input structure of the LQR regulator, we obtain:

$$v = -Ke$$

$$u - u_d = -Ke$$

From here, the final equation for system input can be obtained:

$$u = -Ke + u_d$$

which, considering that $u = \delta_f = \delta$, and that δ_d is the value of the desired steering angle, it becomes:

$$\delta = -Ke + \delta_d$$

The vector e, whose elements are the lateral position error, the lateral velocity error, the yaw angle error and the yaw rate error, thus becomes the new vector of interest. Thanks to this new way of designing the model, the LQR controller, succeeding in itself in minimizing the cost function, will be able to make zero the above mentioned errors and to push the vehicle to follow the pre-established trajectory.

4.5 The State Observer

In the real applications, the state vector is not always directly measurable. When this happens, it is necessary to design a device that allows knowing the $x(t)$ behaviour through the knowledge of the initial conditions, the input and the output of the system.
This kind of state estimator is called Observer, and it is a dynamic system used to find, starting from $y(t)$ and $u(t)$, an estimation of the state, which is called $\hat{x}(t)$.

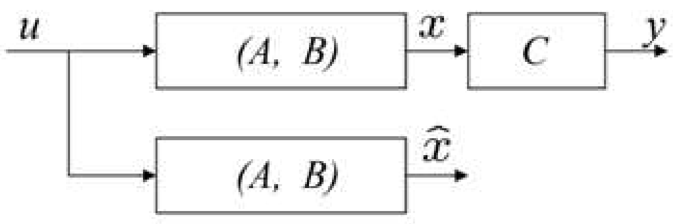

Figure 27 – Observer Scheme

The purpose of the Observer implementation is usually to make the estimation error tend to zero:

$$e = \hat{x} - x$$

$$\lim_{t\to\infty} \|e(t)\| = \lim_{t\to\infty} \|\hat{x}(t) - x(t)\| = 0$$

The error dynamics depends on the eigenvalues of the matrix A: if the system is unstable, the estimation diverges. This means that the error estimation goes to zero, for t tending to infinite, only if the system is asymptotically stable.

In order to take into account the output measurement, it is possible to add a correction factor, which depends on the real output and its estimation:

$$-L(\hat{y} - y)$$

where $L \in R^{nx1}$ is a matrix called Observer Gain matrix. In this way, the system becomes as follows:

$$\begin{cases} \dot{\hat{x}} = A\hat{x} + Bu - L(\hat{y} - y) \\ \hat{y} = C\hat{x} + Du \end{cases}$$

The estimation error will satisfy the following equation:

$$\dot{e} = (A - LC)e$$

It is necessary that the natural modes corresponding to the eigenvalues of $(A - LC)$ are convergent: this means that the system must be asymptotically stable, i.e. the eigenvalues of $(A - LC)$ must have negative real part.

The presence of the estimator does not affect the input-output relationship of the system: the transfer matrix of the system does not change using a feedback of the state x instead of the estimated state.

CHAPTER 5 – CONTROL DESIGN

5.1 An Overview

A control scheme manages, determines actions or regulates the behaviour of devices and systems, through continuous control cycles.

The control scheme adopted for the Path Following problem, as anticipated, allows checking the output errors at the feedback node, obtained from the difference between the desired values and the currently detected values.

For the following study, this difference concerns, in particular, the lateral deviation of the vehicle with respect to the desired trajectory. The control scheme was implemented on the Simulink platform, and the related code was linked to the scheme using MATLAB.

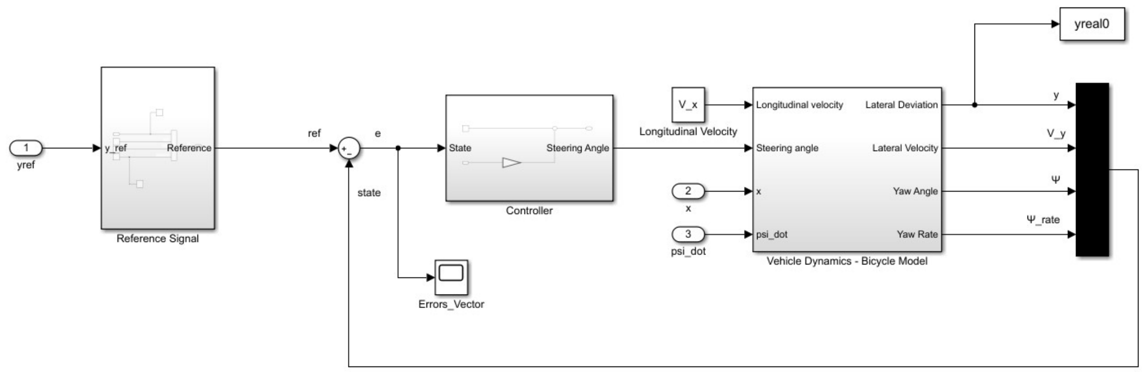

Figure 28 – Simulink LQR Control Scheme

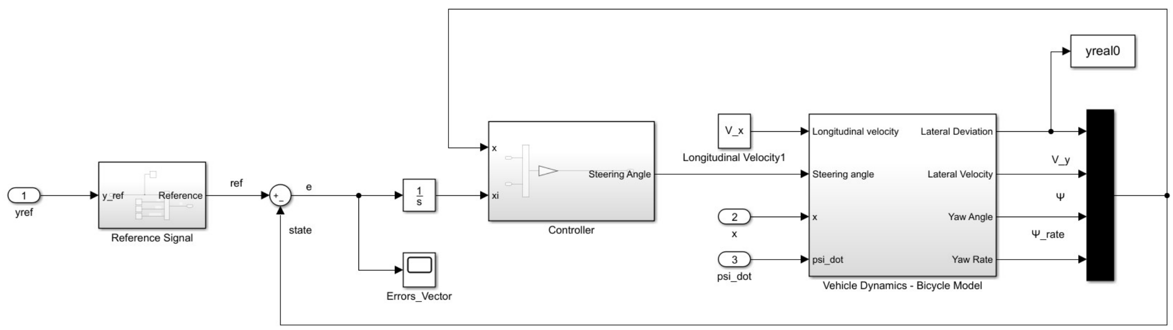

Figure 29 – Simulink LQI Control Scheme

As can be seen from the above diagrams, the system receives as input the lateral position of the vehicle in the body reference frame. It was necessary to distinguish the global reference system from the vehicle one, intended as the point of view of an observer who, positioned on the origin that the two systems share, rotates around the z-axis according to the vehicle, whose longitudinal axis will be parallel to the x-axis of the aforementioned relative system.
At this point, the input enters the block that will generate the reference signal, i.e. a vector of four elements placed in the following order:

$$Reference\ Signal = [y_{ref}, V_{y,ref}, \Psi_{ref}, \dot{\Psi}_{ref}]'$$

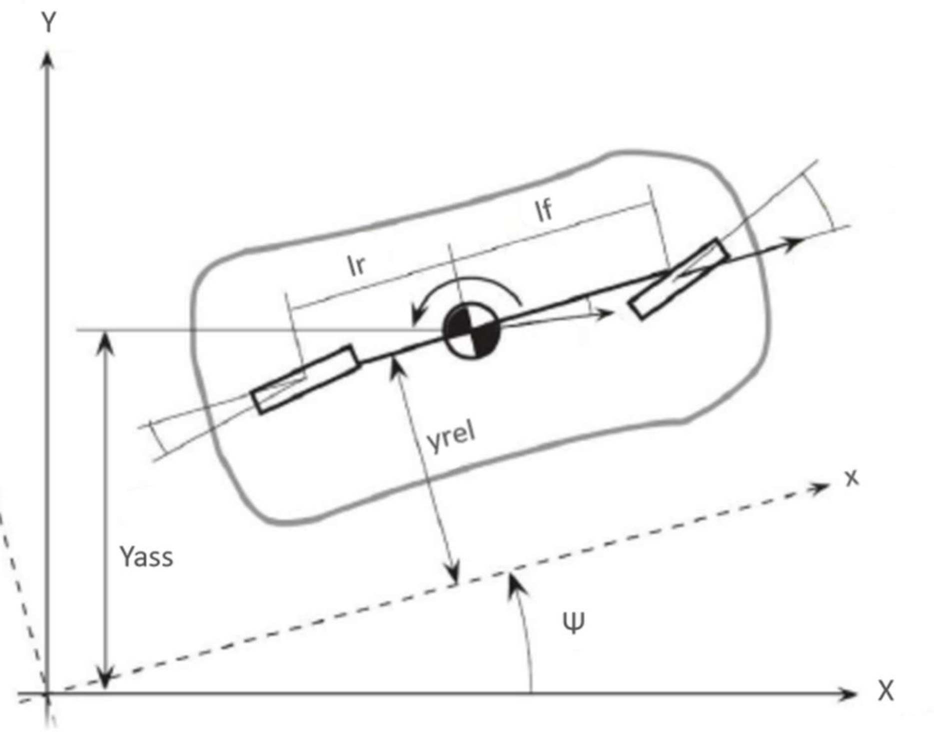

Figure 30 – Body RF

The reference signal is compared to the state vector generated at the output of the dynamics block, whose entity has been discussed in the previous chapters.
From this comparison, the error vector is obtained. This is the vector used by the controller to calculate the optimal steering angle.
Therefore, the control block will receive this vector and will elaborate the control parameters in order to generate in output the steering angle which, together with the predetermined longitudinal speed V_x, will constitute an input for the dynamics block.

Finally, the structure of the bicycle model, which is described inside the dynamics block, will process the state parameters using the equations of motion described in the previous chapters. The same considerations have been made for the schemes which includes the observer estimator.

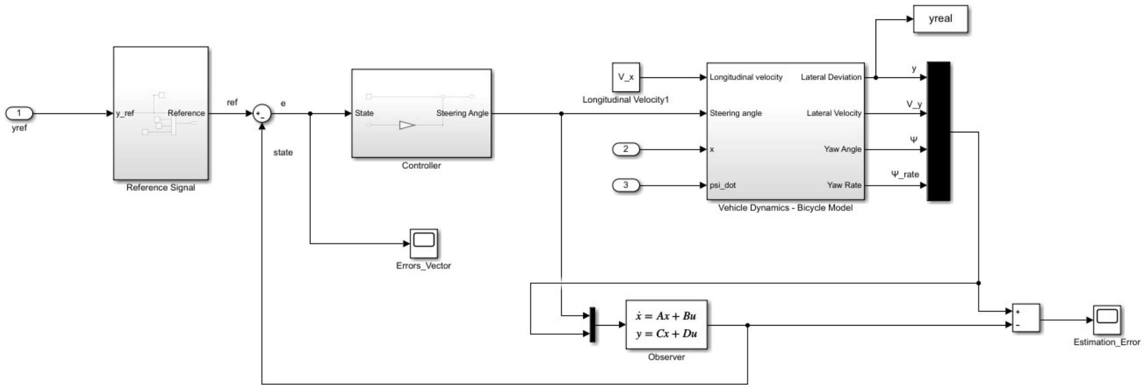

Figure 31 – Simulink LQR Control Scheme with Observer Estimator

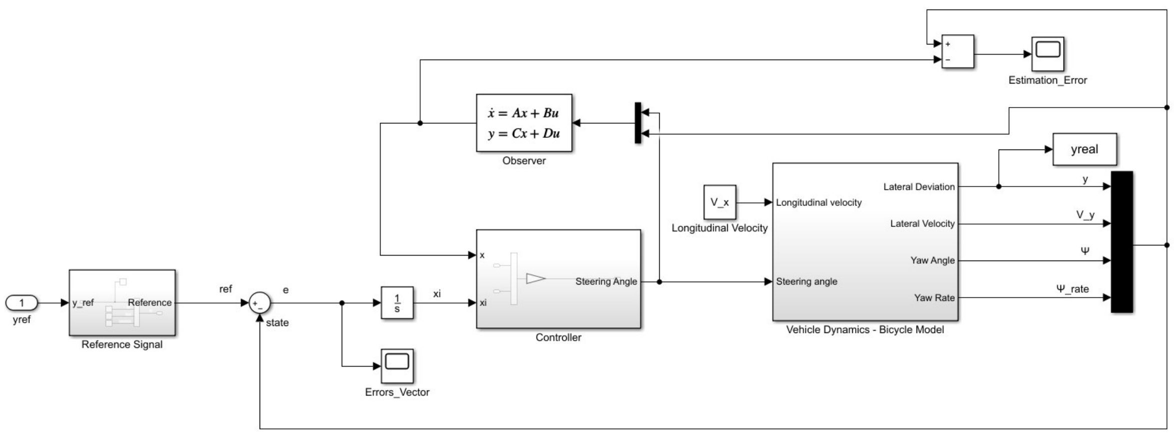

Figure 32 – Simulink LQI Control Scheme with Observer Estimator

5.2 The "Reference Signal" Block

The first interesting block of the schemes presented above is the one that generates the reference signal, which our system has to follow.

In order to simulate the described model, it was necessary to hypothesize the input parameters, i.e. to write a function that describes the trend of the (x,y) coordinates of a plausible trajectory.

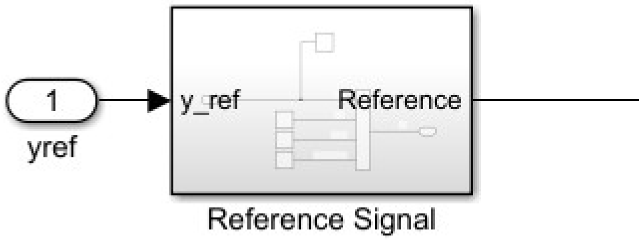

Figure 33 – Reference Signal Block

In particular, the current discussion studies four different paths, which are represented in the figures showed below.

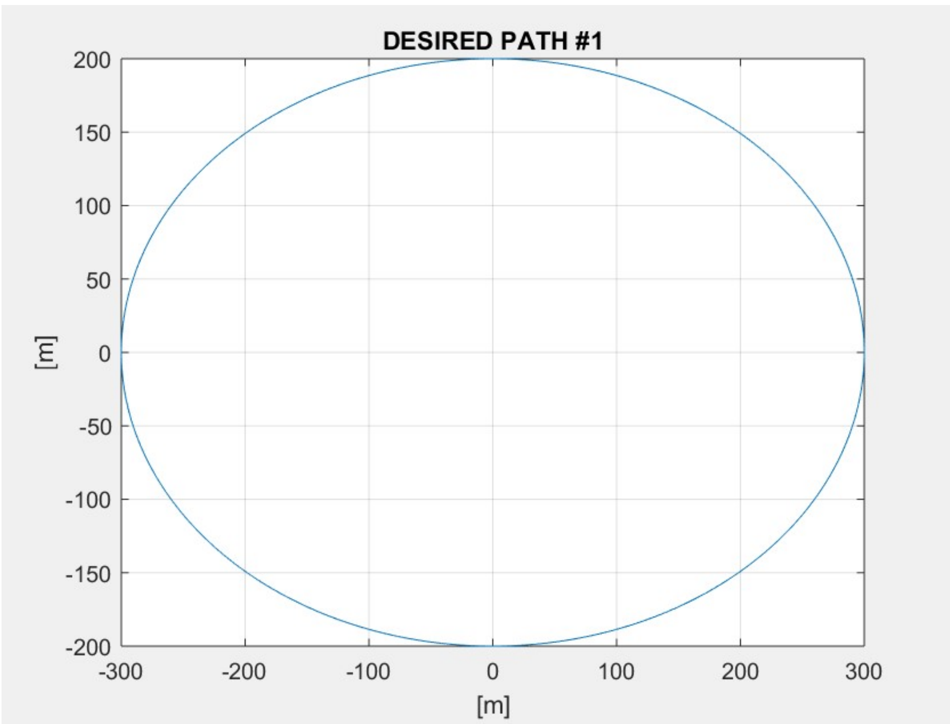

Figure 34 – Absolute Trajectory #1

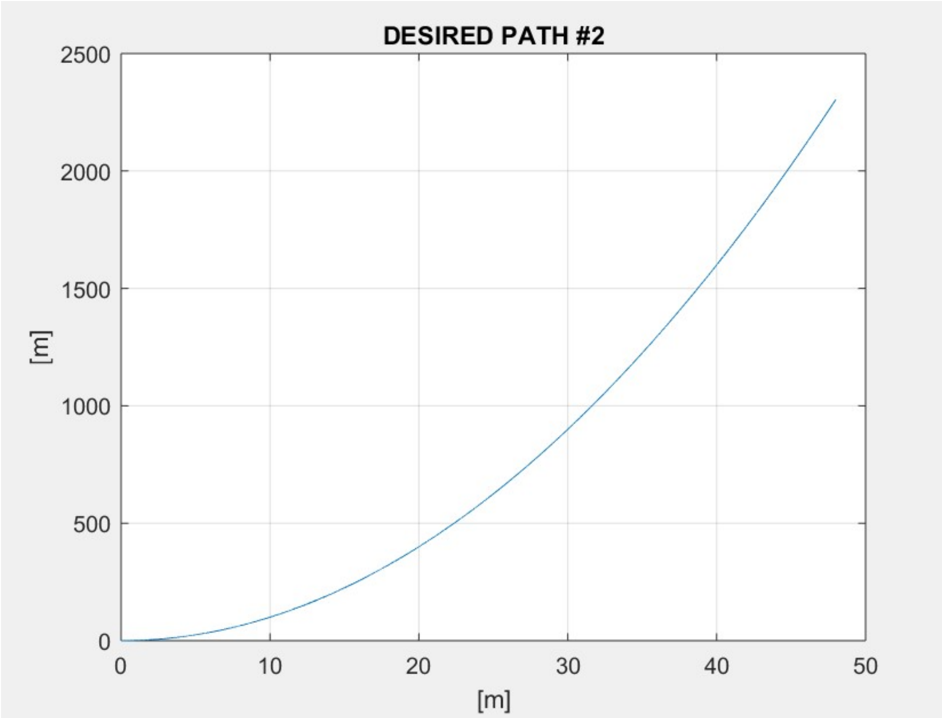

Figure 35 – Absolute Trajectory #2

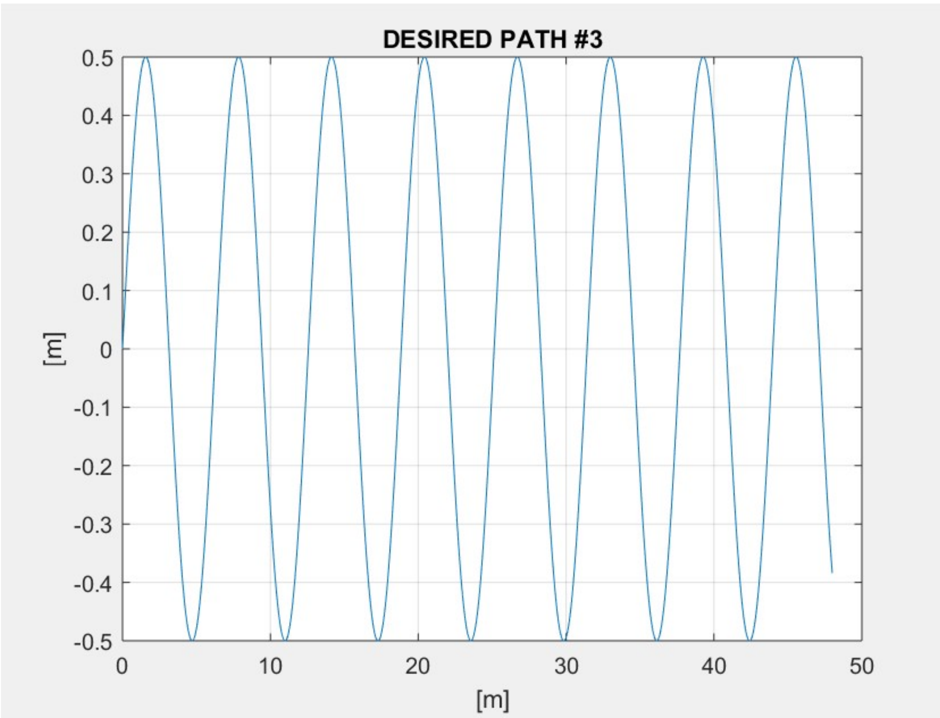

Figure 36 – Absolute Trajectory #3

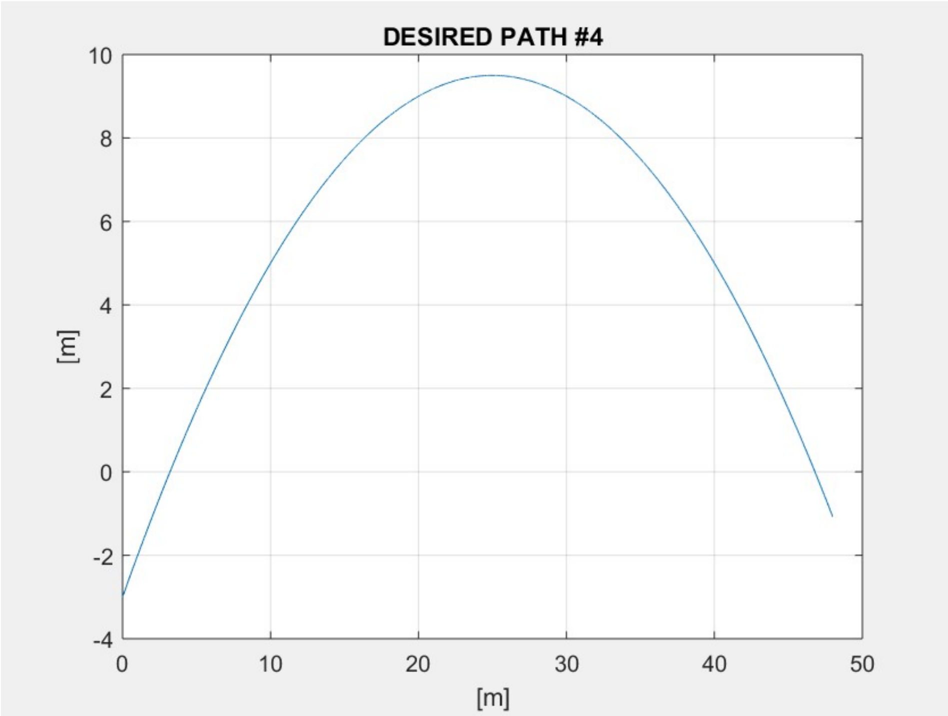

Figure 37 – Absolute Trajectory #4

Starting from this, a transformation was applied to obtain the (x,y) coordinates represented on the reference system which rotates firmly with the vehicle.

The yaw angle has been calculated starting from the inverse function of the tangent, using as argument the division between the ordinate and the abscissa of the current point of the trajectory:

$$\Psi = tg^{-1}\left(\frac{Y_p}{X_p}\right)$$

The yaw angle is useful in the calculation of the rotation matrix required for the transformation process: since these are two systems sharing the same axes origin, a simple rotation will be enough to express one reference frame as function of the other one.

Given a vector $\vec{p}$, represented in an absolute reference system, it is possible to express its coordinates in another reference system, rotated at an α angle with respect to the first one, in the following way:

$$\overrightarrow{p'} = R^T\vec{p}$$

where the matrix R is described as follows:

$$R = \begin{bmatrix} cos\alpha & -sin\alpha & 0 \\ sin\alpha & cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

It is possible to obtain the desired relative behaviour by replacing α with Ψ.

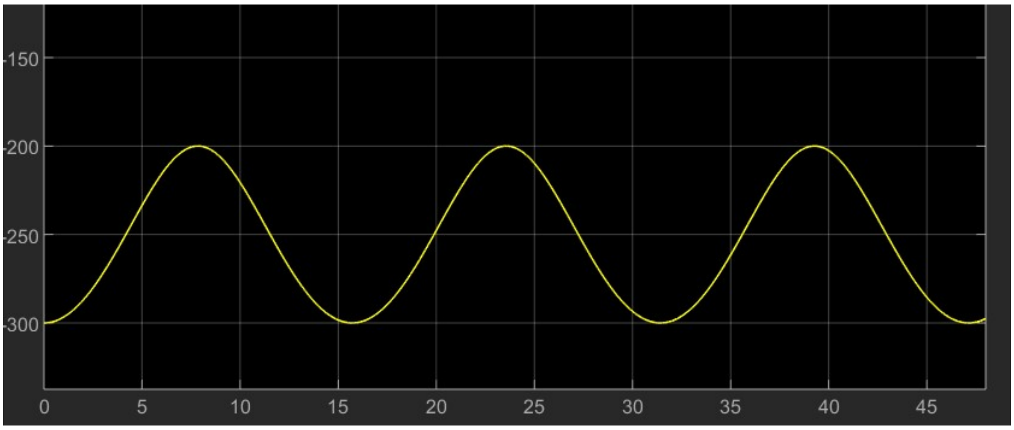

Figure 38 – Relative Trajectory #1

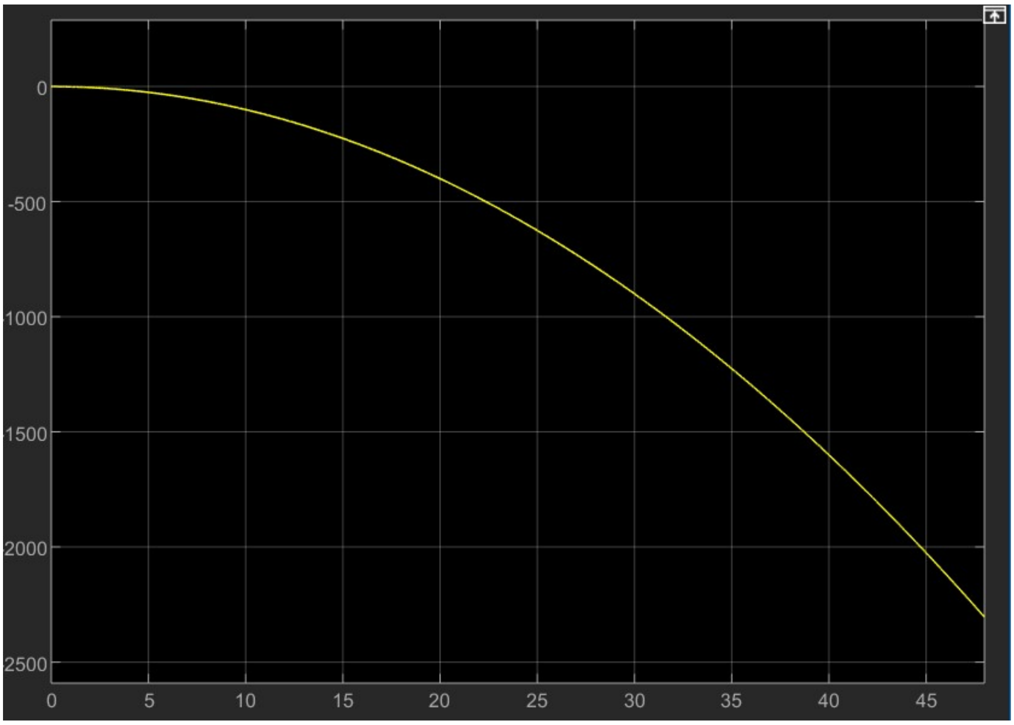

Figure 39 – Relative Trajectory #2

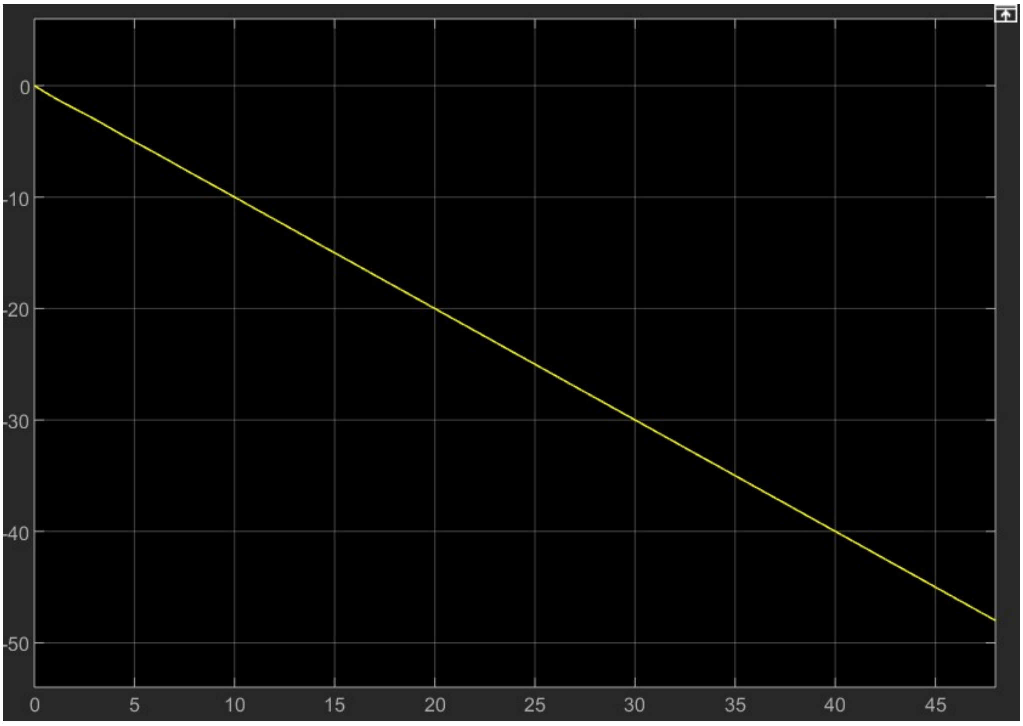

Figure 40 – Relative Trajectory #3

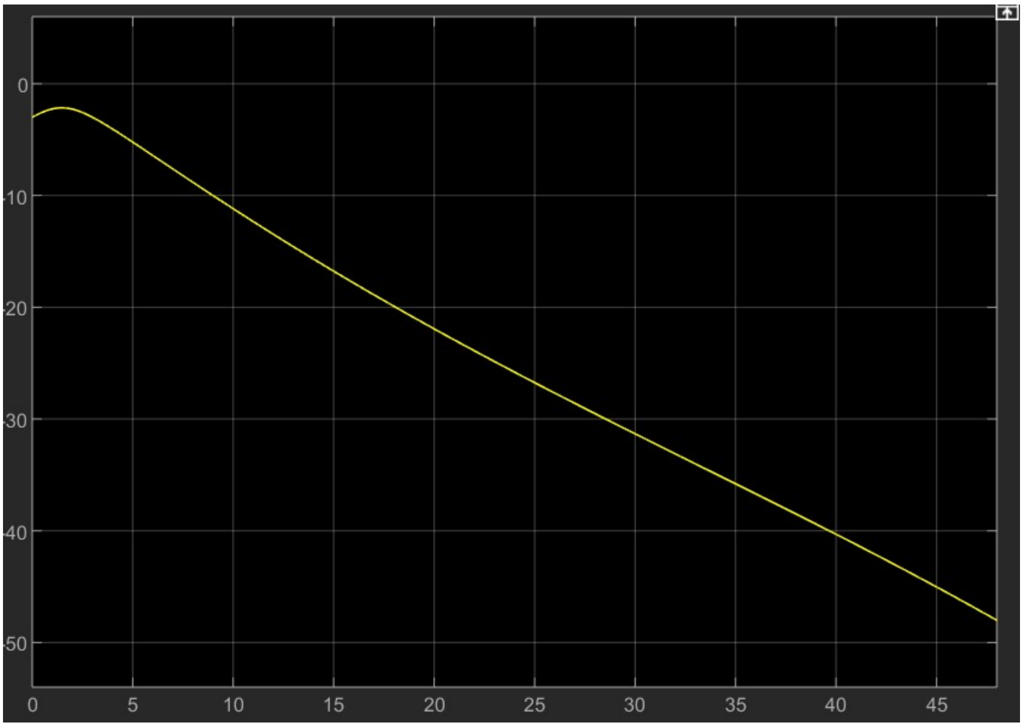

Figure 41 – Relative Trajectory #4

5.3 The “Vehicle Dynamics – Bicycle Model” Block

Regarding vehicle dynamics, it has already been stated that this will be approximated to the dynamics of the bicycle model. The block in question receives the longitudinal velocity, which is a constant value, and the steering angle processed by the controller. The other input parameters are useful to facilitate the computation process.

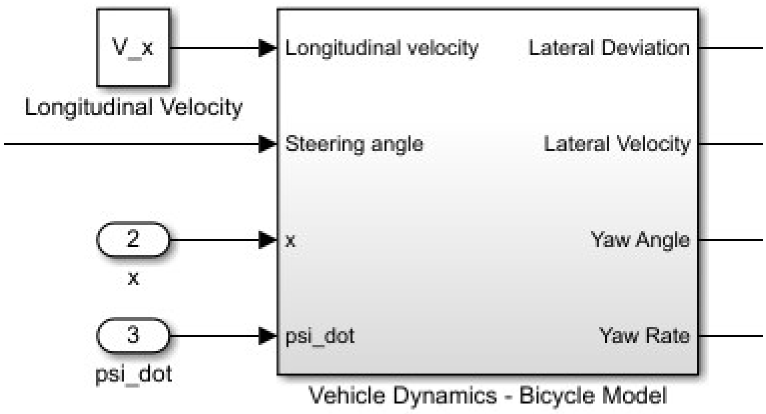

Figure 42 – Dynamics Block

The internal structure of the block was designed to reproduce the state equations presented in the previous chapters.

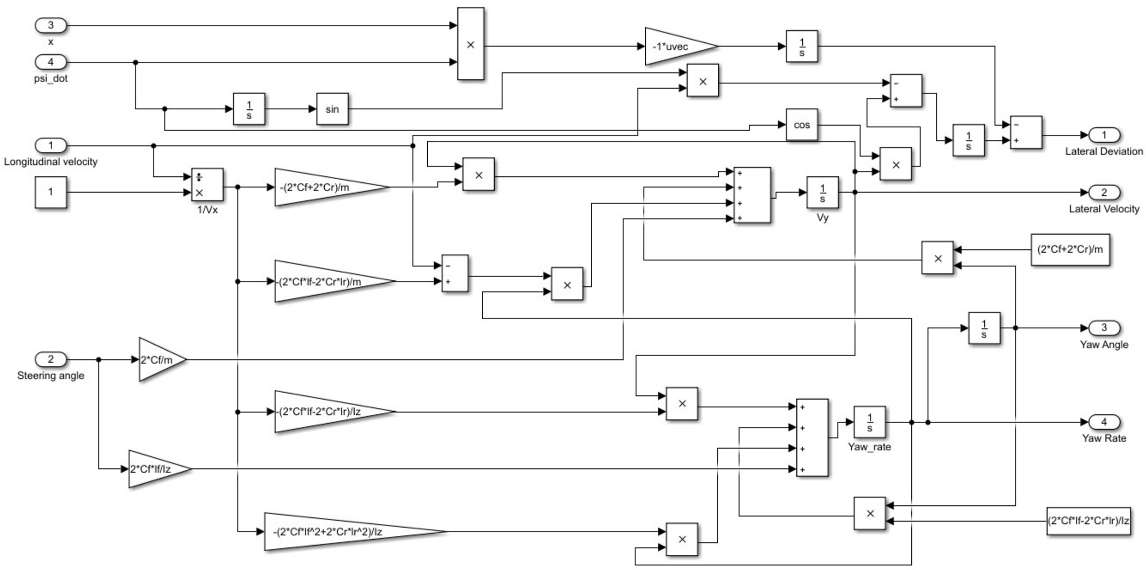

Figure 43 – Bicycle Model Block

5.4 The Controller Block

The most relevant part of the scheme is the controller: from the parameters characterizing the controller, the input of the system is determined, and the same input is the one that will minimize the cost function by making the deviations tend to zero.

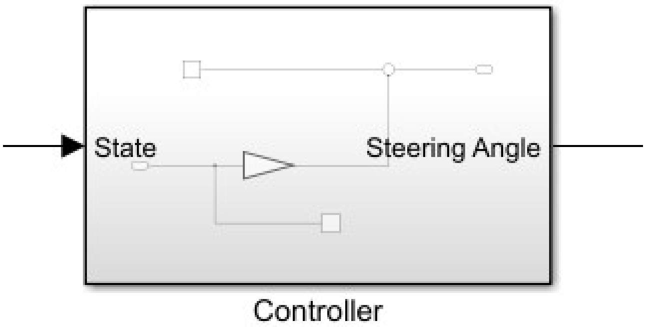

Figure 44 – Control Block

The internal structural is showed below.

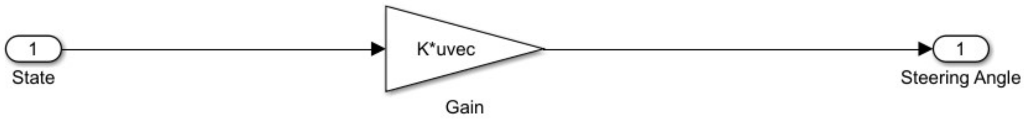

Figure 45 – Controller

CHAPTER 6 – VIRTUAL SIMULATIONS

6.1 The Simulation Approach

For the performance of the simulations, some cases have been considered, which are different for the weight values of the Q matrix. For the LQR variant, four scenarios have been generated, both with and without the state observer. For the LQI variant, however, it was not possible to consider more than one case, since only one of the weights combinations considered could stabilize the system.

To better appreciate the obtained results, it was necessary to go backwards in the calculation of the resulting trajectory: starting from the vehicle coordinates obtained as output of the plant, which are referred to the body system, it was possible to trace the coordinates in the absolute system through the inverse transformation. In this way, it was possible to compare the real trajectories with the reference ones.

6.2 Tests and Results

LQR WITHOUT THE STATE OBSERVER

- Case 1

$$Q = \begin{bmatrix} 1000 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix} \ , \ R = 0.001 \ , \ K = [1000 \quad 99.3 \quad 13.4 \quad 2.3]$$

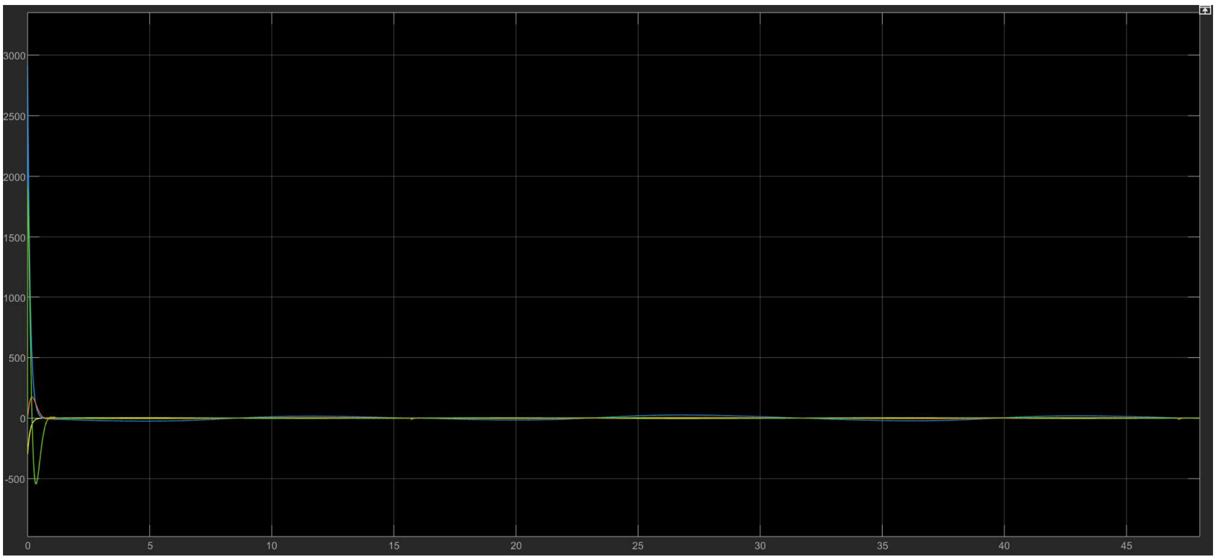

Figure 46 – Case 1: Error Vector

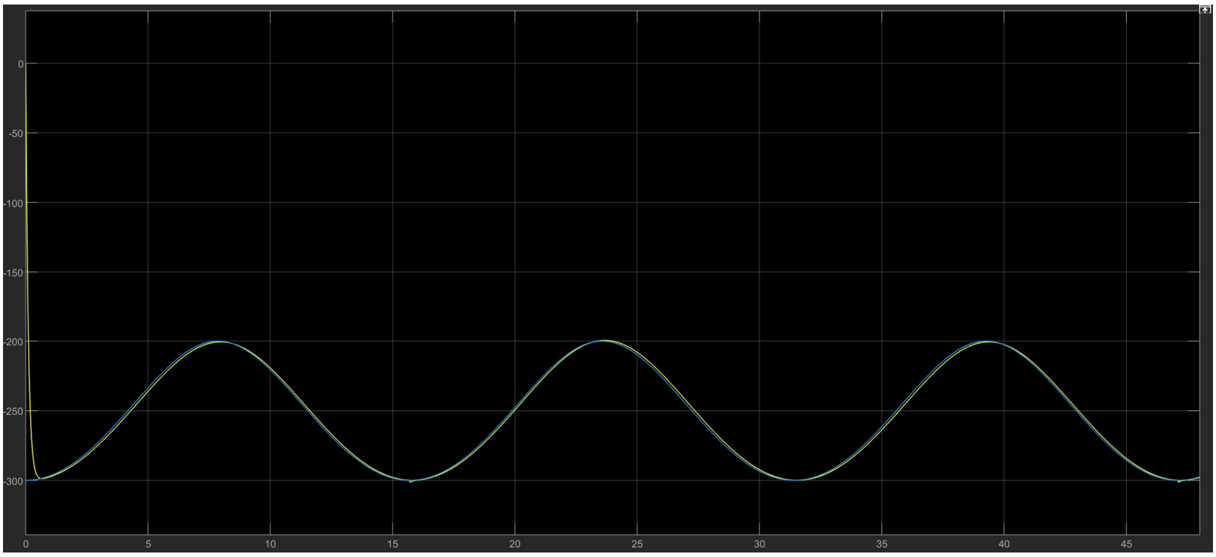

Figure 47 – Case 1: Real y VS Desired y

- <u>Case 2</u>

$$Q = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.05 \end{bmatrix} \quad , \quad R = 0.001 \quad , \quad K = [316.2 \quad 31.3 \quad 5.8 \quad 0.7]$$

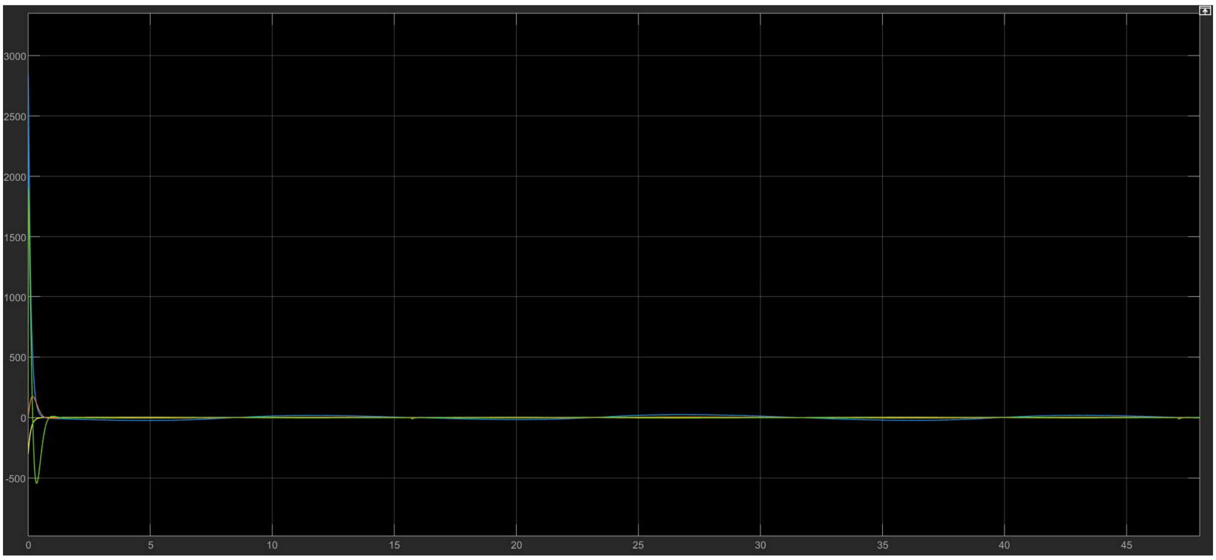

Figure 48 – Case 2: Error Vector

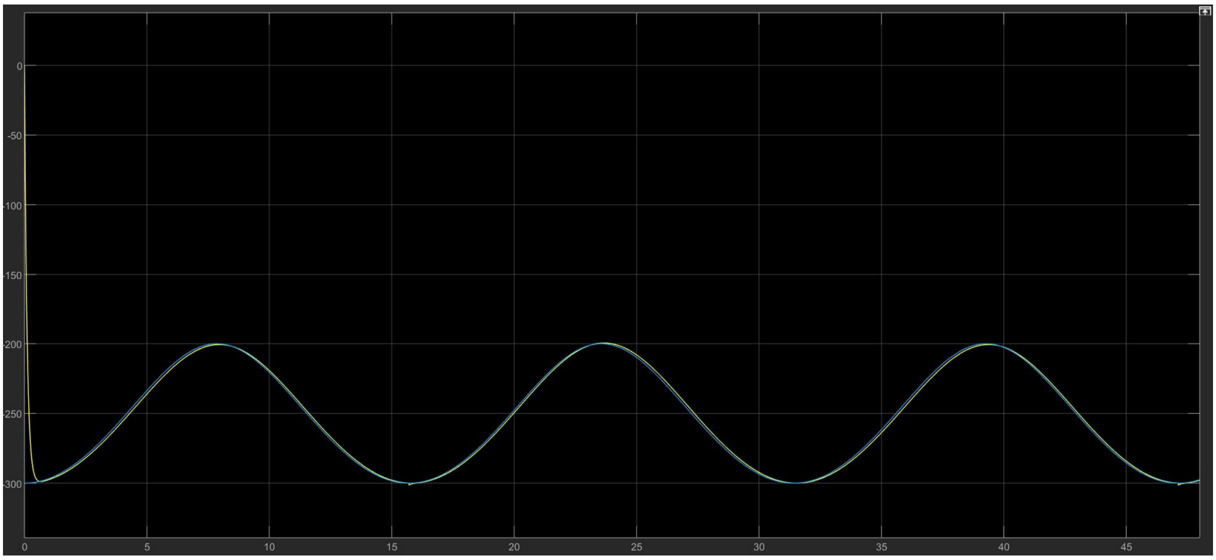

Figure 49 – Case 2: Real y VS Desired y

- <u>Case 3</u>

$$Q = \begin{bmatrix} 500 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} \;,\; R = 0.001 \;,\; K = [707.1 \quad 70.4 \quad 5.5 \quad 0.6]$$

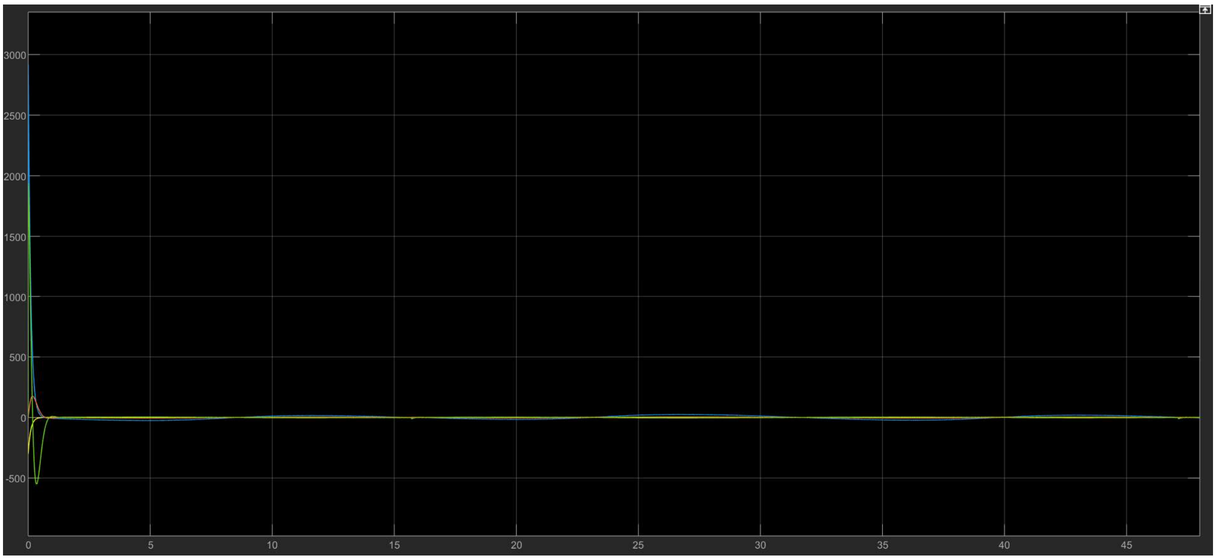

Figure 50 – Case 3: Error Vector

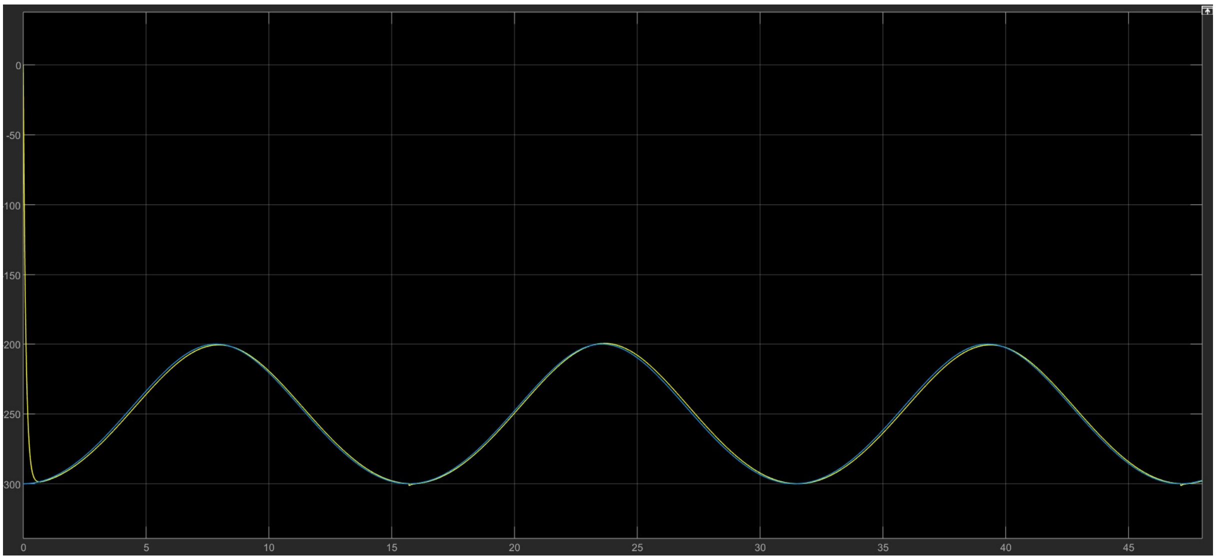

Figure 51 – Case 3: Real y VS Desired y

- <u>Case 4</u>

$$Q = \begin{bmatrix} 50 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} \quad , \quad R = 0.001 \quad , \quad K = [223.6 \quad 22.1 \quad 3.3 \quad 0.2]$$

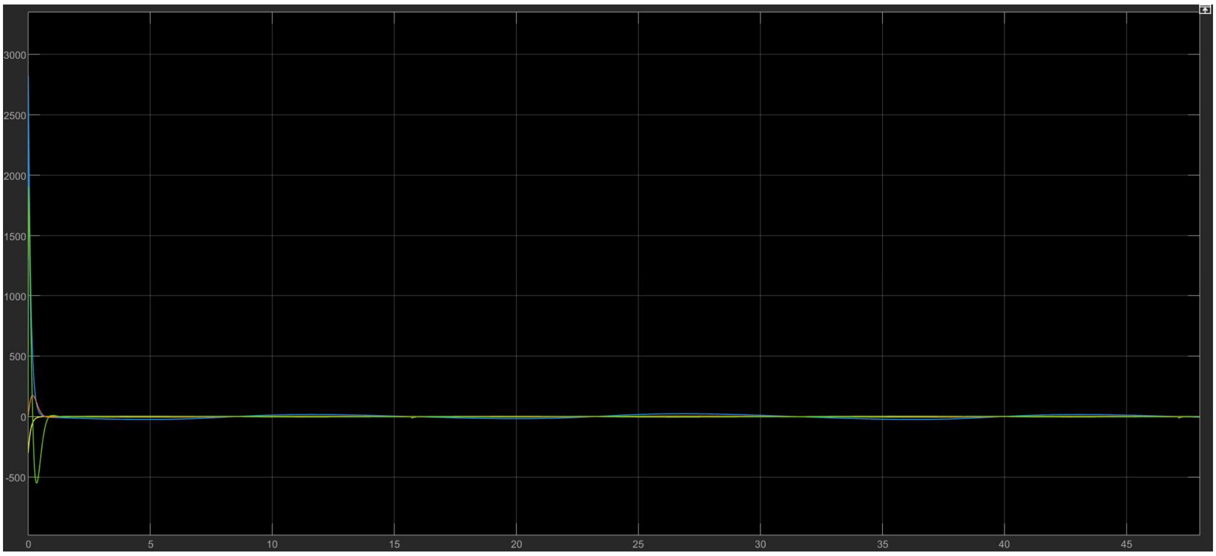

Figure 52 – Case 4: Error Vector

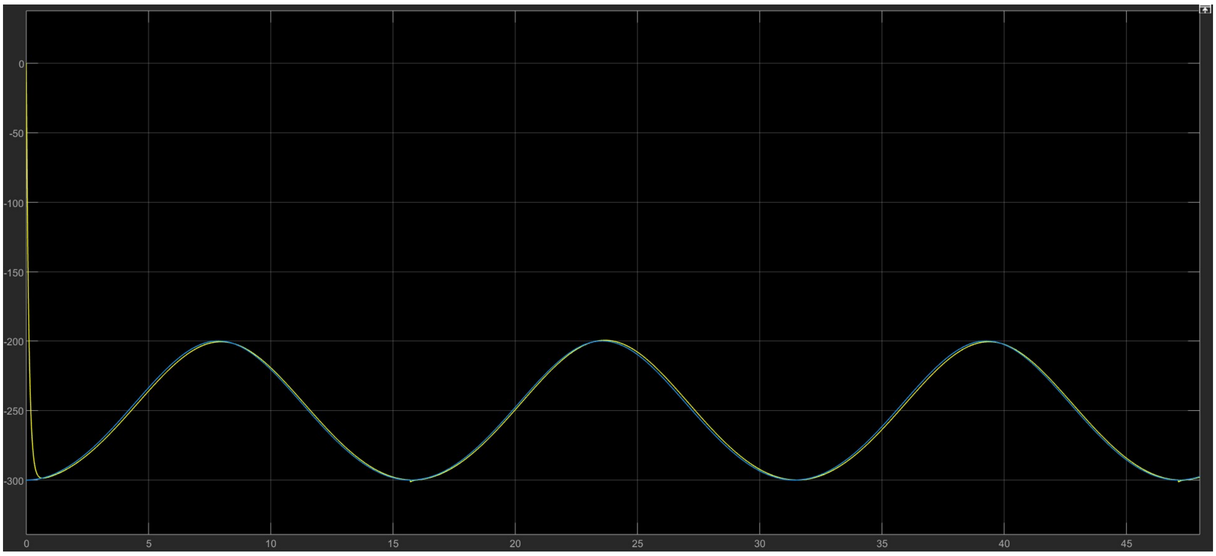

Figure 53 – Case 4: Real y VS Desired y

LQR WITH THE STATE OBSERVER

- Case 1

$$Q = \begin{bmatrix} 1000 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix} \quad , \quad R = 0.001 \quad , \quad K = [1000 \quad 99.3 \quad 13.4 \quad 2.3]$$

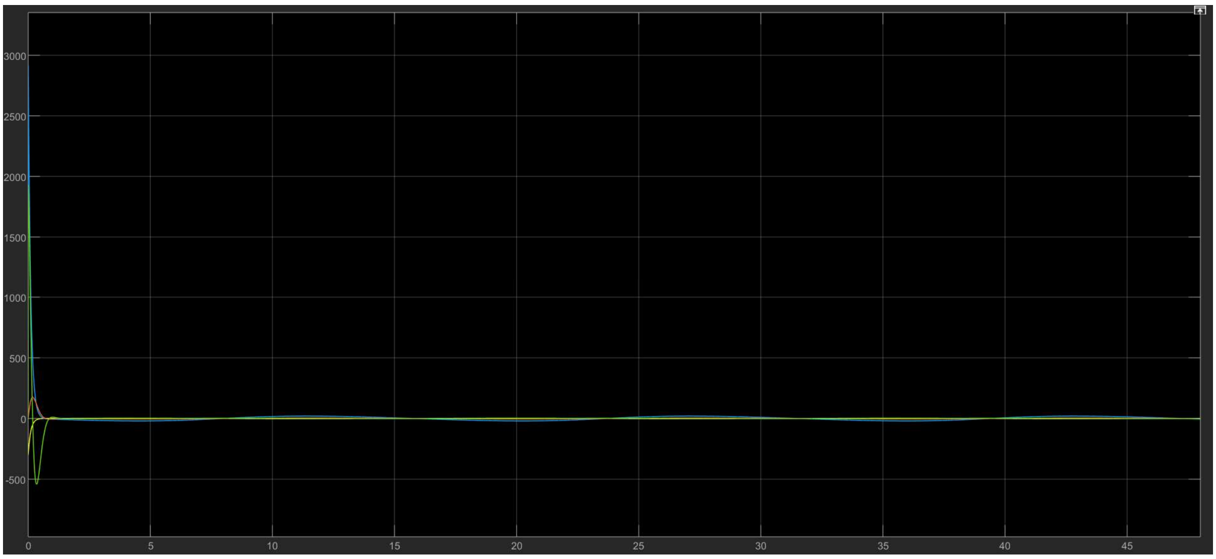

Figure 54 – State Observer Case 1: Error Vector

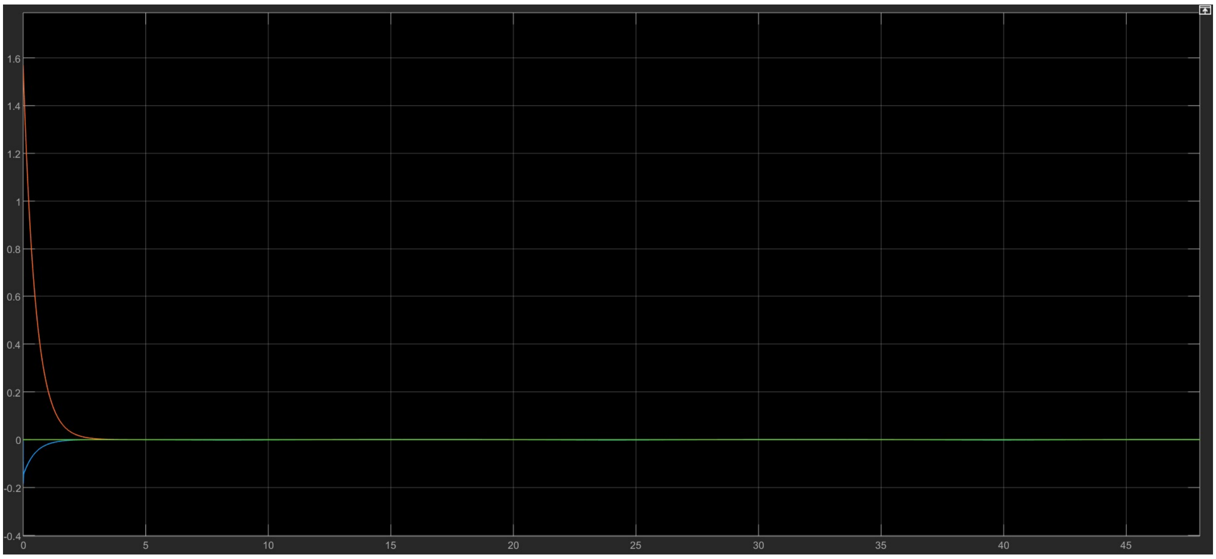

Figure 55 – State Observer Case 1: Estimation Error

- <u>Case 2</u>

$$Q = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.05 \end{bmatrix} \quad , \quad R = 0.001 \quad , \quad K = [316.2 \quad 31.3 \quad 5.8 \quad 0.7]$$

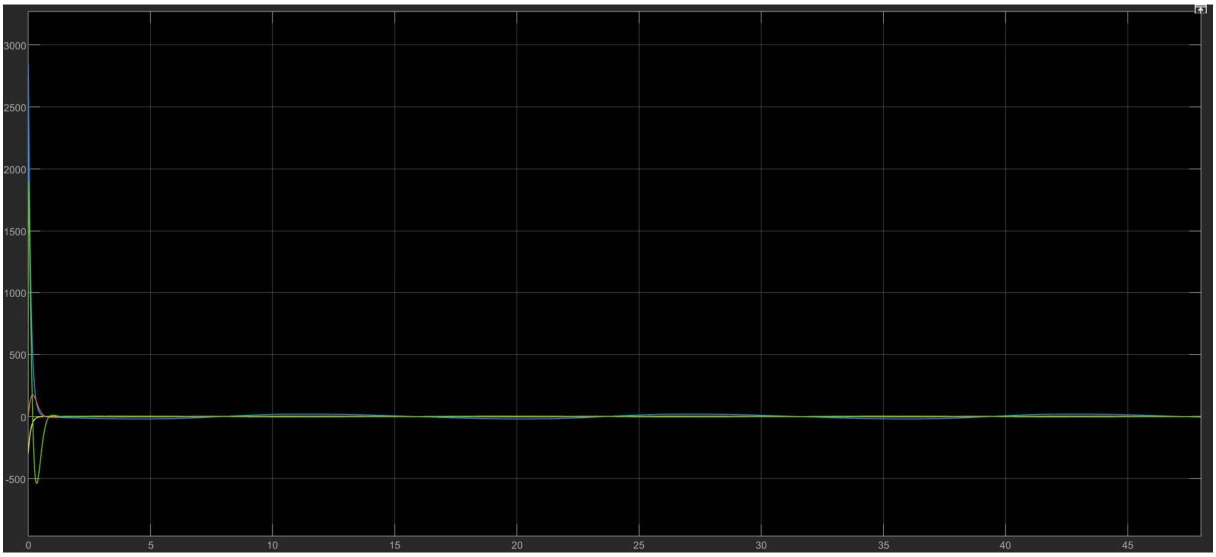

Figure 56 – State Observer Case 2: Error Vector

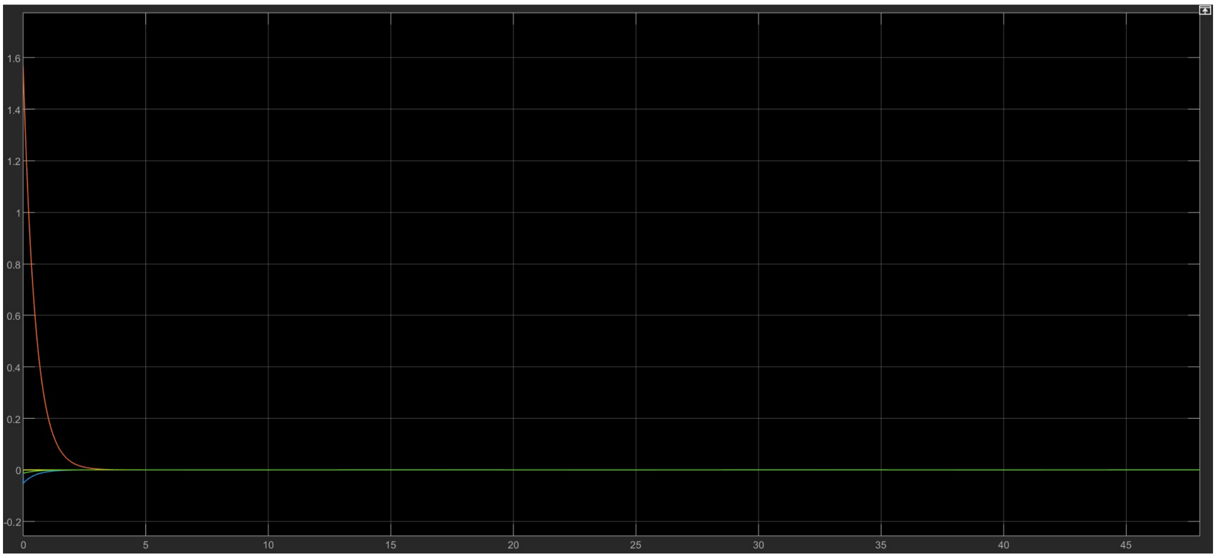

Figure 57 – State Observer Case 2: Estimation Error

- <u>Case 3</u>

$$Q = \begin{bmatrix} 500 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} \quad , \quad R = 0.001 \quad , \quad K = [707.1 \quad 70.4 \quad 5.5 \quad 0.6]$$

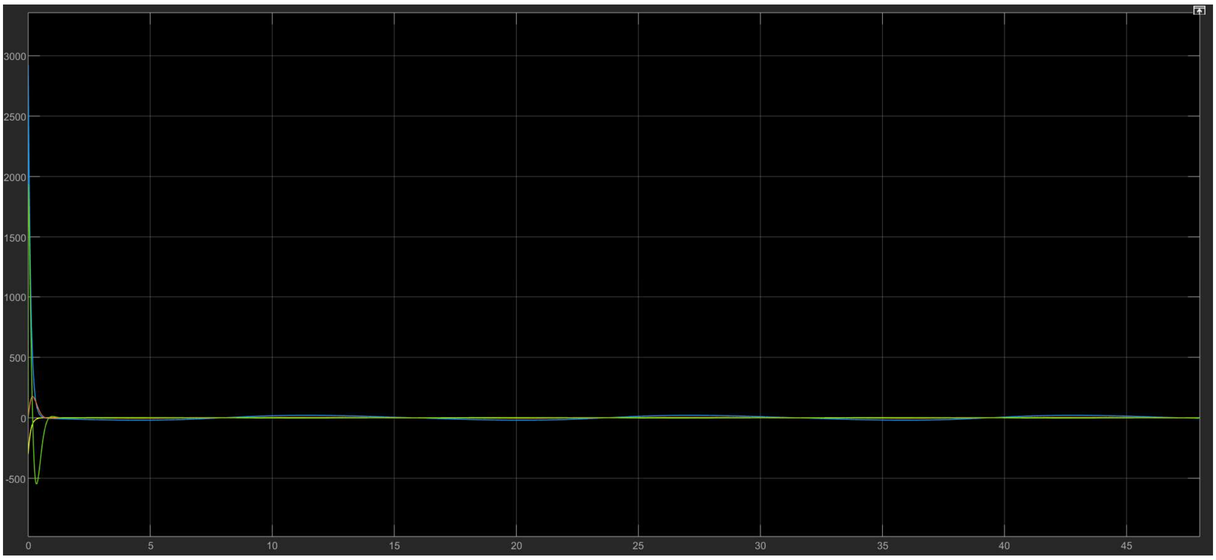

Figure 58 – State Observer Case 3: Error Vector

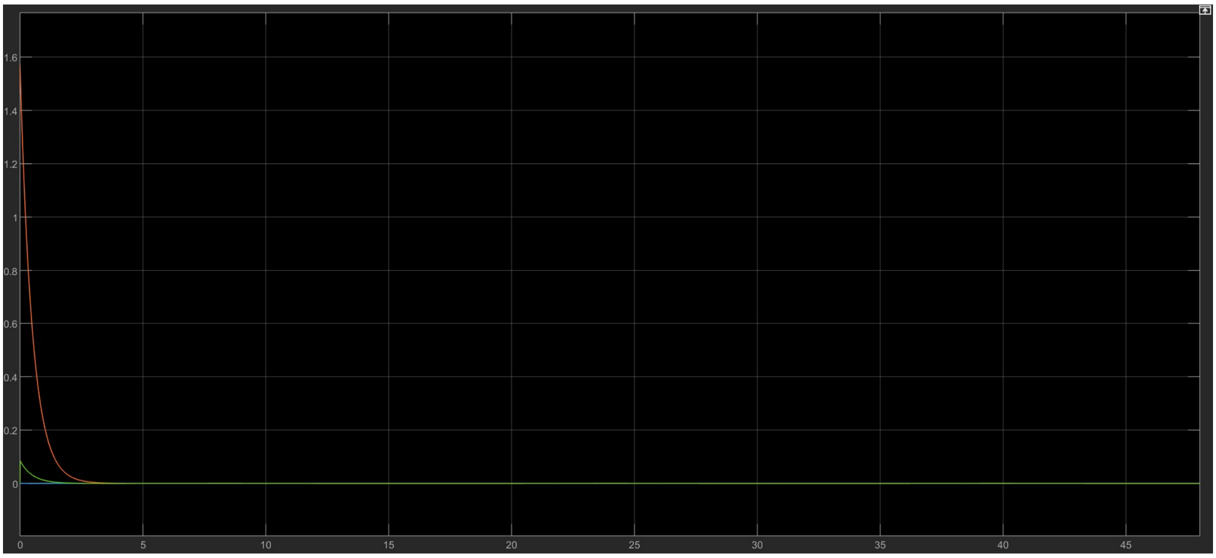

Figure 59 – State Observer Case 3: Estimation Error

- Case 4

$$Q = \begin{bmatrix} 50 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} \quad , \quad R = 0.001 \quad , \quad K = [223.6 \quad 22.1 \quad 3.3 \quad 0.2]$$

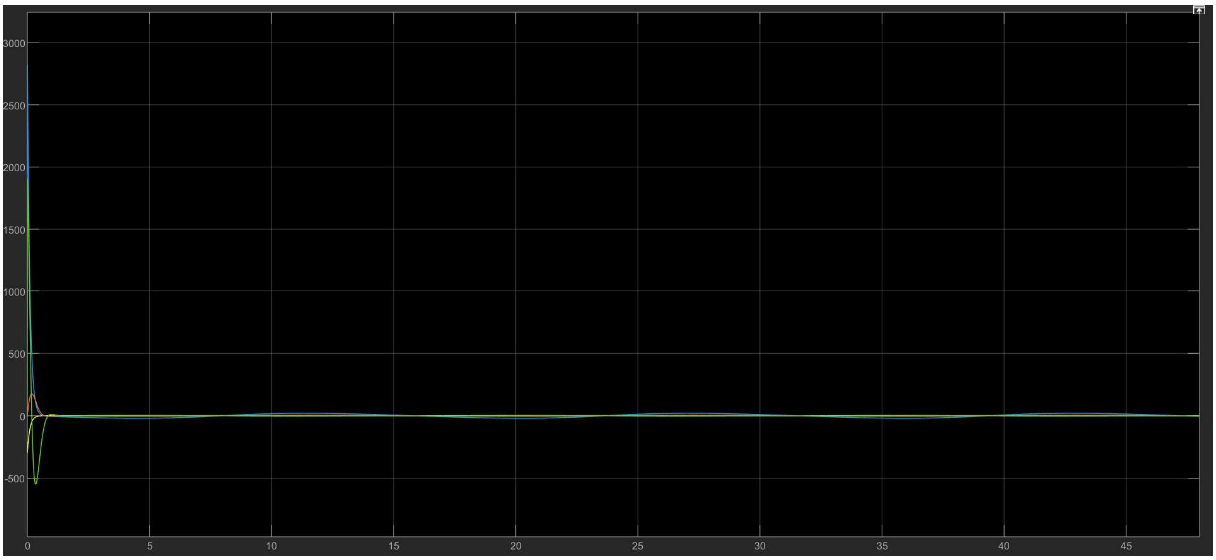

Figure 60 – State Observer Case 4: Error Vector

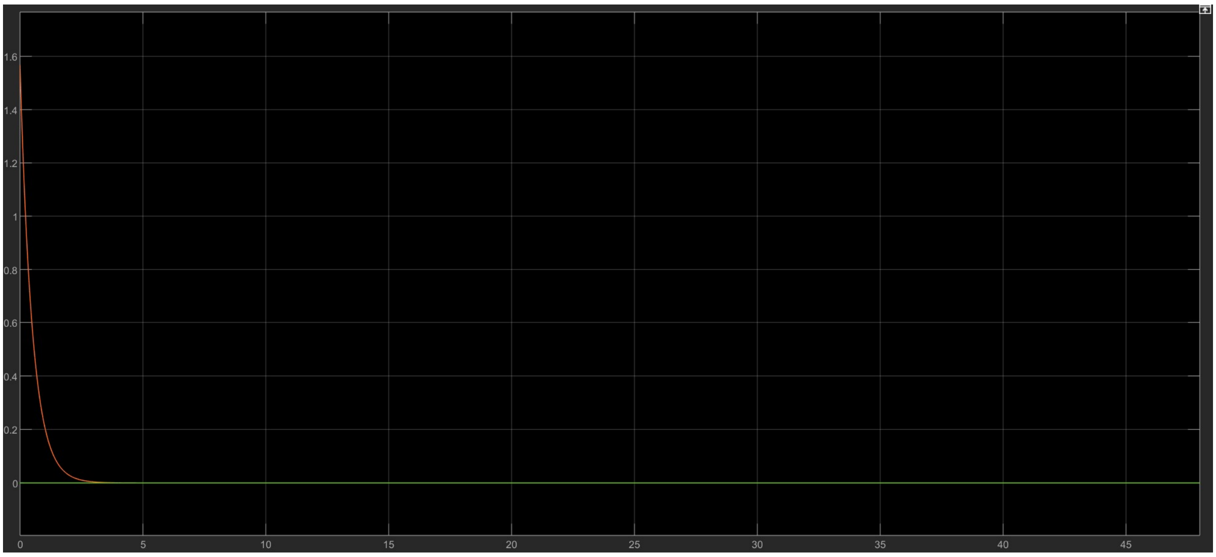

Figure 61 – State Observer Case 4: Estimation Error

LQR BEST RESULTING TRAJECTORIES

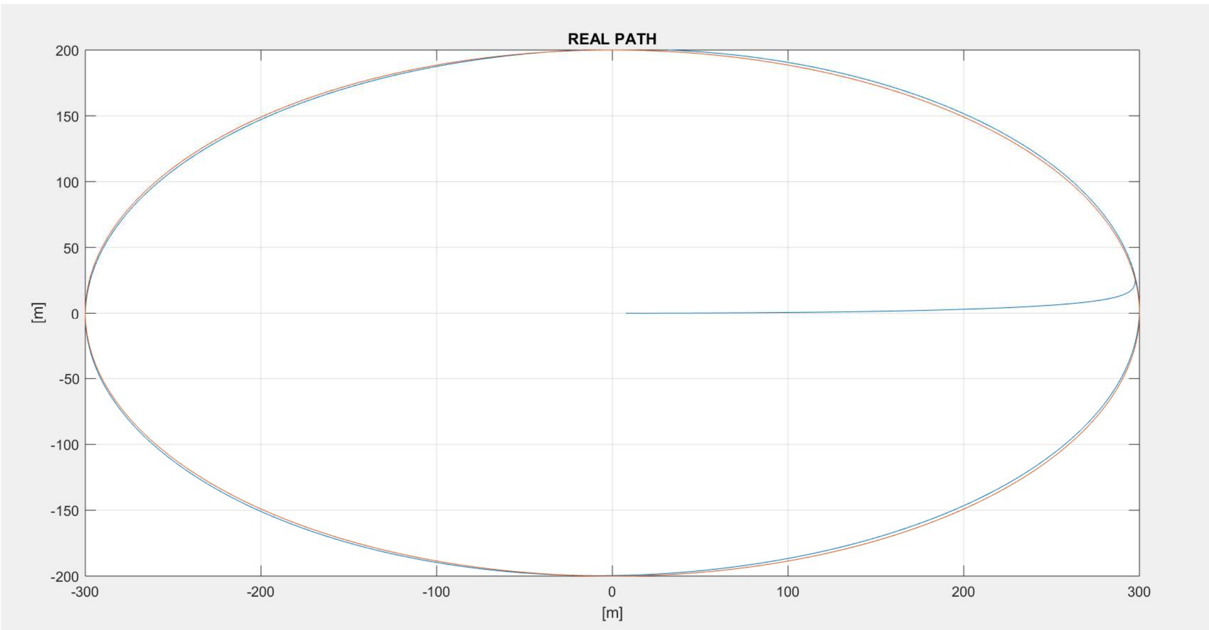

Figure 62 – LQR Trajectory #1

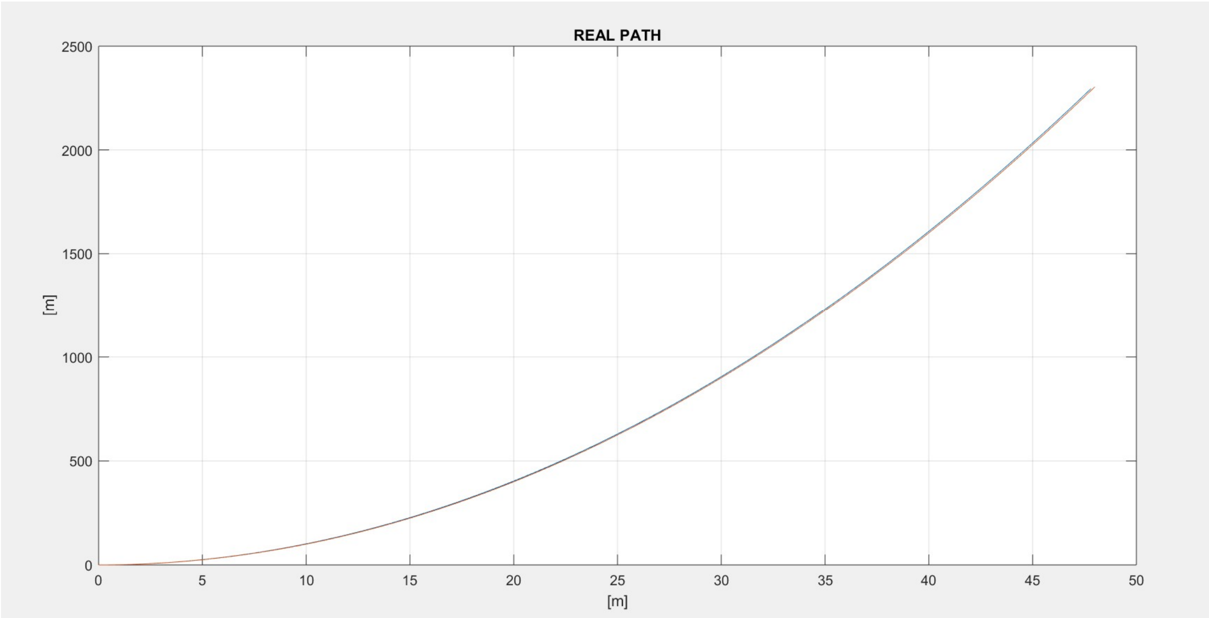

Figure 63 – LQR Trajectory #2

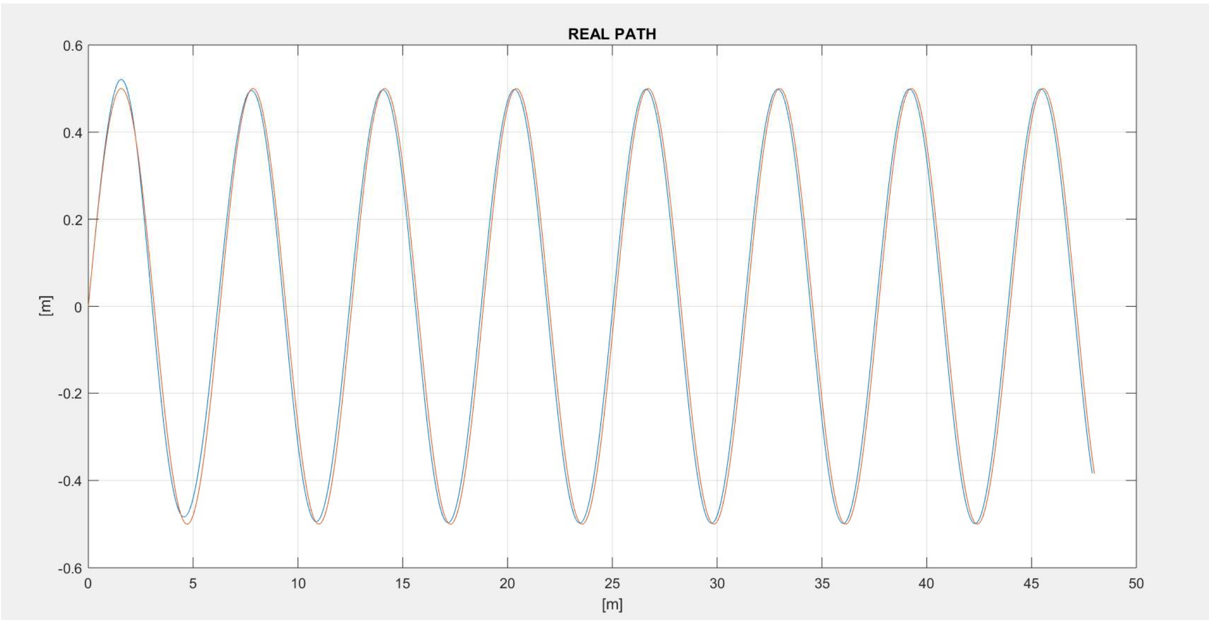

Figure 64 – LQR Trajectory #3

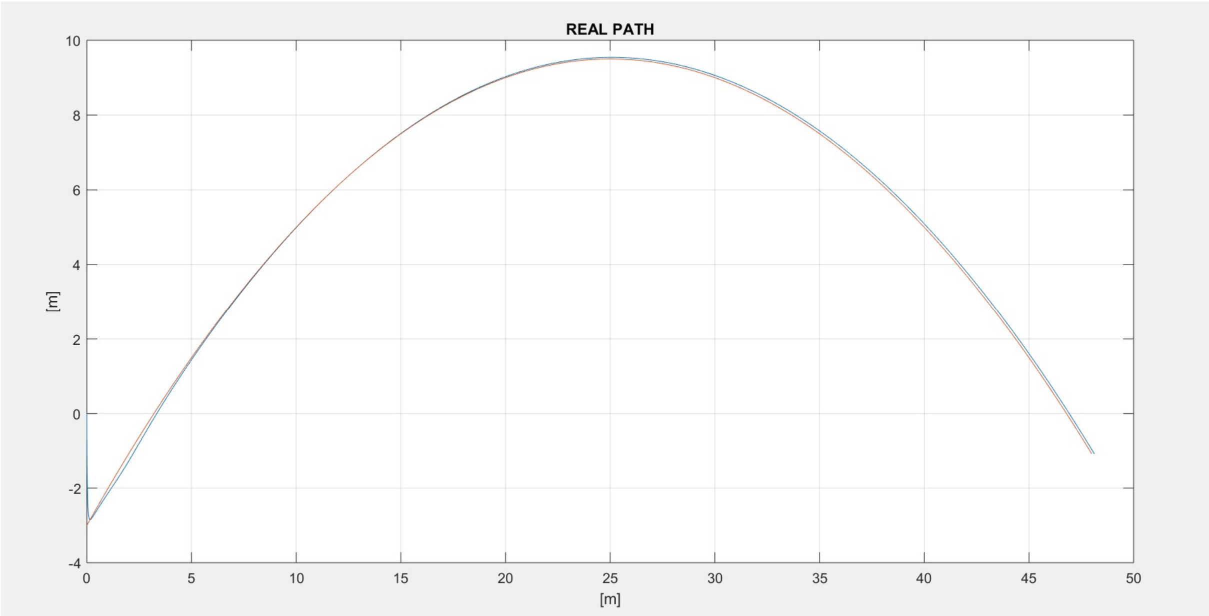

Figure 65 – LQR Trajectory #4

LQI VARIANT

- Case 1

$$Q = \begin{bmatrix} 5000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 5500 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4500 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3500 \end{bmatrix}$$

$$R = 500$$

$$K = [6.6 \quad 2.2 \quad 5 \quad 0.8 \quad -3.5 \quad 0 \quad 0 \quad 0]$$

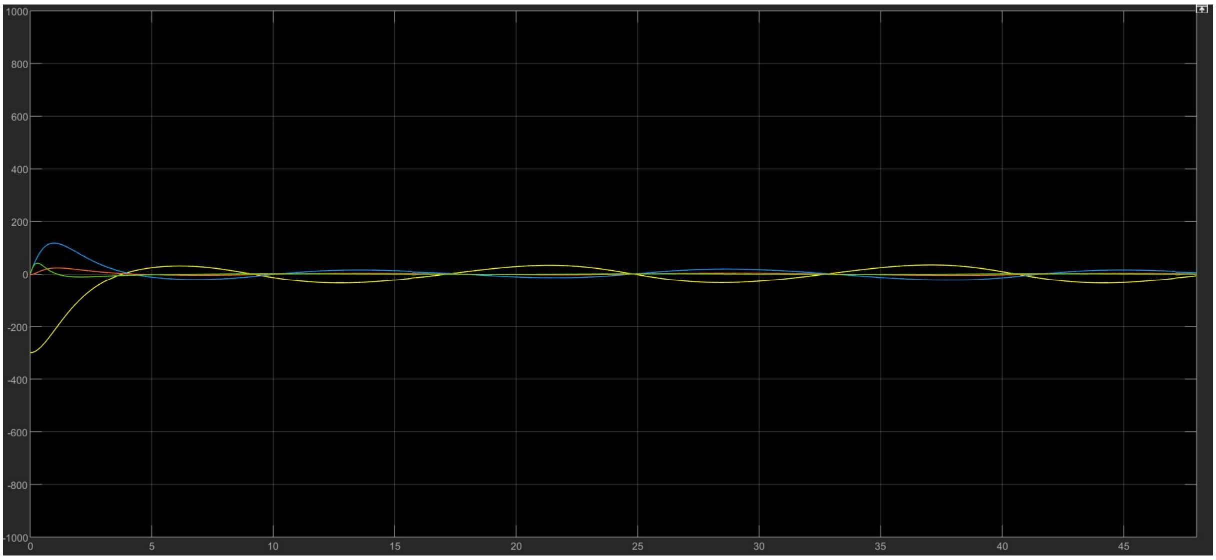

Figure 66 – LQI Case: Error Vector

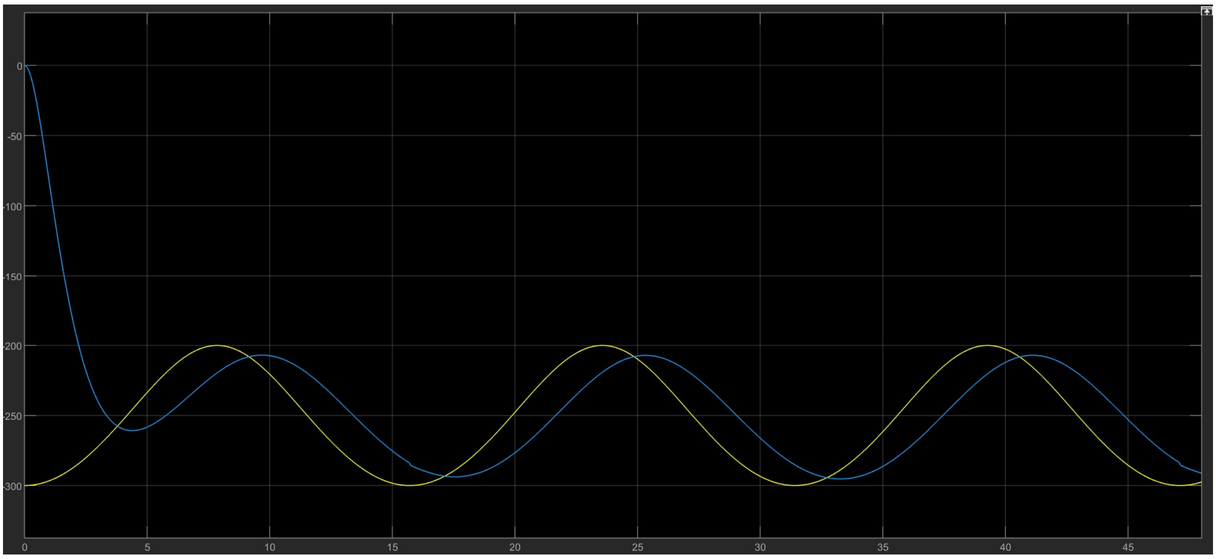

Figure 67 – LQI Case: Real y VS Desired y

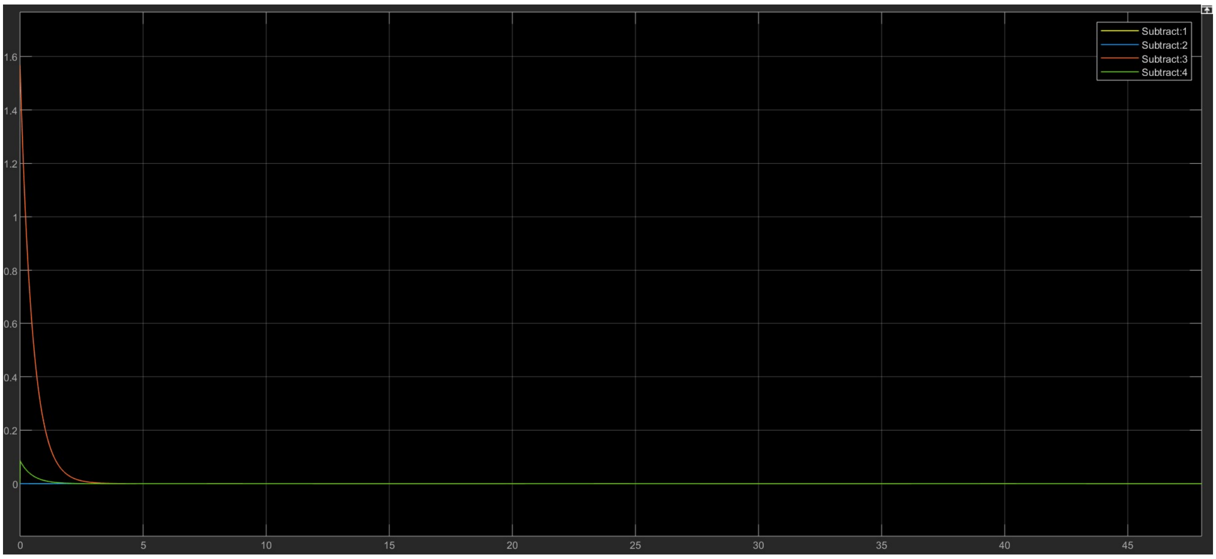

Figure 68 – LQI Observer Case: Estimation Error

LQI BEST RESULTING TRAJECTORIES

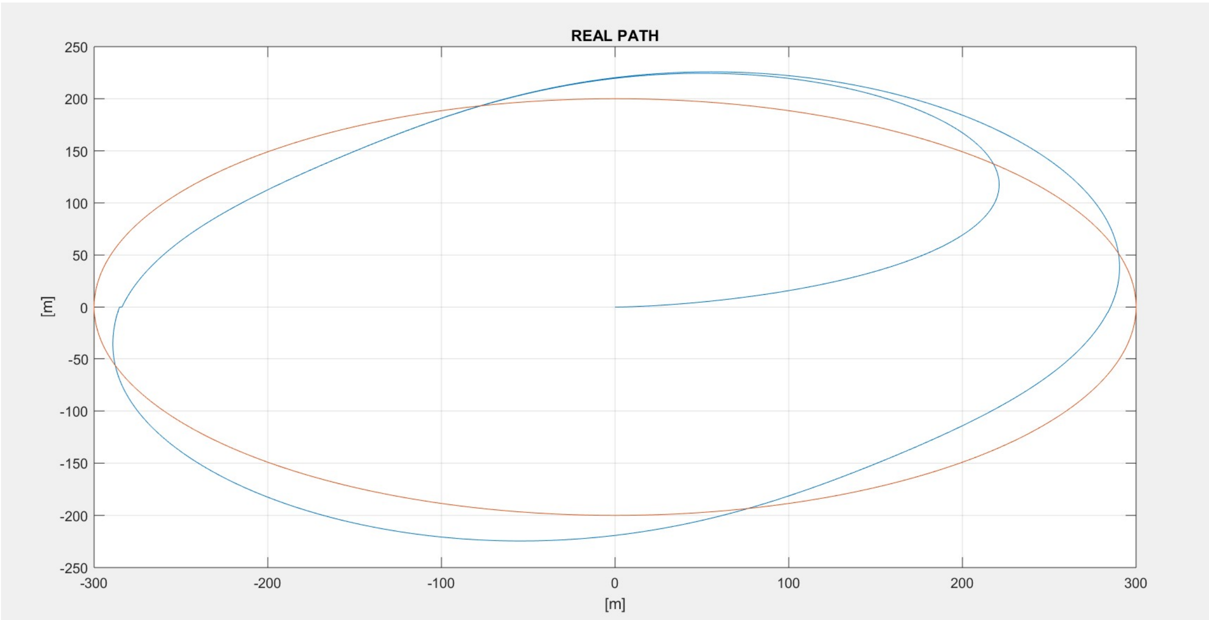

Figure 69 – LQI Trajectory #1

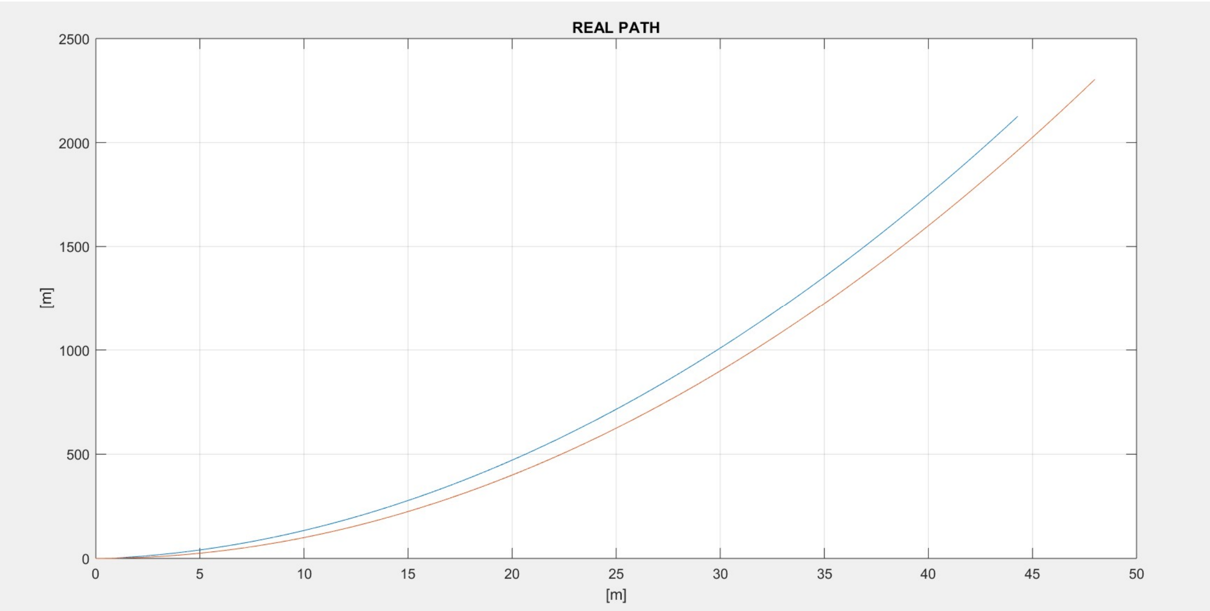

Figure 70 – LQI Trajectory #2

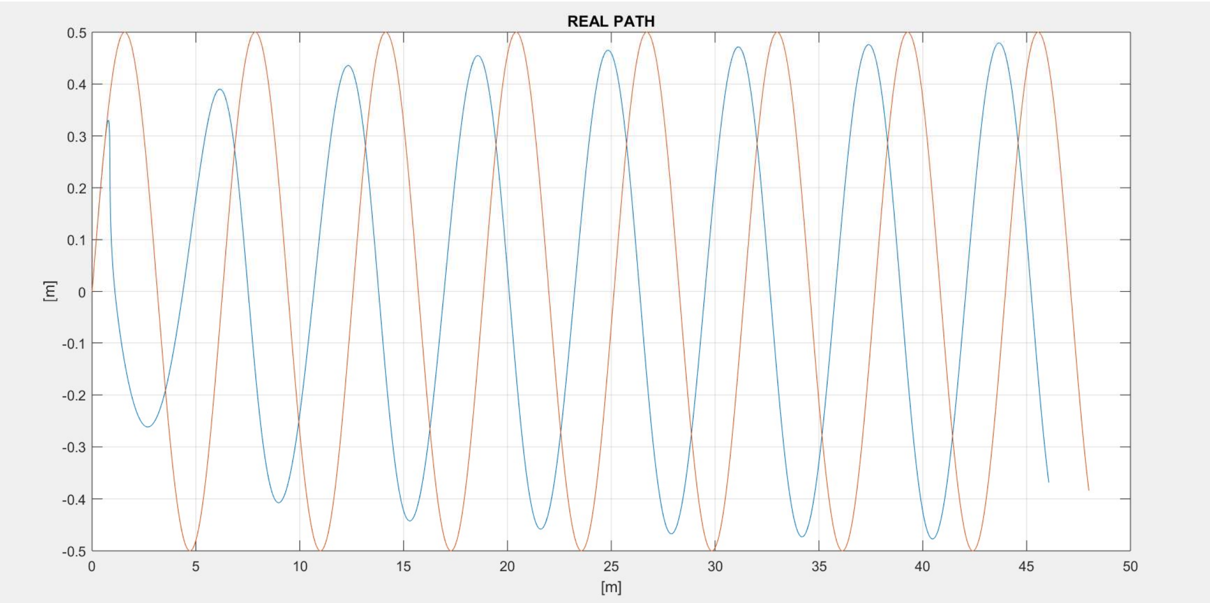

Figure 71 – LQI Trajectory #3

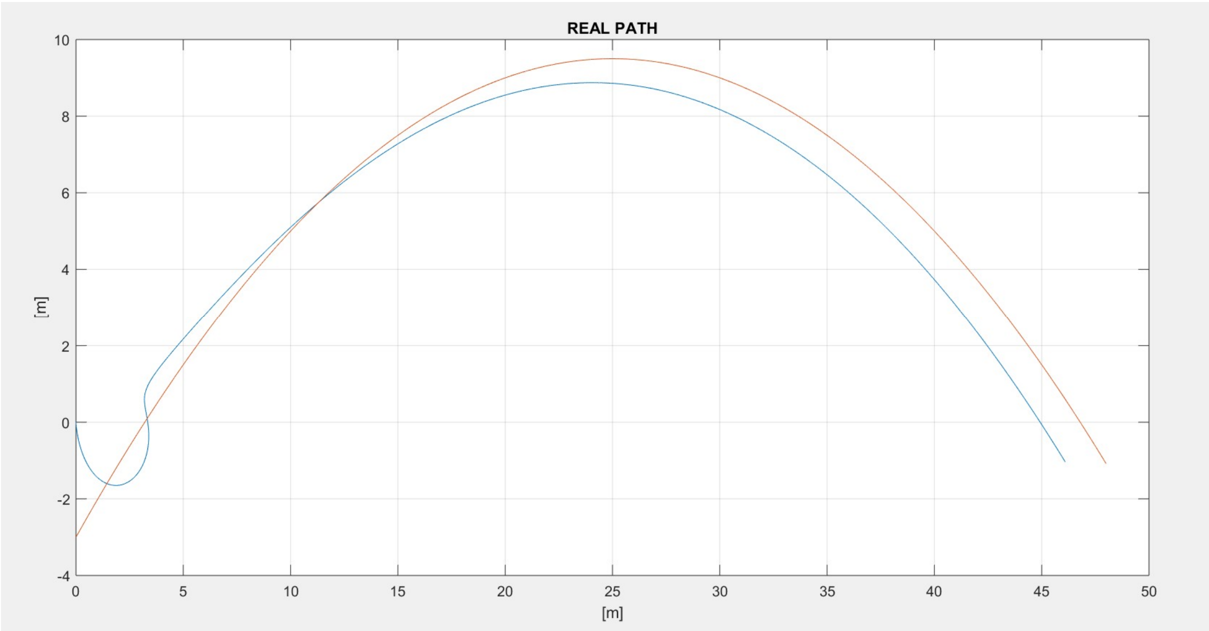

Figure 72 – LQI Trajectory #4

CHAPTER 7 – CONCLUSIONS

This control design work has achieved very satisfying results. As can be seen from the previous chapter, in fact, simulation tests have shown how much the type of control implemented was robust.

In particular, the LQR controller is the most efficient, since the system is easily stabilizable by combinations of weights of the Q matrix of different orders of magnitude. The resulting trajectories, compared to the ideal ones, are very similar. Furthermore, confirming the fact that the system was well implemented, there are no consistent differences between the LQR controller without observer and the one with the state estimation.

The same considerations cannot be made for the LQI controller, because it is very sensitive to changes in the weights of the Q matrix. In fact, only one case was studied because the range of weight values, which manage to stabilize the system, is very restricted. The resulting trajectory, therefore, will not present the same level of accuracy in following the ideal one. Despite this, the result obtained by LQI is generally not a bad one.

The bicycle model has well represented the dynamics of the vehicle, even if it did it in a simplified way.

The reflection carried out by this analysis leads us to say that autonomous driving is certainly not a distant reality. With the right correction terms, and an accurate machine learning process, the world of mobility would be completely revolutionized.

Certainly, a project of this type will be valid and efficient only if all the vehicles will adopt the autonomous driving systems. In fact, if all the vehicles drive autonomously, the car will be able to predict external stimuli much more easily, because those same external stimuli are actions generated by cars adopting the same logic. Human actions, on the other hand, appear to be less obvious from the point of view of a machine. Furthermore, in this case, it is assumed that the vehicles would be able to communicate with each other through an exchange of data and collaborate in avoiding critical situations.

Finally, to raise the level of success, it would be advisable to adapt the urban environment to this new mobility, so that the car can keep a wide variety of scenarios in its database and recognize road signs more easily.

No matter how many years it will take, we should only trust in the technological progress.

APPENDIX

APPENDIX A: LQR SCRIPT

```
clear all
close all
clc

%PARAMETERS

ts=(0:0.003:48)';
global lf lr Cf Cr m g Iz V_x mult L
lf=1.210;
lr=1.220;
m=1404;
g=9.81;
Cf=25000;
Cr=33000;
Iz=2600;
V_x=5;
L=lf+lf;
mult=m*(lr*Cr-lf*Cf)/(2*Cf*Cr*L);
T=48;

%SYSTEM

A=[0, 1, 0, 0;
   0, -(2*Cf+2*Cr)/(m*V_x), (2*Cf+2*Cr)/m, -V_x+(-2*Cf*lf+2*Cr*lr)/(m*V_x);
   0, 0, 0, 1;
   0, -(2*Cf*lf-2*Cr*lr)/(Iz*V_x), (2*Cf*lf-2*Cr*lr)/(Iz), -
(2*Cf*lf^2+2*Cr*lr^2)/(Iz*V_x)];

B=[0;
    2*Cf/m;
    0;
    2*Cf*lf/Iz];

C=eye(4);

D=zeros(4,1);

sys=ss(A,B,C,D);

%TRAJECTORY

t=0;
des_yaw=zeros(1,16001);
ts=(0:0.003:48)';
for i=1:16001
    X(i)=300*cos(0.2*t);
    Y(i)=200*sin(0.2*t);
    des_yaw(i)=atan2(Y(i),X(i))+1.57;
    Rot=[cos(des_yaw(i)), -sin(des_yaw(i)), 0;
        sin(des_yaw(i)), cos(des_yaw(i)), 0;
        0, 0, 1];
    Rt=Rot';
    x(i)=X(i)*Rt(1,1)+Y(i)*Rt(1,2);
    y(i)=X(i)*Rt(2,1)+Y(i)*Rt(2,2);
    t=t+0.003;
end
```

```
psi_dot=[0,diff(des_yaw)/0.003];
X=X';
Y=Y';
x_ref=x';
y_ref=y';
psi_dot_ref=psi_dot';
des_yaw=des_yaw';

des_pos_x=timeseries(x,ts);
des_pos_y=timeseries(y_ref,ts);
des_psi_dot=timeseries(psi_dot_ref,ts);

x_ref=des_pos_x;
y_ref=des_pos_y;
psi_dot_ref=des_psi_dot;

%%LQR WITHOUT OBSERVATOR

q=[1000 10 0 0.5];
r=[0.00001];
rho=100;
Q=diag(q);
R=rho*diag(r);
[K,S,e]=lqry(sys,Q,R);
open('LQR_Simulink'), sim('LQR_Simulink')


yreal=yreal.data;
t=0;
des_yaw=zeros(1,16001);
ts=(0:0.003:48)';

for i=1:16001
    X(i)=300*cos(0.2*t);
    Y(i)=200*sin(0.2*t);
    des_yaw(i)=atan2(Y(i),X(i))+1.57;
    Rot=[cos(des_yaw(i)), -sin(des_yaw(i)), 0;
        sin(des_yaw(i)), cos(des_yaw(i)), 0;
        0, 0, 1];
    Rt=Rot';
    Yreal(i)=(yreal(i)*tan(0.2*t))/(tan(0.2*t)*Rt(2,2)+1.5*Rt(2,1));
    Xreal(i)=(1.5*Yreal(i))/tan(0.2*t);
    t=t+0.003;
end

figure(1),
plot(Xreal,Yreal), title('REAL PATH'), xlabel('[m]'), ylabel('[m]'), grid
on, hold on, plot(X,Y)

%%LQR WITH OBSERVATOR

P_des=single([-2,-2,-2,-2]');
Lo=place(A',C',P_des)';
Ao=double(A-Lo*C);
Bo=double([B Lo]);
Co=C;
Do=double(zeros(4,5));
open('LQR_Obs'), sim('LQR_Obs')

yreal=yreal.data;
```

```
t=0;
des_yaw=zeros(1,16001);
ts=(0:0.003:48)';

for i=1:16001
    X(i)=300*cos(0.2*t);
    Y(i)=200*sin(0.2*t);
    des_yaw(i)=atan2(Y(i),X(i))+1.57;
    Rot=[cos(des_yaw(i)), -sin(des_yaw(i)), 0;
        sin(des_yaw(i)), cos(des_yaw(i)), 0;
        0, 0, 1];
    Rt=Rot';
    Yreal(i)=(yreal(i)*tan(0.2*t))/(tan(0.2*t)*Rt(2,2)+1.5*Rt(2,1));
    Xreal(i)=(1.5*Yreal(i))/tan(0.2*t);
    t=t+0.003;
end

figure(2),
plot(Xreal,Yreal), title('REAL PATH'), xlabel('[m]'), ylabel('[m]'), grid
on, hold on, plot(X,Y)
```

APPENDIX B: LQI SCRIPT

```
clear all
close all
clc

%PARAMETERS

ts=(0:0.003:48)';
global lf lr Cf Cr m g Iz V_x mult L
lf=1.210;
lr=1.220;
m=1404;
g=9.81;
Cf=25000;
Cr=33000;
Iz=2600;
V_x=5;
L=lf+lf;
mult=m*(lr*Cr-lf*Cf)/(2*Cf*Cr*L);
T=48;

%SYSTEM

A=[0, 1, 0, 0;
    0, -(2*Cf+2*Cr)/(m*V_x), (2*Cf+2*Cr)/m, -V_x+(-
2*Cf*lf+2*Cr*lr)/(m*V_x);
    0, 0, 0, 1;
    0, -(2*Cf*lf-2*Cr*lr)/(Iz*V_x), (2*Cf*lf-2*Cr*lr)/(Iz), -
(2*Cf*lf^2+2*Cr*lr^2)/(Iz*V_x)];

B=[0;
    2*Cf/m;
    0;
    2*Cf*lf/Iz];

C=eye(4);

D=zeros(4,1);
```

```
sys=minreal(ss(A,B,C,D));

%TRAJECTORY

t=0;
des_yaw=zeros(1,16001);
ts=(0:0.003:48)';

for i=1:16001
    X(i)=t;
    Y(i)=-0.02*t^2+t-3;
    des_yaw(i)=atan2(Y(i),X(i))+1.57;
    Rot=[cos(des_yaw(i)), -sin(des_yaw(i)), 0;
        sin(des_yaw(i)), cos(des_yaw(i)), 0;
        0, 0, 1];
    Rt=Rot';
    x(i)=X(i)*Rt(1,1)+Y(i)*Rt(1,2);
    y(i)=X(i)*Rt(2,1)+Y(i)*Rt(2,2);
    t=t+0.003;
end

psi_dot=[0,diff(des_yaw)/0.003];
X=X';
Y=Y';
x_ref=x';
y_ref=y';
psi_dot_ref=psi_dot';
des_yaw=des_yaw';

des_pos_x=timeseries(x,ts);
des_pos_y=timeseries(y_ref,ts);
des_psi_dot=timeseries(psi_dot_ref,ts);

x_ref=des_pos_x;
y_ref=des_pos_y;
psi_dot_ref=des_psi_dot;

%%LQI WITHOUT OBSERVATOR

q=[5000 4000 3000 2000 6000 5500 4500 3500];
r=[5];
rho=10^2;
Q=diag(q);
R=rho*diag(r);

[K,S,e]=lqi(sys,Q,R);
open('LQI_No_Obs'), sim('LQI_No_Obs')

yreal0=yreal0.data;
t=0;
des_yaw=zeros(1,16001);
ts=(0:0.003:48)';

for i=1:16001
    X(i)=t;
    Y(i)=-0.02*t^2+t-3;
    des_yaw(i)=atan2(Y(i),X(i))+1.57;
    Rot=[cos(des_yaw(i)), -sin(des_yaw(i)), 0;
        sin(des_yaw(i)), cos(des_yaw(i)), 0;
        0, 0, 1];
```

```
    Rt=Rot';
    Yreal0(i)=(yreal0(i)*(-0.02*t^2+t-3))/(t*Rt(2,1)+Rt(2,2)*(-0.02*t^2+t-
3));
    Xreal0(i)=Yreal0(i)*(t/(-0.02*t^2+t-3));
    t=t+0.003;
end

figure(1),
plot(Xreal0,Yreal0), title('REAL PATH'), xlabel('[m]'), ylabel('[m]'), grid
on, hold on, plot(X,Y)


%%LQI WITH OBSERVATOR

P_des=single([-2.5,-2.5,-2.5,-2.5]');
Lo=place(A',C',P_des)';
Ao=double(A-Lo*C);
Bo=double([B Lo]);
Co=C;
Do=double(zeros(4,5));
open('LQI_Obs'), sim('LQI_Obs')

yreal=yreal.data;
t=0;
des_yaw=zeros(1,16001);
des_steer=zeros(1,16001);
ts=(0:0.003:48)';

for i=1:16001
    X(i)=t;
    Y(i)=-0.02*t^2+t-3;
    des_yaw(i)=atan2(Y(i),X(i))+1.57;
    Rot=[cos(des_yaw(i)), -sin(des_yaw(i)), 0;
        sin(des_yaw(i)), cos(des_yaw(i)), 0;
        0, 0, 1];
    Rt=Rot';
    Yreal(i)=(yreal(i)*(-0.02*t^2+t-3))/(t*Rt(2,1)+Rt(2,2)*(-0.02*t^2+t-
3));
    Xreal(i)=Yreal(i)*(t/(-0.02*t^2+t-3));
    t=t+0.003;
end

figure(2),
plot(Xreal,Yreal), title('REAL PATH'), xlabel('[m]'), ylabel('[m]'), grid
on, hold on, plot(X,Y)
```

BIBLIOGRAPHY

Rajesh Rajamani. (2006). *Vehicle Dynamics and Control*.

OH, S.-W., CHAE, H.-C., YUN, S.-C., & HAN, C.-S. (2005). The Design of a Controller for the Steer-by-Wire System. *JSME International Journal Series C*, *47*(3), 896–907.

Pitre, R. R., Jilkov, V. P., & Li, X. R. (2005). A comparative study of multiple-model algorithms for maneuvering target tracking. *Signal Processing, Sensor Fusion, and Target Recognition XIV*, *5809*, 549.

Raaij, D. Van. (2007). *Application of bilateral control in Steer-by-Wire*.

Hwang, J., Huh, K., Kang, H. J., Yoon, P., Na, H., & Jung, H. (2008). Evaluation of lane keeping assistance controllers in HIL simulations. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, *17*(1 PART 1), 9491–9496.

Monitello, R., Albert, L. W., & Llp, A. A. (1999). Control Review.

Medici, P. (2013). *Teoria dei Veicoli Autonomi*.

Matematica, D., & Universit, S. (2017). *Matematica, automobili e guida autonoma (L'irragionevole efficacia della Matematica) Italo Capuzzo Dolcetta Gnampa-INdAM*.

Murray, R. M. (2006). *CDS 110b: Lecture 2-1 Linear Quadratic Regulators*. (January), 1–14.

Lublin, L., & Athans, M. (2010). Linear quadratic regulator control. *The Control Systems Handbook: Control System Advanced Methods, Second Edition*, 375–400.

Maurer, R. T., & Grazier, K. (2001). Development of core competencies in health care finance. In *The Journal of health administration education*.

Polack, P., Altché, F., De La, A., De La Fortelle, A., & de La Fortelle, A. (2017). *The Kinematic Bicycle Model: a Consistent Model for Planning Feasible Trajectories for Autonomous Vehicles?*

Tomalino, D. R. *Appunti sul progetto per retroazione statica dagli stati, il progetto dell'osservatore e del regolatore*.