Final Master Thesis

MSc in Automotive Engineering

Influence of the Cooperative Adaptive Cruise Control to the traffic flow

CONTENT

Author: Bernat Bosch Romà

Tutor: Arnau Doria Cerezo

Date: September 2017

Overview

The purpose of this study is to investigate the influence of the Cooperative Adaptive Cruise Control (CACC) to the traffic flow. CACC are the techniques that allows the vehicles regulating their speed automatically to avoid collision with precedent vehicles in front incorporating intervehicle communications.

In the first part of the project the mathematical model used to design the CACC algorithms is introduced as well as the mechanical model to perform the vehicle limitations in the simulation stage.

In this work, several simulation models have been build using the Matlab-Simulink environment. Then, the effect of the control parameters has been studied taking into account terms of safety.

In addition, the project analyses how vehicles that incorporates systems with communications act when they circulate with conventional vehicles that do not use Vehicle-to-Vehicle (V2V) communications. The proposed test bench consists in a circular road that allows to observe the effect of increasing the percentage of CACC vehicle to the traffic flow.

Finally, the thesis expounds the possible environmental impact if vehicles in the car fleet incorporate the CACC system.

Table of content

Glossary

(ACC) Adaptive Cruise Control: Vehicles system that automatically adjusts the vehicle speed to maintain a safe distance from vehicles ahead.

(ADAS) Advanced driver-assistance systems: systems to help the driver in the driving process. One example would be the ACC system.

(CACC) Cooperative Adaptive Cruise Control: Extension of ACC system with the possibility to exchange information as position or speed between adjacent vehicles.

(ICE) Internal Combustion Engine: an engine which generates motive power by the burning of petrol, oil, or other fuel with air inside the engine, the hot gases produced being used to drive a piston or do other work as they expand.

(ISA) International Standard Atmospheric: is an atmospheric model of how the pressure, temperature, density, and viscosity of the Earth's atmosphere change over a wide range of altitudes or elevations.

(ITS) Intelligent Transportation System: aims to provide innovative services relating to different modes of transport and traffic management and enable various users to be better informed and make safer, more coordinated, and 'smarter' use of transport networks.

Matlab: it is a high level technical computing language an interactive environment for algorithm development, data analysis, numeric computations and data visualization.

(OEM) Original Equipment manufacturer: It is used to refer to the vehicle companies.

Simulink: it is an environment for multidomain simulation and Model-Based Design for dynamic and embedded systems founded on Matlab. It provides an interactive graphical environment and a customizable set of block libraries with the capability of design, simulate, implement, and test a variety of time-varying systems, including communications, controls, signal processing, video processing, and image processing.

(V2I) Vehicle to Infrastructure: Communication system from a vehicle to the infrastructure that may affect the vehicle and vice versa.

(V2V): Vehicle to Vehicle: is an automobile technology designed to allow automobiles to communications to each other using a wireless network.

1. Preface

1.1. Motivation

During the past few decades, our society has been confronted with problems caused by increasing traffic. This increase in traffic leads to a heavily congested network and has a negative effect on safety, energy consumption and air pollution.

The demand for mobility in today's life brings additional weight on the existing ground transportation infrastructure for which a feasible solution in the near future lies in more efficient use of currently available means of transportation. For this purpose, the development of Intelligent Transportation Systems (ITS) technologies that contribute to improved traffic flow stability, efficiency, fuel economy and safety are needed.

Coordinated driving can reduce fuel consumption reducing inter-vehicle distances and improve traffic flow. Moreover, coordinated driving can improve driving experience by relieving humans from some driving duties and at the same time, by letting an automated system control the vehicle, improve safety. As such goals are not achievable using standard sensor-based Adaptive Cruise Control (ACC), the community started considering Cooperative Adaptive Cruise Control (CACC) [1].

What differentiates a CACC from a standard ACC is the use of wireless communication to share information such as speed and acceleration among vehicles, enabling the possibility to reduce inter-vehicle distance without compromising safety.

1.2. Origin of the project

The project has been developed from a previous bachelor degree final thesis done by Jaume Cartró Benavides [2]. The work consists on a simulation model that recreates the movement of a number of vehicles with different driving profiles in a circular circuit. The model has been developed using the speed for dynamic control and does not take into account the possibility of mixing the vehicles that incorporates the human driver model with the vehicles with ACC model.

2. Introduction

2.1. Objectives of the project

The main objective of the project is the development of a model that recreates a set of vehicles to be able to study the behaviour of different controllers in real traffic situations. Study the behaviour and respective advantages and disadvantages of different controllers and the fact to include communications between vehicles.

The second objective is to study the traffic flow mixing different vehicles models. Given that ACC technology is still growing and the OEM's want to include communications, it will exist a long period of time in which autonomous vehicles and conventional ones share the road. The study will be based on different percentages of vehicles driving in CACC versus ACC mode. The results will show if traffic density is reduced increasing the percentage of vehicles with CACC technology.

2.2. Scope of the project

The scope of the project is to design a model using Matlab-Simulink capable to recreate the vehicles behaviour with different kind of controllers. In addition, are studied the advantages to include the possibility of exchange information between the vehicles their effect in traffic conditions.

To control the vehicles, it only be considered the vehicles in front, therefore this project doesn't consider the possibility to overtake other vehicles or lane changes.

3. State of the art

An ACC system consist in a radar or other sensor capable of measuring the distance to another preceding vehicle on the highway. When there is no vehicle in front, the ACC travels set at speed defined by the user, much like a standard cruise controlled vehicle. However, if a vehicle is detected by the vehicles radar, the ACC system decide if the vehicle can continue travel at desired speed or the vehicle need to reduce for safety.

A CACC system is an evolution of the well-known ACC incorporating communications systems allowing sending or receiving information with other vehicles around.

A concept of CACC system is illustrated in Figure 3.1. This device utilizes information exchange between vehicles using wireless communication besides local sensor measurements that are available in a conventional ACC system. Moreover, allows exchange information between the infrastructures, so called by V2I or Vehicle to Infrastructure but this type of system doesn’t be treated in this project.

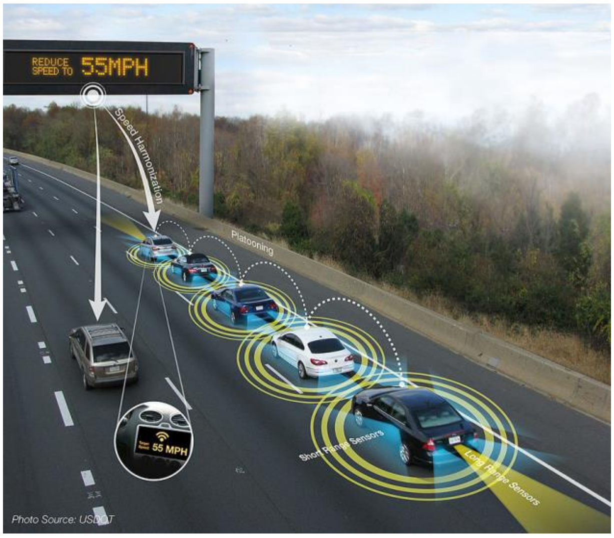


Figure 3.1 CACC Concept [3]

With the additional information because of communication, a CACC equipped vehicle is able to react faster to the behaviour of the surrounding vehicles and, besides, achieves a better synchronized traffic flow while avoiding string instability and reducing inter-vehicle distances. Altogether, CACC can increase traffic flow, reduce travel times and improve ride comfort. A small inter-vehicle distances can be also beneficial for reducing the fuel consumption of a group of vehicles. This benefit in terms of fuel economy is especially apparent for heavy-duty trucks since the aerodynamic drag of a truck is high because the flat frontal surface. Consequently, close distance driving will result in a significant fuel reduction [4].

4. Control design

In this chapter, the design of the control algorithms will be explained. The chapter has been divided in three parts. The first part describes the control problem that this project try to solve. Secondly, the kinematic controllers have been exposed divided in CACC and the typical ACC. Finally, the dynamic model with CACC and ACC variants has been presented.

4.1. Control problem

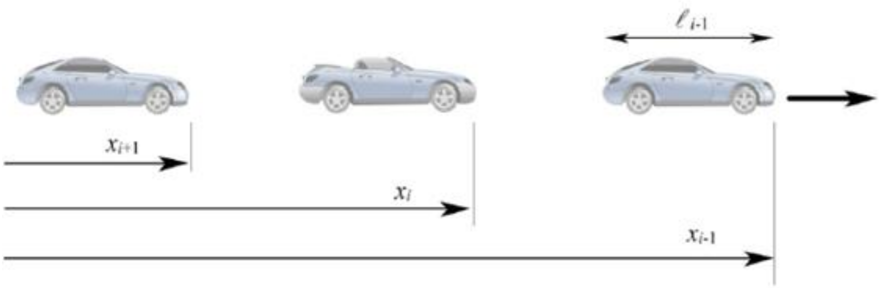


Figure 4.1. Distance between vehicles using relative position [5].

Consider N vehicles moving along the same path. The distance of the ith vehicle (with $i = 1, \ldots, N$) with respect to its predecessor, h_i, is given by:

$$h_i(t) = x_{i-1}(t) - x_i(t) - l_{i-1} \quad \text{(Eq 4.1)}$$

where x_i, x_{i-1} are the absolute positions of the vehicles and l_{i-1} is the length of the $(i-1)$th vehicle. The objective of the adaptive cruise control algorithm is to regulate the distance, h_i to a desired value, given by

$$h_{i,r}(t) = h_{i,o}(t) + k_{i,v} v_i(t) \quad \text{(Eq 4.2)}$$

where $h_{i,o}$ is the standstill distance (available from measurements) and $k_{i,v}$ is the constant time headway (equivalent to the time that the *i*th vehicle takes to arrive at the position of its predecessor). For sake of simplicity, in this paper we consider $h_{i,o} = h_o$ and $k_{i,v} = k_v$ for all the vehicles and we omitted the time dependent $h_i(\mathrm{t}) = h_i$ for following equations.

4.2. Kinematic model

Initially, the study has been designed without considering the vehicle dynamics by the reason of simplicity, therefore the output variable of the control law has been established as the speed.

4.2.1. CACC design

Defining the control error for the *ith* vehicle as

$$e_i = h_r - h_i, \quad \text{(Eq 4.3)}$$

the error dynamics using (Eq 4.1) and (Eq 4.2), results in

$$\dot{e}_i = k_v \dot{v}_i - v_{i-1} + v_i. \quad \text{(Eq 4.4)}$$

To guarantee the closed loop dynamics the error can be written as a first order dynamics

$$\dot{e}_i = -k_p e_i. \quad \text{(Eq 4.5)}$$

where $k_p > 0$ guarantees an asymptotic stability with a time constant $\tau = \frac{1}{k_p}$

Finally, since e_i depends on v_i, from equation (Eq 4.4) and (Eq 4.5) the control law can be rewritten as

$$(1 + k_p k_v) v_i + k_v \dot{v}_i = -k_p (h_o - h_i) + v_{i-1} \quad \text{(Eq 4.6)}$$

or, written as a transfer function

$$v_i(t) = \frac{1}{k_v s + (1 + k_p k_v)} (k_p (h_i - h_o) + v_{i-1}) \quad \text{(Eq 4.7)}$$

where "s" is the time-derivative operator. The control law requires the information of the h_o, v_i and v_{i-1}. Usually, h_o and v_i are available and they are used in the current ADAS systems, but v_{i-1} is subjected to communications among other vehicles.

4.2.2. ACC design

The conventional adaptive cruise control can be implemented using the same control law (Eq 4.7) without taking into consideration the communications between vehicles, (i.e., setting $v_{i-1} = 0$) the control law for an ACC vehicle would be like

$$v_i(t) = \frac{1}{k_v s + (1 + k_p k_v)} k_p (h_i - h_o) \quad \text{(Eq 4.8)}$$

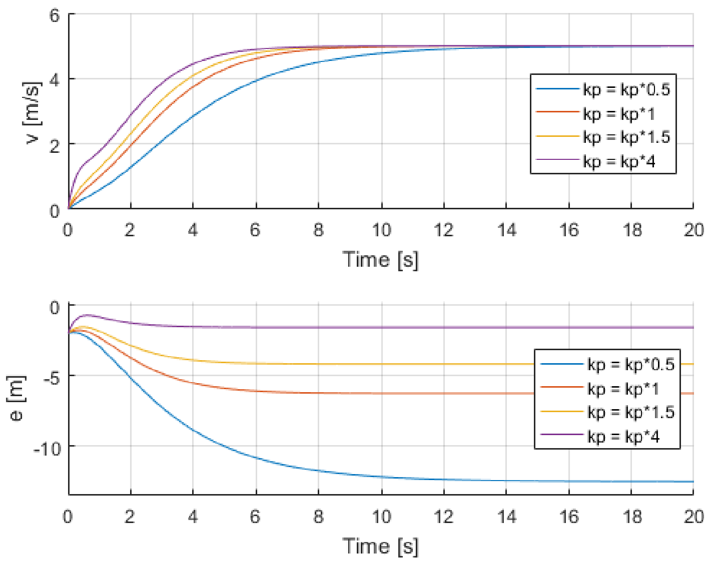


Figure 4.2 Velocity and error for ACC model with different proportional constant

The graphic above shows the results of four vehicles using an ACC model with proportional control and different proportional constant value. The time headway constant (k_v) has set up in 1 second and the proportional gain (k_p) in 0.8. The target speed for following vehicles are 5 m/s.

With a larger proportional constant, the control improves, resulting in quickest way to achieve the target speed value and with a low error result. The error can not reach the zero value and appears a steady state error. On the other hand, with a small k_p, bigger is the steady state error and spend more time to achieve the objective speed.

4.2.3. Integral Action

As shown in Figure 4.2, the knowledge of v_{i-1} is necessary for the proper regulations of vehicle distances. An alternative when the information of the precedent vehicle speed is missing is the use of a dynamic extension (similar to a PI controller). The main function of the dynamic extension is to make sure that the process output agrees with the set point in steady state. Then, the desired second order dynamics is set to

$$\dot{e}_i = -k_p e_i - k_z z_i \quad \text{(Eq 4.9)}$$

$$\dot{z}_i = e_i \quad \text{(Eq 4.10)}$$

that guarantees that $e_i = 0$ is an equilibrium. Then, matching the dynamics of e_i equation (Eq

4.9)
(Eq 4.10) results in

$$\left(1 + k_p k_v\right) v_i + k_v \dot{v}_i = -k_p (h_o - h_i) - k_z z_i \qquad \text{(Eq 4.11)}$$

or using the s-operator

$$v_i(t) = \frac{k_p s + k_z k_v}{k_v s^2 + \left(1 + k_p k_v\right) s + k_z k_v} (h_i - h_o) \qquad \text{(Eq 4.12)}$$

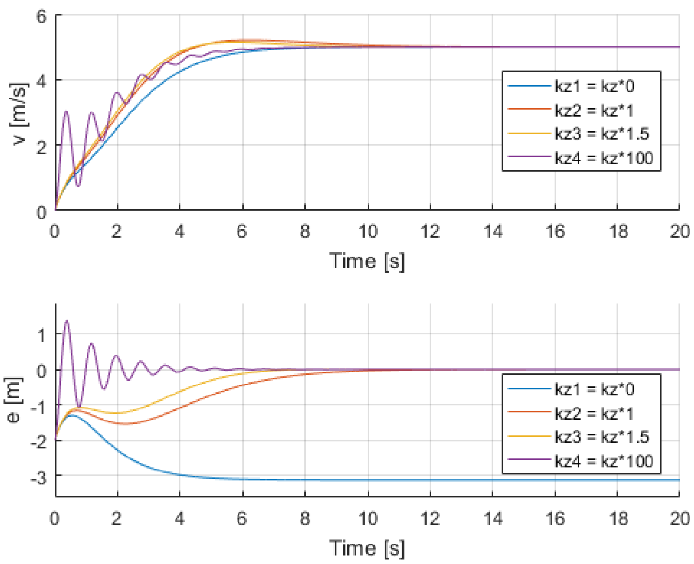


Figure 4.3 Velocity and error for ACC model with different integral constant

The Figure 4.3 illustrates the results of four vehicles using an ACC model with proportional and dynamic extension and different integral constant value. The proportional gain has been set up in 1.6, the time headway constant in 1 second and the integral gain (k_z) has been defined in 0.64.

Adding the integral component, the error achieves the “zero value” (with the integral action always will appear a small steady state error). However, the velocity control suffers and overshoot in comparison with exclusive proportional control. With a large integral constant, the result is reached faster but with a lot of damping. For comparison, in blue, it is added a vehicle with an integral constant equal to zero, equivalent to a vehicle with only proportional component.

4.3. Dynamic model

A better point of view to control the dynamic of the vehicle is, instead using the vehicle speed, the controller uses the engine force that a vehicle need to accelerate. For this approach a new controller has been designed including the vehicle dynamics.

4.3.1. CACC design

Defining the vehicles dynamics as:

$$m\,\dot{v} = u + \sum F \qquad \text{(Eq 4.13)}$$

where m is the mass of the vehicles, u is the force applied to achieve the desired velocity and $\sum F$ are the resistive forces. Replacing (Eq 4.13) into (Eq 4.4) we get

$$\dot{e}_i = \frac{k_v}{m}\left(u_i + \sum F\right) - v_{i-1} + v_i \qquad \text{(Eq 4.14)}$$

Rewriting the equation

$$\dot{e}_i = \frac{k_v}{m} u_i + \delta - v_{i-1} + v_i \qquad \text{(Eq 4.15)}$$

where $\delta = \sum F / m$. It can be considered as an input but in this project, we have considered as unknown because the difficulties to calculate it in the real life. Consequently, in the next related equations will be considered unknown (for simplicity $\sum F = 0$).

On the other hand, similarly to section 4.2.3, it is possible to define the desired error dynamic as a second order dynamics.

$$\begin{aligned} \dot{e}_i &= -k_p e_i - k_z z_i \\ \dot{z}_i &= e_i \end{aligned} \qquad \text{(Eq 4.16)}$$

that guarantees that $e_i = 0$ is an equilibrium. Then, matching the dynamics of e_i in (Eq 4.14) with (Eq 4.16), results in

$$-k_p e_i - k_z z_i = \frac{k_v}{m} u_i - v_{i-1} + v_i \qquad \text{(Eq 4.17)}$$

and the control law can be rewritten as

$$\begin{aligned} u_i &= \frac{m}{k_v}\left(-k_p e_i - k_z z_i + v_{i-1} - v_i\right) \\ \dot{z}_i &= e_i \end{aligned} \qquad \text{(Eq 4.18)}$$

Using state-space representation the error dynamics can be rewritten as

$$\begin{pmatrix} \dot{e}_i \\ \dot{z}_i \end{pmatrix} = \begin{pmatrix} -k_p & -k_z \\ 1 & 0 \end{pmatrix}\begin{pmatrix} e \\ z \end{pmatrix} + \begin{pmatrix} -\tilde{v}_{i-1} \\ 0 \end{pmatrix} \quad \text{(Eq 4.19)}$$

It is possible to calculate the poles calculating previously using the equation $det(A-\lambda I)=0$ where λ are the roots of the characteristic equations or the system eigenvalues A,

$$A = \begin{pmatrix} -k_p & -k_z \\ 1 & 0 \end{pmatrix} \quad \text{(Eq 4.20)}$$

Following the equations (Eq 4.19) and (Eq 4.20) the error dynamics shows two poles at $\lambda = \frac{-k_p \pm \sqrt{k_p^2 - 4k_z}}{2}$

To get a stable system the term inside the square root must be negative, $k_p^2 - 4k_z < 0$. On consequence, it will result on imaginary term. Forcing this conditions the poles result as

$$\lambda = -\sigma \pm j\omega_d = -\frac{k_p}{2} \pm j\frac{\sqrt{4k_z - k_p^2}}{2} \quad \text{(Eq 4.21)}$$

that can be related to the response performance with

$$\sigma = \frac{4}{t_s}\ ;\ \sigma = \frac{k_p}{2} \rightarrow k_p = 2\frac{4}{t_s} \quad \text{(Eq. 4.22)}$$

Where $\sigma = \frac{4}{t_s}$ following the 2% criteria, being t_s the settling time.

Moreover, it is possible to calculate k_z equalling the second term of (Eq 4.21).

$$\omega_d = \frac{\sqrt{4k_z - k_p^2}}{2} \rightarrow k_z = \omega_d^2 + \frac{k_p^2}{4} \quad \text{(Eq 4.23)}$$

Where ω_d is the damped natural frequency and it is function of σ and the overshoot (M_p).

$$\omega_d = -\frac{\sigma\pi}{ln(M_p)} \quad \text{(Eq 4.24)}$$

4.3.2. ACC plus integral action design

As we observed before, the ACC can be implemented without considering the velocity of the precedent vehicle $(v_{i-1}=0)$. This result to be a simplification of the model presented in the equation (Eq 4.18)

$$u_i = \frac{m}{k_v}\left(-k_p e_i - k_z z_i - v_i\right)$$
$$\dot{z}_i = e_i \qquad \text{(Eq 4.25)}$$

5. Mechanical model

In this section, the vehicle model has been designed allowing to make simulations without vehicle restrictions. Nevertheless, it has been added a few mechanical limitations. That can be a logical thinking since it is possible that a vehicle cannot achieve the reference velocity that gives the algorithm.

The mechanical model has a few simplifications to reduce the complexity of the model. Some simplifications have listed below:

- There is no weight distribution.
- There are no inertial forces.
- There are no different drive wheels configurations (all are front wheel drive).

5.1. Longitudinal vehicle dynamics

Consider a vehicle moving on a sloping road as shown in Figure 5.1. The external longitudinal forces acting on the vehicle include tractive forces, rolling resistance, aerodynamic drag forces and climbing resistance forces. These forces are described in detail in the sub-sections that follow.

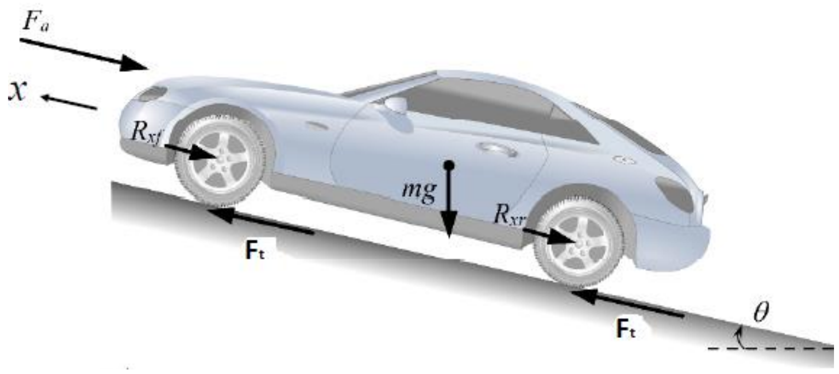


Figure 5.1. Vehicle free body diagram [5].

Using the second Newtons Law and making a force balance along the vehicle longitudinal axis yields

$$M\dot{v} = F_t - F_a - R_R - Mgsin(\theta) \quad \text{(Eq 5.1)}$$

where M means the vehicle mass, F_t is the longitudinal tractive force or longitudinal tire force, F_a is the aerodynamic drag force, R_R represents the force due to the rolling resistance at the tires, g is the gravity acceleration constant and θ means the angle of road inclination on which the vehicle is travelling.

The equation (Eq 5.6) can be interpreted with different meanings. If the sum of the right part terms is less than zero means that the vehicles is braking, when the result is positive means than the vehicle is accelerating and when the sum is equal to zero it means that the car is at constant velocity.

$$F_t - F_a - R_R - Mgsin(\theta)\begin{cases} < 0 \rightarrow the\ vehicle\ brakes \\ = 0 \rightarrow constant\ speed \\ > 0 \rightarrow the\ vehicle\ accelerates \end{cases} \quad \text{(Eq 5.2)}$$

5.1.1. Tractive force

Tractive force come determined by torque, or what it is the same, by the engine power. The vehicles with internal combustion engine incorporate a gearbox to variate the transmission relation and finally achieve the vehicle final movement. Transmission helps us to use the engine at his maximum efficiency, closing to the maximum power curve.

Maxim engine force is defined by following equation

$$F_{engine} = \frac{P_{max}}{v} = \frac{\Gamma}{\omega} \quad \text{(Eq 5.3)}$$

In pictures below the tractive effort do not start at zero speed. Looking carefully the previous equation it is possible to conclude that the force generated by the engine or needed for the vehicle at zero speed is not defined. This speed gap between the nil speed and minimum engine revolutions has been saved by the clutch.

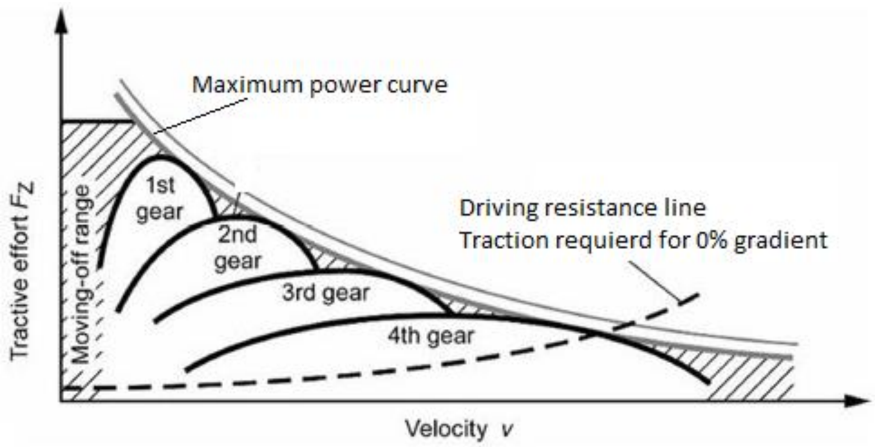


Figure 5.2 Tractive effort versus vehicle speed [6]

The available tractive force is the minimum of the force generated by the engine and the force that the tyre can transmit to the road before slip.

$$F_{engine} = \frac{P}{v} \quad \text{(Eq 5.4)}$$

$$F_{tyre} = \mu N \quad \text{(Eq 5.5)}$$

where P is the engine power, μ is de friction coefficient of the road (the value depends of the road conditions: dry, wet, etc) and N is the normal force.

$$F_{tractive} = min(F_{engine}, F_{tyre}) \quad \text{(Eq 5.6)}$$

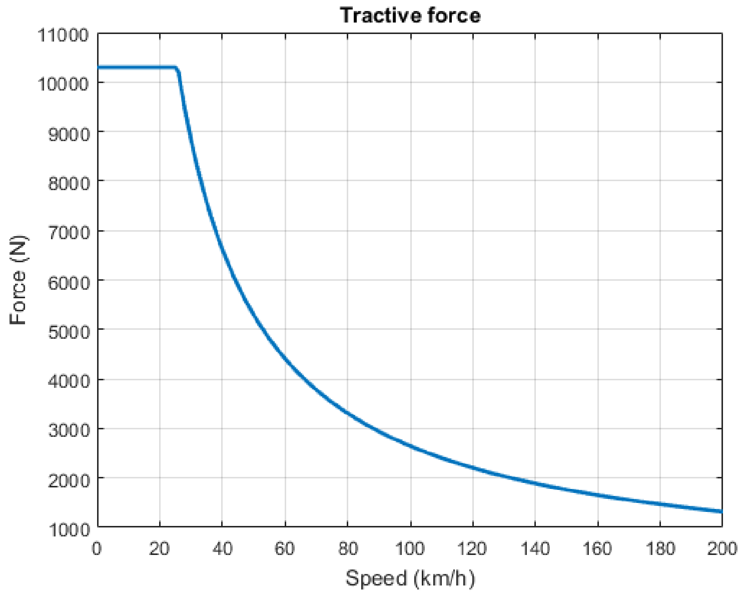


Figure 5.3. Vehicle maximum tractive force

As an example, a simulation has been carried out to observe the maximum traction curve of a vehicle. It has been considered a road friction coefficient of 0.7, a mass of 1500 kg and an engine power of 100 CV.

The tractive force model described above is a simplified model. A conventional car with an internal combustion engine (ICE) and a gearbox do not have this traction curve, more typical in electric motors. The internal combustion engine has low torque at low rotation speeds. Therefore, it is necessary to give a proper slip to the clutch on ICE vehicles. In addition, a transmission is required for changing the vehicle speed.

This supposition has been made in order to simplify the model and because in the next few years the number of electric and hybrid vehicles will grow up replacing the combustion engines.

5.1.2. Rolling resistance

The rolling resistance is the force resisting the motion when the tire rolls on a surface. Due to the vertical force over the tire (normal load) this deforms on the contact patch with the road. The compound used in tires fabrications has a viscoelastic behaviour and the energy spent in deforming the tire material is not completely recovered when the material returns to the original shape.

When the tire is rotating the loss of energy in tire deformation is a non-symmetric distribution as shown in Figure 5.4.

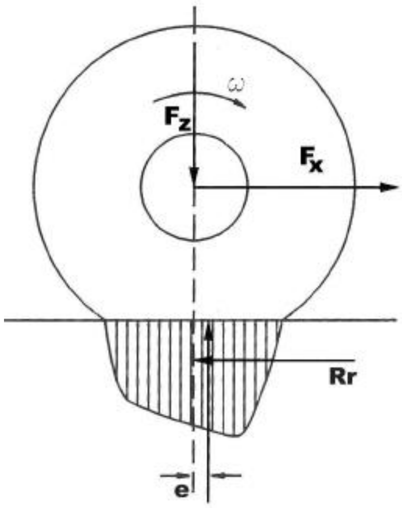


Figure 5.4. Rolling resistance force [7]

The equation that defines the rolling resistance force is:

$$R_R = f_R \cdot M \cdot g \cdot cos(\alpha) \quad \text{(Eq 5.7)}$$

where f_R is the rolling resistance coefficient, M is the weight of the vehicle, g is the gravity and finally α is the ground inclination.

5.1.3. Aerodynamic drag force

The friction between the air and a vehicle moving it suppose a force acting to opposite to the relative motion of the vehicle. The drag force depends on the properties of the fluid (air), the size, shape, and speed of the vehicle. Because the drag force depends on the square of the velocity, the force increases highly when the velocity increases as well while in low speed the drag force will be less important. The aerodynamic force can be represented as:

$$F_a = \frac{1}{2}\,\rho\, C_d\, A_f\,(V_{Vehicle} - V_{wind})^2 \qquad \text{(Eq 5.8)}$$

where ρ is the air density, A_f is the vehicle frontal area, which is the projected area of the vehicle in the travel direction and C_d is the drag force coefficient that is small when the vehicles are well designed in terms of aerodynamics.

Atmospheric conditions affect air density and hence can affect aerodynamic drag, but in this project, it is taken as a constant (1.225 kg/m^3), the density for 15 ºC and at sea level according to ISA (International Standard Atmosphere).

The frontal area A_f is calculated as the multiplication of the width and the height of the vehicle. Additionally, it is multiplicated by a coefficient, always less than one, to be more realistic. Generally, this coefficient is between $0.85 \div 0.9$.

Figure 5.5 shows what is understood as vehicle projected area.

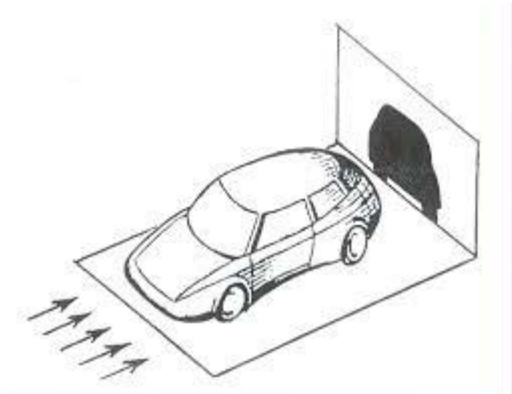


Figure 5.5 Projected frontal area [8]

5.1.4. Climbing resistance

The climbing resistance is just considered when a slope is present. Thus, when a vehicle try to climb an inclined road the self-weight of the vehicle will oppose it.

$$F_g = M \cdot g \cdot sin(\alpha) \qquad \text{(Eq 5.9)}$$

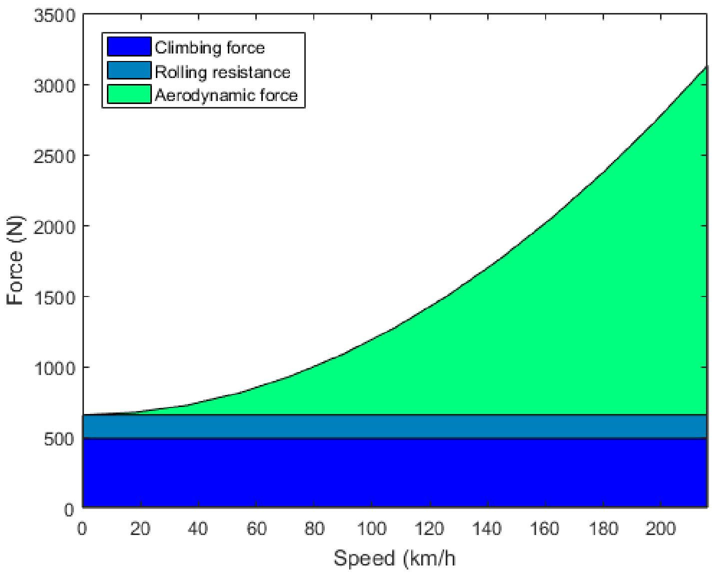


Figure 5.6 Accumulated force versus vehicle speed

In figure above the difference resistive forces explained before are shown in order to see the contribution that everyone has function of vehicle speed.

This simulation has been done with vehicle mass of 1000 kg, a slope of 5%, a drag coefficient of 0.4 and a frontal area of 2.805 m^2.

It shows how, at low speed, aerodynamic force contribution is nil while climbing and rolling resistance forces are constant for any speed. As speed increases drag force increase quadratically being more determinant than the sum of the other two.

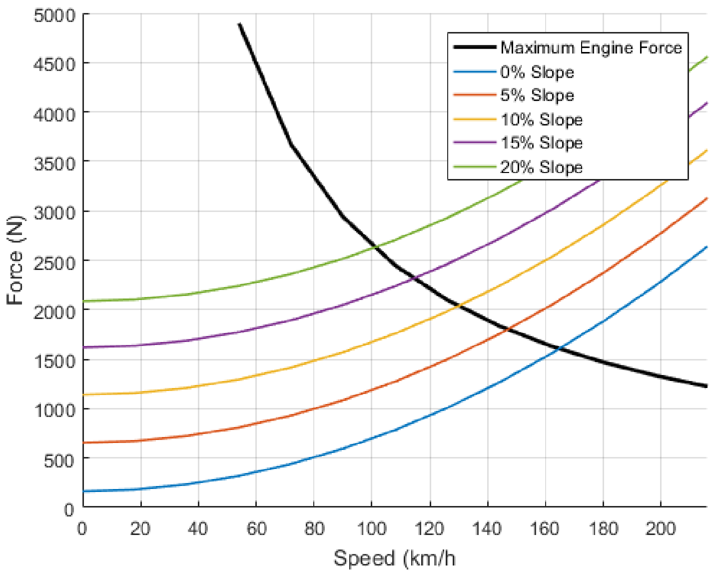


Figure 5.7 Force variation for different slope

The result of different simulations varying slope are shown in Figure 5.7. The parameters of the vehicles are listed on Table 5-1.

Parameter	Value
Power	100 CV
Mass	1000 kg
C_d	0.4
A_f	2.805 m^2
f_R	0.017

Table 5-1. Vehicles parameters

Figure 5.7 shows clearly how the resistive forces increases as slope increases as well. In black, the maximum engine force and when this line cross with a resistive force it is possible to obtain the maximum vehicle speed for a given slope.

Mathematically, it is possible to obtain the theoretical maximum speed equalling (Eq 5.4) and the sum of resistive forces.

$$\frac{P}{v} = \frac{1}{2}\,\rho\, C_d\, A_f\, v^2 + f_R \cdot M \cdot g \cdot cos(\alpha) + M \cdot g \cdot sin(\alpha) \qquad \text{(Eq 5.10)}$$

$$\frac{1}{2}\,\rho\,C_d\,A_f\,v^3 + f_R \cdot M \cdot g \cdot cos(\alpha)v + M \cdot g \cdot sin(\alpha)v - P = 0 \qquad \text{(Eq 5.11)}$$

Rewriting the expression, we obtain a third-degree equation, describing a cubic function. Using the Matlab expressions *roots(x)* it is possible to calculate the roots of a polynomial equation. The roots of this kind of third-degree equation returns two imaginary solutions and one real equation, being this last one, the maximum theoretical speed in meters per second.

6. Model implementation

The software chosen to develop the model has been Matlab because it had been used in previous subjects and is very popular in research and student world. Matlab: is a high level technical computing language, an interactive environment for algorithm development, data analysis, numeric computations and data visualization. The way to save the data in Matlab is through matrix, where in a row it is possible to allocate one value according one variable and in a column other variable. This fact makes very suitable for the implementation of the different models, because, for example, each row of a matrix can represent a different vehicle while each column may represent the vehicle speed for a certain time.

The model itself has been done using a complementary tool of Matlab known as Simulink. It provides an interactive graphical environment and a customizable set of block libraries with the capability of design, simulate, implement, much easier than conventional Matlab.

6.1. Program general structure

The purpose of this chapter is to explain the different programs created to study the different distance control systems. Four main files have been done to understand the behaviour of the vehicles with each control and the combinations in a road of both.

- **Speed_and_error_n_vehicles**: It is a Matlab script created to study the different controls. In this program one is capable to adjust the proportional and integer gains and modify the parameters to observe the variations.
- **Speed_and_error_n_vehicles_communications_comparison**: It is similar than previous program used to compare the controls with communications and without them.
- **Circular_model:** This program has been created to study how the vehicles behave in traffic situations and compare the adaptive cruise control system with the CACC system. At the end, it crates the fundamental diagrams.
- **Normal_distribution**: The program shows the range of vehicle and driving parameters using histograms.

6.2. Speed and error n vehicles

The “Speed_and_error_n_vehicles” is the main file of all the study and contains the different controls used and all the parameters needed to perform simulations and plot the results.

The idea of the file is that, given a leader vehicle it is possible to add a number of following vehicles with distance control system chosen and observe the difference in speed and error of the vehicles chain. Normally, proportional control and proportional integral are compared in each simulation.

This file allows you to see which control has a faster response or has an overshoot. In addition, is useful to decide if certain parameters selected for a control are possible to implement without problems of string stability

The file “Speed_and_error_n_vehicles_communications_comparison” it is almost the same with the difference that incorporates two more controls without communications to compare between all of them, as the program name refers.

6.2.1. Simulink structure

The Simulink structure used in this program it is easily to understand. Can be separated in two main blocks, on one side, the vehicle leader and on the other side there is the following vehicles controls. The Figure 6.1 try to recreate the Simulink file shown in (Annex A1).

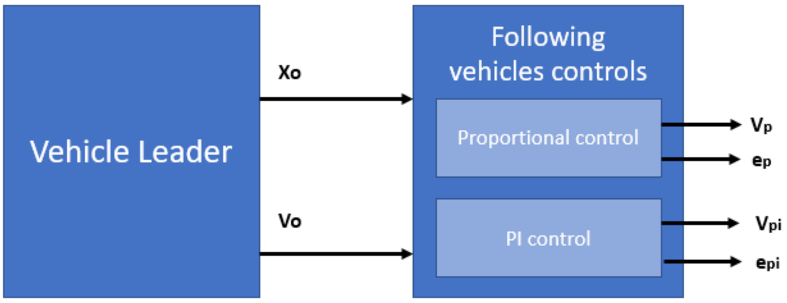


Figure 6.1. Speed and error n vehicles schematic model

First, the vehicles leader speed is created and integrated to obtain the global position. This values are passed to the following vehicles controls and finally they calculate their own speed and the error among other variables such as force or position.

6.2.1.1. Mechanical model implementation

To limit the speed according to vehicles or ground limitations a few blocks has been used, grouped together inside of each vehicle control. This blocks incorporate subsets (Annex A1.5 and A1.6).

The first subset saturates the vehicle force with the limitations. There are two limitations as has been explained in previous chapters. There the engine limitation, the number of horse power limits the vehicle speed. In addition, exist a ground limitations to avoid the slip of the wheels when exceeds the maximum force allowed.

The second subset allow to obtain the vehicle real speed. With the force saturated of previous block and the resistance force it is possible to calculate the real speed of each vehicle.

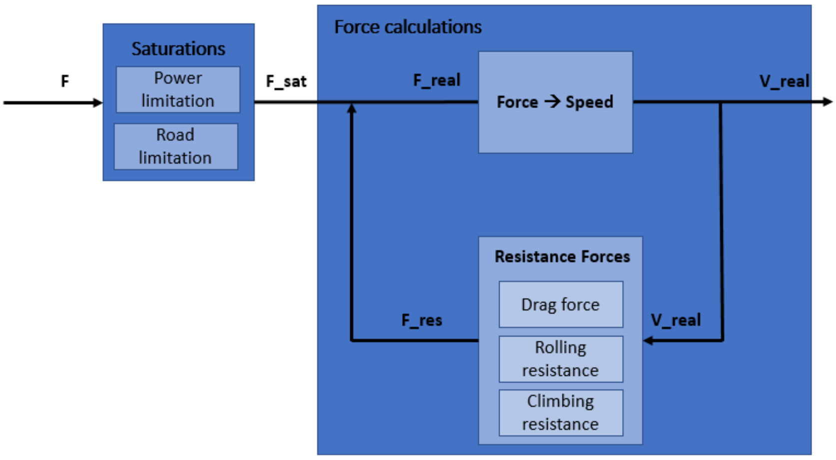


Figure 6.2. Mechanical model implementation scheme

6.3. Fundamental diagram

As explained in previous sections, vehicles including CACC systems will start to appear in our roads and during a certain period of time they will share it with cars that includes conventional adaptive cruise control.

A circular model has been created with the intention to recreate a realistic single lane road. This model allows to evaluate the influence of the control speed system (with or without communication) and the number of vehicles in a close road.

Traffic density (k), traffic flow (q) and average vehicles speed (u) are related in a formula known as continuity equation [9].

$$q = k \cdot u \quad \text{(Eq. 6.1)}$$

The three previous variables can be explained as follows:

- Average speed, $u\ (km/h)$: Represents the mean value of the average speed of all vehicles in the circular model.
- Traffic flow, $q\ (vehicles/h)$: It gives the average the number that pass a cross-section during a unit of time. In this case, per hours.

- Traffic density, k $(vehicles/km)$: It gives the rate of the number of vehicles circulating per distance unit.

The aspect of the fundamental diagram is shown in picture below:

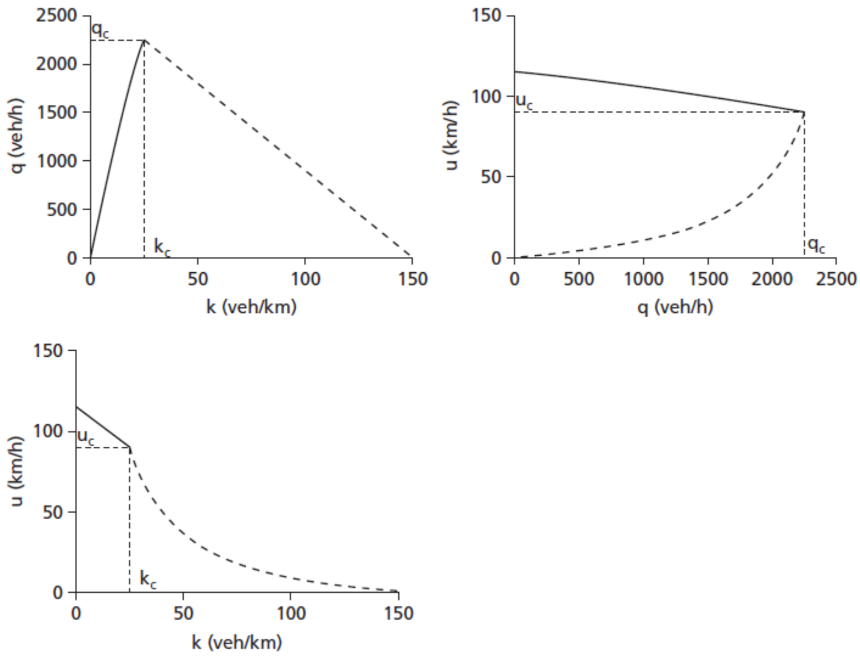


Figure 6.3. Theoretical fundamental diagram [9]

The points related to Figure 6.3 (q_c, k_c and u_c) correspond to the roadway capacity, the critical density and critical velocity respectively. These points show when traffic jams occur and, when that happens it means that average speed starts to decrease rapidly.

The file in charge to create the fundamental diagrams is the Matlab script called "Circular _model.m". To make the fundamental diagrams the program has two main loops explained in detail below:

1. The first one is responsible to increase the percentage of vehicles with cooperative adaptive cruise control system starting from zero percent (all vehicles with ACC system) until one hundred percent (all the vehicles with CACC system).
2. The second loop have the responsibility to increment the number of vehicles in the same circular circuit. So, maintaining constant the length of the circuit but increasing the number of vehicles it results that the traffic density is also increasing. The traffic density used has been for $2\ vehicles/km$ to $120\ vehicles/km$.

For each simulation, all vehicles parameter has been calculated again to recreate different scenarios and try to represent more accurately the real traffic conditions.

6.3.1. System type vector

An important vector used on circular model it has been the vector known as "System type". This vector is in charge to determine which vehicles have the ACC system or the CACC system and assign one position to the vehicles. This vector has been developed by Marc Fernandez in his final project thesis [4].

One problem was to ensure a certain number of vehicles that would have a given system based on a percentage. This obstacle would not have been so important if we had treat with thousands of cars but the difficulty is magnified when dealing with units or tens because percentages may not be representative.

The solution found was using a Matlab command named "round". As its name suggests this command round a number to nearest integer.

Therefore, the steps to obtain the System type vector are the following:

1. Is obtained the integer number of vehicles that use CACC using the command "round".
2. Is created a random vector to assign the position for the vehicles that use the CACC system. The previous step determines the number of vehicles with CACC system and the same positions as the number calculated before determines the position of the vehicles inside the vector.
3. Is created a vector fill of zeros to assign the dimension of System type vector.
4. Is allocated the vehicles that use the CACC system in the vector according the positions of the random vehicle.

Finally, what is obtained is a vector with one column and many rows as vehicles exist fill with zeros (ACC vehicles) and ones (CACC vehicles) in random positions.

6.3.2. Simulink structure

A simplification of the Simulink model is presented in Figure 6.4. The first block (starting left to right) is the main block of the program. It contains the controllers, a system that recreates an ACC control and other which represents the CACC system. In addition, this block incorporates the subset responsible to mix the two kinds of vehicles. Secondly, appears the mechanical model with the vehicle limitations already explained in 6.2.1.1. After that, there is a block in charge to transform the longitudinal position of the vehicles into circular position. Finally, there are a few blocks responsible to save the speed and the position among other, for future graphics.

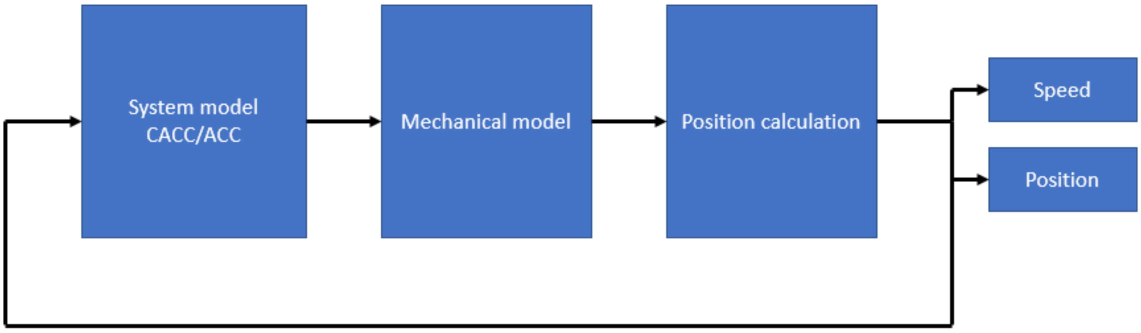


Figure 6.4. Circular model Simulink scheme

6.3.2.1. CACC and ACC driver combinations

One of the principal objective of the project is to study the behaviour of the vehicles in combination of divers with different control systems and how coexist in the same circuit.

To combine the vehicles that uses the two-different distance control system it has been used the System type vector explained before.

A scheme of the blocks used in Simulink environment is represented in Figure 6.5. For the CACC vehicles branch it has been used the Drivers system but in the ACC branch, the opposite vector has been used.

So, the result of using this system is that one can obtain the force of the vehicles with CACC and the other positions filled of zeros and the opposite vector with ACC vehicles force and other positions filled of zeros. These two vectors have opposite values, when one has ones the other has zeros and vice versa. Finally, when the two vectors are added result in a vector with the proper combination of vehicles with ACC and CACC force.

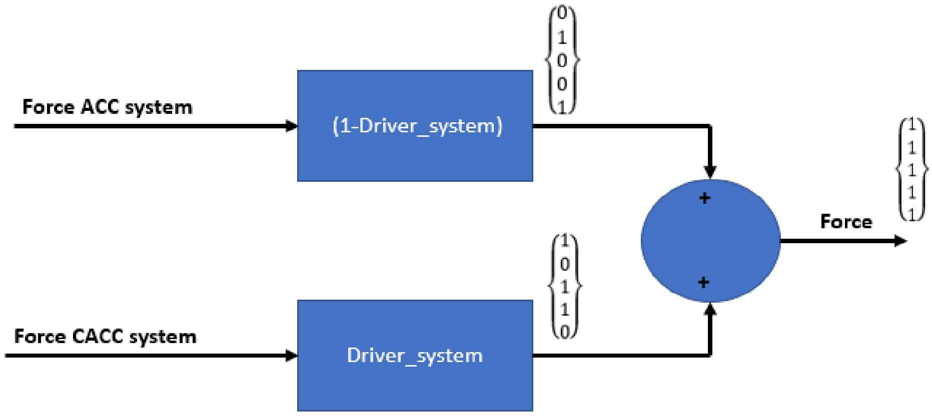


Figure 6.5. Combination of vehicles with ACC and CACC system

6.4. Vehicles and driver parameters

The vehicles that OEM's sell in the market do not have the same characteristics. It is possible to see vehicles with variations in height or width among others. The same happens in the driving profile of each person. A driver with a risky behaviour will leave less safety distance and will have a maximum acceleration higher than a calmer driver.

To implement a different driving profile characteristics into Matlab, the normal distribution has been added to the model. The normal distributions can be characterized with a mean value and a deviation associated. To introduce this distribution in Matlab code, *"randn"* code has been used. This command returns a random scalar drawn from the standard normal distribution. Table 6-1 illustrates the mean value and the corresponding deviations used for circular model simulations.

	Mean value	Deviation
Width (m)	1.9	0.2
Length (m)	4.0	0.5
Height (m)	1.5	0.2
Mass (kg)	1500	100

Engine power (W)	73500	12500
Safety distance (m)	2.0	0.2
Time headway (s)	1.3	0.1
Maximum acceleration (m/s^2)	1.3	0.3
Minimum acceleration (m/s^2)	-3.5	0.4

Table 6-1. Mean values and deviations for vehicles and driving profiles parameters

The MATLAB function “Normal_distribution.m” generates the histograms that shows the distributions of each parameter for a population of thousand vehicles.

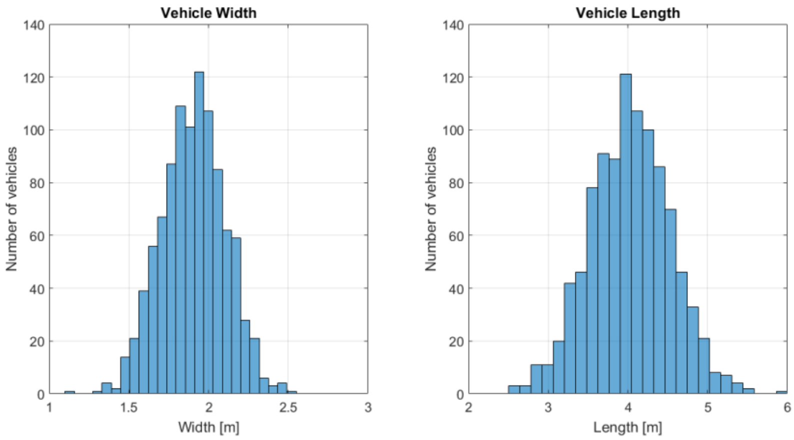


Figure 6.6 Vehicle width and length variation

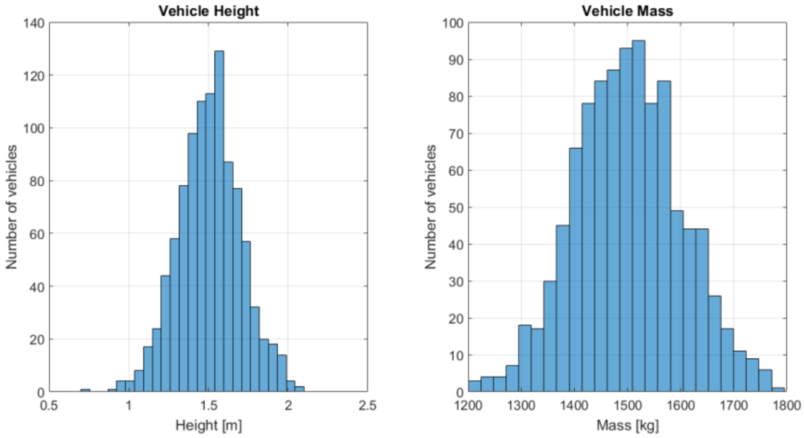


Figure 6.7 Vehicle height and mass variation

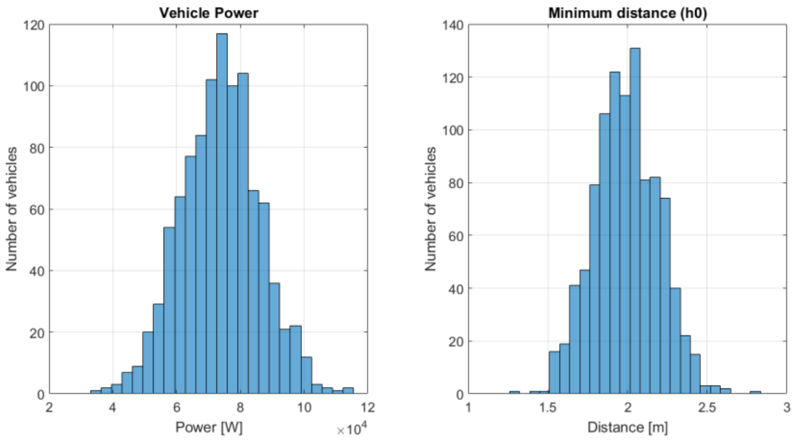


Figure 6.8 Engine power and minimum distance between vehicles variation

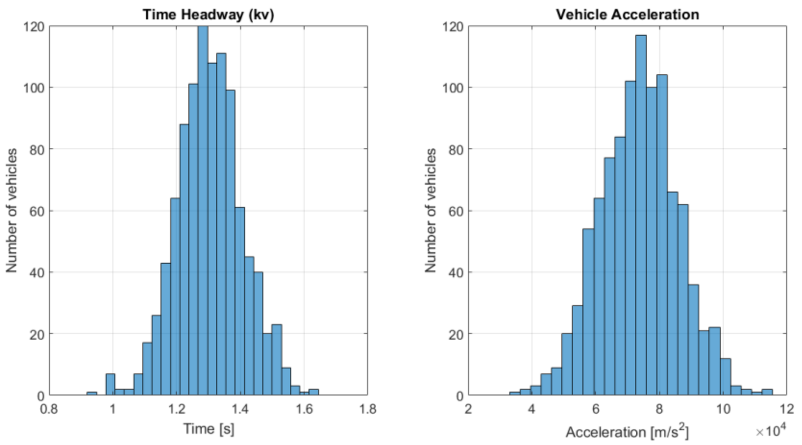


Figure 6.9 Time headway constant and acceleration variation

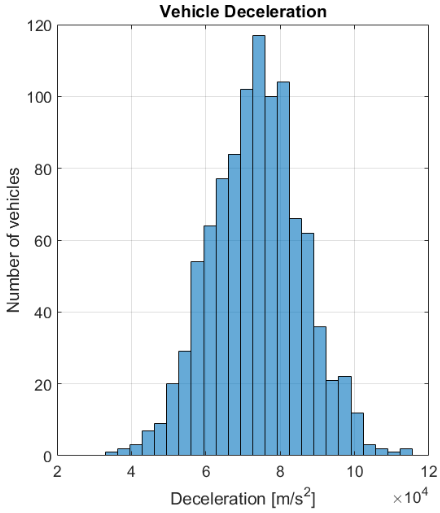


Figure 6.10. Deceleration variation

7. Simulations

In this section, the different control designs described in Section 4 are implemented. Moreover, one of the main objectives of this master's thesis is to obtain a program capable of simulate different conditions with a few parameters, being two of them, the number of vehicles and the possibility of the vehicles to transfer information related with speed and position between them. The information of the vehicles it is saved in matrix to be more easily to plot the velocity, error, among others described before.

7.1. String Stability

In this chapter, the string stability is validated when exist communications between vehicles. The term string is used to define a group of vehicles and string stability means any variable (speed, acceleration, etc) of an individual vehicle in a string do not amplify when the number of vehicles of the string increases [10].

An example which can be used to explain string stability is showed in Figure 7.1, where the leading vehicle (black line) accelerates until 30 km/h and it is possible to see different response of the following vehicles depending on whether the string is stable or not. In Figure 7.1 a), the picture shows that the platoon is string stable: the acceleration of the leading vehicle is not amplified through the following vehicles and the acceleration of following vehicle is smooth without any fluctuation of the speed. While in Figure 7.1 b), the platoon is considered of being not string stable because of the following vehicles accelerate more than the leading vehicle. Although the speed of following vehicles approach to the leading vehicles speed, their speed fluctuates too much and if the number of vehicles increase, the fluctuations increase as well. During the period of fluctuation, the distance of vehicles also fluctuates and collisions are more likely to happen, in other words, safety is reduced.

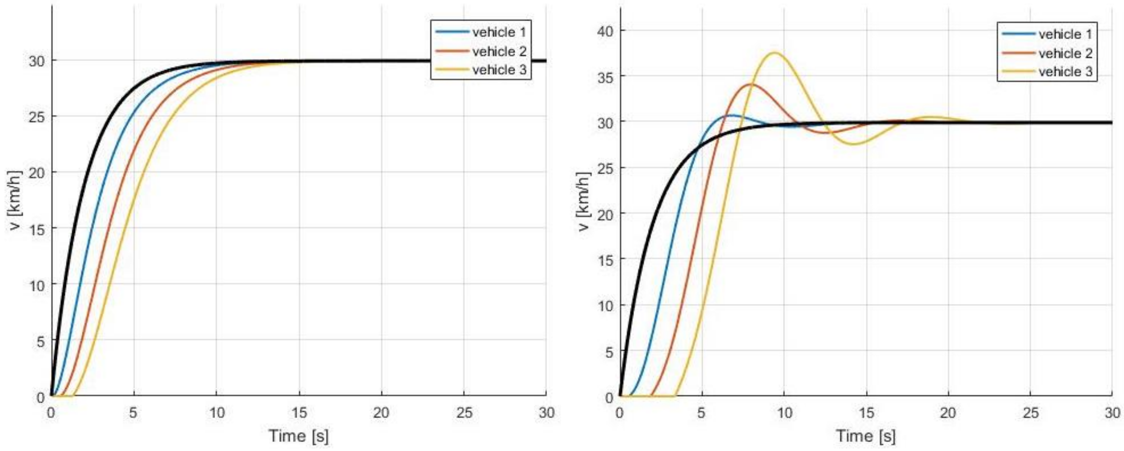


Figure 7.1 a) Vehicles with CACC; b) Vehicles without communication between them

String stability can be guaranteed if the information of the platoon leader and the preceding vehicle is used, and the information of the leader and the preceding vehicle can be collected by communication.

One important parameter that can define if a group of cars achieve the string stability is the time headway (k_v). This parameter defines the time needed for a vehicle to arrive at the position of its predecessor.

In Figure 7.3 it is possible to observe how evolve the maximum speed of vehicles string when the time headway parameter increases. If a vehicle achieves higher speed than the leader can cause a collision, so the minimum time headway is taken when the speed of the follower vehicle coincides with the leader. For example, according graphics below, for a group of 3 vehicles (one leader and two followers) with ACC the minimum time headway allow would be 1.5 second while for a group of two cars the time headway should be 0.8 seconds. On the other side, the vehicle with CACC has the same speed as vehicle leader no matter the time headway chosen.

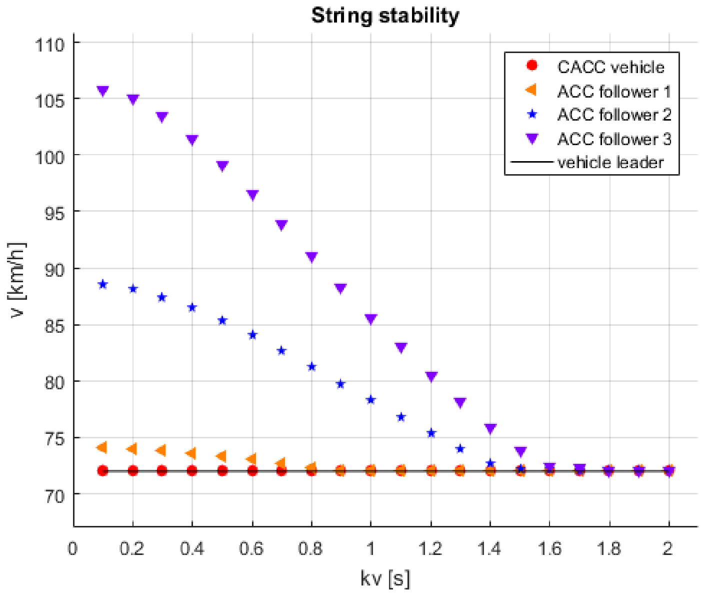


Figure 7.2. Evolution of maximum speed vs time headway parameter for different vehicles.

This parameter also plays an important role to reduce the environmental impact. As smaller it is, less space between two vehicles, and less drag force what is equivalent to a decrease fuel consumption and emissions.

String stability is an essential requirement for the design of vehicle following control systems that aim for short distance following. It has been shown that Cooperative Adaptive Cruise Control (CACC), which is based on common ACC sensors and a wireless inter-vehicle communication link, allows for time headway parameters significantly smaller than ACC vehicles while maintaining string stability [11].

As we have seen in this section, the string stability can be measured using vehicle speed. However, also other measures can be used for this purpose, such as relative position and acceleration.

7.2. First approach

To start understanding how the different controllers designed in the previous chapter work, some simple models have been implemented. This studied models are ACC, ACC+PI (ACC + dynamic extension) and CACC they are designed according to the equations (Eq 4.8), (Eq 4.12) and (Eq 4.7) respectively. All these three controllers have been designed using the “s” operator and considering one single vehicle after the leader.

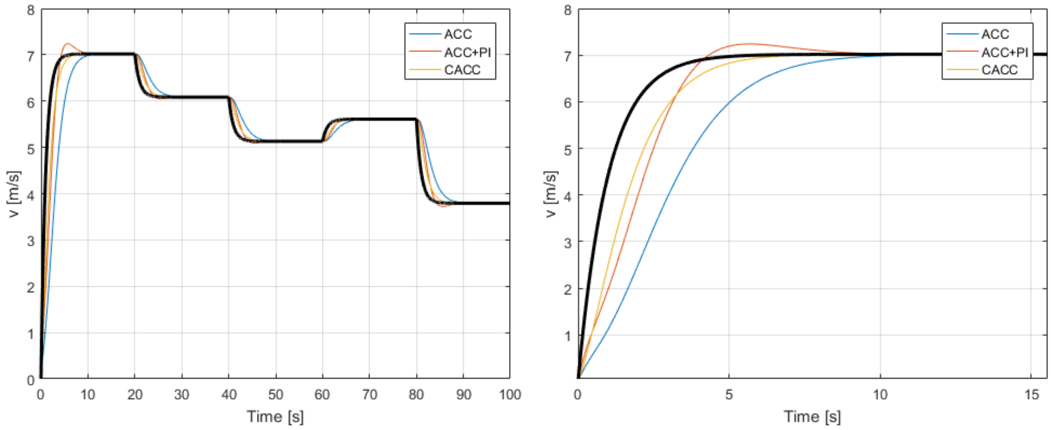


Figure 7.3 a) Speed variation for an input signal; b) Detail of speed variation

The Figure 7.3 a) shows an input signal that represent the speed of a vehicle (black line) while the other colour lines are the speed response using the controllers described above. As shown there is a difference between all controllers submitted to the same input signal. CACC have a better response than the others.

In Figure 7.3 b) it is presented a detail of the speed-time plot showing clearly the difference response for an input signal. The first impression is that all controllers reproduce quite good the input signal after a certain time. The slower controller is the ACC, followed by ACCPI and the CACC being the quickest to reach the input signal. The ACC+PI also reach the input signal without considerable delay but it has an overshoot. This problem can be solved changing the tuning of the controller but requiring an extra effort to find the correct parameters. It must be considered because this kind of result can cause a crash between to vehicles and this result must be completely banned.

As we explained before the ACC+PI does not accomplish the string stability condition. Without this condition, if the group of cars has a great number or the overshoot is too high, it is possible to have a crash between two vehicles, what is completely banned.

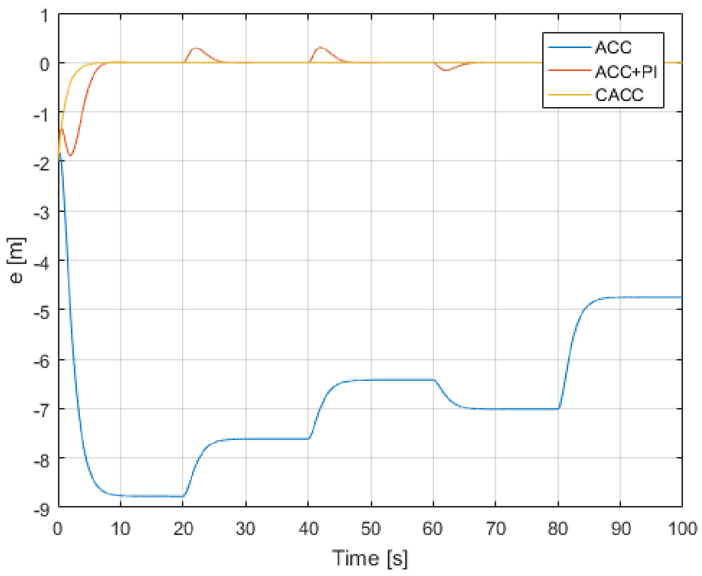


Figure 7.4 Control error

The figure above present the error defined in (Eq 4.3). It shows that in ACC+PI and CACC models there is no error while in ACC there is an error function of leader speed. The error increases the distance from zero when leader vehicle increase the speed as well.

7.3. Force controller

For clarification to the reader, the Matlab model has been explained not starting with full model. Instead, different upgrades have been added to the model. Firstly, a simply model only with the control it is showed. Secondly, the resistive forces have appended and finally, the saturations have been included to the model.

To show the differences between using a single proportional control or a proportional integral control the following simulations showed the two versions.

7.3.1. Proportional / Dynamic extension

All simulations have been made with the same parameters showed in Table 7-1 unless otherwise it is said.

Parameter	Value
Vehicles length (L)	4 m
Vehicles mass $(Mass)$	1500 kg

Safety distance (h_0)	2 m
Settling time (t_s)	5 s
Overshoot (M_p)	5 %
time headway (k_v)	1 s
Proportional gain (k_p)	0.8
Integral gain (k_z)	0.8638

Table 7-1. Vehicle and control parameters

In Figure 7.5 vehicle errors and speeds using a proportional control is shown. It is possible to see that all vehicles have the same error and it goes to zero value without overshoot.

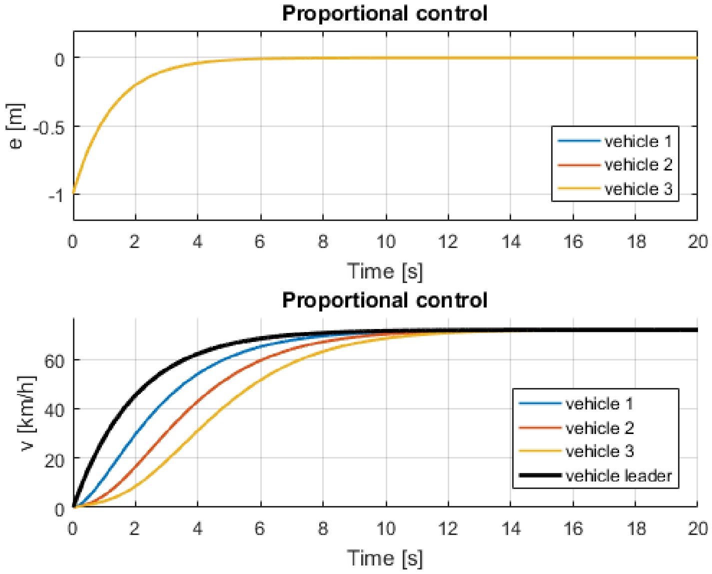


Figure 7.5. Vehicle errors and speeds using a proportional control

When an integral component is included in the controller, it can appear an overshot on the signal output as shown in Figure 7.6. The error of the three vehicles achieve zero value like pictures before and reach the vehicle leader speed faster than the proportional controller.

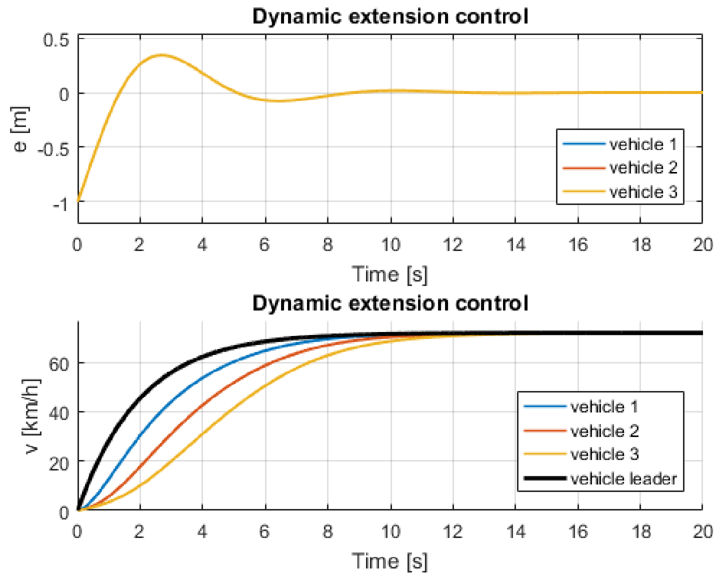


Figure 7.6. Vehicle errors and speeds using a proportional integral controller

As we mentioned before, there is a difference when the vehicles have communications between them. Figure 7.7 show the variation in the error and the vehicles speed for one vehicle that follow a leader. The vehicles that use a dynamic extension controller are represented in warm colours while in cold colours the vehicles which only use a proportional controller. The vehicles without communications are illustrated with dashed line and the cars that includes a CACC models are showed with solid line.

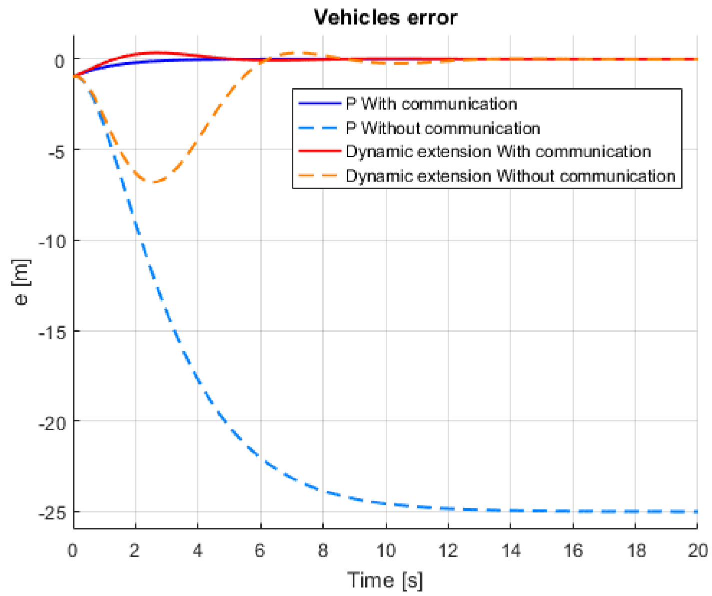


Figure 7.7 Vehicle errors with and without communications and different controller

The results show that proportional control without communications (a conventional ACC model) do not achieve the goal for the error, stablished in zero. On the other hand, the rest of the options evaluated reach the aim sooner or later. A characteristic behaviour is overbed in the dynamic extension controller without communication, decreasing the error until approximately one second to finally get the zero value.

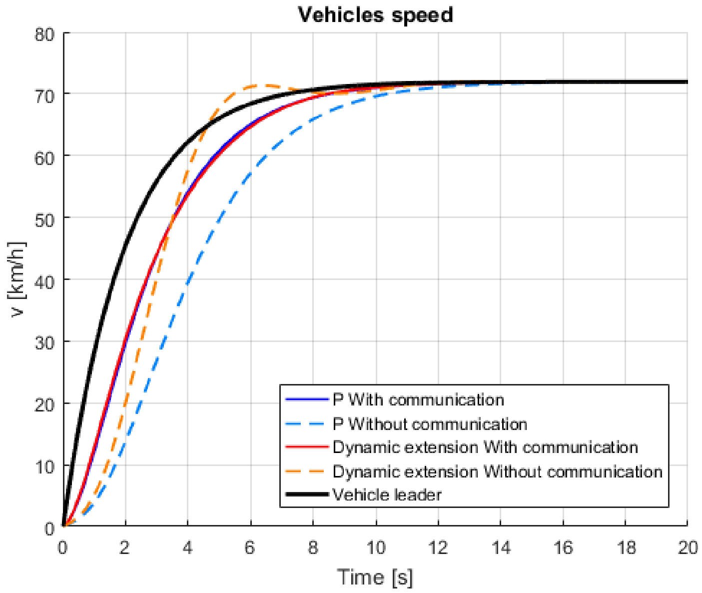


Figure 7.8 Vehicle velocity with and without communications and different controller

In Figure 7.8 is presented the vehicles speed using the same legend of picture before. It is possible to deduce different conclusions observing showed results. Firstly, the proportional controller with communications reach the vehicle leader velocity faster than the same control without communications. Moreover, both options of dynamic extension control (with and without communications) achieve the target speed, more or less at same time however, without taking account the possibility of communications among vehicles, the follower vehicle have an overshoot. As we explained before, this phenomenon could say that this model doesn't accomplish the string stability and further investigation must be done. Finally, the results between using a single proportional or a dynamic extension controller do not show significant variations.

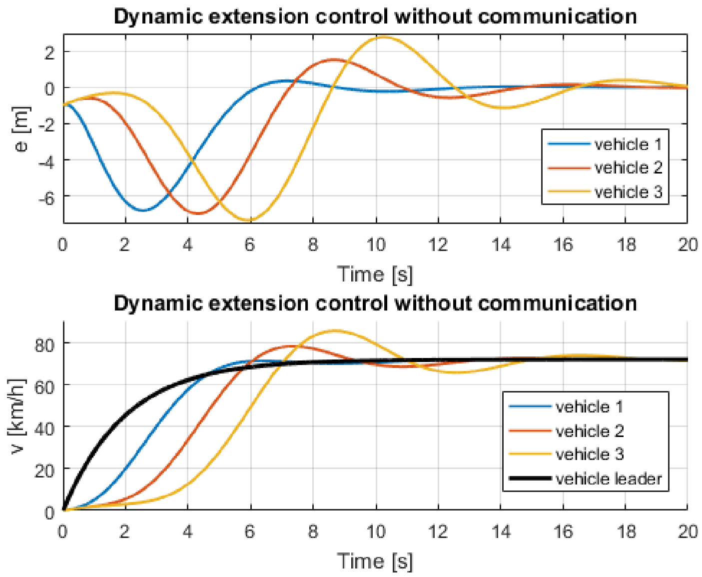


Figure 7.9. Vehicle errors and speeds using a dynamic extension controller without communications.

As mentioned previously, a study of different vehicles platooning using a proportional integral control without communications has been made to know if the model achieves the string stability. It is easy to answer the question seeing the results of Figure 7.9. The pictures clearly show how the error or the speed increases the maximum values when the number of vehicles of the platoon increases as well.

A final study has been done to ensure if the dynamic extension control without communications accomplish the string stability. The gain parameters have calculated according to (Eq. 4.22) and (Eq 4.23) and the number of vehicles has increased to five.

Parameter	Value
Proportional gain (k_p)	1.6
Integral gain (k_z)	1.3438

Table 7-2. Dynamic extension parameters

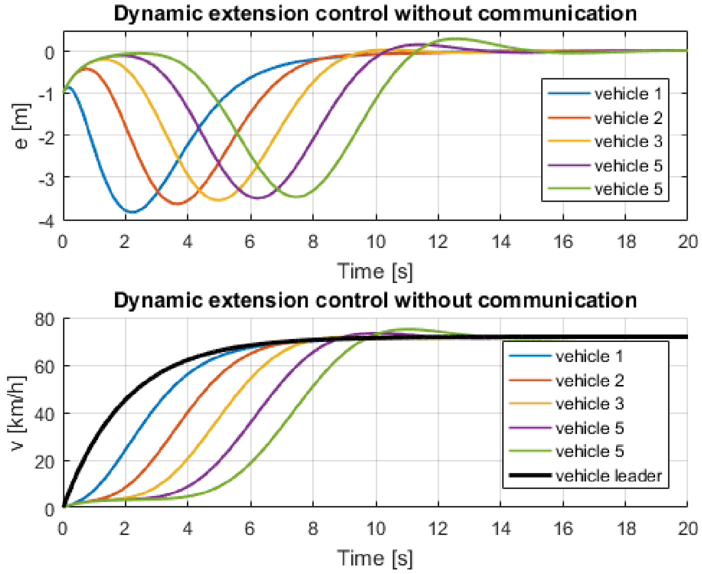


Figure 7.10. Vehicle errors and speeds using a dynamic extension controller without communications.

The previous figure shows how the gain parameters and inter-vehicle communication link influence to the vehicle response. To conclude, this model can not be used in real traffic conditions to avoid collisions between vehicles.

7.3.2. Resistive forces

In this point the resistive forces has added to the model. A block that contains the three forces explained on Chapter 5 it has been introduced after the controller. The vehicle and control parameters are the same used in Table 7-1.

Parameter	Value
Drag coefficient (C_d)	0.4
Frontal area (A_f)	2.805 m^2
Rolling resistance (f_R)	0.017
Ground inclination (α)	0 º

Table 7-3. Forces Parameters

It is possible to see on the image below that now, adding a nonlinear disturbance, like the aerodynamic drag force, appears a steady state error using proportional controller. This steady state error is the main cause of the necessity to implement a proportional integral control instead using a simpler control. The main difference between the proportional control with resistive forces compared with the proportional control without resistive forces showed in Figure 7.5, leaving aside, the steady state error, is that the error for the vehicles are not the same as we have seen before.

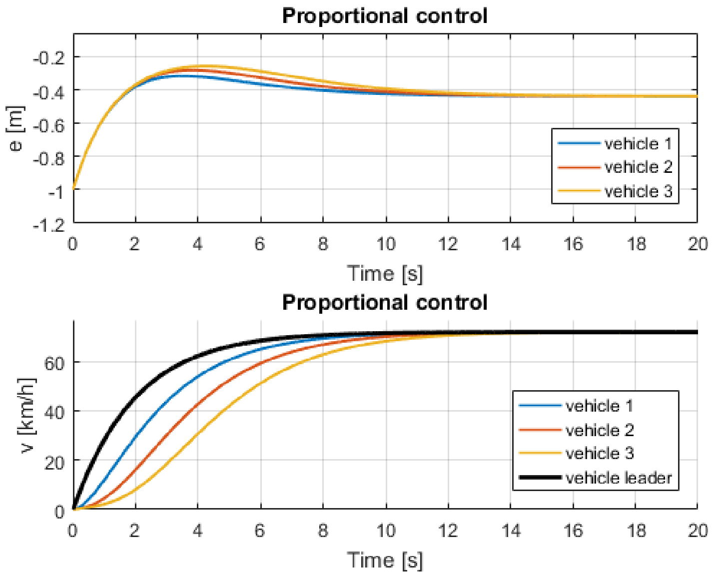


Figure 7.11. Vehicle errors and speeds using a proportional control and resistive forces included.

The result of a proportional integral control is illustrated in Figure 7.12. Adding an integral component, the steady state error observed in Figure 7.11 disappear, achieving the target value for the error. The value of the error is not the same for the different vehicles as we mentioned with the proportional control.

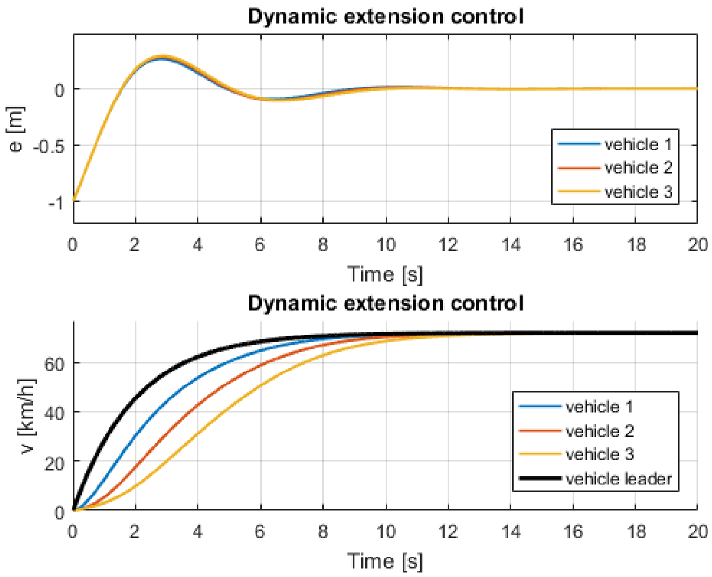


Figure 7.12. Vehicle errors and speeds using a proportional integral control and resistive forces included.

The same study has been done comparing the models including communications and the models without them.

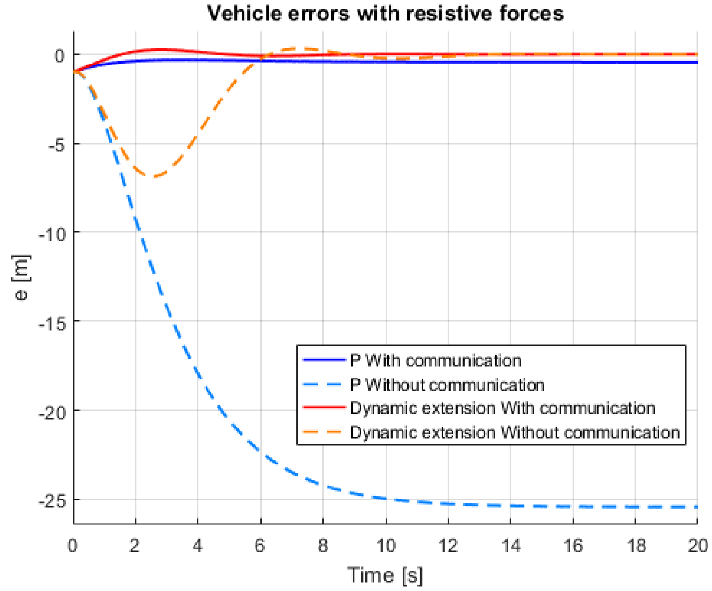


Figure 7.13. Vehicles errors with and without communications, different controller and resistive forces

The results show similar values that Chapter 7.3.1 with differences in proportional controller existing a stationary error that makes impossible to reach the value of the error.

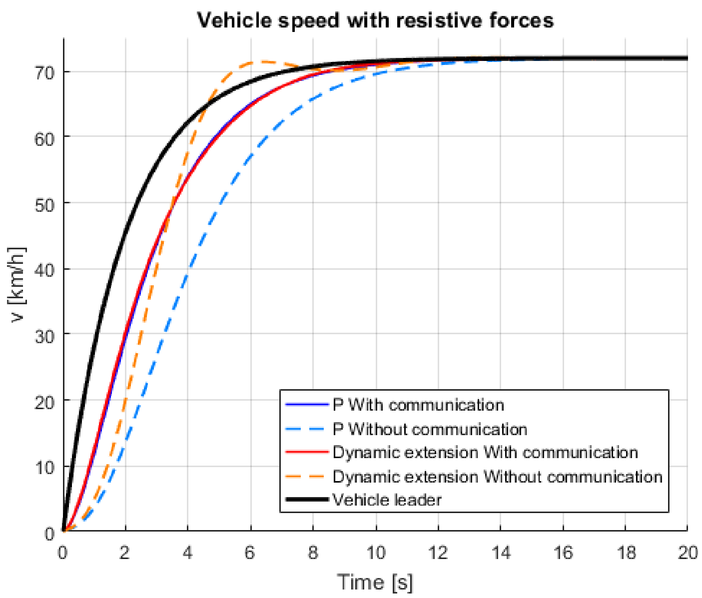


Figure 7.14. Vehicle velocity with and without communications, different controllers and resistive forces.

The graphic above present similar results as previous chapter achieving proportional and PI controllers with communications almost at the same time the speed target and PI in absence of communications reaching faster than the other controllers but having an overshoot instead. To discard this last type of control further investigation must be done.

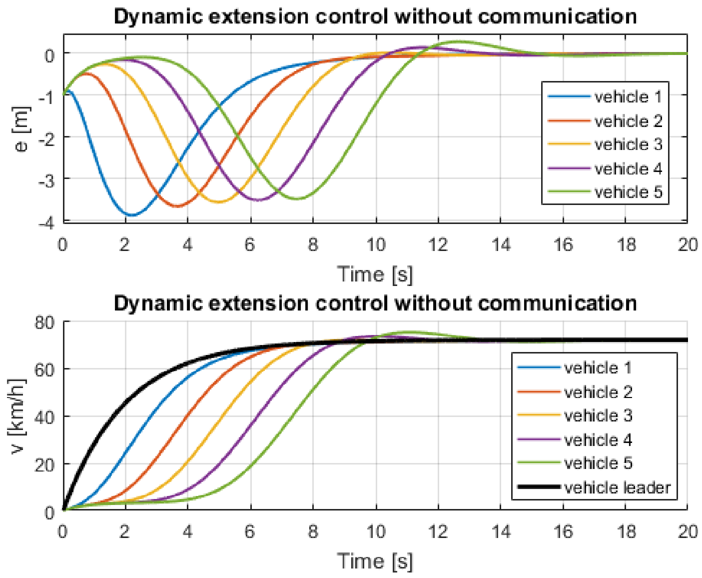


Figure 7.15. Vehicle errors and speeds using a dynamic extension controller without communications

As highlighted in the previous paragraph, a study of five vehicles forming a group using a proportional integral control without communications with the same control gains used in Table 7-2 has been made to know if the model achieves the string stability. The picture shows how maximum the error and speed augment when the number of vehicles also increase. Consequently, this controller does not accomplish the string stability.

7.3.3. Saturation

As a final step, the saturations have been added to the model. There are different saturations as we explained in previous chapters. One saturation has included to banned the possibility that cars taken a negative velocity. It is a reasonable supposition to think that vehicles can use the rear gear in a traffic jam scenario. The second saturation and third saturations are explained on tractive force chapter.

Parameter	Value
Engine power (P)	100 CV
Friction coefficient (μ)	0.7

Table 7-4 Saturation Parameters

The following simulations has been made with the parameters given in Table 7-1 and Table 7-3.

When saturation was introduced, controller stability problems were found when demanding very high accelerations from the leading vehicle and the other vehicles could not cope.

For this reason, the parameter that regulates that the leading vehicle reaches a certain speed with a set time has had to be modified. A second order equation has been introduced to control this parameter based on the maximum speed assigned to the leading vehicle. It is not uncommon to think that you are introducing a physical limitation, i. e. when you must go at a high speed, it will take you longer to reach it than if you want to reach a lower speed.

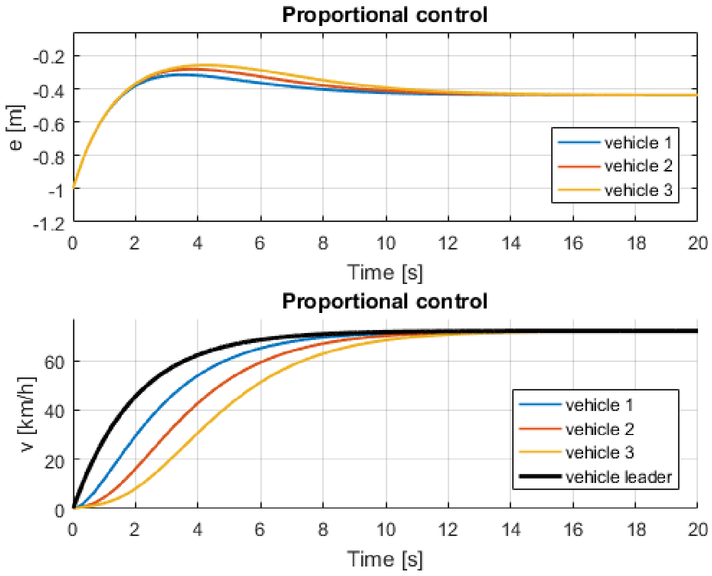


Figure 7.16. Vehicle errors and speeds using a proportional control

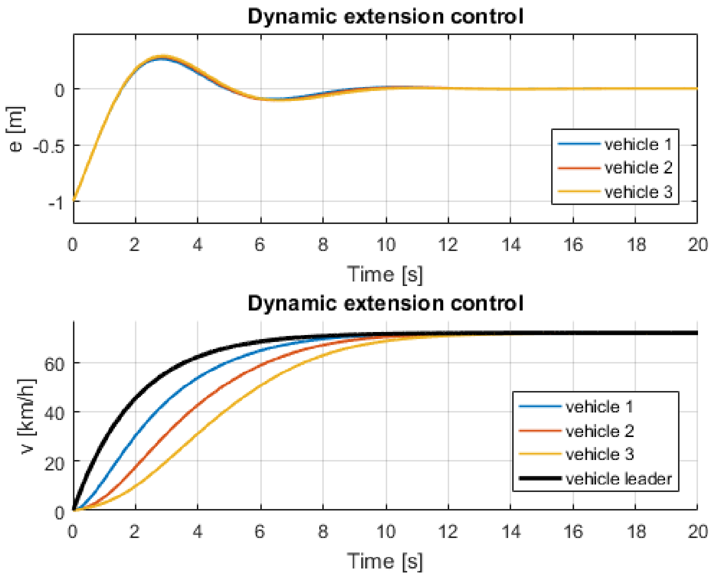


Figure 7.17. Vehicle errors and speeds using a proportional integral controller

The Figure 7.16 and Figure 7.17 do not present significant differences between the same figures of previous chapter, Figure 7.11 and Figure 7.12. Only the variations of using a curve than a step function of input signal.

To see the results of adding the saturations a simulation with a vehicle leader capable to reach a maximum speed of 180 km/h has done. The follower vehicle has been simulated with the parameters of Table 5-1.

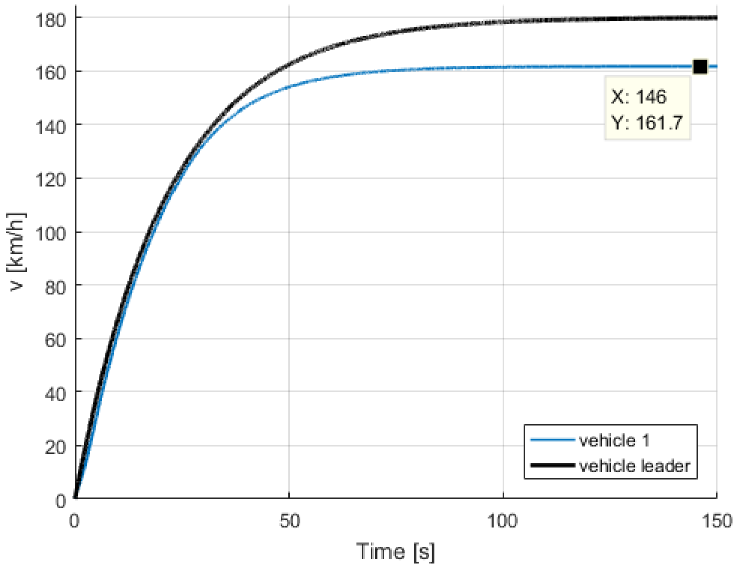


Figure 7.18. Vehicle maximum speed with saturation

On Figure 7.18 it is illustrated the vehicle leader achieving the maximum speed introduced as a target speed while the follower vehicle can not reach the velocity of the leader, obtaining an upper limit speed of 161.7 km/h according to simulation.

Using the expression (Eq 5.11) it is possible to check the results:

(1) $\frac{1}{2}\, 1.225 \cdot\ 0{,}4 \cdot\ 2{,}8005 \cdot\ v^3 + 0{,}017 \cdot 1500 \cdot 9{,}81 \cdot cos(0)v + 1500 \cdot 9{,}81 \cdot sin(0)v - 73500 = 0$

(2) $0{,}6872v^3 + 250{,}155v - 73500 = 0$

(3) $v_1 = -22{,}46 + 43{,}32i\ m/s; v_2 = -22{,}46 - 43{,}32i\ m/s; v_3 = 44{,}91\ m/s$

(4) $v_3 = 44{,}91\ m/s \rightarrow\ 161{,}9\ km/h$

7.4. Circular Model

The main idea of this chapter is to simulate how vehicles incorporating systems with communications act when they circulate with other vehicles which do not transmit information in a circular circuit Figure 7.19. In addition, it has been tried to recreate the fundamental diagrams with different percentage of vehicles with CACC.

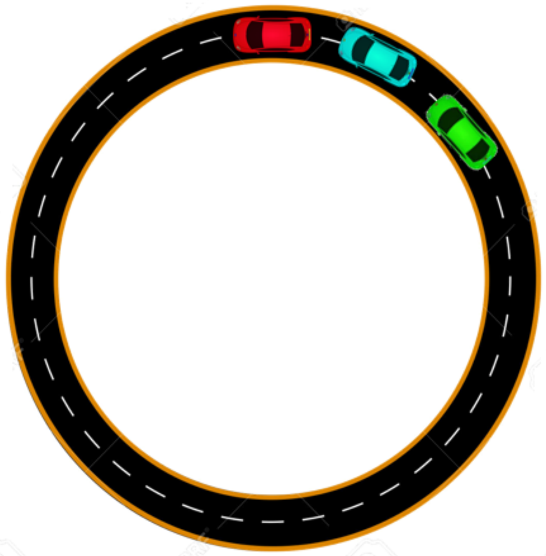


Figure 7.19. Circular model scheme.

The vehicles have been set into the circular road leaving always the same distance between them to avoid initial crashes as the previous figure shows.

7.4.1. Initial simulations

For purpose of understand the behaviour of the two systems compared (ACC and CACC) in a circular road, three simulations have been carried out. First figure shows a simulation with all the vehicles using the non-communication cruise control system, where the black line represents the average speed of all the vehicles. Secondly, Figure 7.21 show a mix of vehicles with the ACC system (dashed line) and the other half with CACC system (solid line). Finally, the last figure of the chapter represents a simulation with all the vehicles using the system that incorporate communications.

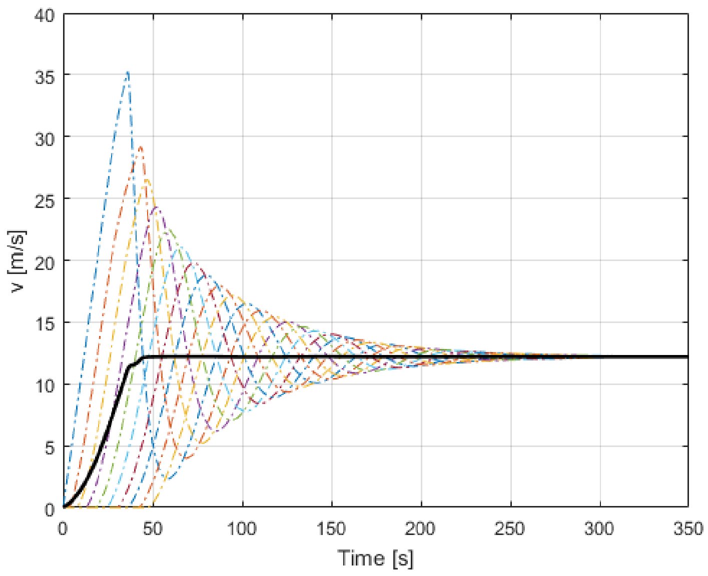


Figure 7.20. Vehicle speeds with all vehicles equipped with ACC system

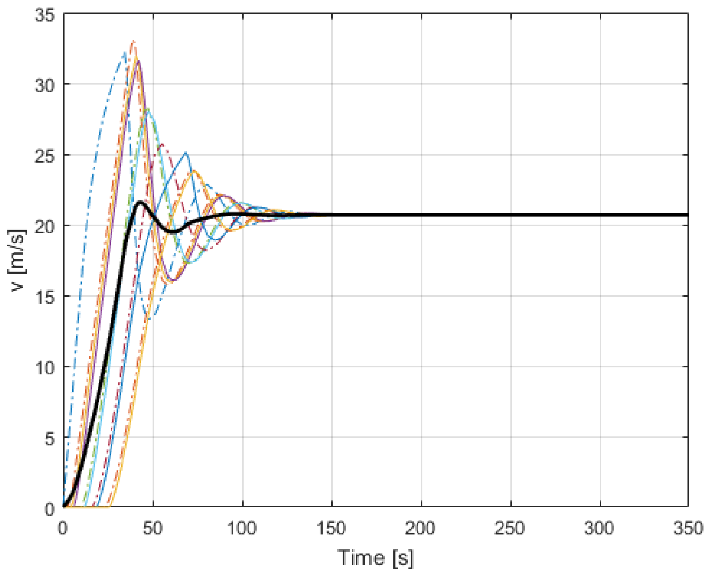


Figure 7.21. Vehicle speeds with 50% of vehicles using CACC system

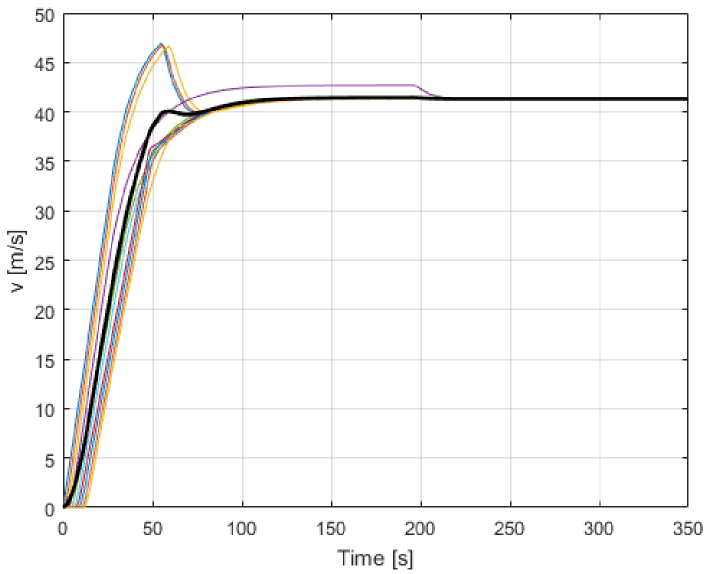


Figure 7.22. Vehicle speeds with all vehicles equipped with CACC system

It is possible to see in, Figure 7.22, how three vehicles reduce their speed. This is since the variability all the vehicles, more precisely, the vehicles power. In previous chapter, has been explained how is possible to calculate theoretically the maximum speed of a vehicle using his engine power and the force that act against its movement.

7.4.2. Traffic fluidity

Based on fundamental diagrams explained in previous chapters, varying the number of vehicles and different percentage of vehicles with CACC the traffic flow, traffic density and average speed has been evaluated. The radius of the circuit has maintained in 159 meters in order to have the circuit distance constant. The simulations have been performed increasing the number of vehicles from 2 to 120 with steps of 2 vehicles and increasing the percentage the vehicles with CACC form 0% until 100% with steps of 10%.

The results of this simulations are shown in the following graphics:

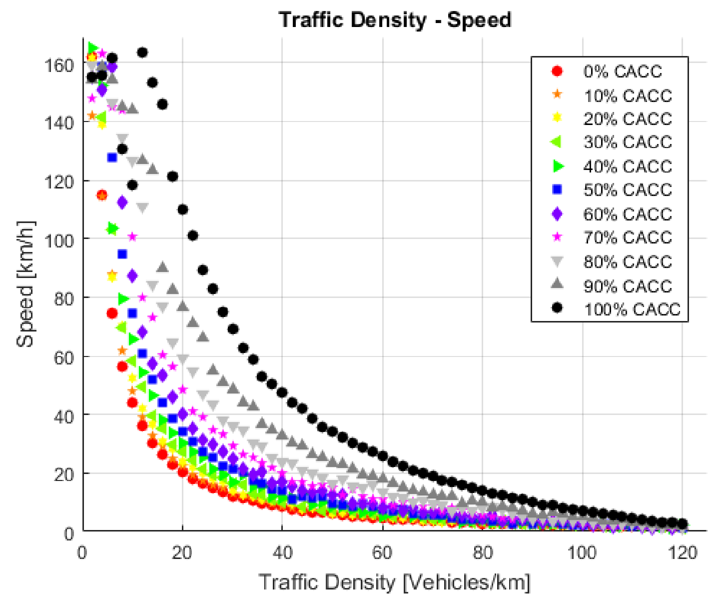


Figure 7.23. Fundamental diagram Traffic density vs. Average speed

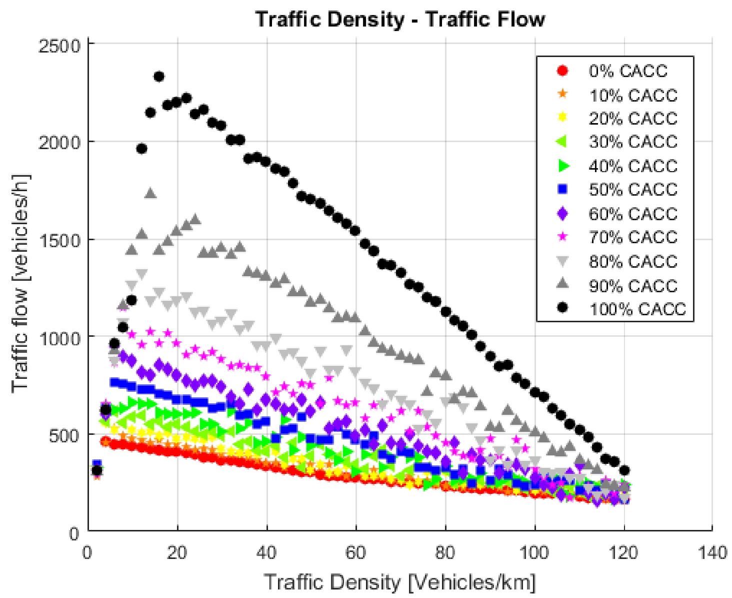


Figure 7.24. Fundamental diagram Traffic flow vs. Traffic density

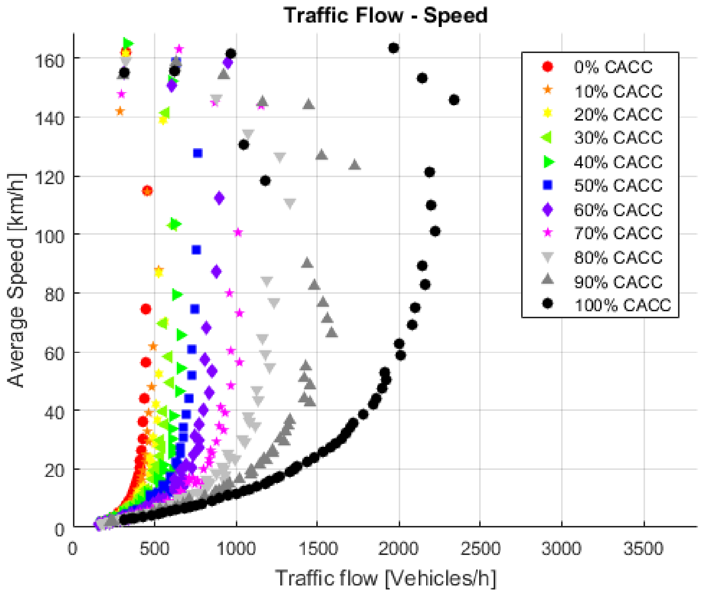


Figure 7.25. Fundamental diagram Speed vs. Traffic flow

As it can be observed in the previous figures, different percentages of vehicles equipped with CACC bring into different traffic behaviors.

The main conclusion is that the critical points commented in section 6.3 improve when increasing the percentage of CACC vehicles. Therefore, the traffic jams are reduced as increase the number of vehicles with intercommunications systems.

On traffic density – speed graphic, the vehicles that includes CACC system reach the maximum speed until 12 Veh/km. Nevertheless, when all the vehicles have ACC system it seems that with 2 Veh/km the average speed start to reduce rapidly.

A similar phenomenon is observed in Figure 7.24. The vehicles with CACC increase the traffic flow until a traffic density of 12 Veh/km and then starts decreasing while when all the vehicles have adaptive cruise control system traffic flow starts to decreasing with only 2 Veh/km.

As it is possible to see, there are some points, particularly with 100% of the vehicles including CACC system, that do not follow the general tendency. This is due to the variability in the parameters such as engine power which has been commented in past chapters. When in a circular road, without possibility to overtake, all the vehicles have limited by the maximum speed of the slower vehicle of the group. So, the discordant points of the simulations re due to vehicles with little engine power.

8. Environmental impact

The project has not carried out a detailed study of the environmental impact of introducing the cooperative adaptive cruise control system in comparison with traditional adaptive cruise control. However, it can be assumed that the increase in the percentage of vehicles with wireless communication can be assumed to reduce traffic jams as shown in the figures that recreate the fundamentals graphs. In addition, the CACC controller has the ability to reduce headway time by decreasing vehicle spacing. This would result in a significant increase in highway capacity and decrease of fuel consumption and emissions of heavy-duty vehicles can also be expected.

On the other side, it is possible to consider the environmental impact caused to create the project itself. Since this has been a theoretical project, any physical prototype has been needed. The only aspects that should be taken into account to evaluate the carbon footprint are the electrical consumption of the computer and the displacement of the engineer.

9. Planning and timing

In this chapter the timing is presented. In Figure 9.1 the forecast timing before the project started with the different actions to do is shown. It was defined five main actions. Firstly, a collection of information about the topic using technic articles and previous final graduate thesis. Secondly, the mathematic development to implement. After that, the main part of the project separated in two main blocks. On one hand, there is the time required for programing in Matlab-Simulink for linear vehicle simulations and on the other hand, appears the time necessary for programming the circular model to get the fundamental diagrams. Finally, there is the time needed to write the project.

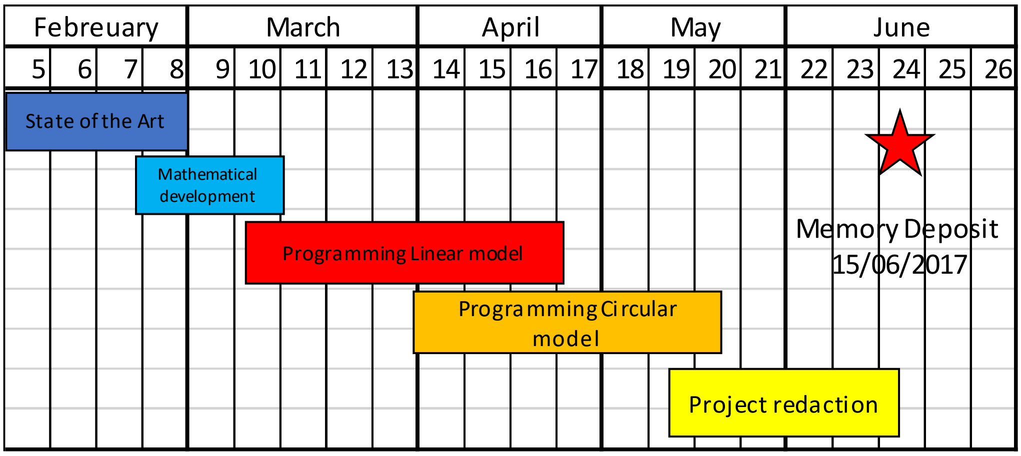


Figure 9.1. Forecast timing

It was not possible to meet the previous timing due some problems to implement the controllers in the lineal model. Since that reason all the project had been affected and delayed until to deposit the memory in the extended time.

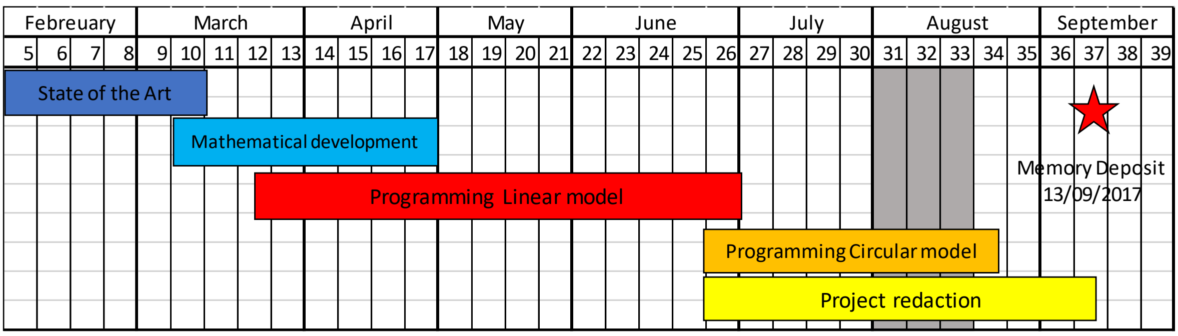


Figure 9.2. Real timing

10. Budget

The estimated budget for this project is as follows in Table 10-1.

Units	Concept	Unitary cost	Total
1	MATLAB and Simulink Student suite	2000 €	2000 € [12]
1	Simulink	3000 €	3000 € [12]
1	Microsoft Office licence	123.14€	123,14€ [13]
1	PC with mid-range performance	650 €	650 €
60	Initial research and documentation	10 €/h	600 €
250	Programming	10 €/h	2500 €
50	Project report redaction	10 €/h	500 €
Subtotal			9.373,14 €
VAT (21 %)			1.968.36€
TOTAL			**13.341,50 €**

Table 10-1 Estimated Project Budget

The salary of the engineer with a university degree has been extracted from BOE number 15 of January 18th of 2017 according to article 33 [14]. The project development budget is set at thirteen thousand, three hundred forty-one euros and fifty cents.

Conclusions

In this project, has been developed two main simulation models to recreate the behaviour of the vehicles that incorporates the adaptive cruise control system and the future generation of the same system that will incorporate communications.

The first model allows to study different control algorithms and how the tuning of the parameters affect to the cruise control system. Several options have been compared and have been validated for possible market introduction with the restriction of string stability. It has been shown that, for the same time headway, vehicles with CACC accomplish the string stability while ACC vehicle do not.

The second model analysed how vehicles perform in real traffic situations. A road model, that recreates a circular circuit has built. Several simulations have been carried out with a certain number of vehicles, varying the percentage of cars with CACC system in the same circuit with other cars with the conventional ACC system.

The main conclusion of the project is that vehicles with CACC system improves traffic flow and average speed in traffic jams. This benefits are due to faster response to reach the error set point in comparison with the system that do not incorporates communications.

Even though an in-depth study of the possible environmental impact if all the vehicles were equipped with ACC system has not been carried out, it can be considered beneficial for air pollutant levels. As shown in the fundamental graphs, traffic jams could be reduced, increasing the average speed and traffic flow maintaining the same vehicle density.

As a proposal for future research, the introduction of the possibility to overtake for certain vehicles when they find a slower car in front of them would recreate more realistically most of the current roads. Another proposal would be study how affects the inter-vehicle distance in drag force to obtain an exact environmental impact for this technology.

Acknowledgements

I would like to express my deep gratitude to Professor Arnau Doria, my project director, for their patient guidance, enthusiastic encouragement and useful critiques of this research work.

Finally, I would like to thank my girlfriend for their support and encouragement throughout my study.

Bibliography

Bibliographic references

[1] V. Milanés, S. E. Shladover, J. Spring, and C. Nowakowski, “Cooperative Adaptive Cruise Control in Real Traffic Situations,” vol. 15, no. 1, pp. 296–305, 2014.

[2] J. Cartró Benavides, “Simulació de la fluència del trànsit de vehicles comparant el comportament humà i l’impacte de vehicles autònoms,” (Bachelor’s thesis) Universitat Politècnica de Catalunya, 2017.

[3] U.S. Department of Transportation, “CACC system concept,” 2015. [Online]. Available: http://www.its.dot.gov/image_gallery/image36.htm. [Accessed: 20-Jul-2017].

[4] M. Fernandez, “Energy efficiency comparison between human drivers an adaptive curise control system,” (Bachelor’s thesis) Universitat Politència de Catalunya, 2017.

[5] R. Rajamini, *Vehicle Dynamics and Control*, 2nd ed. Minneapolis: Springer, 2006.

[6] M. Roshan, “Why do internal combustion vehicle engines require transmission gears to optimize power output but electrical vehicle motors (i.e. cars, locomotives) don’t?,” *Quora*, 2015. [Online]. Available: https://www.quora.com/Why-do-internal-combustion-vehicle-engines-require-transmission-gears-to-optimize-power-output-but-electrical-vehicle-motors-i-e-cars-locomotives-dont. [Accessed: 04-Jul-2017].

[7] D. Gallegos, “Neumáticos. Resistencia a la rodadura.” Universitat Politècnica de Catalunya.

[8] “Projected Frontal Area.” [Online]. Available: http://photobucket.com/gallery/user/aerohead2/media/cGF0aDovVW50aXRsZWQ0XzI3LmpwZw==/?ref=. [Accessed: 10-Jul-2017].

[9] S. Hoogendoorn and V. Knoop, “Traffic flow theory and modelling,” in *The transport system and transport policy: an introduction*, Edward Elg., B. Van Wee, J. A. Annema, and D. Banister, Eds. Cheltenham, UK, 2013, p. 399.

[10] C. Lei, “Cooperative Adaptive Cruise Control model study based on traffic & network simulation,” no. January. pp. 5–9, 2011.

[11] J. Ploeg, B. T. M. Scheepers, E. Van Nunen, N. Van De Wouw, and H. Nijmeijer, “Design and Experimental Evaluation of Cooperative Adaptive Cruise Control,” 2011.

[12] “New License for MATLAB R2017a - MathWorks España.” [Online]. Available: https://es.mathworks.com/store/link/products/standard/ML. [Accessed: 09-Sep-2017].

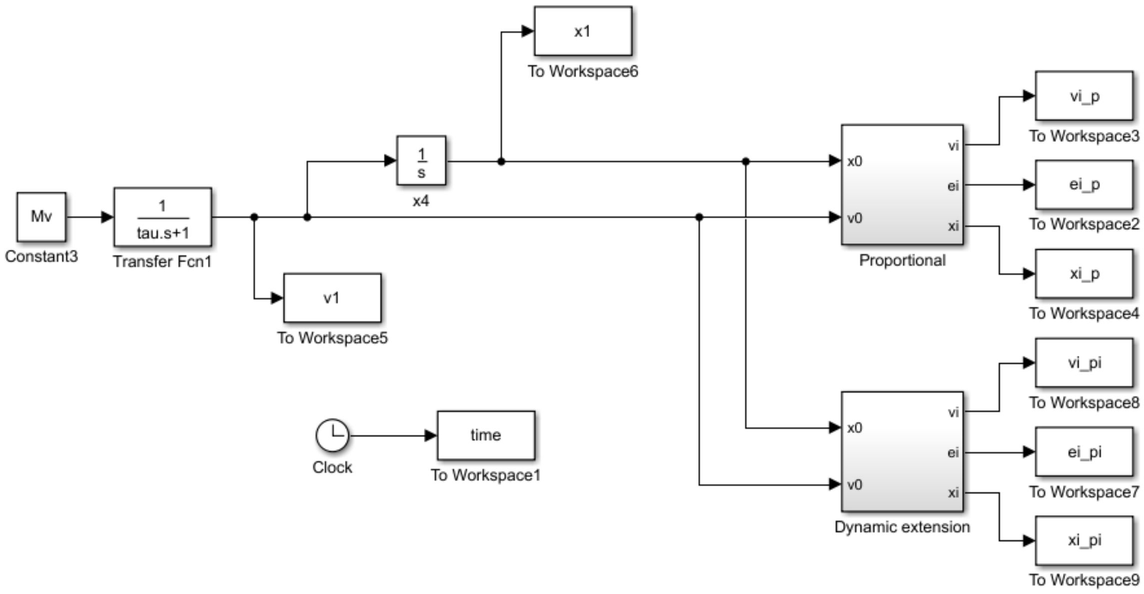
[13] “Microsoft Office para el hogar, estudiantes y profesionales: Tienda Microsoft España.” [Online]. Available: https://www.microsoft.com/es-es/store/b/office?icid=CNavSoftwareOffice&invsrc=search&OCID=AID620886_SEM_WXYvowAAAHCBgQPA:20170909070718:s&ef_id=WXYvowAAAHCBgQPA:20170909070718:s&s_kwcid=AL!4249!3!214112895586!p!!!!microsoft

office&ef_id=WXYvowAAAHCBgQPA. [Accessed: 09-Sep-2017].

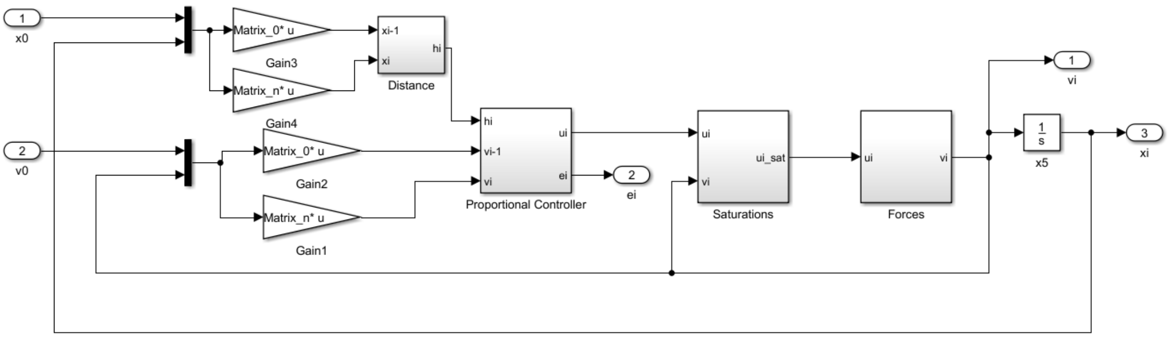
[14] "«BOE» núm. 15, de 18 de enero de 2017, páginas 4356 a 4382." pp. 11370–11421, 2017.

Annex A. Simulink model

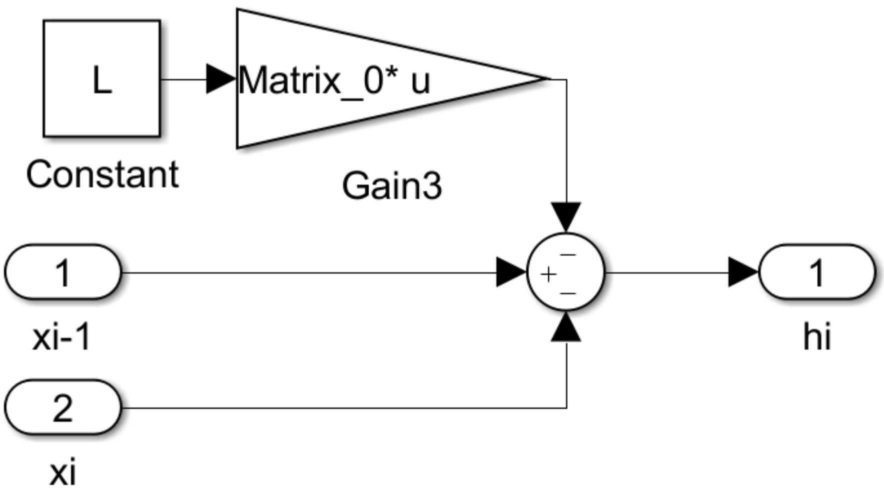
A1. General view. Speed_and_error_n_vehicles



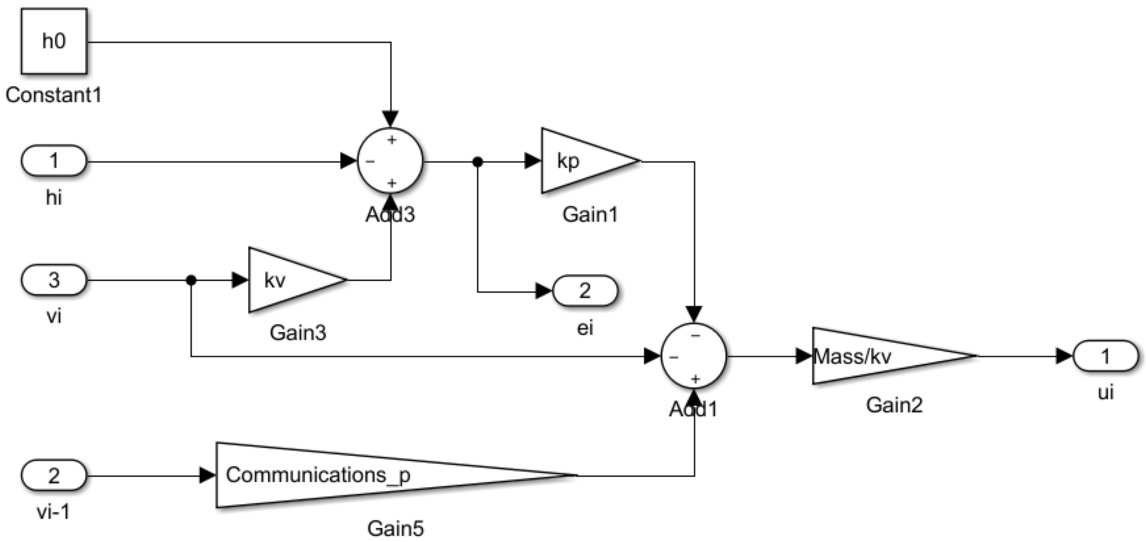
A1.1. Proportional system



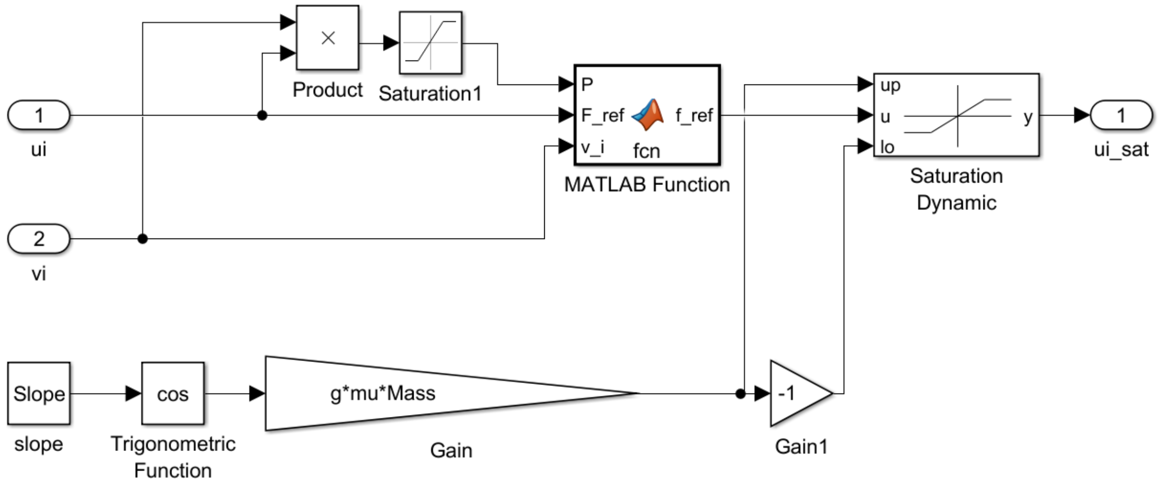
A1.2. Proportional system. Distance calculation.



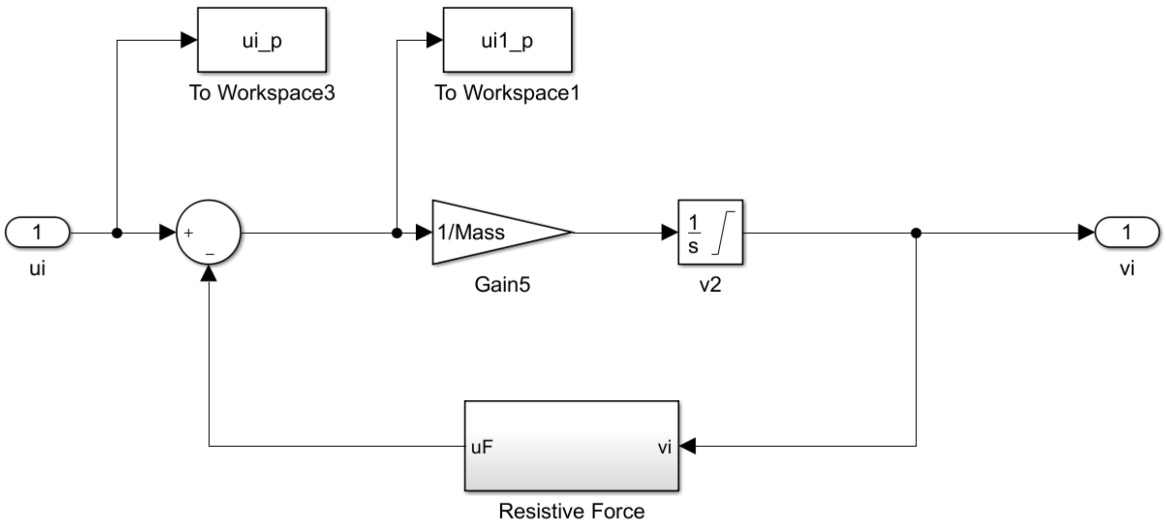
A1.3 Proportional system. Controller



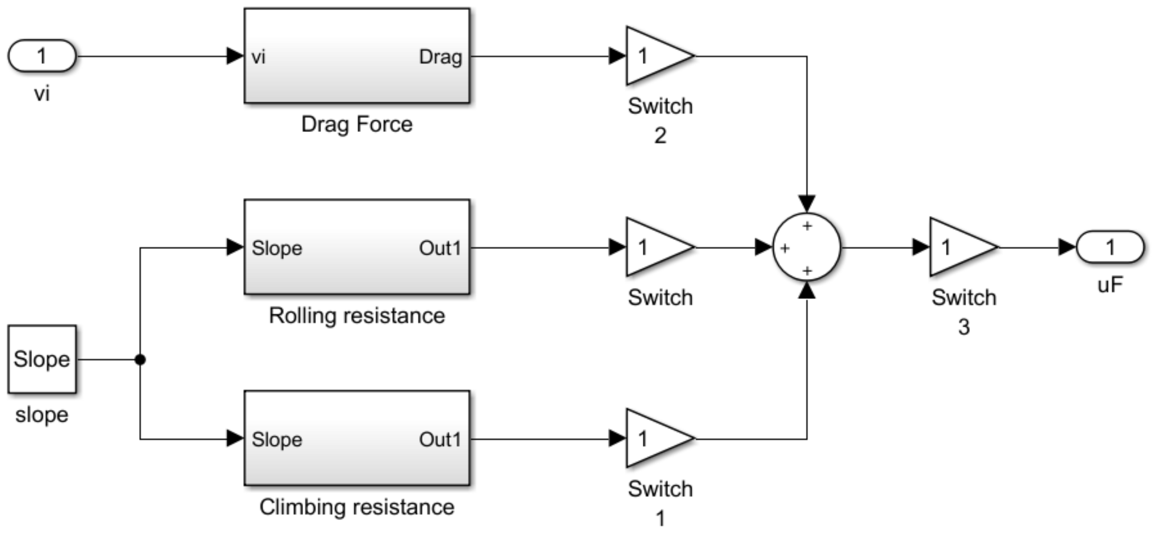
A1.4. Proportional system. Saturations.



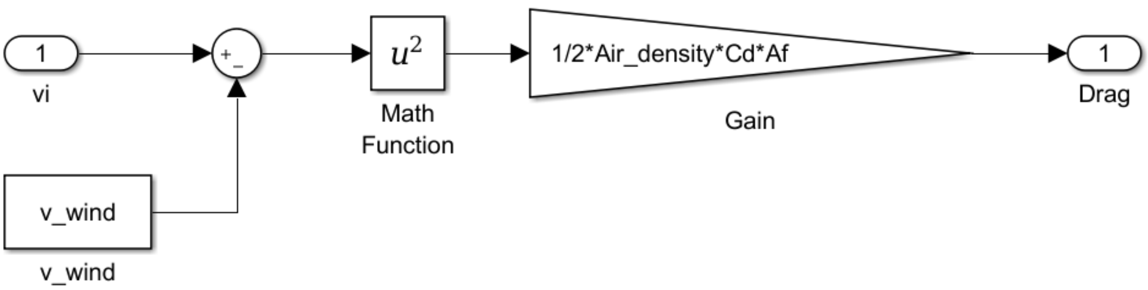
A1.5. Proportional system. Forces



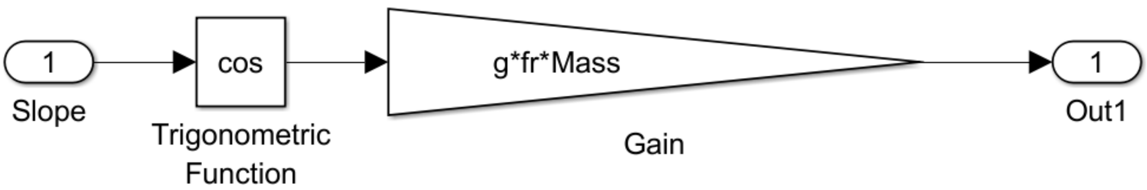
A1.6. Proportional system. Forces. Resistive force



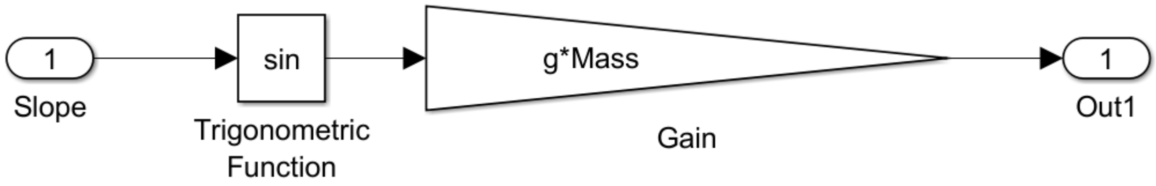
A1.7 Proportional system. Forces. Resistive force. Drag force



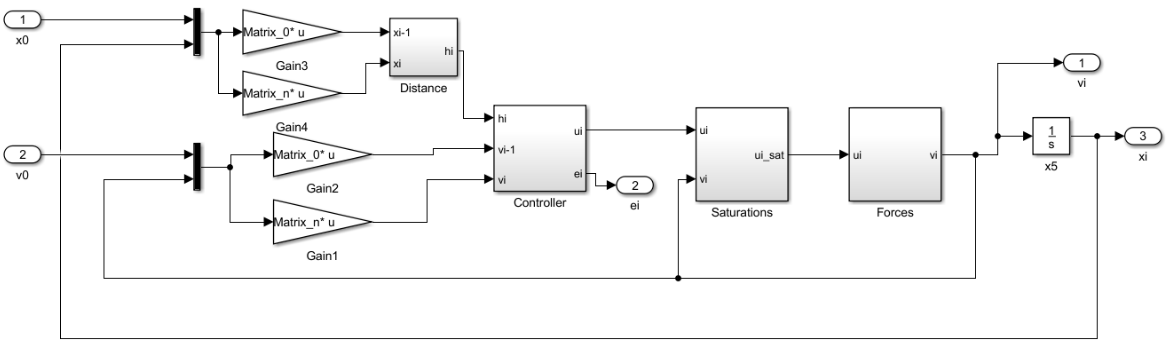
A1.8 Proportional system. Forces. Resistive force. Rolling resistance



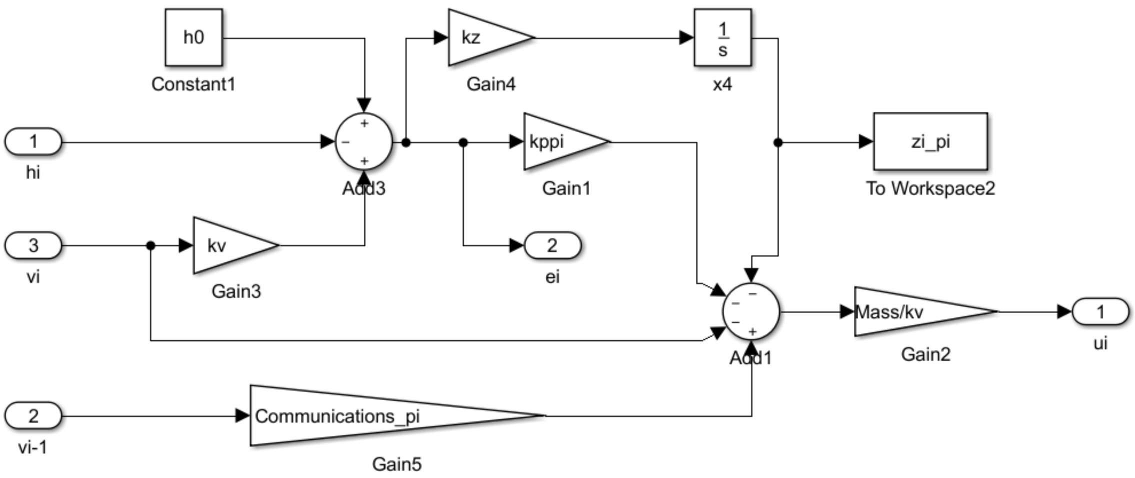
A1.9 Proportional system. Forces. Resistive force. Climbing resistance



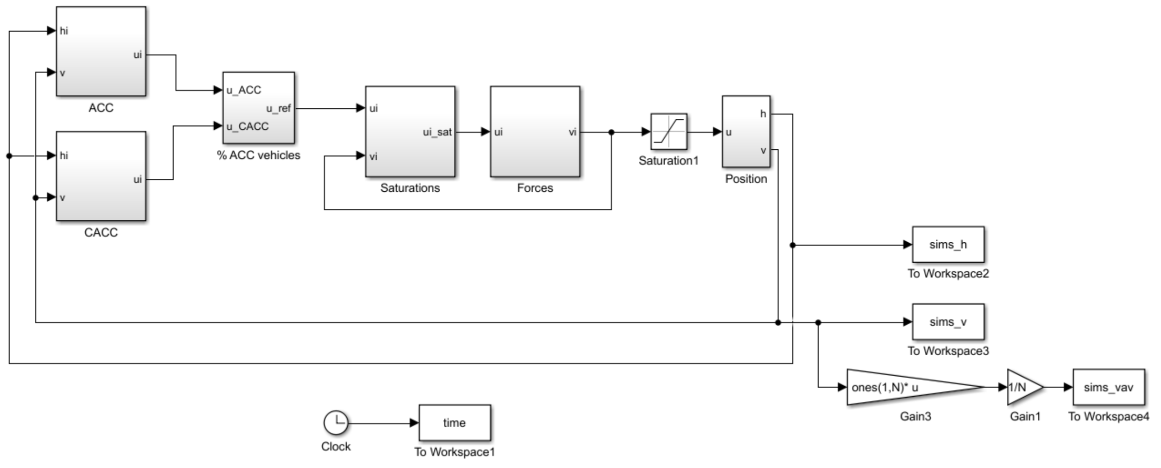
A1.10. Dynamic extension system



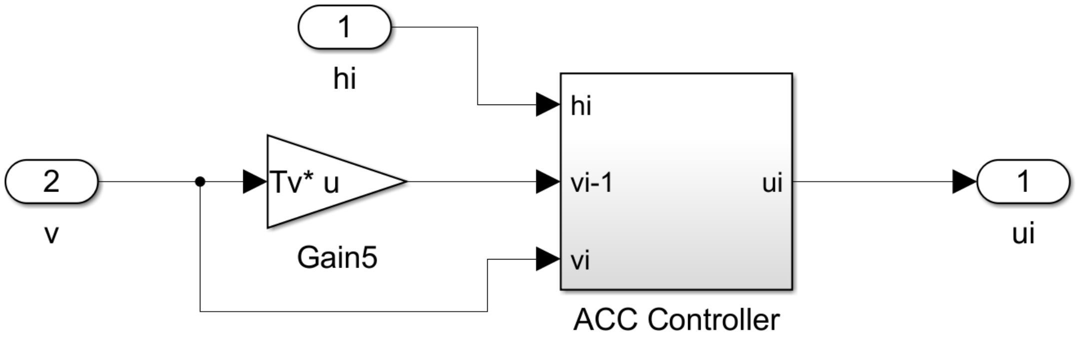
A1.11. Dynamic extension system. Controller



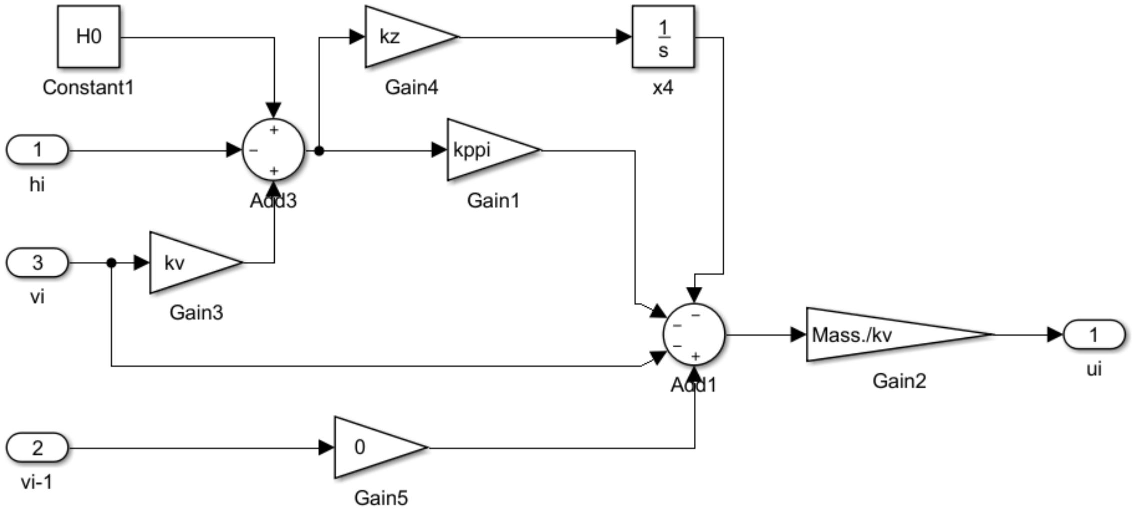
A2. General overview. Circular_model



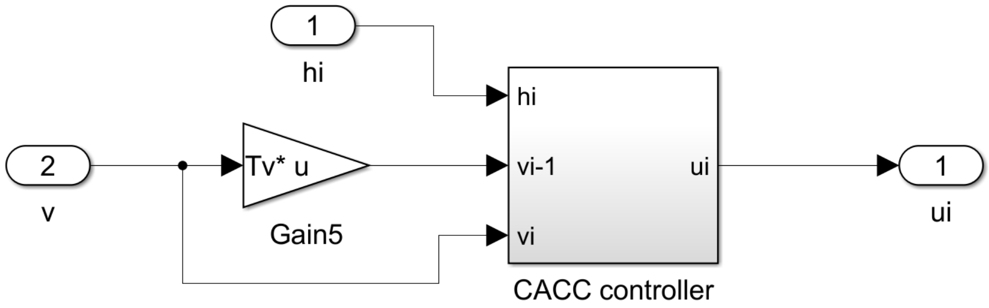
A2.1. ACC.system.



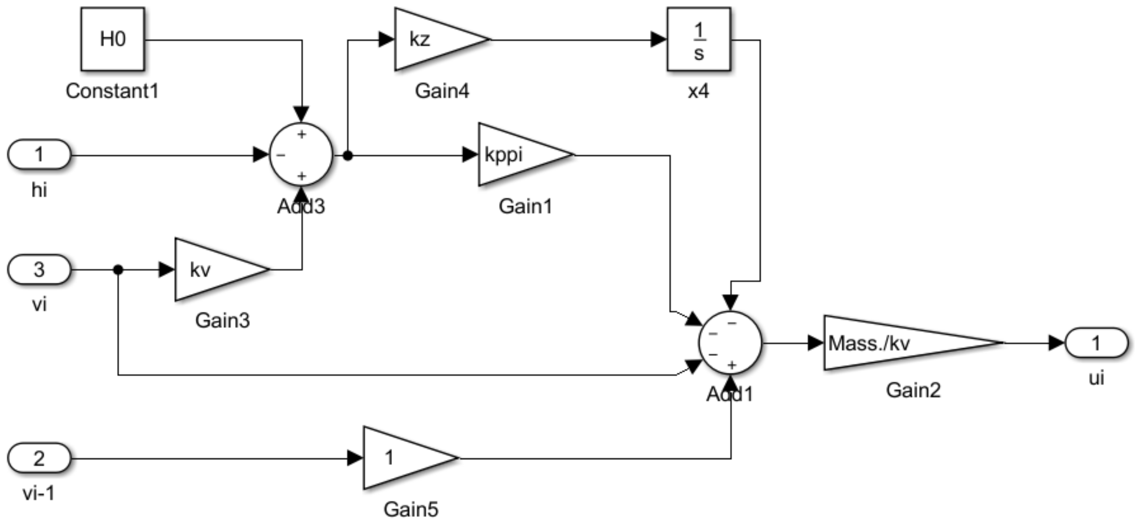
A2.2. ACC system. ACC controller.



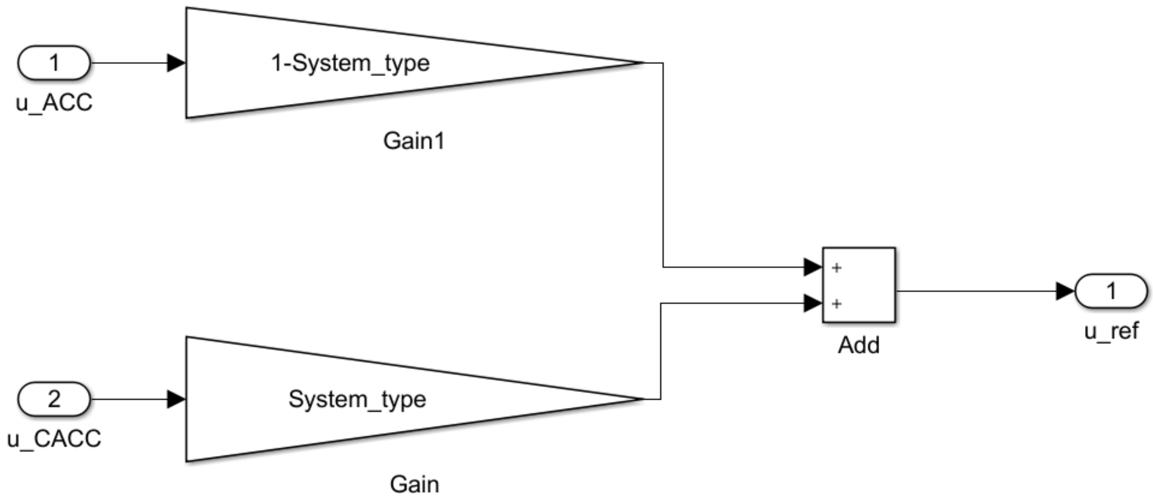
A2.3. CACC.system



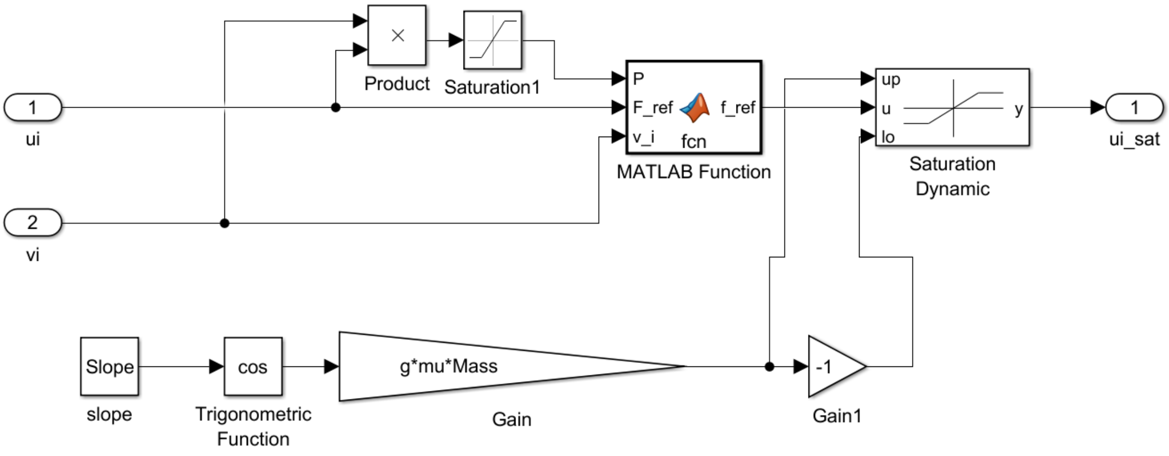
A2.4. CACC system. CACC controller



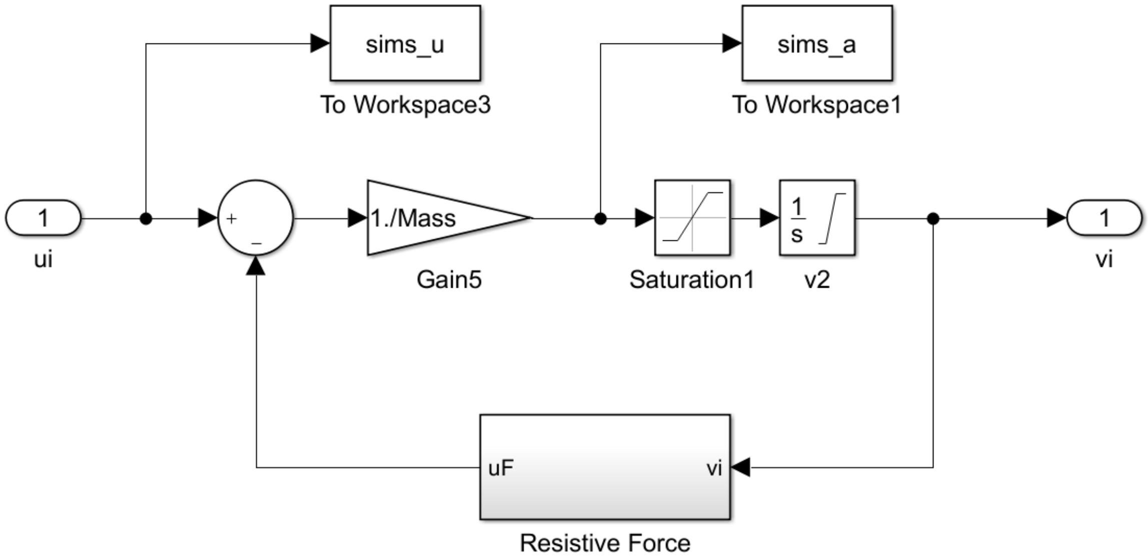
A2.5. % ACC vehicles



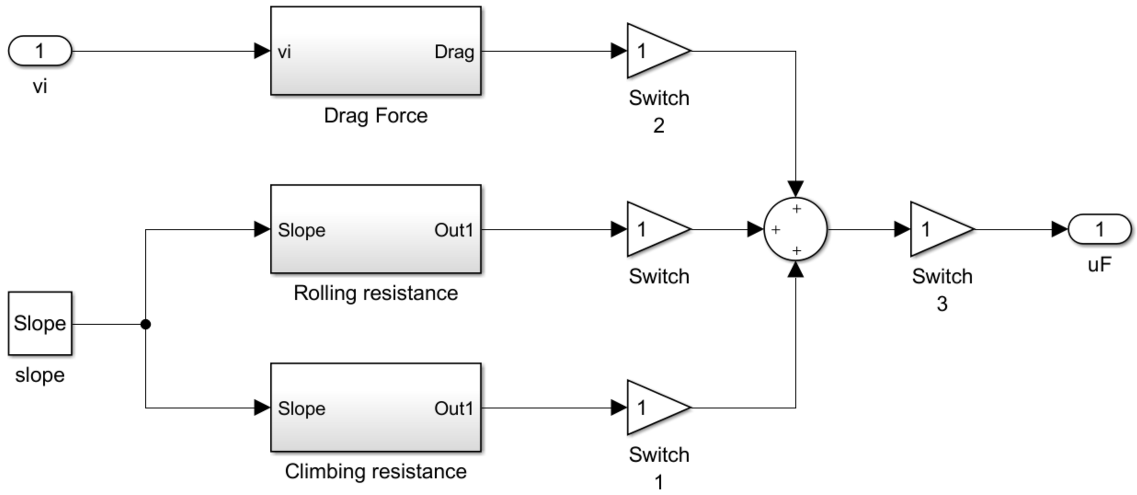
A2.6. Saturations



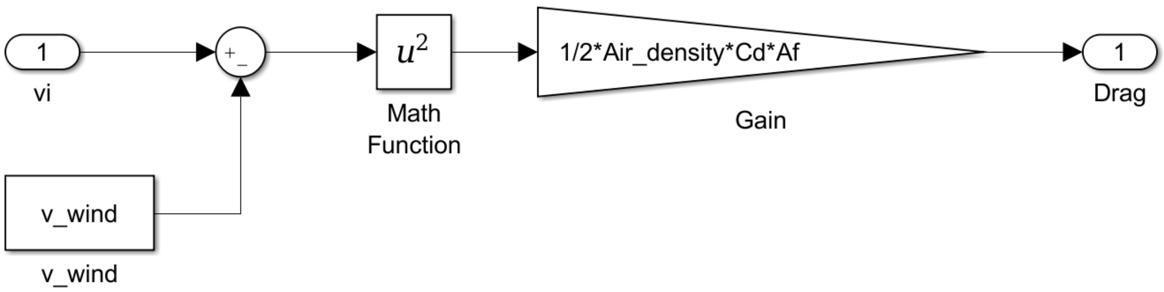
A2.7 Forces.



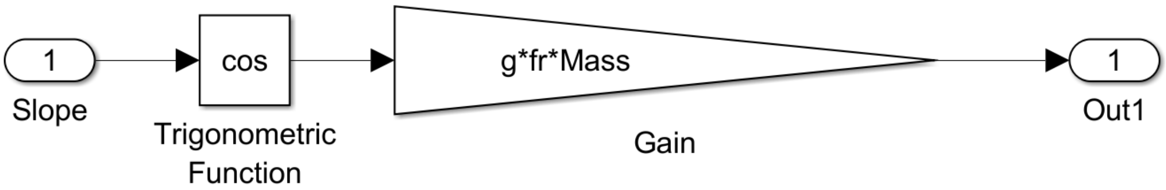
A2.8. Forces. Resistive forces.



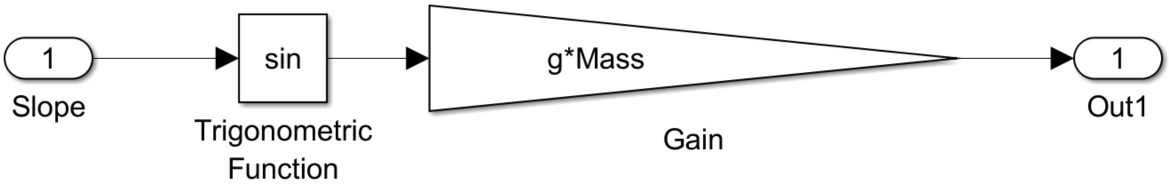
A2.9. Forces. Resistive forces. Drag force



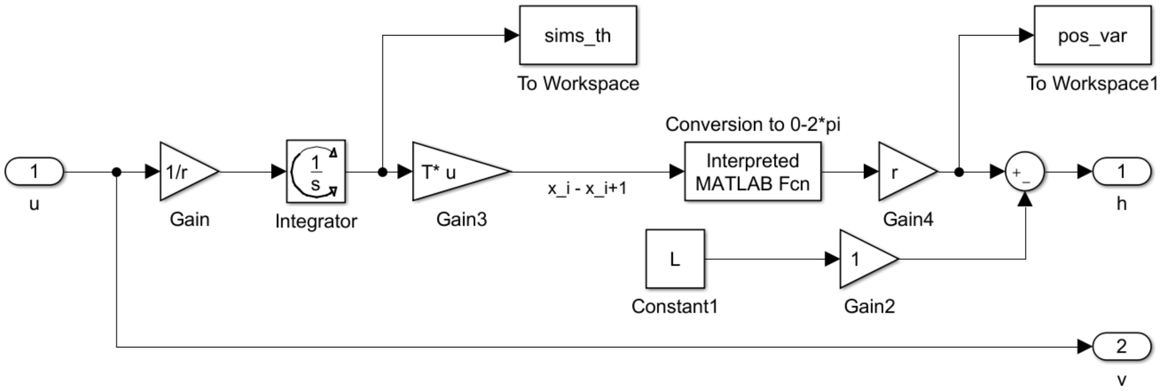
A2.10. Forces. Resistive forces. Rolling resistance



A2.11. Forces. Resistive forces. Climbing resistance



A2.12. Position calculation



Annex B. Matlab code

B1. Speed_and_error_run_n_vehicles.m

```
clear all;close all; clc

% Parameters
C=1e3;
r=C/2/pi;
l=4;
h0=2;
Nmax=C/(l+h0);
vmax=22*10; %22
kp=0.1; %0.1
%% Control parameter
Mp=0.05; % overshoot
ts=5; % settling time (s)
sigma = 4/ts;
omega_d= -sigma*pi()/log(Mp); %damped natural frequency (Hz)

%kp=4/ts;
kppi=2*4/ts*0.1;
%kppi = kp;
%kp = kppi;
kz=(omega_d^2+kppi^2/4)*0;

%% RESISTANCE FORCES
%% ----------------------------------------------------------------------
--
%% Drag Force
v_wind = 0;
Air_density = 1.225; % Air density (kg/m^3)
Cd = 0.4;           % Drag coeficient
%% Rolling Resistance Force & Climbing resistance
g = 9.81;           % Gravity constant (m/s^2)
fr = 0.017;         % Rolling resistance coeficient
Slope = 0;

%% Road frictional coeficient
mu = 0.7;
%% Vehicles parameters
% Average Values
Average_Minimum_Space = 2; % h0 [m]
Average_Time_Headway = 1.3; % kv [s]
Average_Vehicles_Acceleration = 1.3; % amax [m/s^2]
Average_Vehicles_Deceleration = -3.5; % amin [m/s^2]
Average_Mass = 1500; %[kg]
Average_Width = 1.9; %[m];
Average_Height = 1.5; %[m];
Average_Longitude = 4; %[m]
Average_Power = 73500; %[W]

% Deviation values
Deviation_Minimum_Space = 0.2; % [m]
```

```
Deviation_Time_Headway = 0.1; % [s]
Deviation_Acceleration = 0.3; % [m/s^2]
Deviation_Vehicles_Deceleration = 0.4; % [m/s^2]
Deviation_Mass = 100; %[kg]
Deviation_Width = 0.2; %[m];
Deviation_Heigth = 0.2; %[m];
Deviation_Longitude = 0.5; %[m]
Deviation_Power = 12500; %[W]
%% Simulation parameter
tfinal=350;
maxstepsize=1e-1;
%%
tic
j=1;

ii = 1;

for P_cacc=0:10:100 %percentage of CACC vehicles
    jj = 1;
    %P_cacc = 10;
    for N=5:5:120 %120 %number of vehicles
        %N=5;
        clear T
        clear Tv
        clear th0

        v0=0*ones(N,1); %Initial speed

        Vmax=vmax*ones(N,1); %Maximum speed
        % Vehicles parameters
        L = Average_Longitude+randn(N,1)*Deviation_Longitude;
        H0 = Average_Minimum_Space+randn(N,1)*Deviation_Minimum_Space;
        amax =
Average_Vehicles_Acceleration+randn(N,1)*Deviation_Acceleration;
        amin =
Average_Vehicles_Deceleration+randn(N,1)*Deviation_Vehicles_Deceleration;
        kv = Average_Time_Headway+randn(N,1)*Deviation_Time_Headway;
        Mass = Average_Mass+randn(N,1)*Deviation_Mass;
        Power_vehicles = Average_Power+randn(N,1)*Deviation_Power;
        Width_vehicles = Average_Width+randn(N,1)*Deviation_Width;
        Height_vehicles = Average_Height+randn(N,1)*Deviation_Heigth;
        Af = 0.85 * Width_vehicles .* Height_vehicles;

        %Percentage of vehicles with CACC system
        n_vehicles_CACC = round(N*P_cacc/100);
        random_vector = randperm(N)';
        System_type = zeros(N,1);
        System_type(random_vector(1:n_vehicles_CACC)) = 1;

        %Vehicles Initial Position
        initial_distance = 0.1;
        th0(1)=0;
        for e = 2:N
            th0(e)= th0(e-1)+initial_distance+L(e);
        end
        th0=th0/r;
```

```
        th0=fliplr(th0);

        for i=1:N
%              th0(i)=(1+0.1)/r*(N-i);
            T(i,i)=-1;
            if i==1
                T(i,N)=1;
                Tv(i,N)=1;
            else
                T(i,i-1)=1;
                Tv(i,i-1)=1;
            end
        end

        sim('Circular_model')
        D(ii,jj)=N/C;
        Vav(ii,jj)=sims_vav(end)
        disp(['End simulation num.' num2str(j) ';' '  Vehicles percentage 
CACC: ' num2str(P_cacc) '% ;' ' (' num2str(N) ' vehicles)'])
        j=j+1;

        jj=jj+1;
    end
    ii=ii+1;
    end
toc

%sum(sims_h(end,:))+l*N

%% Continuity equation
u = Vav*3.6; %Speed [km/h]
k = D*1e3; %Traffic Density [Vehicles/km]
q = u.*k; %Traffic Flow [Vehicles/h]
%%
%Plots
close all
figure (2)
for i=1:N
    if System_type(i)==1
        plot(time,sims_th(:,i),'-.')
        hold on
    else
       plot(time,sims_th(:,i))
       hold on
    end
end
grid on
xlabel('Time [s]')
ylabel('\theta [rad]')

figure (3)
for i=1:N
    if System_type(i)==1
        plot(time,sims_h(:,i),'-.')
        hold on
    else
```

```
        plot(time,sims_h(:,i))
        hold on
    end
end
%plot(time,sims_h1)
grid on
xlabel('Time [s]')
ylabel('h [m]')

figure (4)
for i=1:N
    if System_type(i)==1
        plot(time,sims_v(:,i),'-.')
        hold on
    else
        plot(time,sims_v(:,i))
        hold on
    end
end
%plot([0 time(end)],[vmax vmax],'g','Linewidth',2)
plot(time,sims_vav,'k','Linewidth',2)
grid on
xlabel('Time [s]')
ylabel('v [m/s]')

%% Fundamental diagram Plots
marker_color = {'[1 0 0]','[1 0.5 0]','[1 1 0]','[0.5 1 0]','[0 1 0]','[0
0 1]','[0.5 0 1]','[1 0 1]','[0.75 0.75 0.75]','[0.5 0.5 0.5]','[0 0
0]'};

figure
hold on
scatter(k(1,:),u(1,:),'MarkerFaceColor',marker_color{1},'MarkerEdgeColor'
,'none','Marker','o');
scatter(k(2,:),u(2,:),'MarkerFaceColor',marker_color{2},'MarkerEdgeColor'
,'none','Marker','p');
scatter(k(3,:),u(3,:),'MarkerFaceColor',marker_color{3},'MarkerEdgeColor'
,'none','Marker','h');
scatter(k(4,:),u(4,:),'MarkerFaceColor',marker_color{4},'MarkerEdgeColor'
,'none','Marker','<');
scatter(k(5,:),u(5,:),'MarkerFaceColor',marker_color{5},'MarkerEdgeColor'
,'none','Marker','>');
scatter(k(6,:),u(6,:),'MarkerFaceColor',marker_color{6},'MarkerEdgeColor'
,'none','Marker','s');
scatter(k(7,:),u(7,:),'MarkerFaceColor',marker_color{7},'MarkerEdgeColor'
,'none','Marker','d');
scatter(k(8,:),u(8,:),'MarkerFaceColor',marker_color{8},'MarkerEdgeColor'
,'none','Marker','p');
scatter(k(9,:),u(9,:),'MarkerFaceColor',marker_color{9},'MarkerEdgeColor'
,'none','Marker','v');
scatter(k(10,:),u(10,:),'MarkerFaceColor',marker_color{10},'MarkerEdgeCol
or','none','Marker','^');
scatter(k(11,:),u(11,:),'MarkerFaceColor',marker_color{11},'MarkerEdgeCol
or','none','Marker','o');
grid on
xlabel('Traffic Density [Vehicles/km]')
ylabel('Speed [km/h]')
```

```
axis([0,max(k(11,:))+5,0,max(u(11,:))+5]);
legend('0% CACC','10% CACC','20% CACC','30% CACC','40% CACC','50%
CACC','60% CACC','70% CACC','80% CACC','90% CACC','100% CACC');
title('Traffic Density - Speed')
hold off
%
figure
hold on
scatter(k(1,:),q(1,:),'MarkerFaceColor',marker_color{1},'MarkerEdgeColor'
,'none','Marker','o');
scatter(k(2,:),q(2,:),'MarkerFaceColor',marker_color{2},'MarkerEdgeColor'
,'none','Marker','p');
scatter(k(3,:),q(3,:),'MarkerFaceColor',marker_color{3},'MarkerEdgeColor'
,'none','Marker','h');
scatter(k(4,:),q(4,:),'MarkerFaceColor',marker_color{4},'MarkerEdgeColor'
,'none','Marker','<');
scatter(k(5,:),q(5,:),'MarkerFaceColor',marker_color{5},'MarkerEdgeColor'
,'none','Marker','>');
scatter(k(6,:),q(6,:),'MarkerFaceColor',marker_color{6},'MarkerEdgeColor'
,'none','Marker','s');
scatter(k(7,:),q(7,:),'MarkerFaceColor',marker_color{7},'MarkerEdgeColor'
,'none','Marker','d');
scatter(k(8,:),q(8,:),'MarkerFaceColor',marker_color{8},'MarkerEdgeColor'
,'none','Marker','p');
scatter(k(9,:),q(9,:),'MarkerFaceColor',marker_color{9},'MarkerEdgeColor'
,'none','Marker','v');
scatter(k(10,:),q(10,:),'MarkerFaceColor',marker_color{10},'MarkerEdgeCol
or','none','Marker','^');
scatter(k(11,:),q(11,:),'MarkerFaceColor',marker_color{11},'MarkerEdgeCol
or','none','Marker','o');
grid on
xlabel('Traffic Density [Vehicles/km]')
ylabel('Traffic flow [vehicles/h]')
axis([0,max(k(11,:))+5,0,max(q(11,:))+100]);
legend('0% CACC','10% CACC','20% CACC','30% CACC','40% CACC','50%
CACC','60% CACC','70% CACC','80% CACC','90% CACC','100% CACC');
title('Traffic Density - Traffic Flow')
hold off
%
figure
hold on
scatter(q(1,:),u(1,:),'MarkerFaceColor',marker_color{1},'MarkerEdgeColor'
,'none','Marker','o');
scatter(q(2,:),u(2,:),'MarkerFaceColor',marker_color{2},'MarkerEdgeColor'
,'none','Marker','p');
scatter(q(3,:),u(3,:),'MarkerFaceColor',marker_color{3},'MarkerEdgeColor'
,'none','Marker','h');
scatter(q(4,:),u(4,:),'MarkerFaceColor',marker_color{4},'MarkerEdgeColor'
,'none','Marker','<');
scatter(q(5,:),u(5,:),'MarkerFaceColor',marker_color{5},'MarkerEdgeColor'
,'none','Marker','>');
scatter(q(6,:),u(6,:),'MarkerFaceColor',marker_color{6},'MarkerEdgeColor'
,'none','Marker','s');
scatter(q(7,:),u(7,:),'MarkerFaceColor',marker_color{7},'MarkerEdgeColor'
,'none','Marker','d');
scatter(q(8,:),u(8,:),'MarkerFaceColor',marker_color{8},'MarkerEdgeColor'
,'none','Marker','p');
```

```
scatter(q(9,:),u(9,:),'MarkerFaceColor',marker_color{9},'MarkerEdgeColor'
,'none','Marker','v');
scatter(q(10,:),u(10,:),'MarkerFaceColor',marker_color{10},'MarkerEdgeCol
or','none','Marker','^');
scatter(q(11,:),u(11,:),'MarkerFaceColor',marker_color{11},'MarkerEdgeCol
or','none','Marker','o');
grid on
xlabel('Traffic flow [Vehicles/h]')
ylabel('Average Speed [km/h]')
axis([0,max(q(11,:))+5,0,max(u(11,:))+5]);
legend('0% CACC','10% CACC','20% CACC','30% CACC','40% CACC','50%
CACC','60% CACC','70% CACC','80% CACC','90% CACC','100% CACC');
title('Traffic Flow - Speed')
hold off
```

B2. Circular_model.m

```
 clear all;close all; clc

% Parameters
C=1e3;
r=C/2/pi;
l=4;
h0=2;
Nmax=C/(l+h0);
vmax=22*10; %22
kp=0.1; %0.1
%% Control parameter
Mp=0.05; % overshoot
ts=5; % settling time (s)
sigma = 4/ts;
omega_d= -sigma*pi()/log(Mp); %damped natural frequency (Hz)

%kp=4/ts;
kppi=2*4/ts*0.1;
%kppi = kp;
%kp = kppi;
kz=(omega_d^2+kppi^2/4)*0;

%% RESISTANCE FORCES
%% -----------------------------------------------------------------------
--
%% Drag Force
v_wind = 0;
Air_density = 1.225; % Air density (kg/m^3)
Cd = 0.4;        % Drag coeficient
%% Rolling Resistance Force & Climbing resistance
g = 9.81;        % Gravity constant (m/s^2)
fr = 0.017;      % Rolling resistance coeficient
Slope = 0;

%% Road frictional coeficient
mu = 0.7;
```

```
%% Vehicles parameters
% Average Values
Average_Minimum_Space = 2; % h0 [m]
Average_Time_Headway = 1.3; % kv [s]
Average_Vehicles_Acceleration = 1.3; % amax [m/s^2]
Average_Vehicles_Deceleration = -3.5; % amin [m/s^2]
Average_Mass = 1500; %[kg]
Average_Width = 1.9; %[m];
Average_Height = 1.5; %[m];
Average_Longitude = 4; %[m]
Average_Power = 73500; %[W]

% Deviation values
Deviation_Minimum_Space = 0.2; % [m]
Deviation_Time_Headway = 0.1; % [s]
Deviation_Acceleration = 0.3; % [m/s^2]
Deviation_Vehicles_Deceleration = 0.4; % [m/s^2]
Deviation_Mass = 100; %[kg]
Deviation_Width = 0.2; %[m];
Deviation_Heigth = 0.2; %[m];
Deviation_Longitude = 0.5; %[m]
Deviation_Power = 12500; %[W]
%% Simulation parameter
tfinal=350;
maxstepsize=1e-1;
%%
tic
j=1;

ii = 1;

for P_cacc=0:10:100 %percentage of CACC vehicles
    jj = 1;
    %P_cacc = 10;
    for N=5:5:120 %120 %number of vehicles
        %N=5;
        clear T
        clear Tv
        clear th0

        v0=0*ones(N,1); %Initial speed

        Vmax=vmax*ones(N,1); %Maximum speed
        % Vehicles parameters
        L = Average_Longitude+randn(N,1)*Deviation_Longitude;
        H0 = Average_Minimum_Space+randn(N,1)*Deviation_Minimum_Space;
        amax =
Average_Vehicles_Acceleration+randn(N,1)*Deviation_Acceleration;
        amin =
Average_Vehicles_Deceleration+randn(N,1)*Deviation_Vehicles_Deceleration;
        kv = Average_Time_Headway+randn(N,1)*Deviation_Time_Headway;
        Mass = Average_Mass+randn(N,1)*Deviation_Mass;
        Power_vehicles = Average_Power+randn(N,1)*Deviation_Power;
        Width_vehicles = Average_Width+randn(N,1)*Deviation_Width;
        Height_vehicles = Average_Height+randn(N,1)*Deviation_Heigth;
        Af = 0.85 * Width_vehicles .* Height_vehicles;
```

```
        %Percentage of vehicles with CACC system
        n_vehicles_CACC = round(N*P_cacc/100);
        random_vector = randperm(N)';
        System_type = zeros(N,1);
        System_type(random_vector(1:n_vehicles_CACC)) = 1;

        %Vehicles Initial Position
        initial_distance = 0.1;
        th0(1)=0;
        for e = 2:N
            th0(e)= th0(e-1)+initial_distance+L(e);
        end
        th0=th0/r;
        th0=fliplr(th0);

        for i=1:N
%               th0(i)=(l+0.1)/r*(N-i);
            T(i,i)=-1;
            if i==1
                T(i,N)=1;
                Tv(i,N)=1;
            else
                T(i,i-1)=1;
                Tv(i,i-1)=1;
            end
        end

        sim('Circular_model')
        D(ii,jj)=N/C;
        Vav(ii,jj)=sims_vav(end)
        disp(['End simulation num.' num2str(j) ';' '  Vehicles percentage
CACC: ' num2str(P_cacc) '% ;' ' (' num2str(N) ' vehicles)'])
        j=j+1;

        jj=jj+1;
    end
    ii=ii+1;
    end
toc

%sum(sims_h(end,:))+l*N

%% Continuity equation
u = Vav*3.6; %Speed [km/h]
k = D*1e3; %Traffic Density [Vehicles/km]
q = u.*k; %Traffic Flow [Vehicles/h]
%%
%Plots
close all
figure (2)
for i=1:N
    if System_type(i)==1
        plot(time,sims_th(:,i),'-.')
        hold on
    else
```

```
        plot(time,sims_th(:,i))
        hold on
    end
end
grid on
xlabel('Time [s]')
ylabel('\theta [rad]')

figure (3)
for i=1:N
    if System_type(i)==1
        plot(time,sims_h(:,i),'-.')
        hold on
    else
        plot(time,sims_h(:,i))
        hold on
    end
end
%plot(time,sims_h1)
grid on
xlabel('Time [s]')
ylabel('h [m]')

figure (4)
for i=1:N
    if System_type(i)==1
        plot(time,sims_v(:,i),'-.')
        hold on
    else
        plot(time,sims_v(:,i))
        hold on
    end
end
%plot([0 time(end)],[vmax vmax],'g','Linewidth',2)
plot(time,sims_vav,'k','Linewidth',2)
grid on
xlabel('Time [s]')
ylabel('v [m/s]')

%% Fundamental diagram Plots
marker_color = {'[1 0 0]','[1 0.5 0]','[1 1 0]','[0.5 1 0]','[0 1 0]','[0
0 1]','[0.5 0 1]','[1 0 1]','[0.75 0.75 0.75]','[0.5 0.5 0.5]','[0 0
0]'};

figure
hold on
scatter(k(1,:),u(1,:),'MarkerFaceColor',marker_color{1},'MarkerEdgeColor'
,'none','Marker','o');
scatter(k(2,:),u(2,:),'MarkerFaceColor',marker_color{2},'MarkerEdgeColor'
,'none','Marker','p');
scatter(k(3,:),u(3,:),'MarkerFaceColor',marker_color{3},'MarkerEdgeColor'
,'none','Marker','h');
scatter(k(4,:),u(4,:),'MarkerFaceColor',marker_color{4},'MarkerEdgeColor'
,'none','Marker','<');
scatter(k(5,:),u(5,:),'MarkerFaceColor',marker_color{5},'MarkerEdgeColor'
,'none','Marker','>');
```

```
scatter(k(6,:),u(6,:),'MarkerFaceColor',marker_color{6},'MarkerEdgeColor'
,'none','Marker','s');
scatter(k(7,:),u(7,:),'MarkerFaceColor',marker_color{7},'MarkerEdgeColor'
,'none','Marker','d');
scatter(k(8,:),u(8,:),'MarkerFaceColor',marker_color{8},'MarkerEdgeColor'
,'none','Marker','p');
scatter(k(9,:),u(9,:),'MarkerFaceColor',marker_color{9},'MarkerEdgeColor'
,'none','Marker','v');
scatter(k(10,:),u(10,:),'MarkerFaceColor',marker_color{10},'MarkerEdgeCol
or','none','Marker','^');
scatter(k(11,:),u(11,:),'MarkerFaceColor',marker_color{11},'MarkerEdgeCol
or','none','Marker','o');
grid on
xlabel('Traffic Density [Vehicles/km]')
ylabel('Speed [km/h]')
axis([0,max(k(11,:))+5,0,max(u(11,:))+5]);
legend('0% CACC','10% CACC','20% CACC','30% CACC','40% CACC','50%
CACC','60% CACC','70% CACC','80% CACC','90% CACC','100% CACC');
title('Traffic Density - Speed')
hold off
%
figure
hold on
scatter(k(1,:),q(1,:),'MarkerFaceColor',marker_color{1},'MarkerEdgeColor'
,'none','Marker','o');
scatter(k(2,:),q(2,:),'MarkerFaceColor',marker_color{2},'MarkerEdgeColor'
,'none','Marker','p');
scatter(k(3,:),q(3,:),'MarkerFaceColor',marker_color{3},'MarkerEdgeColor'
,'none','Marker','h');
scatter(k(4,:),q(4,:),'MarkerFaceColor',marker_color{4},'MarkerEdgeColor'
,'none','Marker','<');
scatter(k(5,:),q(5,:),'MarkerFaceColor',marker_color{5},'MarkerEdgeColor'
,'none','Marker','>');
scatter(k(6,:),q(6,:),'MarkerFaceColor',marker_color{6},'MarkerEdgeColor'
,'none','Marker','s');
scatter(k(7,:),q(7,:),'MarkerFaceColor',marker_color{7},'MarkerEdgeColor'
,'none','Marker','d');
scatter(k(8,:),q(8,:),'MarkerFaceColor',marker_color{8},'MarkerEdgeColor'
,'none','Marker','p');
scatter(k(9,:),q(9,:),'MarkerFaceColor',marker_color{9},'MarkerEdgeColor'
,'none','Marker','v');
scatter(k(10,:),q(10,:),'MarkerFaceColor',marker_color{10},'MarkerEdgeCol
or','none','Marker','^');
scatter(k(11,:),q(11,:),'MarkerFaceColor',marker_color{11},'MarkerEdgeCol
or','none','Marker','o');
grid on
xlabel('Traffic Density [Vehicles/km]')
ylabel('Traffic flow [vehicles/h]')
axis([0,max(k(11,:))+5,0,max(q(11,:))+100]);
legend('0% CACC','10% CACC','20% CACC','30% CACC','40% CACC','50%
CACC','60% CACC','70% CACC','80% CACC','90% CACC','100% CACC');
title('Traffic Density - Traffic Flow')
hold off
%
figure
hold on
scatter(q(1,:),u(1,:),'MarkerFaceColor',marker_color{1},'MarkerEdgeColor'
,'none','Marker','o');
```

```
scatter(q(2,:),u(2,:),'MarkerFaceColor',marker_color{2},'MarkerEdgeColor'
,'none','Marker','p');
scatter(q(3,:),u(3,:),'MarkerFaceColor',marker_color{3},'MarkerEdgeColor'
,'none','Marker','h');
scatter(q(4,:),u(4,:),'MarkerFaceColor',marker_color{4},'MarkerEdgeColor'
,'none','Marker','<');
scatter(q(5,:),u(5,:),'MarkerFaceColor',marker_color{5},'MarkerEdgeColor'
,'none','Marker','>');
scatter(q(6,:),u(6,:),'MarkerFaceColor',marker_color{6},'MarkerEdgeColor'
,'none','Marker','s');
scatter(q(7,:),u(7,:),'MarkerFaceColor',marker_color{7},'MarkerEdgeColor'
,'none','Marker','d');
scatter(q(8,:),u(8,:),'MarkerFaceColor',marker_color{8},'MarkerEdgeColor'
,'none','Marker','p');
scatter(q(9,:),u(9,:),'MarkerFaceColor',marker_color{9},'MarkerEdgeColor'
,'none','Marker','v');
scatter(q(10,:),u(10,:),'MarkerFaceColor',marker_color{10},'MarkerEdgeCol
or','none','Marker','^');
scatter(q(11,:),u(11,:),'MarkerFaceColor',marker_color{11},'MarkerEdgeCol
or','none','Marker','o');
grid on
xlabel('Traffic flow [Vehicles/h]')
ylabel('Average Speed [km/h]')
axis([0,max(q(11,:))+5,0,max(u(11,:))+5]);
legend('0% CACC','10% CACC','20% CACC','30% CACC','40% CACC','50%
CACC','60% CACC','70% CACC','80% CACC','90% CACC','100% CACC');
title('Traffic Flow - Speed')
hold off
```

B3. Normal distributions

```
clear all; clc; close all;

N = 10000; % Number of vehicles

%% Vehicle Parameters (Average values)
Average_Minimum_Space = 2; % h0 [m]
Average_Time_Headway = 1.3; % kv [s]
Average_Vehicles_Acceleration = 1.3; % amax [m/s^2]
Average_Vehicles_Deceleration = -3.5; % amin [m/s^2]
Average_Mass = 1500; %[kg]
Average_Width = 1.9; %[m];
Average_Height = 1.5; %[m];
Average_Longitude = 4; %[m]
Average_Power = 73500; %[W]

%Deviations Values
Deviation_Minimum_Space = 0.2; % [m]
Deviation_Time_Headway = 0.1; % [s]
Deviation_Acceleration = 0.3; % [m/s^2]
Deviation_Vehicles_Deceleration = 0.4; % [m/s^2]
Deviation_Mass = 100; %[kg]
Deviation_Width = 0.2; %[m];
Deviation_Heigth = 0.2; %[m];
Deviation_Longitude = 0.5; %[m]
Deviation_Power = 12500; %[W]
```

```
for e = 1:N
    % Driving Profile Parameters (with normal distribution)
    Vehicle_Matrix(e,6) = Deviation_Minimum_Space.*randn +
Average_Minimum_Space;
    Vehicle_Matrix(e,7) = Deviation_Time_Headway.*randn +
Average_Time_Headway;
    Vehicle_Matrix(e,8) = Deviation_Acceleration.*randn +
Average_Vehicles_Acceleration;
    Vehicle_Matrix(e,9) = Deviation_Vehicles_Deceleration.*randn +
Average_Vehicles_Deceleration;

    % Vehicle Parameters (with normal distribution)
    Vehicle_Matrix(e,1) = Average_Width + Deviation_Width.*randn;
    Vehicle_Matrix(e,2) = Average_Longitude + Deviation_Longitude.*randn;
    Vehicle_Matrix(e,3) = Average_Height + Deviation_Heigth.*randn;
    Vehicle_Matrix(e,4) = Average_Mass + Deviation_Mass.*randn;
    Vehicle_Matrix(e,5) = Average_Power + Deviation_Power*randn;

end

Width = Vehicle_Matrix(:,1);
Length = Vehicle_Matrix(:,2);
Height = Vehicle_Matrix(:,3);
Mass = Vehicle_Matrix(:,4);
Power = Vehicle_Matrix(:,5);
Minimum_distance = Vehicle_Matrix(:,6);
Time_headway = Vehicle_Matrix(:,7);
Acceleration = Vehicle_Matrix(:,8);
Deceleration = Vehicle_Matrix(:,9);

%% PLOTS
figure
subplot(1,2,1)
histogram(Width,25);hold on;grid on;
%histogram(Minimum_distance,35);
set(gca,'fontsize',14);
title('Vehicle Width');
xlabel('Width [m]');ylabel('Number of vehicles');

subplot(1,2,2)
histogram(Length,25);hold on;grid on;
%histogram(Minimum_distance,35);
set(gca,'fontsize',14);
title('Vehicle Length');
xlabel('Length [m]');ylabel('Number of vehicles');

figure
subplot(1,2,1)
histogram(Height,25);hold on;grid on;
%histogram(Time_headway,20);
set(gca,'fontsize',14);
title('Vehicle Height');
xlabel('Height [m]');ylabel('Number of vehicles');

subplot(1,2,2)
```

```
histogram(Mass,25);hold on;grid on;
%histogram(Minimum_distance,35);
set(gca,'fontsize',14);
title('Vehicle Mass');
xlabel('Mass [kg]');ylabel('Number of vehicles');

figure
subplot(1,2,1)
histogram(Power,25);hold on;grid on;
%histogram(Minimum_distance,35);
set(gca,'fontsize',14);
title('Vehicle Power');
xlabel('Power [W]');ylabel('Number of vehicles');

subplot(1,2,2)
histogram(Minimum_distance,25);hold on;grid on;
%histogram(Minimum_distance,35);
set(gca,'fontsize',14);
title('Minimum distance (h0)');
xlabel('Distance [m]');ylabel('Number of vehicles');

figure
subplot(1,2,1)
histogram(Time_headway,25);hold on;grid on;
%histogram(Time_headway,20);
set(gca,'fontsize',14);
title('Time Headway (kv)');
xlabel('Time [s]');ylabel('Number of vehicles');

subplot(1,2,2)
histogram(Power,25);hold on;grid on;
%histogram(Minimum_distance,35);
set(gca,'fontsize',14);
title('Vehicle Acceleration');
xlabel('Acceleration [m/s^2]');ylabel('Number of vehicles');

figure
subplot(1,2,1)
histogram(Power,25);hold on;grid on;
%histogram(Minimum_distance,35);
set(gca,'fontsize',14);
title('Vehicle Deceleration');
xlabel('Deceleration [m/s^2]');ylabel('Number of vehicles');
```