POLITECNICO DI TORINO

Master's Degree in Mechatronic Engineering

Master's Degree Thesis

Energy flow management of P2 hybrid vehicle based on ADAS sensors

Academic Supervisor

Prof. Angelo BONFITTO

Internship Supervisor

Ing. Alessandro DAMINO

Candidate

Emanuele LAVENIA

October A.Y. 2021/2022

Abstract

The optimization of consumption is one of the main challenges of today's automotive world. The increasing imposed restrictions require seeking means to reduce consumption. The proposed work aims to reduce the consumption of an existing vehicle through the exploitation of currently available technologies. The vehicle is characterized by a P2 parallel type full-hybrid electric architecture. In this context three possible approaches are used.

The first involves the study and comparison between the automatic transmission present in the standard vehicle and a Dual Clutch Transmission (DCT). This comparison wanted to evaluate the efficiency performance among different control logics implemented in the DCT and the automatic transmission. For the evaluation, the energy required for the imposition of a load state is measured. Also, regarding the DCT logics an estimator capable of obtaining the future next gear request imposed by the ECMS-GC control logic was implemented.

The second study is the creation of control logics that is able to reduce consumption by supervising the human driver to discourage its imposed inefficiencies speed profiles. This supervisor is created by exploiting a fuzzy logic that is capable of categorize the speed profiles using the input information coming from the ADAS sensor integration. The discouragement of the inefficient speed pattern is implemented by a dynamic calibration of the throttle PID controller.

The third aims to create a logic capable of imposing a longitudinal speed profile that can improve the efficiency of the vehicle. This logic is implemented taking the braking scenario as a reference. In this context, the algorithm developed imposes a profile capable of maximizing the regenerated energy during braking. The implementation of this logic must be evaluated in terms of real-time implementation: imposing a solution capable of obtaining the overall maximum of energy recovered during braking would be too time-consuming. To overcome this drawback, the implemented algorithm is able to obtain the speed profile in less time, which however is only sub-optimal. The results of these logics will examine the amount of energy recovered.

Table of Contents

List of Tables

List of Figures

Chapter 1

Background

The automotive sector is now considered one of the main causes of the production of global pollutants. Therefore, the increasing number of vehicles and the future perspectives of an increase in registrations are leading to urgent initiatives to limit emissions as much as possible. Specifically, global organizations impose restrictions in terms of emissions on each registered vehicle, this leads car manufacturers to increase the technological effort for reduce the gas emissions of the produced vehicles.The applied technologies provide for the evaluation of different approaches capable of obtaining an effective improvement in optimizing the actual consumption of the vehicle. In this context, the project examined the two main alternatives for reducing consumption: a) optimization of car components, b) optimization of the car's control logic.

Optimization of car components

Optimization of the components characterizing the car [1][2], includes the evaluation of all the equipment that is present in the vehicle to characterize their relationship with the fuel consumption. This characterization allows to evaluate whether to replace any components to make the vehicle more efficient. The approaches that aims to modify the combustion engine motorization are included in this optimization method. These changes have taken up a lot of space in recent decades as thermal machines turn out to be the main source of emissions of pollutants from the car. Taking into account this aspect, the inclusion of an electric motor that goes to support (Hybrid Electric Vehicle) or replace (Full Electric Vehicle) the thermal engine is positively evaluated.

In this context, it is of particular interest to consider the disadvantages brought by electric motorization. The problems related to the storage of battery energy involve both a high increase in the weight of the car to ensure greater autonomy and significantly longer charging times compared to thermal vehicles. Taking into account this criticality, hybrid electric vehicles are evaluated as the optimal solution for the transition from cars totally powered by thermal engines to fully electric ones. They allow you to take advantage of the main characteristics of both types of motorization:

- Extend autonomy through fuel in the tank;
- Energy recovery during braking, taking advantage of the reversibility characteristic of the electric motor;
- Flexibility in providing power, this aspect is of high importance since better flexibility in the delivery of power flows can be exploited to improve the consumption optimization of the car.

The disadvantages hybrid electric power unity are:

- Larger weight of the car given by the presence of the two engines and the battery;
- complex mechanical components for coupling the electric machine inside the driveline;
- Higher maintenance demand due to the increasing of components number present in the car.

Optimization of vehicle control logics

Optimization of the logic implemented [3] to control the components and/or the car in its entirety, this approach involves improving the actual consumption of fuel, through the optimization of the operation of the individual components and/or their overall operation. This method allows to avoid critical issues and/or inefficiencies of the components that increase the total consumption of the car. In this sense it is possible to elaborate two types of logics: a) causal logics, b) non-causal logics.
In causal logics [4][5], cause-effect relationships between a given physical phenomenon and its response are exploited. This approach allows evaluating what effects will give an action to the result in order to extrapolate a relationship that allows to obtain values capable of being optimal snapshots in relation to the imposed problem. Types of causal logics are widely used because they guarantee better implementation and implementation in real-time systems: having low computational execution times and NOT requiring future information for the evaluation of the effect.

On the other hand, non-causal control logics [6][7], obtain the result by analyzing every possible evolution in a given time horizon by calculating the certain relationship that we want to maximize and/or minimize for each evolution. It is called 'Non-causal' because it obtains the result by analyzing all possible evolutions, finding the one that respects the required relationship, and only as a consequence imposes the cause. This type of approach is able to provide the absolute optimal type in the given time horizon examined but requires high computational demand for the calculation as well as the knowledge of the future evolution of the phenomenon. these logics are then used only as a reference and are difficult to implement in real-time.

1.0.1 Project framework

The project, carried out through funding from the Piedmont region [8] to the Politecnico di Torino [9], with Podium Engineering partners [10] aims to reduce the consumption of a Light Duty Vehicle through the exploitation of the points just exposed. Specifically, the project plans to:

- Hybridize the vehicle in order to exploit the characteristics of that type of motorization, making it homologable at the European level;
- Implementation of high-level logics for the reduction of consumption by exploiting information from appropriate ADAS sensors, sensor configuration analysis to obtain variables for the definition of controls;
- Implementation of a sensor fusion on the signals obtained from the sensors in order to exploit the best characteristics of each sensor and/or compensate for worse characteristics by adding redundancy.

The project in question is called AutoECO.
The thesis work concerns the validation of the model of the hybrid vehicle, to validate the results to predict any real test results.
The efficiency comparison of the motion transfer between the base automatic transmission located in the drivetrain and a Dual-Clutch transmission (DCT) with different control logics.
The definition of high-level control algorithms capable of exploiting the signals coming from the configuration of ADAS sensors located in the car for the consumption reduction.

Sections

Specifically, the project in question is composed of the following characteristic sections: First section in which the development environment of the simulation model and the modeling of the vehicle are explained. Also referring to the sensor configuration examined for the characterizing simulation.

The second section in which there is the work of modeling a DCT, to validate the simulation part so as to be able to understand the differences with the previously validated one and consider the variations that, in the first approximation, this variation would impose on the overall system. In addition, a procedure for estimating the next gear was evaluated.

The third section concerns the control logic called 'P-DRIVER', which is a logic capable of evaluating the future states of the car using the information coming from the sensors and then processed by the speed profiling logic. It acts in parallel with the input provided by a real driver, recognizing driving patterns that are considered incorrect, devaluing their implementation by dynamically calibrating the actual response of the car.

The fourth section deals with the logic of speed profiling. In fact, it is considered as the highest logic level and is able to create a speed trajectory in relation to a future state perceived by what the configuration of sensors present in the car is. Specifically, the section shows an approach capable of partially minimizing the losses that occur during the braking of the car in question.

The fifth section will collect the deductions found in the previous ones and the results obtained will be the probable improvements.

Chapter 2

Vehicle model

As explained in chapter 1, one of the main points for emissions reductions is to evaluate different types of vehicle motorization. Another way is making power delivery more efficient by exploiting the characteristics of the vehicle in question. The overall efficiency of the vehicle generally represents how much power is not used by the overall system during a given operation. This magnitude is characterized by the components, so taking into account the types of motorization can be evaluated:

- For the thermal machine the values of the Brake Specific Fuel Consumption (BSFC);
- Efficiency for the electric part.

These two can be considered as maps that characterize each working point of the machine, considering a point in a Torque-Speed diagram uniquely associates the value of these quantities.

BSFC

It is characteristic of thermal machines and relates how much energy a thermal engine can generate concerning the number of grams of fuel that must be supplied at the input.
It is then defined as the ratio between the grams per second and the power generated at the output of the engine, referring to the specific working point (2.1):

$$BSFC_{i,j} = \frac{\dot{m}_f}{T_{ICE}(i) \cdot \omega(j)} \tag{2.1}$$

A three-dimensional curve is then defined and it associates each value to the specific point. For better visualization, the two-dimensional figure with the presence of contour lines is shown below Figure 2.1:

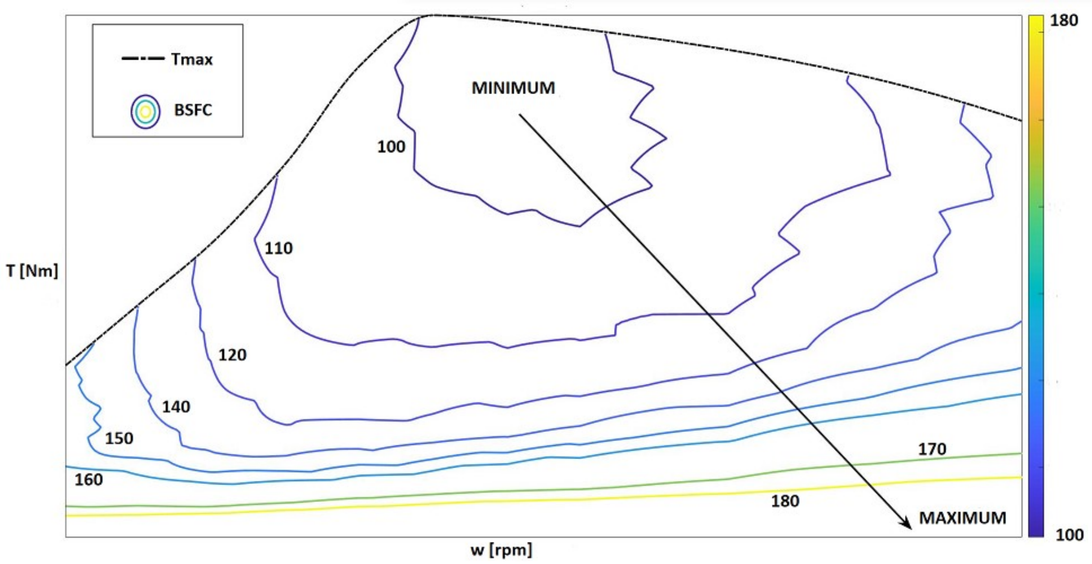

Figure 2.1: the Brake Specific Fuel Consumption (BSFC) related to a specific engine (expressed in $[\frac{g}{kW \cdot h}]$), for confidentiality reasons a normalized map is shown

Such a map then shows how efficient an operating point of the thermal machine is. Specifically, this surface is of particular interest because it is able to indicate the point where the machine has its best functioning. This point is represented by the absolute minimum present on the map.

Electrical drive efficiency

As far as an electric drive is concerned, the actual efficiency is characterized by the transfer of the electrical power of the battery to the mechanical power output from the actual electric machine. It is then obtained from the presence of a specific efficiency map for each component of this chain, considering the generic electric drive chain in Figure 2.2:

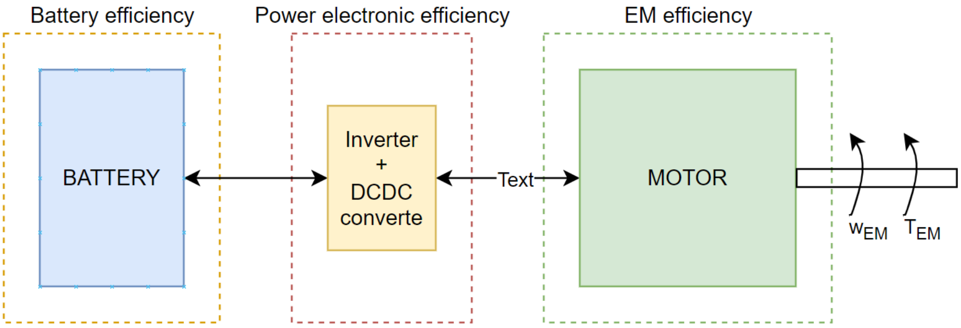

Figure 2.2: Electric drive schematic

Considering the concatenate elements, the components show then a generic efficiency associated with the whole (2.2) chain obtained:

$$\eta_{TOT} = \eta_{BATTERY} \cdot \eta_{PowerElectronics} \cdot \eta_{MOTOR} \tag{2.2}$$

As for the BSFC, this characteristic value depends on the actual working point of the total drive and therefore is characterized by a surface in which defined the torque-speed values of the machine. It is described by a certain efficiency of the map Figure 2.3.

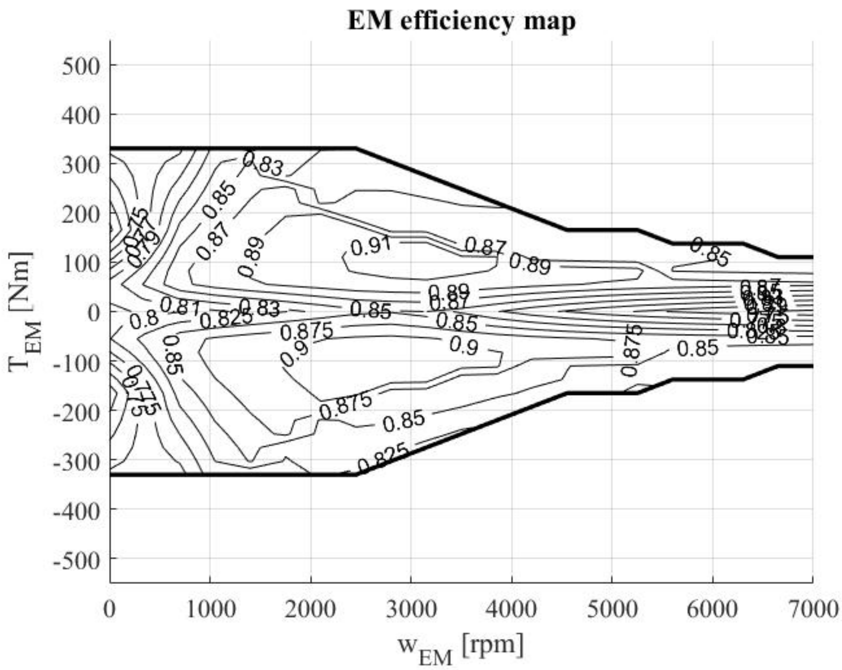

Figure 2.3: Electrical Motor (EM) efficiency map

The map shows the absolute maximum point (greater efficiency) that coincides with the best working point.

2.1 Hybrid Electric Vehicles (HEV)

The most convenient points are different. Often one of the best improvements is to use both a thermal machine and an electric machine in order to take advantage of both motorizations:

- General consumption reduction thanks to the use of the electric machine;
- Improve overall efficiency by joining different working points in the considered performance maps;
- More versatility in propulsion by referring to different driving scenarios;
- Versatility in refueling/recharging times.

The use of hybrid architectures is therefore becoming increasingly widely used within the global automotive scene, both for their strong application qualities and as a bridge between pure thermal power unity to pure electric power unity that in fact will be imposed as standard in the coming decades.

Hybrid configurations

The hybrid vehicles present are characterized by how the power flow moves from the power unit to the wheels [3]. Such types of combinations can be many and are distinguished mainly in:

- Series hybrids Figure 2.4, in which the electric car is the only one that delivers effective torque to the wheels and the thermal machine is employed using a cascade generator to provide power to the electric car and/or the battery present in the car. This type of configuration is mainly used for overcoming the problem of electric motorization, i.e. charging. Having a thermal machine means providing power to recharge the battery when needed. Generally, this configuration allows the use of the thermal machine in a more efficient way or where the value of BSFC is lower in order to obtain a sort of load shifting of the power;

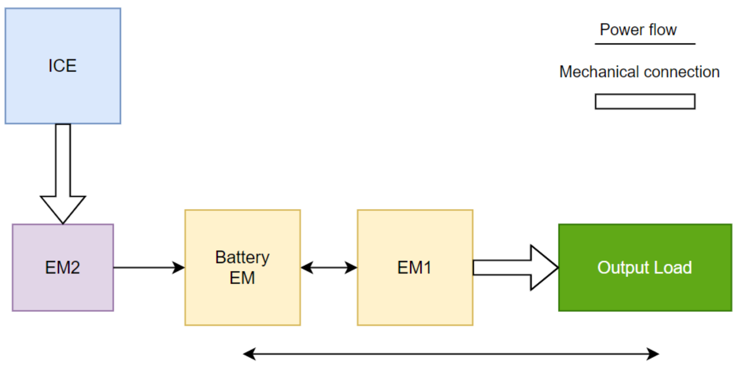

Figure 2.4: Series hybrid electric vehicle configuration

- Parallel hybrid Figure 2.5, it represents the more versatile of the two and also the most widely used in the automotive world. It can decouple the operation of the machines present to be able to deliver power with both machines to the wheels or even just one at a time, both electrical and thermal. Considering how versatile this type of configuration is, it must be evaluated how a possible split control logic must be advanced to make the machine's power delivery as efficient as possible. The parallel configuration allows different versions. The scheme representing the configuration is shown below;

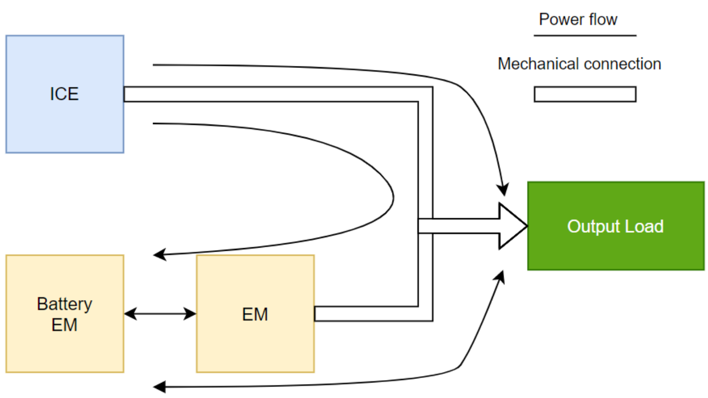

Figure 2.5: Parallel hybrid electric configuration

- Hybrid of mixed type (series-parallel) Figure 2.6 through the use of mechanical equipment of specific coupling manages to be considered at the same time both as a series vehicle and as a parallel vehicle. It, therefore, allows the transfer of the power flow in a much more versatile way. The diagram for the energy flow in question is then shown below.

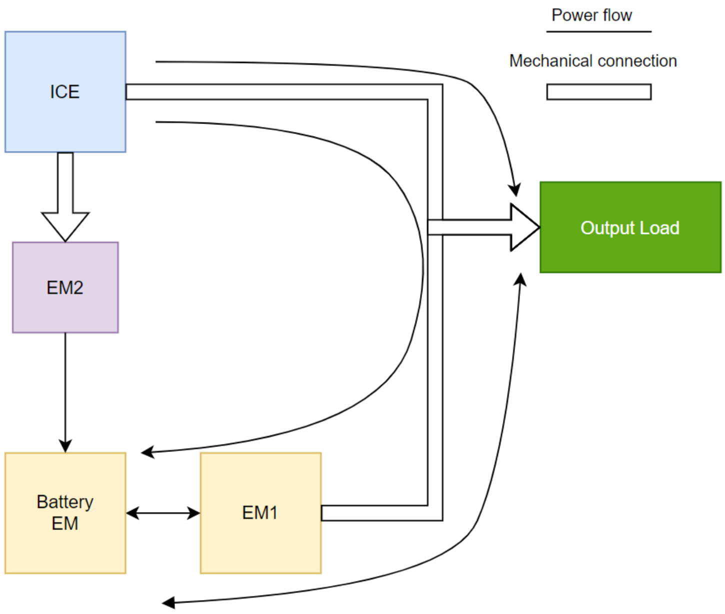

Figure 2.6: Series-parallel hybrid electric configuration

Another possible characterization of hybrid vehicles must be carried out about the type of battery present in the car, specifically the capacity of the latter. This characterization is of fundamental importance because it represents the limit up to which the electric machine can go, to obtain power under normal conditions. Together with the maximum power that can be delivered by the electric motor, it characterizes the type of hybrid. Among the various characterizations Figure 2.7 [3]:

- Micro Hybrid, vehicles with the presence of a battery with voltages typically below 15V and characterized by the presence of small electric machines capable of providing power in the order of few kW;
- Mild Hybrids, characterized by the presence of batteries at higher voltages >40V. They have the presence of electric motors capable of recovering energy during braking and capable of being used for short stretches as suppliers of power to the wheels;
- Full Hybrid, characterized by voltages usually >100V and large capacities, accompanied by the presence of electric cars of large enough size to allow the electric motion of the car only. They allow having a range in km of electric only;
- Plug-in hybrid allows the battery in the car to be recharged by means of

external sockets and chargers. They represent the closest category to electric motorization, capable of conspicuous autonomy in full electric.

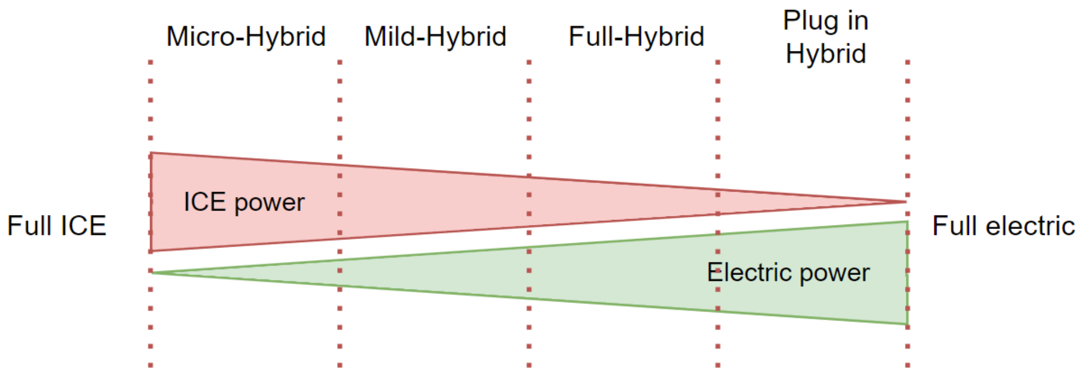

Figure 2.7: Hybrid electric vehicle power classification

2.2 Parallel-Hybrid vehicles

As explained above, the type of hybrid widely used is that of the parallel type because it allows through the use of straightforward mechanical equipment a wide range of possible combinations of power flows.
However, this is related to the type of parallelism that the two engines have inside the vehicle, in fact the position of the electric motor provides different types of parallelism. Each of the following positions has different cases in which the electric machine will have to interact differently. Referring to the following possible combinations Figure 2.8:

- P0, it provides for the insertion of the electric machine inside the heat engine often placing it as alternator. This type of configuration is mainly used for increasing the performance of the thermal machine, especially related to response times. Another of their characteristics is that they can be used as a starter engine;
- P1, this type of system of the electric drive provides for a calettatura of the latter at the exit of the thermal machine. Specifically, this configuration could provide for the calettatura on the flywheel. This type of arrangement allows the obtaining of an electric torque that can only be used at the same speed as the thermal machine. This configuration allows to recharge the battery under braking and to deliver small amounts of power electric;

- P2, the electric machine is located in the primary of the transmission. It therefore provides for a total decoupling given by the friction between the engines and motor. This allows the use of the thermal machine and the electric machine at different speeds (if the clutch is open). This is very relevant for the optimization of the working point. This configuration also allows the use of the electric machine alone without dragging the thermal machine;

- P3, the positioning of the electric motor is provided to the secondary of the gearbox present in the car. So it brings with it all the advantages of the P2 arrangement with the only flaw of having an imposition of speed that is obtained through the speed to the wheels;

- P4, this configuration provides for the insertion of the electric machine in the axis of the wheels, thus allowing a total decoupling between the electric machine and the thermal machine. Such a configuration is often used for the creation of all-wheel drive vehicles. Also in the P4 configuration there are possible solutions for in-wheel electric motors.

In the following, combinations of a parallel hybrid are shown:

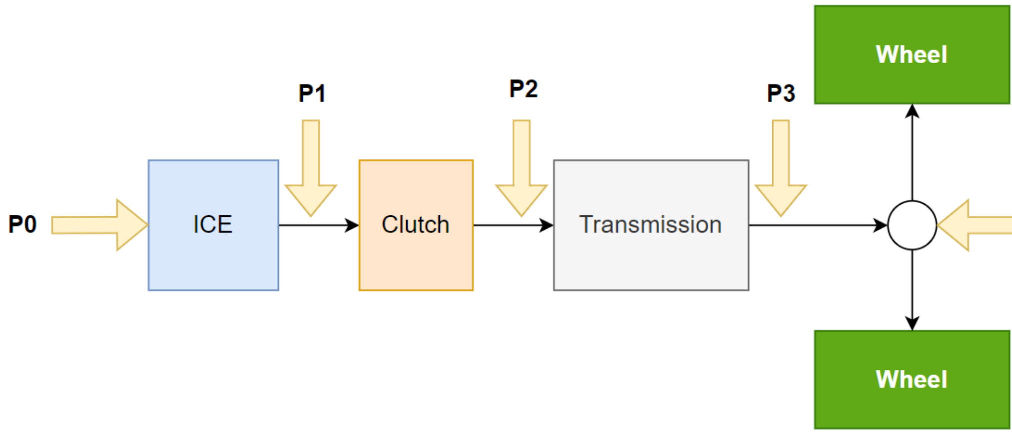

Figure 2.8: Possible positions of the electric motor in a parallel hybrid electric vehicle

2.3 Vehicle plant model and characteristics

For the evaluation of the results obtained, the car under consideration is characterized as a hybrid vehicle with an electric motor located in the P2 position.

Specifically, the modeled type is that of a full-Hybrid electric vehicle.
The plant in question models both the presence of the electric drive and that of the thermal machine to evaluate the actual power flows and consequently to be able to compute numerically both fuel consumption and the state of charge of the battery. The characterizing subsystems, therefore, come to be:

- Engine (ICE), characterized by a system capable of imposing a torque given by the high-level logic in the form of throttle. Modeling involves the presence of an idle velocity maintenance system. A first version of what is the resistant pair of the ICE is also modeled. To obtain the value of the crankshaft speeds, a block belonging to the Simscape tool is used, capable of giving the angular velocity as output. This information, set torque and relative shaft speed are used for calculating the fuel consumed. Specifically, this quantity is obtained through the $\frac{g}{s}$ consumption map that associates consumption with the specific working point $T - \omega$.;

- Electric drive, formed by the presence of an electric machine and the battery. Specifically, the battery is modeled using the specific block of Simscape, in which the type of internal model and the characteristic values such as voltage (E [V]) capacity (C [Ah]) are set. The battery is therefore capable of both delivering and absorbing energy, thus exploiting the reversibility capacity of the electric machine. The modeling of the electric drive is then completed by inserting a system capable of delivering a torque imposed by the control logic. To take into account the real drive including all the components, also considering the non-modeled components, an overall efficiency map is incorporated into the model for the entire drive chain, this map associates the overall value of the chain associated with the output working point. Through the use of this model, it is possible to evaluate the state of charge of the battery and/or the energy that it delivers or absorbs in a given period;

- Transmission, this system will be exposed later when it will be compared with the modeling of a different type of transmission;

- Longitudinal model, such a system associates a certain force applied to the wheels of the car with its longitudinal speed. For the modeling the Simscape block 'Vehicle Body' is exploited. This can take into consideration other aspects for the calculation of the speed such as the inclination from the roadway and the wind. Inside, the vehicle characteristic are inserted.

In the following are reported all the vehicle parameters Table 2.1:

Elements	Characteristics
ICE	$T_{MAX} = 390Nm(1700rpm) \quad n_{MAX} = 4000rpm$
EM	$T_{MAX} = 330Nm(0-2700rpm) \quad n_{MAX} = 7000rpm$
Transmission	Torque Converter + automatic 8 Gearbox
Coast down	$f_0 = -401.33N \quad f_1 = -0.0133N\cdot\frac{h}{km} \quad f_2 = -0.067N\cdot\frac{h^2}{km^2}$
Vehicle Mass	M=3175 kg
Battery	Capacity=40 $A \cdot h$ CapacitykWh=14 $kW \cdot h$
Wheels	Inertia= 0.8 $kg \cdot m^2$ Radius =0.35 m

Table 2.1: Vehicle Parameters, include principal characteristics of the hardware located in the considered vehicle configuration

2.3.1 Vehicle plant and control system setup

The model explained above makes possible to assess how a change in control can affect the overall consumption of the vehicle. It is therefore important to create an appropriate integration between all the possible systems examined.
The integration involves two different feedback loops, called respectively:

- Internal loop, related to low-level logics. It is the loop characterized by the measurement of variables inside the car (eg. Engine speed, engine torque, instantaneous car speed, etc.). It is explained specifically later, in the section on the P-DRIVER;
- External loop, related to the measurements perceived by the ADAS sensors. This loop allows the activation and implementation of the logic for the reduction of consumption elaborated during the thesis work.

ADAS SENSORS

The configuration under consideration for the drafting of the thesis work has been hypothesized to consist of:

- Stereocamera;
- Radar.

For the development of the logics during the work carried out, the following assumptions were made on the type of information capable of being found from these logics:

- The current speed of the vehicle, to be redundant with the speed obtained by other methods;
- Speed of a possible vehicle ahead;
- The relative distance of a given vehicle ahead;
- Road signs.

System schematic integration

The integration of all these systems is then shown below. This overview shows the schematic of the architectures created during the thesis work Figure 2.9.

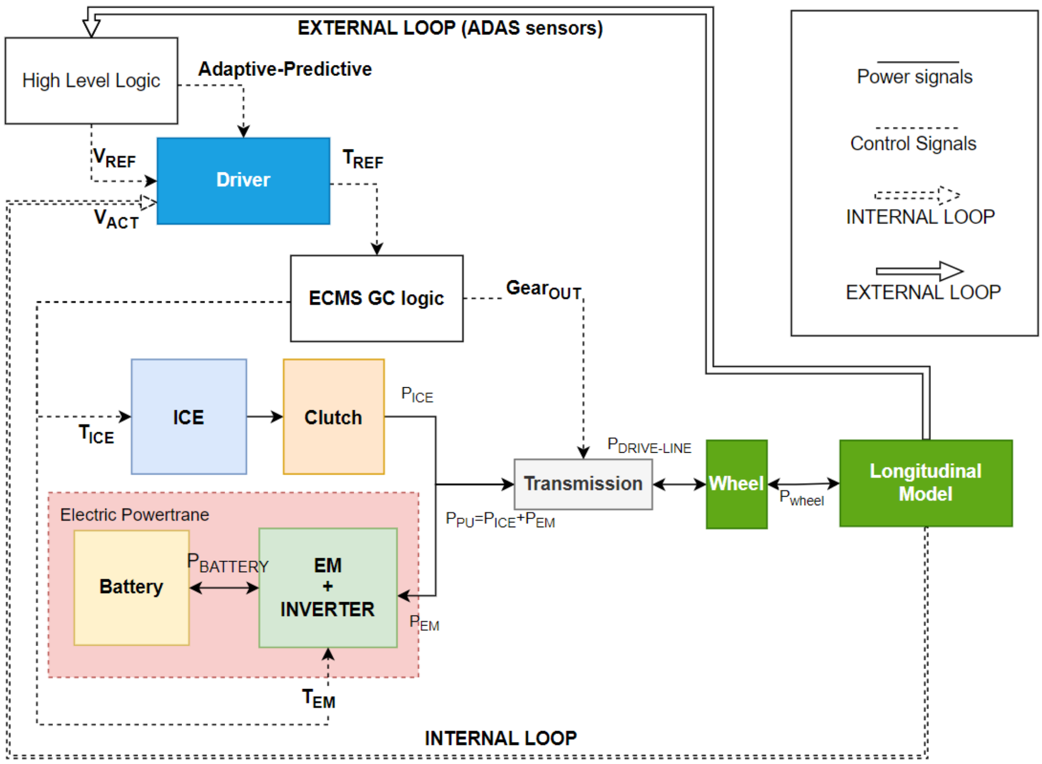

Figure 2.9: Complete Schematic Integration of vehicle plant and control model

Chapter 3

Control Logic of a Dual-Clutch Transmission (DCT)

3.1 Introduction

The characterization of a subset present in the car is of fundamental importance to understand how much the latter influences the overall dynamics of the results obtained.
It is therefore essential to evaluate qualitatively the modification of elements present in the car. In the thesis work carried out the evaluation between the past automatic gear (with torque converter) transmission and a Dual Clutch Transmission (DCT).
Firstly, it is relevant to evaluate what differences would cause, in relation to the driveline integration, the replacement of the transmission, previously present and modeled, with a Dual-Clutch Transmission.
The comparison is carried out evaluating different kinds of DCT control strategy during gear shift.
Starting from what was the already present model of transmission, a first approximation of the aforementioned transmission was therefore developed.

3.2 State of art

The driveline the apparatus that allows the transfer of power from the power unity to the wheels. It includes all the rotating elements and gears present in the car and its realization is one of the main challenges of the current automotive world.
The modeling of the driveline is dependent on the configuration of the vehicle,

specifically the modeling of a hybrid vehicle is influenced by what type of interconnection the electric car has with the thermal engine. Taking into consideration configurations of a parallel type hybrid vehicle such as P0, P1 and P2, considering the position of both the electric and thermal motor, it is clear how the driveline including the gearbox is common to both elements of the power unity.
It is therefore important to create a system that is as stable and robust as possible, capable of not creating critical issues in the vehicle's transmission and power split. The DCT involves the use of two separate clutches capable of transmitting torque at the same time by exploiting a partial slip of them. This leads to quite evident advantages [11][12]:

- Driving comfort;
- Partial decrease in consumption.

DCT is also advantageous because it allows the exceeding of the efficiency limits imposed by the torque converter and the torque transfer limits imposed by the possible use of a CVT [13].

3.2.1 Transmission: torque converter and gearbox

The starting point for DCT modeling within the driveline involved a study of what transmission modeling previously was. For this reason, it is important to analyze the previous transmission model integrated into the overall system in order to make the new model created integrable.
The system taken as a baseline wants to model a transmission related to an automatic gearbox with the following characteristics:

- Presence of Torque Converter;
- Gearbox with 8 gears.

The Torque Converter is of fundamental importance in automatic transmissions since it is the element that interconnects the load, which can be seen as the wheels and the thermal engine (ICE).
This coupling is relevant during the low-speed regimes of the thermal machine, i.e. when demand for load torque cannot be delivered by the ICE.
The Torque Converter therefore through the exploitation of hydraulic mechanisms allows a peak torque delivery at low speeds. It also makes possible, by its construction, a slippage between its primary and secondary part. The main characteristic parts of a torque converter are [14]:

- The impeller, connected to the driving side of the system, i.e. to the engine;
- The turbine, connected to the load;
- Stator, an element not connected to the two previous ones, which allows the flow of oil inside the component so that the rotation is possible only in the direction of rotation of the torque converter.

The presence of the incompressible liquid (oil) inside the component allows the transfer of motion. It should be noted that this system allows the engine to run even with the turbine locked, which is possible with the car braked and the engine running [14]. The operation of the component is therefore related to slippage that is the difference between the impeller and turbine speeds, when the slip is high then the transmissible torque is maximum. As another important system, the presence of a clutch in parallel to the torque converter is also envisaged, because recognizing the inefficient nature of the system, once the load has been into kinematic motion and the impeller is brought to full speed, the system is excluded[14]. Simulink simulation environment using the Simscape tool was used to model this system. Specifically for the modeling of the torque converter, there is a block that already groups the dynamics of the component. To exclude the component, the clutch is added in parallel with a logic control for its engagement.
For the modeling of the gearbox, it was considered appropriate to use a CVT discretizing the ratio in order to make it equivalent to an 8-speed gearbox.
The overall system is then schematically represented in the following figure Figure 3.1.

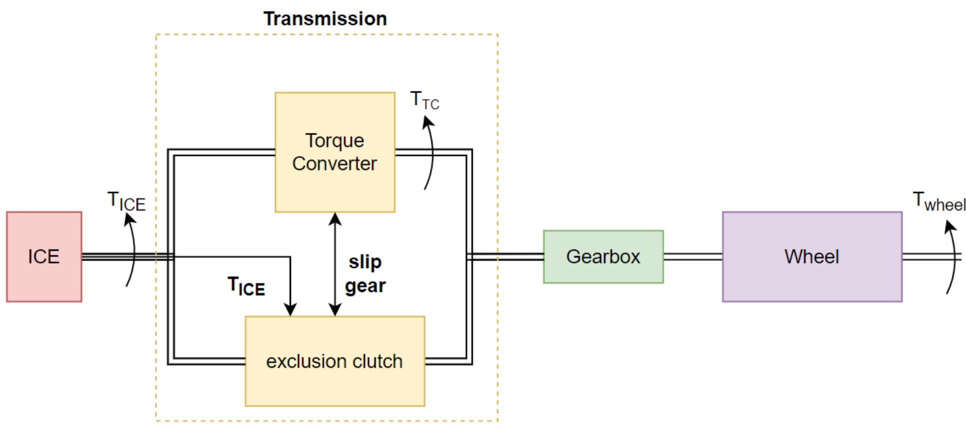

Figure 3.1: Schematic of drivetrain with automatic gearbox and torque converter

3.3 DCT implementation

3.3.1 DCT model

The modeling of the vehicle was entirely carried out in Matlab/Simulink environment, specifically using the tool called Simscape [15].
Simscape allows the rapid modeling of complex systems. Specifically it is widely used because it is able to facilitate the modeling and interconnection of different components acting in different physical systems. Through the use of visual blocks it is able to integrate the dynamic equations of the fundamental components it uses. The appropriate interconnection of the blocks allows the modeling of more complex components.
The modeled DCT will replace the automatic transmission including torque converter previously present inside the vehicle. The presence of the torque converts allows the previous transmission to take on smoother trends thanks to its constructive characteristics that allow to dampen changes in relative speed.
The DCT provides for the presence of the two clutches individually linked to odd or even ratios. The following figure shows the modeling of the transmission as a whole Figure 3.2.

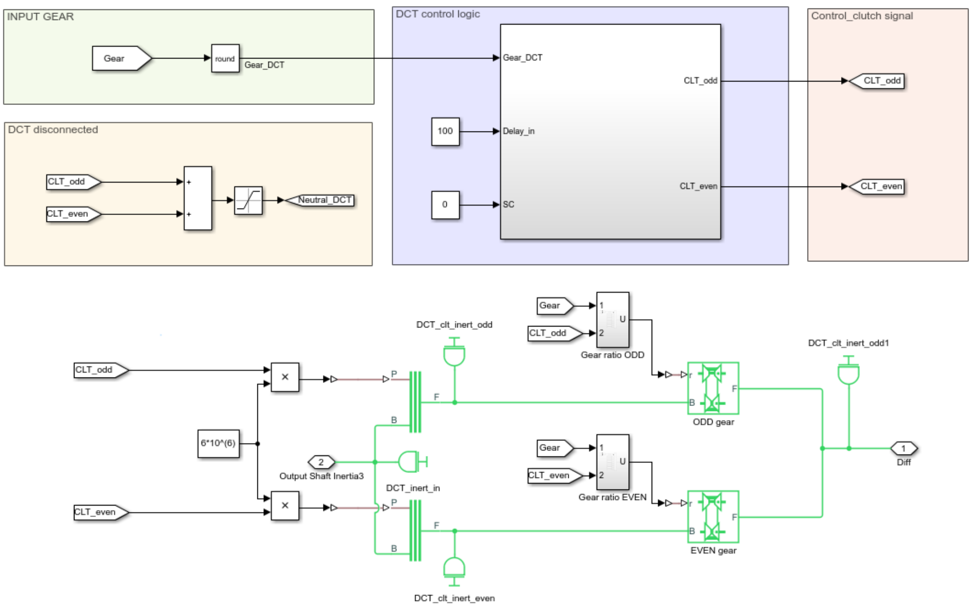

Figure 3.2: Dual-Clutch Transmission (DCT) complete model and control system

The two clutches have been modeled using Simscape's "Disk Friction Clutch" block. It represents a friction model that can be controlled by the pressure signal present in the P input of the component. The engagement pressure can be defined as a value inside the block and represents the pressure in Pa that allows the clutch to start its operation. Another parameter that can be set is the relative speed threshold. It imposes the relative speed at which the clutch will be engaged and therefore for which threshold it will be possible to transmit torque. Once engaged, it remains in its state as long as the pressure is still the activation pressure and if the transmitted torque is not greater than the torque that can be transmitted by the static friction threshold. The equations characterizing the torque transfer of the block are the following:

- Kinetick friction, when a single axis is spinning (3.1)(3.2);

$$\tau_k = \mu \cdot \omega + \tau_{contact} \tag{3.1}$$

$$\tau_{contact} = k_k \cdot D \cdot N \cdot r_{eff} \cdot P_{fric} \cdot A \tag{3.2}$$

- Static friction, when the clutch is locked and both axes are spinning (τ_s is the static friction torque limit) (3.3).

$$\tau_s = k_s \cdot D \cdot N \cdot r_{eff} \cdot P_{fric} \cdot A >= 0 \tag{3.3}$$

The maximum torque values that can be transferred from the clutch are to be considered both positive and negative.
To complete and try to apply more accurately the real operation of the DCT, the system has been completed by inserting the following blocks appropriately in the following Simscape blocks:

- The inertias are modeled using 'Inertial' block in Simscape tool with fixed values and are present to limit the accelerations of the various rotating elements. They model the following equation (3.4):

  $$T = J \cdot \frac{d\omega}{dt} \tag{3.4}$$

  For the simulations carried out, the inertia block was used with a single graphic port, but with appropriate adjustments the results are completely equivalent using the two graphics port.

- The Variable ratio transmission block models dynamic motion transfer between its input and output. It is represented by a conical gearbox. This is a compromise between modeling specifically every single relationship present, which is a lower level type of modeling and still maintaining a good fidelity in the results obtained from the present model. The block models the following equations (3.5) (3.6):

$$\omega_B = +g_{VB} \cdot \omega_V \tag{3.5}$$

Where ω_B and ω_v are the input/output rotational speed;

$$\tau_V = +g_{VB} \cdot \tau_B \tag{3.6}$$

Where τ_B and τ_v are the input/output torques.
It should be noted that the equations maintain a positive sign because although the block allows to invert the motions between the two speeds and torques, the DCT models only the 'positive' transmission ratios. The g_{VB} parameter, seen as an input, represents the translation value imposed on the block. For the modeling of the DCT in question, which has fixed ratios, a transmission ratio was modeled that varies only in a discrete and discontinuous way between the gear ratios that will then occur in the real car. The discretization of the ratio to be given in input was done by means of a suitably built switch case Figure 3.3.

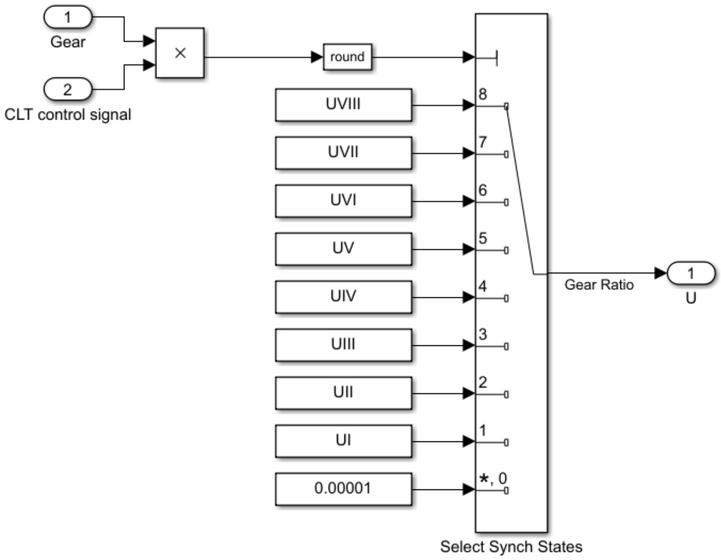

Figure 3.3: Gear ratio selector for integration in the DCT model

As shown in the previous figure, in input there is the gear decided by the higher level logic and the control signal coming from the DCT logic, therefore it will be possible to select a ratio, different from neutral, only if the respective branch (even or odd) is activated by the low data logic at the DCT.

3.3.2 DCT control logics

As written in [13], the actuation control of the two clutches present in a DCT is essential for correct operation. A basic check can be carried out through the presence of a logic capable of disengaging and engaging the clutches with well-defined timing. If suitably modeled and with the presence of a suitable control logic, a DCT [16] is capable of transferring torque even during the disengagement of one of the two previously engaged clutches. We therefore introduce the fact that the 2 clutches present during a gear change do not have discrete states (0 or 1) but are actually in intermediate states, for a short but very precise period of time. For the low-level logic to be integrated into the DCT model, two different possibilities of engagement were evaluated:

- A logic that foresees the disengagement of both clutches during the gear change;
- A more complete logic capable of partializing the engagement of the clutches during the transients of the gear change.

For the design of the control logic, an algorithm capable of implementing command timings, synchronizing control signals and actions must be provided. For this reason it is used a tool present in the Simulink environment capable of performing the previous functions. The tool in question is called Stateflow. Stateflow allows to use graphic blocks to model transitions, timings, control actions in relation to predefined or variable inputs and/or timings.
These blocks are the points in which it is possible to perform actions within Stateflow. They are used to define and/or impose values on variables.
The variables present within the tool must be defined in what way they will be used during the simulation of the created control. In the application, only variables of the type have been defined: a) input, b) output.
Where the inputs are all the input signals that are used to decide in what state the diagram will enter the created environment. To define the types of variables, using the 'Symbols Plan' within the tool in question, the following image Figure 3.4 shows the specific screen.
The interconnections are defined within the diagram formed by the various states. Specifically, the following can be applied within the interconnections:

- Timings using the following command (3.7)
$$after(Delay, sec) \tag{3.7}$$
- Access conditions using the following command (3.8)
$$[(Variable_1)(Typecondition)(ConditionValue)] \tag{3.8}$$

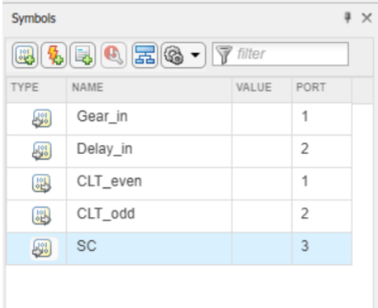

Figure 3.4: Symbols plan: Stateflow input-output variable definition tool

The built control takes as input the signal that indicates the gear controller wants to engage. The shifting logic is implemented by a higher level logic which is not taken into consideration for the modeling of this part of the design.
The signal indicating the gear represents an integer that goes from 0, representing neutral gear, to the last gear, excluding reverse.
In the application case, a 7 or 8-speed gearbox was modeled to verify how the DCT affected the overall performance and consumption of the vehicle. In any case, whether it is an 8-speed or 7-speed gearbox, the model is completely equivalent (for simplicity, we only show how the control of an 8-speed gear has been modeled).
The logic is divided into 2 completely distinguishable blocks:

- first relating to neutral gear;
- second relative to all the other gears present.

The prototyping envisages going to disengage both clutches in the event of a neutral request from the higher level logic. In this case the control signals will disengage both clutches indefinitely until a signal other than neutral arrives as indicated gear. The following output values of the signals are then imposed (3.9)(3.10):

$$CLT_{odd} = 0 \tag{3.9}$$

$$CLT_{even} = 0 \tag{3.10}$$

The second part of the control prototyping involves the decision to close one of the two clutches in relation to which gear is required to engage. This means that thanks to the presence of what can be approximated to a switch case, there will be different choices in relation to which gear will be required. Specifically, there will be a division between 2 main types of output:

- Odd gears;
- Even gears.

This provides for the engagement of the odd clutch only when the required gear is an odd gear, vice versa it will engage the even clutch in the event of an even gear request. In both cases, once one of the two clutches is totally engaged, there will be no changes for a range of time equal to the range for which the gear required by the high-level logic will remain the one currently engaged.
This logic is shown in the following Figure 3.5:

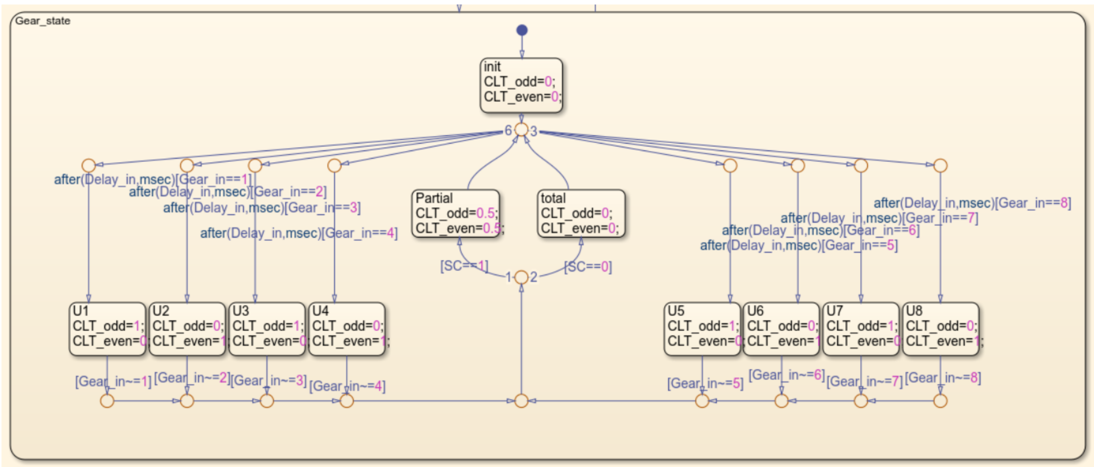

Figure 3.5: Stateflow state machine implementation for clutches control

The presence of 8 choice branches is shown, a number relating to the fact that the present logic considers 8 gear ratios (the schematic would be completely identical to the case of a 7 gear ratio with the sole exclusion of the branch relating to the eighth gear).
The shifting logic can be briefly summarized with the following Table 3.1:

Gear number	1	2	3	4	5	6	7	8
State	odd	even	odd	even	odd	even	odd	even
CLT_{odd}	1	0	1	0	1	0	1	0
CLT_{even}	0	1	0	1	0	1	0	1

Table 3.1: Input-output DCT clutches control state machine, it include all the possible clutches command combination in relation to the selected input gear

The last device inserted within these control logics related to transmission is the presence of the following input variable called SC. It can assume the values 0 or 1 and is used for identifying 2 types of transmission operation:

- $SC = 1$ at the gear change where there is a partial engagement of both clutches. It foresees that therefore there will always be a torque transferred to the wheels coming from the Power Unit;
- $SC = 0$ when gear changes both clutches will always disconnect. In both control configurations the transient time is modeled through a timing of the choice. This timing is intended to approximate the delays that a gearbox has in implementing a change in the gear ratio.

For the modeling of this transmission it was assumed that the gearbox actuation time is constant regardless of the gear change required by the high-level logic.

3.3.3 DCT integration in the complete vehicle plant model

Since the presented model and the control of the DCT are only an approximation of reality, its integration into the overall system provides for some tricks that cannot be neglected.
One of the challenges given by the integration of the previous system is certainly given by the presence of a delay between the actual request of the gear change and its real engagement, this delay is invalid both in the case of clutches partially engaged together and clutches both open during the change of gear ratio.
This is because the high level logic present in the torque split control also provides the gear to engage [17]. This in turn defines the split system also given the thermal engine and electric motor speeds.
By imposing constant wheel speed for an instant (legitimate assumption considering that the speed of rotation of the wheels is only proportional to the speed of the car itself), it is possible to calculate that the speed of the two machines present in car follows the following proportionality (3.11):

$$\omega_{EM} = \omega_{ICE} = \omega_{diff} \cdot Gear_{ratio} \tag{3.11}$$

The variation of the gear ratio therefore causes strong fluctuations in speed to the thermal and electric motor, given by the fact that torque is still transferred by the Power Unit. But during the shifting transient the equivalent inertia seen by the engine is much smaller than the usual one. This entails the need to insert control signals that during shifting regulate delivery so as not to create actual inconsistencies.
The modeled shifting time still needs attention since the high-level logic, not

perceiving variations in the speed of the Power Unity, arrives at some critical points in which it is unable to define optimal working points for the gearbox. This kind of problem related to gear change has been solved by imposing an estimate of the engine speed (both thermal and electric) during the gear change.
The estimate is based on the logic given by the high-level control, the engine speed will be given by the transient phase of the shifting once is over, so as to make the high-level controller compute the working points in the correct and most stable way possible.
For the implementation of this logic, the Enabled Subsystems Figure 3.6 have been exploited, capable of keeping their output constant once the enabling signal connected to the block becomes 0.

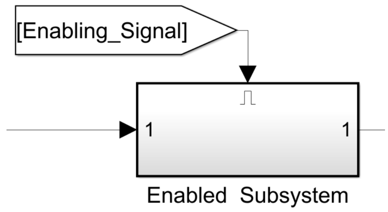

Figure 3.6: Enabling Simulink subsystem

Using this property through the activation variable $NeutralDCT$ is possible to keep the output signals from the subsystems constant in order to calculate the speed in the best possible way and then send it to the split control system. Once a change arrives, the estimate block proceeds with the following mathematical operations (3.12):

$$\omega_{estimation} = \frac{\omega_{preGearchange}}{Gear_{ratiopreGearchange}} \cdot Gear_{ratioactual} \tag{3.12}$$

The speed estimation system that will be used as input for the subsequent high-level logic is therefore resulting as the following Figure 3.7:

Note that the equation shown above is integrated into the subsystem called $'wduringGearchange'$.
The inputs of this system are:

- Rotation speed measured at the shaft before the gear ratio;
- Signal coming from high-level logic that imposes the gear to engage;
- The Neautral DCT enabling signal coming from the low level control present in the DCT when a gear shift is in progress.

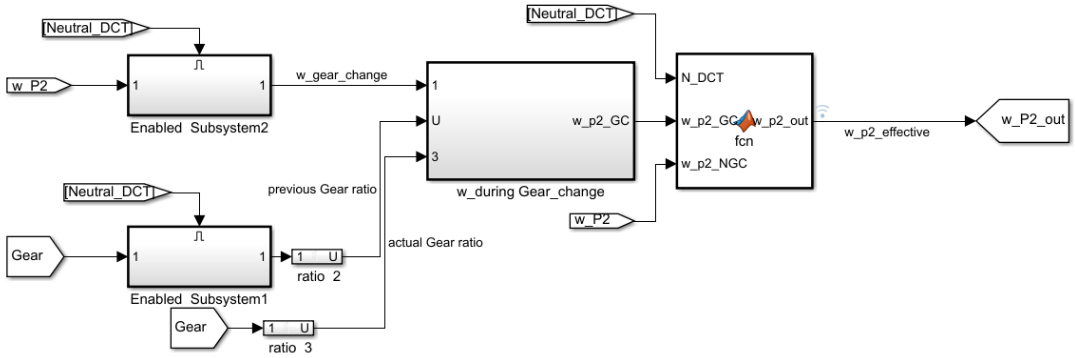

Figure 3.7: Complete setup for speed estimation during gear shift

3.3.4 ECMS-Gear Change control logic future gear estimation

One of the most important characteristics for which a dual-clutch transmission is used is certainly its short shifting time in relation to traditional gearboxes. This happens because the presence of two detached parts of the ratios allows a pre-engaging [18] of the ratios which is attached to the disengaged clutch.
It therefore happens that, once defined the current relationship as a 'state', it is relevant to create a logic capable of recognizing, with a good approximation, the successive states that will occur, with a recognition time long enough to allow the true mechanical engagement of the estimated relationship.
This kind of logic can be assumed from the following Figure 3.8:

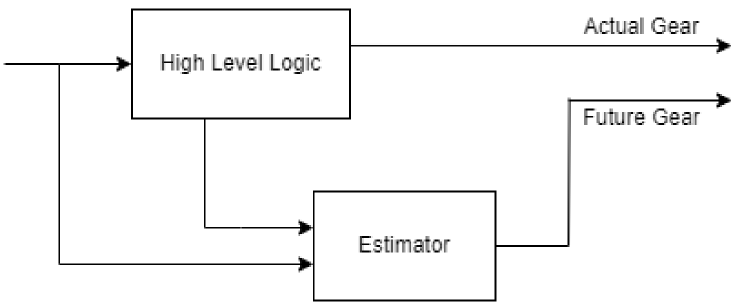

Figure 3.8: Future Gear estimation schematic representation

As can be seen, the presence of the estimator that defines the successive states is fundamental for correct functioning. The problem is therefore posed of creating a predictive system capable of providing the ratio to the gearbox to be engaged in the immediate future. To evaluate the types of estimators to be created, however, it is advisable to consider what kind of high-level logic provides the gearshift command signal.

High level control resume, torque split and gear change logic

The high-level logic provides for the drafting of an advanced logic for the torque split (ECMS [17]) capable of setting the torque states present in the car and imposing the gear at which the torque actuations must be given. The output states are all functions of the current state in which the vehicle is located and in general the input-outputs of the algorithm are represented by the following illustrative Table 3.2:

	1	2	3	4	5	6	7	8
INPUT	T_{req}	w_{ICE}	SoC	w_{EM}	$Gear_{in}$	T_{ICEavl}	$T_{ICE_{past}}$	$T_{EM_{avl}}$

	1	2	3	4
OUTPUT	T_{ICEOUT}	T_{EMOUT}	T_{break}	$Gear_{OUT}$

Table 3.2: Input-output signals of the high level logic control system

The algorithm follows the block diagram which shows how in relation to the input states that define the working point of the current machine. This algorithm estimates a cost function in relation to how much fuel is consumed by the thermal machine (ICE) and how much 'equivalent fuel' is consumed or gained (in case of regenerative braking) using the electric car.
This function is progressively obtained for all possible gear ratios starting from the initial state and finally it is evaluated which, among the total combinations is the most suitable to provide the best and more efficient split. The following schematic tries to summarize the operation just explained, Figure 3.9:

It is therefore clear how complex is the study of an estimator capable of providing an appropriate estimate of all the input states present in this logic.

Estimator definitions and implementation

The estimator is an algorithm capable of estimating the subsequent state of a variable. It is widely used in the world of engineering and investments as it is capable of giving a prediction of future states in the immediate temporal vicinity of the instant under consideration.

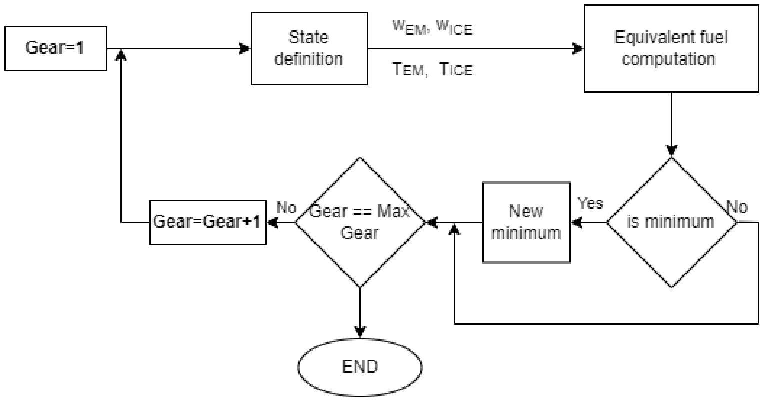

Figure 3.9: Flow chart of Equivalent Consumption Minimization Strategy (ECMS) logic

The estimator is generally an element that responds to the following equation (3.13):

$$stateOUT_{future} = f(stateOUT_{actual}, stateIN_{actual}) \tag{3.13}$$

where $stateIN_{actual}$ is relative to the input state and $stateOUT_{actual}$ is relative to the computed output state of the considered logic. Knowing these two inputs, the estimator returns the future state outputs with a certain degree of approximation. Taking the case study, considering the presence of a high number of variables both in input and in output, it is very difficult at the computational and analytical level to find a closed solution of the estimator. We therefore consider some approximations:

- Approximation of the variables characterizing the state;
- Definitions of particular cases related to the physical application.

As a first analysis for the definition of the fundamental input states, it is considered how the high-level algorithm calculates the weights for the definition of the split and how these weights are used to define the actual split and the gear to be engaged. Considering how the cost function of logic (3.14) is defined:

$$\dot{m_{eq}} = \dot{m_f}(w_{ICE}, T_{ICE}, T_{EM}) + \dot{m_{EM}}(w_{EM}, T_{EM}, T_{ICE}, SoC) \tag{3.14}$$

The weight of the electric part is a function also dependent on the state of the current SoC, which is suitably calculated to keep the battery at the desired state of charge (especially in the context of the homologation cycle this takes on a lot of weight).
The states that define what the response of the high-level control logic will be, can

be considered the following:

- wheel speed;
- torque required at the wheels;
- weighted state of charge function $f(SoC)$.

The second analysis must be carried out contextualizing the physics in which the estimate to be made is required. Considering the fact, the goal is to estimate the next gear of the gearshifting logic in order to be able to pre-engage the required gear and reduce the gearshifting times. Two notable cases can be incorporated into the logic of estimation in so that they are as precise as possible:

- Once the first gear has been engaged, the next gear will be neutral or the second, so the pre-engagement of the second gear can take place immediately;
- Once the last available gear has been engaged among the available ratios, the only possible gear to be engaged will be obviously the gear immediately adjacent in descending direction.

Starting from considering the initial states taken as dominant variables for the definition of the output, the algorithm of prediction of the exchange ratio evaluates how they will evolve in the future.
An estimate of these variables is then made by evaluating various factors that characterize their trend over time.
Since signals are not constant in time and assuming that each of these quantities and their derivative are continuous, the approach provides for the approximation of the function to its taylor [19] series at the point t_0 (t_0 time in which each signal is evaluated).
We therefore find a formulation of the following type (3.15):

$$f(t) = \sum_{k=1}^{\infty} \frac{f^{(k)}(t_0)}{n!} \cdot (t - t_0)^k \tag{3.15}$$

Assuming that the function in question has infinite derivatives.
The current problem is then moved to the choice of which degree is appropriate for the development of the series in question, then find the compromise between the following:

- The greater the development of the series, the greater it is capable of representing the reference variable correctly and precisely, taking into account that (assuming still infinite differentiability of the function) the variable would be perfectly reconstructed for k which tends to infinity;
- The development of the series requires the computation of the derivative, the latter increasing the order of the series becomes more subject to the presence of noises. During the reconstruction of the signal through development these superimposed noises lead to errors in the subsequent estimation of the desired signal.

The second problem represents the bottleneck of the estimation logic taken into consideration.
It was decided to reach a compromise that would allow to obtain results that are still worthy of note. Therefore evaluating:

- Stop at a level of development that is not too high but that is nevertheless capable of explaining the future state of the variable considered;
- Setting and design of filters capable of obviating the noise problem.

For the characterization of the development related to each variable it was evaluated to what extent they varied over time so as not to overestimate the associated development.
The first variable taken into consideration is the torque required from the wheels. Characterizing this variable, the equation of the car dynamics was evaluated (3.16):

$$J \cdot \dot{\omega_{wheel}} = T_{wheel} - T_f \tag{3.16}$$

Considering as a first approximation constant and negligible elements related to friction and viscosity, the following proportionality is found (3.17):

$$T_{wheel} \, \alpha \, \dot{\omega_{wheel}} \tag{3.17}$$

With the exception of multiplicative constants, there is therefore a direct relationship with the angular acceleration. Therefore, comparing the dynamism of the required torque and the speed of the wheels, it can be seen that the torque is characterized by variations of greater intensity.
Therefore, for the estimation of the variable associated with the torque, the development must be of a higher order than that related to the speed. The approach used therefore envisages the following developments by associating Taylor (3.18)(3.19)(3.20):

- wheel velocity;

$$\omega_{wheelf}(t) = \omega_{wheel}(t_0) + \dot{\omega_{wheel}}(t_0) \cdot (t - t_0) \tag{3.18}$$

- SoC function;

$$f(SoC(t))_f = f(SoC(t_0)) + f(\dot{SoC}(t_0)) \cdot (t - t_0) \tag{3.19}$$

- wheel requested torque.

$$T_{wheelf}(t) = T_{wheel}(t_0) + \dot{T_{wheel}}(t_0) \cdot (t - t_0) + \frac{\ddot{T_{wheel}}(t_0)}{2} \cdot (t - t_0)^2 \tag{3.20}$$

Considering a not excessively high prediction time, the noise associated with the derivation is considered negligible for the two lower-order developments. On the contrary, for the future estimate of the torque a filter is provided to limit the noise. The implementation of Taylor's development is carried out through the use of an appropriate Matlab function. The system for estimating the input variables can therefore be seen in the following image Figure 3.10:

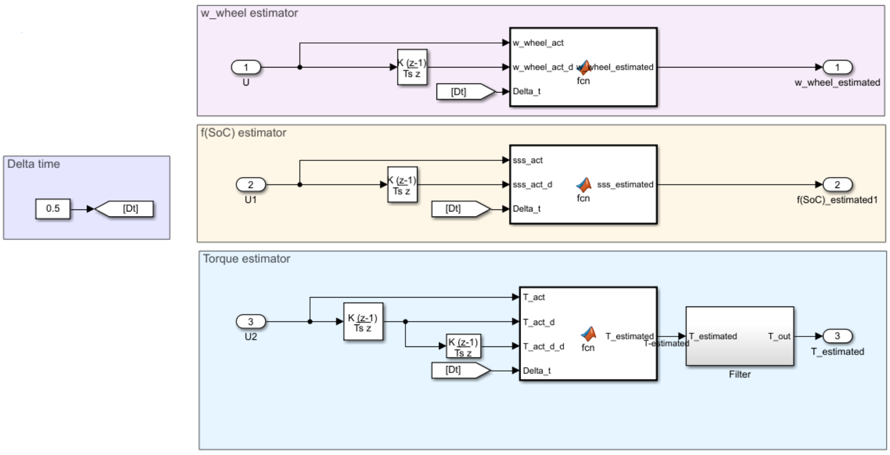

Figure 3.10: Simulink implementation for the variable future estimation using derivatives

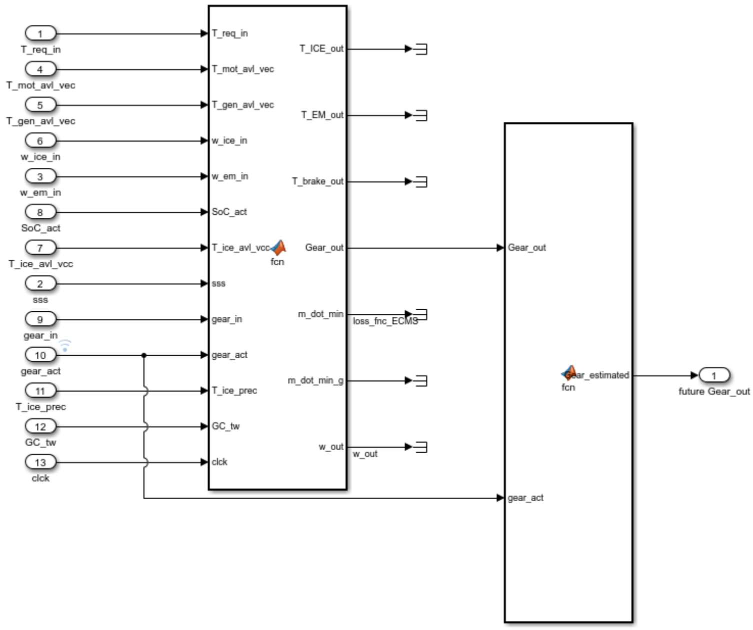

Figure 3.11: Function integration for computing the notable gear estimation

The estimator is therefore created through the parallel use of the existing high level logic, having however as input states the estimates obtained from the previous step and leaving the remaining input states unchanged. Finally, a Matlab function is inserted which recognizes the notable cases at the extremes of the gear changes Figure 3.11.
Taking into consideration the presence of ADAS in the car, it is natural to think that these estimated quantities can be somehow extrapolated from the data relating to the external environment. A first finding is that the speed of the wheels (in non-slip conditions) is proportional to the speed of the car. Knowing therefore a possible speed profile obtained through the use of ADAS, the estimate relating to the speed of the wheels improves considerably.
Other additions can be proposed in the same way for the estimation of T_{wheel} and the $f(SoC)$ but these are not covered in this paper.

3.4 Results and comparison between DCT control logics and transmission composed by torque convert and automatic gearbox

For the validation of the model built, its operation in a well-defined situation was evaluated, then comparing the results with the system previously present within

the model.
The estimation of the results is carried out by evaluating factors characteristic of motion.
An overall system is then established for the simulation of some reference signals and the subsequent evaluation of the characteristic outputs.

Validation and simulazion

In the Matlab/Simulink environment, an overall evaluation system is then prepared Figure 3.12:

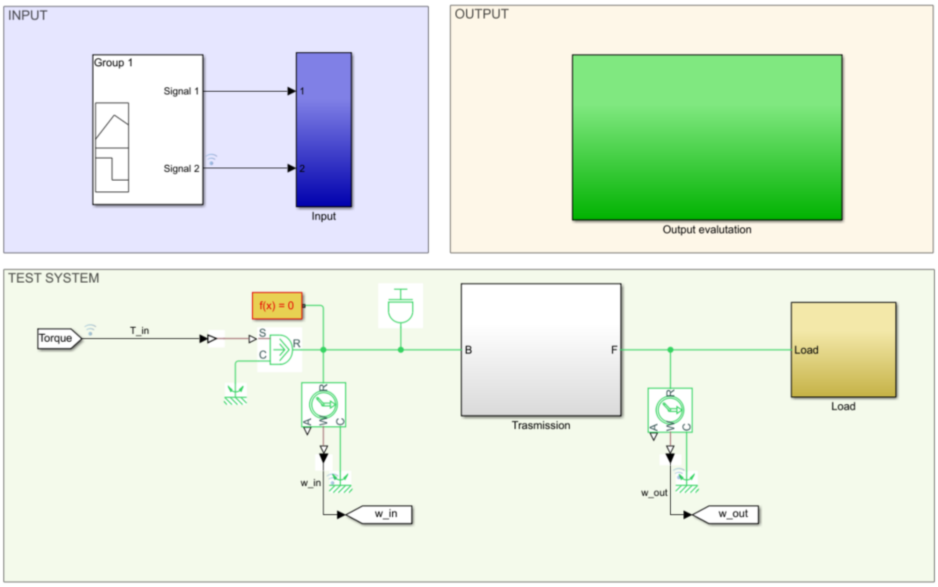

Figure 3.12: Transmission model test setup

Within this, it can be noted the presence of:

- Ideal torque source, it is capable of imposing the given torque value. This is considered as the input of the system;
- Inertia, explained earlier. Characteristic values have been assigned specifically for load modeling and to limit the output speed from the ideal torque source;
- Ideal speed sensors, they are used to obtain the speed of the transmission shaft so that it can be evaluated both in input and in output of the modelled

system .

System input

The inputs of the system will be tabulated below. They have been used so that we can evaluate the behavior in both increasing and decreasing shifting gear. To evaluate the response of the model Table 3.3.

$Gear_{in}$	T_{in}
0-1-2-1	0-1 Nm

Table 3.3: Input of the test signals for the comparison among the DCT implemented control logics and the torque converter + automatic gearbox transmission

For the definition of the inputs, the Signal Builder block is used. It allows the definition of completely custom signals in a precise way.
The gears imposed during the complete simulation of the system are shown below Figure 3.13.

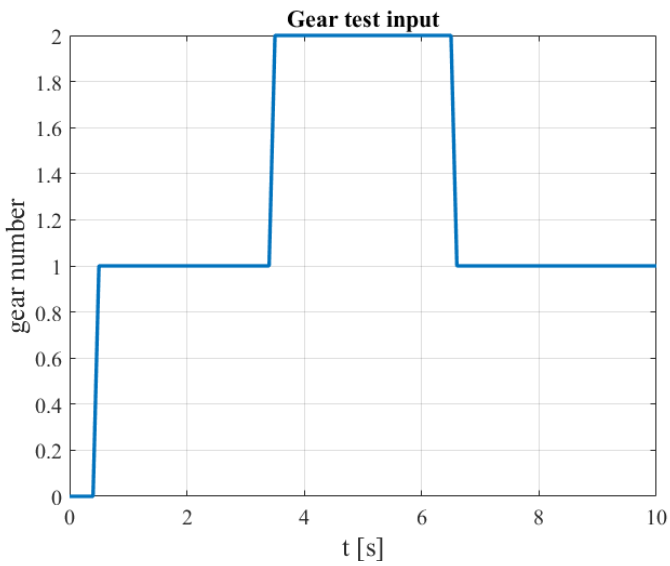

Figure 3.13: Gear input test

Specifically, the following quantities have been washed out and evaluated Table 3.4:

w_{in}	w_{out}	$Clutch_{even}$	$Clutch_{odd}$

Table 3.4: Output Transmission test signal, they are used for the evaluation of the transmissions model and control performances

For the evaluation of these, there are 2 possible cases of operation of the modeled DCT to evaluate the differences.
In the first case, a total disconnection of both clutches is imposed for a well-specified time interval. In the second, the behavior is shown when the control related to the clutches allows an intermediate value in which both clutches are both partially engaged.
The result of the two simulations is then shown below Figure 3.14:

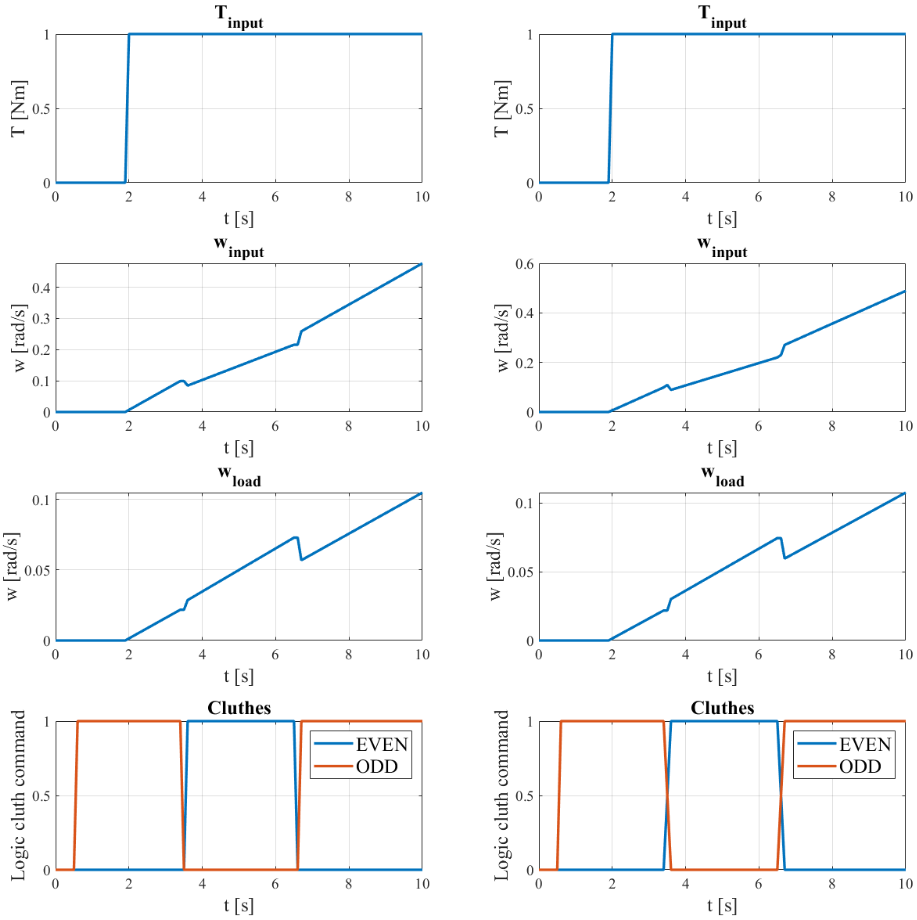

Figure 3.14: Comparison not-engaged and partial engaged cluthes during gear shift

Specifically, it is noted that the differences in the comparison are marginal in the shift situations, in which the second allows a slightly more continuous transfer of torque. The dashed line shows the ideal trend that should have the angular velocities in the input-output. However, it is clear how the behavior tends to be similar to the ideal one Figure 3.15.

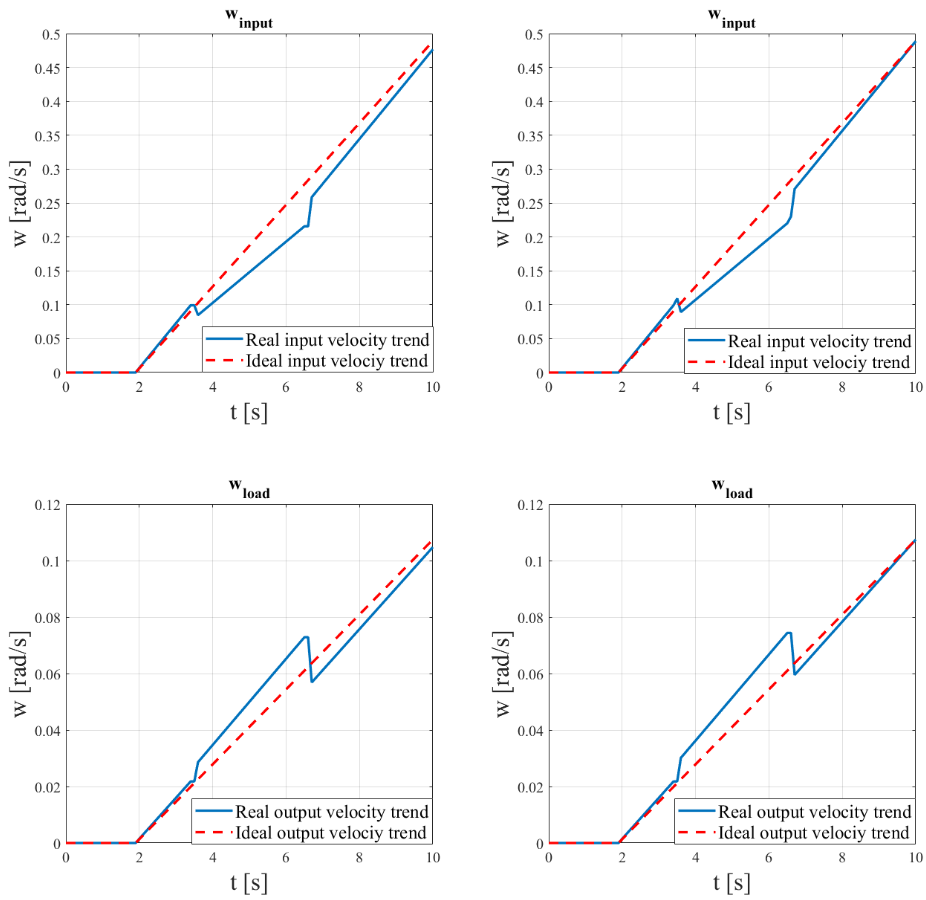

Figure 3.15: Motion transfer comparison between the two implemented DCT control logics

As a second comparison, reference is made to the previous transmission model, which is characterized by the presence of a Torque Converter (TC) with the gearbox downstream. Comparing the two trends shows a particular difference. The TC+Gear configuration shows a strongly non-linear relationship between the input speed and the output speed. This occurs due to the presence of the Torque Converter that adds a speed shift between its primary and its secondary, which transfers the motion by dragging. This phenomenon also leads to having a higher input speed at the same final angular velocity of the load Figure 3.16.

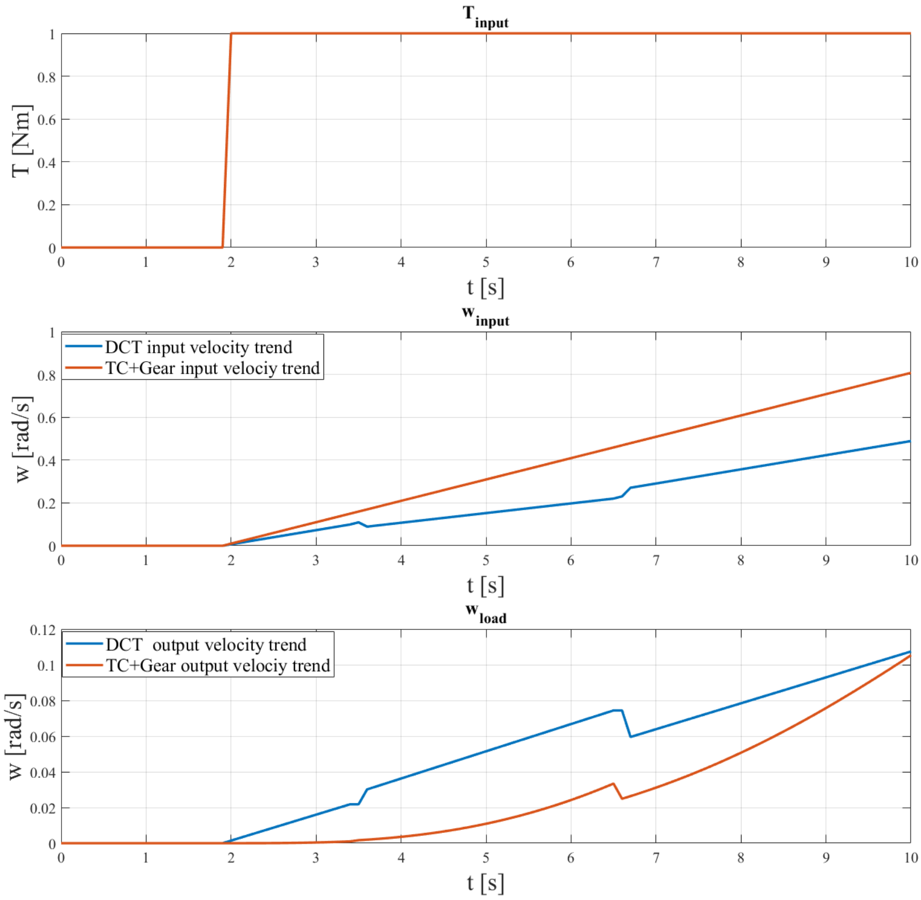

Figure 3.16: Motion transfer comparison between automatic gear with torque converter and DCT

For the evaluation of the efficiency concerning the 3 configurations shown, the actual energy transfer that allows a certain load condition must be evaluated. This assessment was made using the same system shown in Figure 3.12. The energy fed into the system by the 'source' (in a real vehicle is represented by the motorization) is then evaluated about obtaining a final speed of the load (rotational speed of $\omega_{load} = 1{,}72rpm$).
The results of this evaluation have therefore been collected and shown in the following table, they are expressed as a relative percentage deviation from the basic configuration (torque converter + automatic transmission) Table 3.5:

Configuration	E_{source} [J]	improve %	improve [J]
Automatic gearbox + torque converter	4.545	baseline	baseline
DCT configuration all clutches disengaged during gear shift	3.302	27,35	-1,24
DCT configuration clutches partially engaged during gear shift	4.545	26,6	-1,21

Table 3.5: Output Transmission test signal, they are used for the evaluation of the transmissions model and control performances

The results show that the DCT configuration is more efficient, as it manages to impose a certain state on the load with less energy. In relation to the type of control configuration present in the DCT, the differences show how the control with partial presence engagement of both clutches is more efficient.

3.5 Future Gear Estimator results

For the validation of the results related to the estimator, it is important to evaluate different dynamics of use of the car to test critical issues related to the proposed solution.
It is therefore imposed as a speed profile that is related to the homologation procedure where the vehicle in question belongs, considered as a varied speed trend and approximate to what could really be normal use of the car.
The results are then obtained and collected below Figure 3.17:

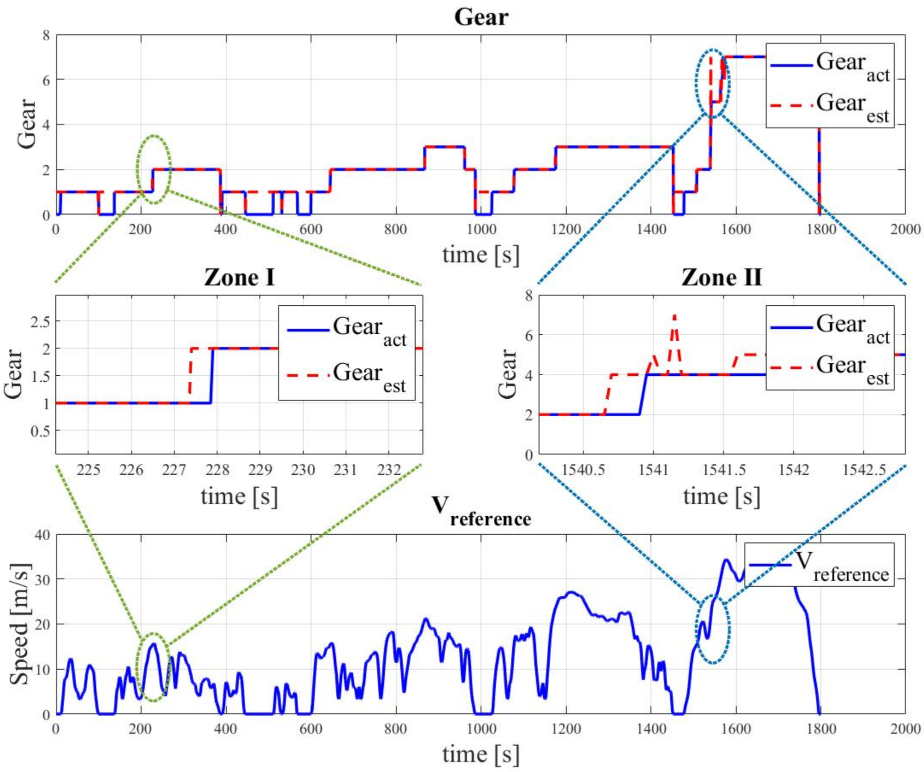

Figure 3.17: Next Gear Estimator results

- A Zone I is characterized by the correct evaluation of the next gear. This correct evaluation is provided by the logic implemented about 0.5 seconds before the real change takes place;
- A Zone II in which there are strong oscillations obtained from the estimator. The zoomed image explains how the obtaining of the future gear takes place partially correctly but there are oscillations.

The proposed method represents a first approach to the estimation of the next gear. Appreciable results are obtained in situations where there are no large dynamics of the vehicle. However, it comes to incur problems during phases in which torques and speed tend to have high harmonic content. In those cases, the estimate is negatively influenced.

Chapter 4

Adaptive fuzzy logic supervisor on the throttle controller

4.1 Introduction

The overall control architecture examined by the vehicle is characterized by the presence of an external and an internal feedback. Both were created appropriately to exploit the dynamics of the vehicle and to better follow the external references appropriately perceived by the ADAS sensors located in the car.

4.2 State of art

4.2.1 Internal loop throttle controller: 'Driver'

The driver is widely used as:

- Internal loop in control architectures with the presence of ACC to improve stability and improve low-level actuations;
- As a bridge for the simulation of a car in order to impose a predefined speed or acceleration profile and evaluate vehicle performance, an element of fundamental importance when designing vehicles in order to evaluate any problems and estimate consumption during homologation cycles.

[20] It explains specifically how a driver is used and how it is exploited within a simulation environment in order to highlight the performance of a vehicle. It deals

with the modeling of a driver that acts on both lateral and longitudinal dynamics.

[21] It shows how the implementation of a suitably calibrated driver is of vital importance to allow a tracking of the reference profiles. It shows how the exploitation of an adaptive PID by means of suitable algorithms is an improvement in the performance of the overall driver.
Drive modeling can take into consideration all the dynamics of a vehicle such as:

- lateral dynamics;
- longitudinal dynamics.

The modeling of a driver for lateral dynamics should be used if the application in question needs the characterization of this dynamics, such as for example evaluating the stability control during a curve, where an appropriate steering profile must therefore be imposed by have that kind of dynamic.
The present raises the problem of characterizing a driver for longitudinal dynamics, capable of making the vehicle follow a specific speed or longitudinal acceleration profile/throttle, this when in the modeling of the vehicle only the longitudinal dynamics is characterized.
For the creation of this control architecture, the first step was to evaluate the implementation variable. It takes into account the method by which the vehicle changes its current state. The evaluation of this aspect therefore considers the implementation of the state to be controlled and the relationship that exists between it and the implementation command. When a driver is created there are two types of reference commands:

- Speed;
- Acceleration.

Both have similar modeling architecture characterizations with some differences.

Speed feedback throttle control model

To create a driver capable of varying the throttle by means of a speed reference signal, the deviation between the two state variables, reference speed and current speed, is evaluated, in order to determine what the magnitude that the control must have to get the appropriate reference.
The modeling therefore takes the following relation (4.1):

$$\Delta v_{err} = v_{ref} - v_{act} \tag{4.1}$$

As this error increases, the output required by the control grows in reference to the fact that the greater the difference, the greater the acceleration or deceleration required to produce this variation. For the modeling it is then used with a suitably calibrated PID controller to guarantee the imposed dynamics.
The generic formulation of the PID is the following (4.2);

$$throttle = (P + I \cdot \frac{1}{s} + D \cdot \frac{N}{1 + N \cdot \frac{1}{s}}) \cdot \Delta v_{err} \tag{4.2}$$

The presence of the derivative part is omitted in this architecture as the presence of only the proportional (P) and integral (I) part guarantees the required performance.
The final formulation then comes to be (4.3):

$$throttle = (P + I \cdot \frac{1}{s}) \cdot \Delta v_{err} \tag{4.3}$$

The modeling is therefore as follows Figure 4.1:

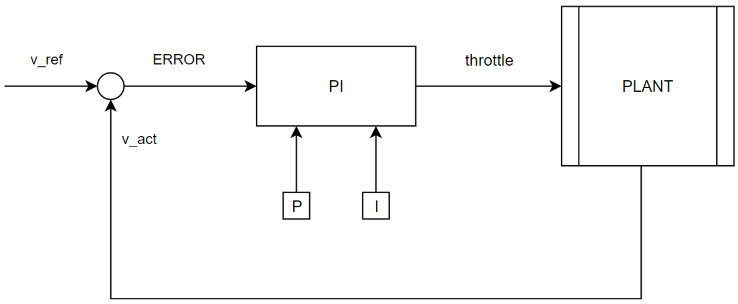

Figure 4.1: Throttle control schematic model using speed feedback

The gains characterizing the control represent the aggressiveness with which the driver will follow the requested reference.
This type of driver modeling is commonly used for prototyping and simulation of cycles and homologation procedures, as it allows to give an imposed speed profile as opposed to that of WLTP.
One of the main problems related to the modeling of this internal loop is its adaptability to the possible presence of a high-level logic that gives the speed profile since in very common application areas the profile is not provided through references in velocity of the center of mass but in acceleration required by the vehicle.
The problem then becomes to adapt this internal loop to a different reference signal.

Driver aggressivity

During the pursuit of a certain type of speed profiling, how this pursuit is carried out, that is, how quickly and with what error, as mentioned, are dictated by what kind of aggression the internal controller implements.
The presence of this parameter therefore leads to having to characterize how and with what flow rates the consumptions are affected by giving the same reference as input and evaluating the response of the overall system in terms of time.

[22] Taking as data the new European RDE homologation cycles carried out in Portugal, It studies that correlation between the consumption of vehicles driven aggressively and not aggressively. It assessed how an aggressive driver causes an increase in consumption of up to 7% compared to the non-aggressive case.

[23] It starts from data obtained from several real drivers to extrapolate the different driving patterns in order to evaluate consumption and evaluate what type of pattern causes an increase in consumption.

4.2.2 Fuzzy logic

The usual control architectures base their algorithms and in general their characterization on a mathematical analysis of the analyze type on the system in which they will be used.
This type of approach, if applied to very complex systems or systems with many intrinsic uncertainties in their modeling, leads to difficult problems of characterizing an appropriate control.
The presence of control architectures capable of solving this problem is therefore becoming increasingly useful in the world of controls, precisely because of their ability to be tolerant to the type of system used.
A Fuzzy control logic is therefore proposed, capable of obviating the problem of analytical analysis of the driver in its modeling. This logic acts in a predictive way on the aggressiveness that the internal throttle control present in the system has on the implementation of the throttle of the car. The problem therefore tries to dynamically vary the torque required by the vehicle to avoid known situations of higher consumption in specific contexts.

In the text [24] it is presented how different types of aaggressivity affect the consumption of a hybrid vehicle in relation to the driving context used. It therefore generates a method for identifying the driving style and adapting it to the scenario in order to reduce the consumption of the vehicle. They are obtained through a

Fuzzy algorithm and one using complex Genetic Algorithm (GA).

The control logic developed through the use of ADAS sensors wants to dynamically calibrate the response of the vehicle through the use of a Fuzzy logic.
The presence of ADAS is capable of providing information on the future of the path in which the car will be used. Recognizing the characteristic factors, the throttle re-calibration allows a reduction in consumption in of the track.

Fuzzy theory

The Fuzzy logic, as opposed to the multivalue and binary ones, allows to obtain continuously 'fuzzy' values included in an interval [0,1].

[25] The definition of this logic also allows to have statistics bases and for this reason it is used in many applications ranging from finance to engineering applications.
The use of this logic in the theory of controls therefore allows the creation of a robust control system and at the same time immediate in development and calibration.

[26] It explains how a fuzzy control logic must be implemented. In the specific case, it is pointed out how the logic was implemented in the case study under examination. To define the complete logic, mandatory steps must be followed, shown in the following Figure 4.2:

Specifically, we identify how each of these badges is important for the implementation of the logic. fuzzification is the first phase that allows to associate an input variable with a characteristic fuzzy value. Here, different possibilities of characteristic ranges deriving from the input quantity are defined.
It also defines the type of relationship between the input quantity and the associated Fuzzy input variable.
The second step taken into consideration shows how the input variables are processed and in turn used for the decision. Here, the presence of the decision part comes into play. It is characterized by real rules in which the input signals obtained from the input fuzzification are related to the rules chosen by those who decide the characteristics of the control. Examples of rules are shown that you can insert (4.4):

$$if(ConditionA \ \& \ ConditionB) \rightarrow C1 \tag{4.4}$$

As shown, the rules that characterize the bonds are quite similar to the conditions

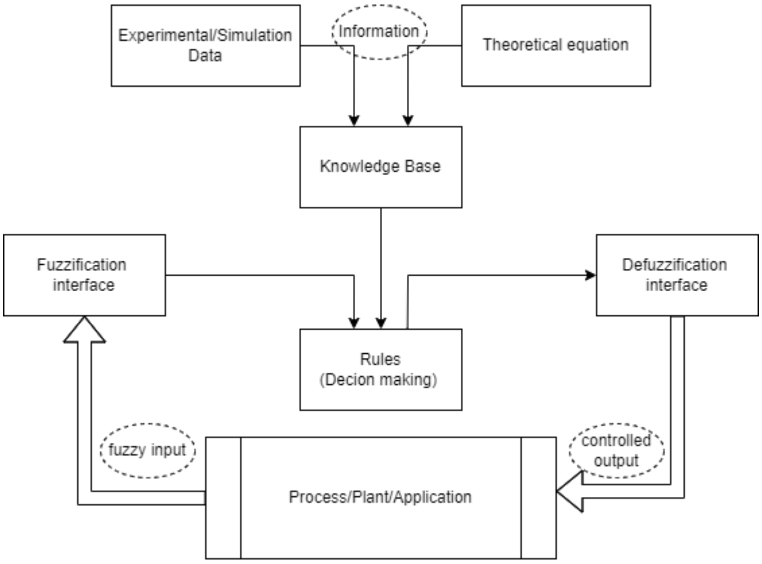

Figure 4.2: Fuzzy logic control general schema

of Boolean algebra but with the difference that even the possible output state is kept nuanced.
Assuming that the fuzzy logic has only a scalar value output, the number of rules present is in relation to the number of inputs present inside the fuzzy. This implies that the insertion of many input signals makes the architecture created very difficult at the computational level. Therefore, assuming that at the exit from fuzzification each input signal is characterized by 7 membership function Table 4.1:

$\frac{A}{B}$	HL	L	ML	M	MH	H	HH
HL	1	2	3	4	5	6	7
L	8	9	10	11	12	13	14
ML	15	16	17	18	19	20	21
M	22	23	24	25	26	27	28
MH	29	30	31	32	33	34	35
H	36	37	38	39	40	41	42
HH	43	44	45	46	47	48	49

Table 4.1: Fuzzy rules complete possible combination using 2 input signals, which are obtained both using 7 fuzzification membership functions

The possible combinations of signals fall into the order $n \cdot n$ that is, the logic would allow 49 possible association values in the rules present (49 different combinations).

Here, the problem arises when making the control logic light enough to be applicable to real time systems. For this reason, a clustering is usually carried out between the possible rules present so that by defuzzyfication the result at the computational level is acceptable for the application.

Fuzzification

The first step to accomplish in using a fuzzy controller is to 'blur' the signals of interest, thus blurring signals that can initially be considered sharp.
This procedure is usually carried out by special softwares that facilitate the conversion. The latter is performed by associating the clear signal through the presence of one or more fuzzy membership functions. They can be assumed through knowledge of the system in question or through classification approaches (ML).
The possible creations of fuzzy membership functions are defined by specific mathematical properties. The most used are characterized by the following functions Figure 4.3:

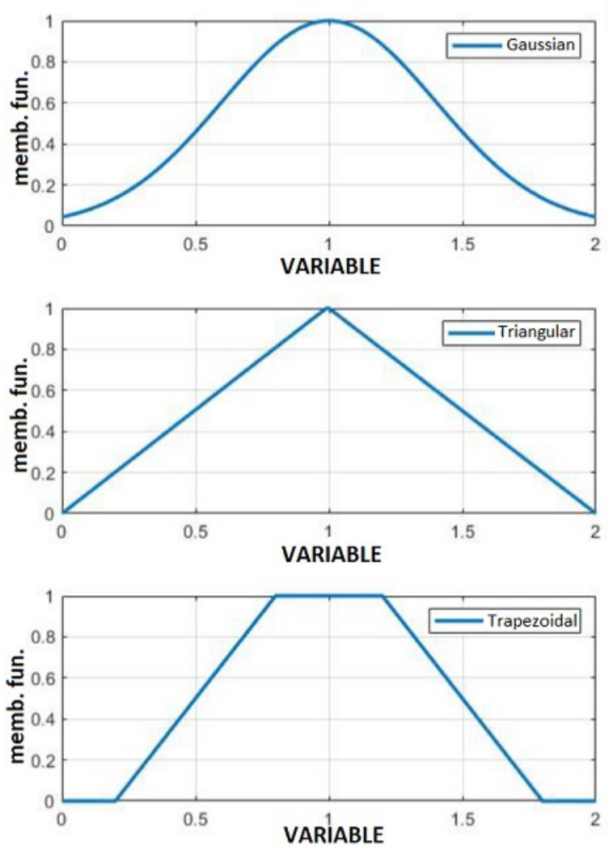

Figure 4.3: Fuzzy single membership function example

The previous images correspond to the following possible fuzzy membership functions, obviously note that the possible functions are many and here are just listed some of the possible types.

- Gaussian;
- Triangular;
- Trapezoidal.

The presented therefore allow to convert possible discrete signals into signals that vary continuously so that they can subsequently be used during the other phases of the logic.

Rules

The decision-making part of Fuzzy logic follows normal human language, setting rules and conditions to specific situations and associating appropriate consequences.

[27] explains in detail how to interpret fuzzy rules and associations between them and a non-fuzzy logic.
The drafting of the rules represents one of the greater degrees of freedom of fuzzy. It allows to associate an output with a very precise combination of conditions. The basis of which the rules are chosen is strongly dependent on various factors such as:

- Previous knowledge of the systems, capable of providing general lines on the fields of action of the application;
- Experimental phase of data collection of the system, essential for understanding system dynamics and analyzing what kind of rules and signals to use;
- Theoretical analysis of the system, a possible step to be carried out in the case of systems that are not too complex and which allows to find theoretical relationships capable of providing information.

Even one of the preceding procedures greatly facilitates the process of forming the logic in question.
By interpreting this information, it allows the drafting of appropriate rules and the subsequent defuzzification of the signal. Specifically, the rules have an example in the formulation shown in (4.4).

The number of possible rules to insert depends on how membership functions are used:

- number of input membership functions (fuzzification signal);
- number of output membership functions (defuzzification signal);
- number of inputs;
- number of output.

A relevant problem that arises already in this part, during the drafting of the rules to be considered in the created logic, is a possible reduction of the rules in order to avoid computational problems. The study part of the system comes in handy which allows the reduction of the fuzzy membership functions and the consequent decrease of the rules to be inserted.

Defuzzification

The section relating to Defuzzification is also of important weight within the creation of the fuzzy controller because it is the section that allows to pass from the fuzzy world to the real variable to be used in the control architecture used.

[28] shows how defuzzification is a very delicate topic and how the use of mathematical methods greatly modifies the behavior of the created logic, modifying both computational and mathematical characteristics.

As in the case of fuzzification, the main point is to associate a real variable with a corresponding fuzzy. In this situation the characterization is not unique having the possibility of associating the same set (or combinations of them) to several possible real values.
Specifically, different characteristics of the obtainable result are encountered. The imposed rules and the previous knowledge of the system make possible to characterize the membership functions, so they must be converted by means of possible mathematical operations or relations created in a thoughtful and univocal way the different combinations obtained from the rules. Using trapezoidal fuzzy membership functions as typology (the characterization is completely equivalent with other types of functions), the fuzzy extrapolates information from the rules to assign a certain amount of the output membership function (4.5)(4.6):

$$if(ConditionA\ \ \&\ \ ConditionB) \rightarrow C \tag{4.5}$$

$$if(ConditionE\ \ \&\ \ ConditionL) \rightarrow D \tag{4.6}$$

The membership function are then defined as (4.7)

$$\mu_1(C) \qquad\qquad \mu_2(D) \tag{4.7}$$

For each, this function is then defined with the corresponding variable. Each trapezoid is then sectioned by a straight line which decreases its 'height'. An area relative to the specific membership function is then obtained Figure 4.4.

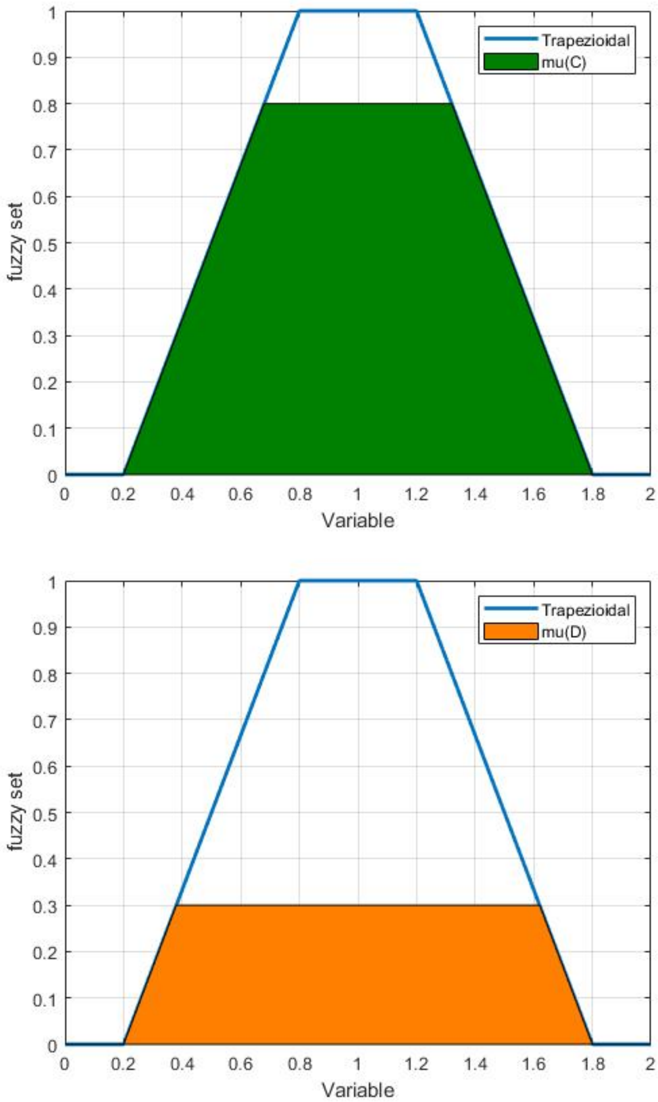

Figure 4.4: Defuzzification membership function area

Obtained all the corresponding areas. The next step is determining a value to associate to them. The associative possibilities are many and all explained in [28]. Specifically, the most used are:

- Center of Gravity (CoG) or Centroid, the most commonly used;
- MOM (half of the maximum) usually used for symmetrical membership functions;
- FOM Choose the first of the first as an element;
- LOM Takes the last of the highs as an element.

Note that: MOM and CoG consider the overall area to give a result, on the contrary FOM and LOM use only the maxima and consequentially lose information.
However, they come to be useful in the case of systems where computational time is of high importance as they are certainly more computationally light than the former.

4.3 Adaptive throttle control model implementation

4.3.1 Acceleration feedback throttle control model

Considering therefore that the integration with an ACC occurs through an accelerating reference, the first step to be carried out is that of creating a 'Driver' capable of acting on this state.
Taking a PID into consideration in this type of architecture too, it is evaluated the difference (or error) between the reference and the feedback coming from the longitudinal model and we act in the same way on the throttle.
Although it would be possible to remedy the presence of an internal loop in the case of an accelerating reference, since the throttle can be considered proportional to the acceleration, the presence of the internal loop guarantees better performance and better stability of the overall system, improving the 'tracking' of the system to the imposed references.
The formulation is therefore based as in the previous case on the error between reference and current value (4.8):

$$\Delta a_{err} = a_{ref} - a_{act} \tag{4.8}$$

As in the previous case, a PID controller is used.
In this modeling an issue concerns the acceleration measuring of the vehicle. Since it has a high dynamics, the problem related to the measurement noise arises. Therefore, suitably calibrated filters are set up which guarantee an adequate final dynamics and an appropriate reduction of the noise.
The overall modeling of this control is then presented Figure 4.5:

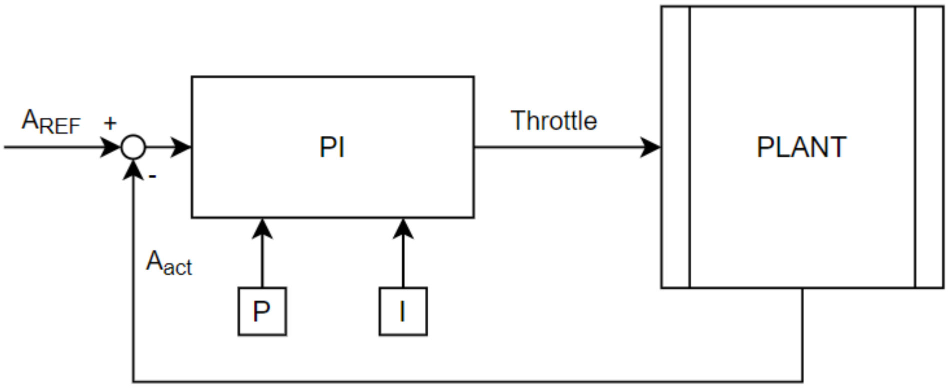

Figure 4.5: Throttle control model using acceleration feedback

Also in this modeling the presence of the P gain affects the aggressiveness of the signal tracking.

4.3.2 Adaptive PID-fuzzy throttle controller

As described, the change in aggressiveness causes a change in consumption.
The controller therefore poses itself as a high-level logic capable of recognizing future driving patterns and adapting aggressiveness as a consequence.
This control architecture is implemented using fuzzy logic supervisor.

Input signal setup: ADAS sensor data extrapolation and fuzzy input signal definition

For the recognition of the future pattern, suitable ADAS sensors are used, capable of recognizing the surrounding environment. Therefore, by implementing appropriate recognition software, it is possible to extrapolate the following information:

- Horizontal and Vertical Signage;
- Any obstacles inside the carriageway;
- Relative position of the preceding vehicle and / or objects placed in the immediate vicinity of the vehicle;
- Relative speed relative to vehicles in the seats in front of the car in question.

Using this data, the high-level logic is able to provide a vector of future speed that we will travel in order to give a forecast of the speed profile of our car. Using this vector (4.9):

$$v_{vec} = [v(t_0), v(t_1), v(t_2), ..., v(t_N)] \tag{4.9}$$

It is possible to extrapolate different parameters characterizing the future profile. The proposed work is to define the appropriate parameters to be analyzed and calibrated a fuzzy logic capable of supplying a correlated aggressiveness that reduces consumption without significantly affecting performance.
The evaluation of the inputs fell into signals that made possible to identify ineffective behaviors in order to penalize them by making the pursuit as slow as possible. The choice fell into the following:

- This formulation (4.10) allows to associate with which deviation (in module) the initial speed deviates from the average calculated in the forecast;

$$\Delta v_{mean} = |\frac{1}{N} \sum_{k=1}^{\frac{time}{step_{size}}} v_{vec}(k) - v_{vec}(1)| \tag{4.10}$$

  Where N is the number of total samples

- This equation (5.17) evaluates the difference (in module) between the initial speed value and the final speed value foreseen by the high level logic;

$$\Delta v = |v_{vec}(N) - v_{vec}(1)| \tag{4.11}$$

- The current vehicle speed is evaluated (4.12).

$$v_{actual} = v_{vec}(1) \tag{4.12}$$

The quantities were used to characterize the following types of speed profiling which is evaluated as erroneous and expensive Figure 4.6:

- Harmonic phenomena present on speed;
- Short-lived impulsive phenomena present on the speed.

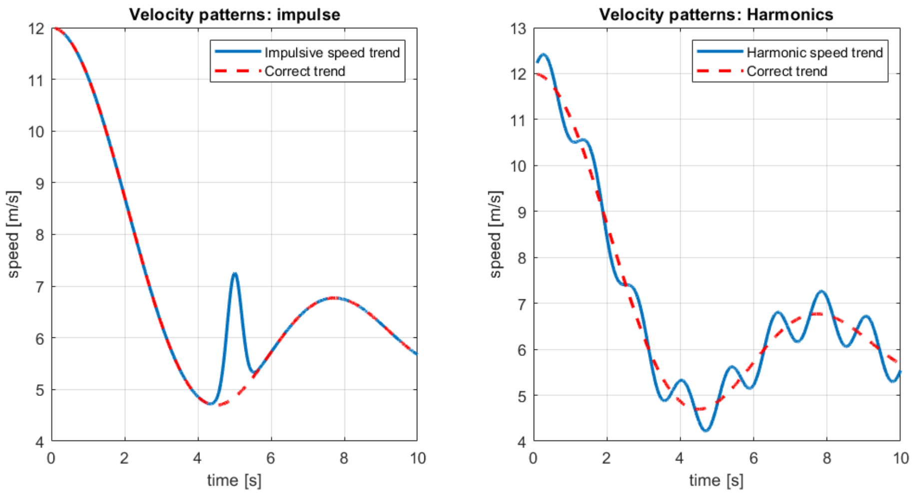

Figure 4.6: Possible inefficient speed patterns

The procedure to be implemented is the one previously explained, therefore the fuzzification of the signals in question takes place. For the choice, two different membership functions were evaluated for the characterization of the input signals, by giving a 'Low' or 'High' value as a fuzzy membership function if the input value exceeds a certain threshold of interest.
This procedure was repeated for all three input signals. The choice fell on two membership functions as it can be considered a fair compromise between an accurate subdivision of the signals and quite low computational level.
A vector that estimates the future profiling of the vehicle is then evaluated, by calculating the signals previously shown, their magnitude is evaluated in order to elaborate a threshold that will divide the signals into 'Low' and 'High'. To obtain these signals, the behavior of the vehicle is simulated in order to obtain the trend of the signals treated.
Taking in consideration this simulation, the European WLTC homologatio is imposed as speed profile. It is considered appropriate given its great variety of possible sub-scenarios present inside.
Evaluating the notions acquired from this information, the fuzzy membership functions used are then created. The type of functions has fallen into the trapezoidal ones and the various thresholds have been created and set in order to obtain the desired values with the pre-established operating conditions and limits.
The fuzzy membership functions obtained for the signals are shown below Figure 4.7;

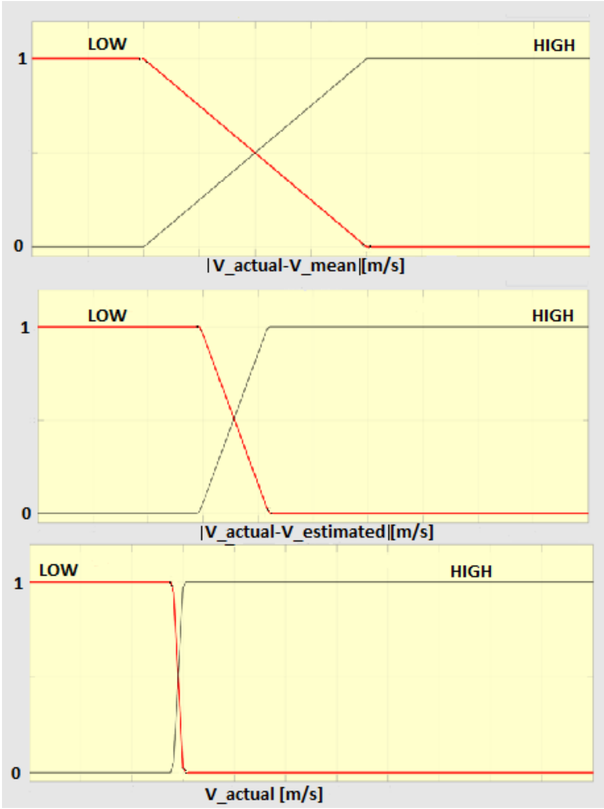

Figure 4.7: Fuzzy membership function input signals

Obtaining these therefore concludes the fuzzification process of the signals to be processed by means of the fuzzy controller.
As previously described, the implementation variable of this fuzzy control logic is the proportional gain present in the PID located in the internal point of the total control architecture present. The rules with which the possible aggressivity value is changed and for the decision of how many possible membership functions characterize the aggresivity.
Therefore, the presence of a complex system would not guarantee high enough performance to be computed in any hardware. The choice of possible functions therefore falls into a range of three cases:

- LOW aggressiveness (identified as L), set in cases in which incorrect behaviors are recognized by the driver in order to dampen them;
- MEDIUM aggressiveness (identified as M), value to be set in 'gray' driving areas where the behavior is not very inefficient;
- HIGH aggressiveness (identified as H), to be used in case of driving profiles recognized as appropriate and efficient.

Knowing how the three input signals are fuzzified (each one by means of 2 fuzzy membership functions), it is possible to calculate all the possible combinations to which associate the aggressiveness value. Considering this information, the obtainable combinations are (4.13):

$$Combination = 2^3 = 8 \tag{4.13}$$

The following possibilities are then tabulated by identifying 1 = $High$ and 0 = Low for the specific membership function of the input signal Table 4.2:

Combination	Δv_{mean}	Δ v	Velocity	Aggressivity
1	0	0	0	L
2	0	0	1	M
3	0	1	0	L
4	0	1	1	M
5	1	0	0	L
6	1	0	1	M
7	1	1	0	M
8	1	1	1	H

Table 4.2: Fuzzy logic rules implemented in the considered adaptive supervisor

The values associated with the corresponding aggressiveness level in the two cases of internal loop (speed-acceleration) taken into consideration are tabulated below Table 4.3

————	Internal acceleration loop	Internal velocity loop
LOW	0.1	0.05
MEDIUM	0.2	0.15
HIGH	0.3	0.25

Table 4.3: Proportionals (P) gain imposed by the fuzzy logic supervisor used in the throttle PID controller

In relation to the defuzzification section, it examines the type of membership functions are used for the output signal (which for the application in question is of the symmetrical type), the MOM method is used for defuzzification.

Integration of the fuzzy logic adaptive supervisor on the PID controller

The control architecture is then integrated into the overall system. The previously developed logic is then implemented using a tool present in the Matlab/Simulink environment capable of providing assistance in the training and implementation of all the processes for creating a fuzzy logic. The tool in question is the following "fuzzyLogicDesigner".
For integration, the system from which the information vector on the future speed profile arrives is obtained from the higher level logic present in the final control.
The output of this logic instead, as explained, dynamically controls the proportional part.
However, varying dynamically could pose a problem related to possible jumps in the delivery of the throttle and consequently of the torque to the wheels. For this reason, rate limiters have been included in the final architecture that will dampen the speed of these possible variations.
A block diagram is then shown representing the overall system Figure 4.8 in question and how it has been specifically integrated, the steps ranging from the perception of the signals that allow the estimation of the future speed to the dynamic variation of the previously explained architecture are noted.

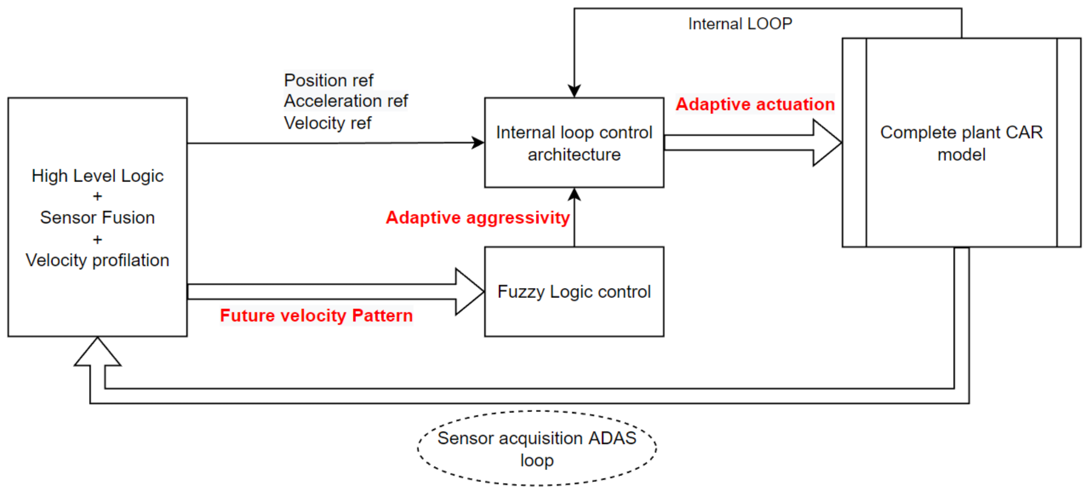

Figure 4.8: Global High level Control architecture of the supervisor logic system

A determining factor for the reduction in consumption is the premonition interval considered. Since the future speed is difficult to characterize and estimate (given the high number of possible patterns that a vehicle can have), long forecasts are complex to obtain. The possibility of having a better estimate of the future improves the capabilities of the architecture in question.

4.4 Comparison between adaptive and non adaptive control system

In order to obtain the results of the control architecture created, a scenario is examined. It considers two cars one in front of the other, the one behind is the examined vehicle. This scenario allows the obtaining of the future speed vector in relation to the speed of the vehicle to be followed.
The following are then defined:

- The first (Lead vehicle) that proceeds in front has the WLTP speed profile related to the specific vehicle category;

- A second (Ego vehicle) is characterized by the presence of sensors for perception. It is the vehicle to which the algorithms are applied and whose performance in terms of emissions are evaluated.

The simulation takes into account the consumption of the ego vehicle. Specifically, the results obtained are compared with both an internal adaptive and non-adaptive control loop.
The quantities examined are:

- The speed that the ego vehicle implements;

- Trend and the final value of the state of charge of the battery;

- Consumption related to CO_2/km produced by the vehicle.

Specifically, the previous list is shown in the Figure 4.9:

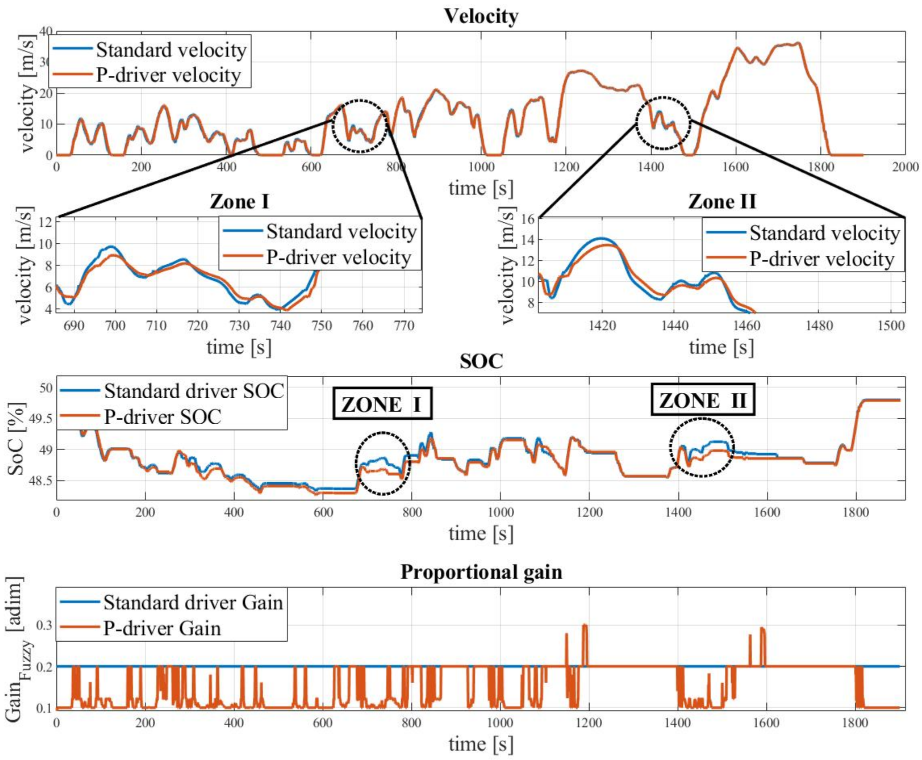

Figure 4.9: Comparison Adaptive/NON-adaptive control

Speed trends tend to discourage the presence of oscillation. It is noted that both in ZONE I and in ZONE II the harmonic trends present tend to be damped at least partially. At the same time in these two zones examined the state of charge tends to be different from the case with non-adaptive control. This shows how making this part adaptive involves a different demand in terms of energy.
Of particular relevance is the presence of an equal final state of charge of the battery between the two cases examined. This makes consumption comparable by evaluating only the CO_2 value per km produced in this simulation. In relation to emissions, the following Table 4.4 is then shown:

————	SOC_{final}	$kG_{CO_2}/km[g/km]$
Standard Driver	49.8%	260.79
Predictive Driver	49.8%	259.53

Table 4.4: Adaptive and non-adaptive internal loop results collector

The actual improvement in terms of emissions is therefore expressed (4.14):

$$Improvements = \Delta CO_2 = 1.43\frac{g}{km} \quad Improvements\% = 0.54\% \tag{4.14}$$

This result is related to the driving cycle imposed by the lead vehicle and therefore will be different if its speed changes. Concerning this possible diversity of results, the simulation was carried out by imposing a speed profile equal to the WLTP considering this procedure close to the real use (in terms of speed profiles).

Chapter 5

Longitudinal speed profiling

Consumption resulting from inefficient driving style are among the main reason of an increase in consumption, especially when these driving styles are accompanied by an inappropriate speed pattern. By imposing a certain distance to travel x_f, there are many possible speed combinations such that (5.1):

$$x_f = \int_{t_0}^{t_f} v(t) \cdot dt \tag{5.1}$$

From this formulation it is possible to see how, for a given distance, obtaining any speed profile does not guarantee a minimization of consumption.
This statement takes hold considering many non-negligible aspects of daily driving and how a car behaves under certain stresses. The previous chapter set the goal of introducing a system to cushion speed profiles evaluated as incorrect through a high-level logic in order to make these trends less harmful.
However, that type of system is, as explained, only damping as it cannot completely remedy bad driving profiles imposed by the driver.
The problem posed is obtaining an optimization of the speed profile.
Real cases present during normal driving and different situations characterizing the normal behavior of a driver with their car are therefore included. Some of the possible situations that can be encountered normally during common driving are therefore listed below:

- Slowdown and restart and/or stop due to traffic;
- Slowdown and restart and/or stop due to vertical and/or horizontal signs;
- Slowdown in the vicinity of curves.

The previous situations confer different types of approach to the problem that are possible to be integrated into high-level controls capable of imposing the driving profile.
One of the main tasks to have is the definition of the signals with which the speed profiling takes place. It should be noted that the presence of ADAS sensors is currently the main method for obtaining external information. This information is then processed and stored so that it can be used for the production of the speed profile. Once a possible speed profile has been obtained, the problem involves imposing a very specific profile capable of maximizing efficiency by applying the boundary conditions.
A certain speed profile translates into an acceleration actuation to be carried out according to the cinematic relationship (5.2) :

$$a = \frac{dv(t)}{dt} \tag{5.2}$$

The high-level controller must therefore impose an appropriate speed profile by implementing a certain longitudinal acceleration as a command variable. The longitudinal acceleration can be considered proportional in a first approximation to a torque required at the wheels, and consequently to the power unity of the car. The problem is then translated into optimizing the implementation that will have on the power unit. Considering how the total torque is expressed as (5.3):

$$\begin{cases} T_{tot} = T_{req} - T_{fric} \\ T_{req} \quad \alpha \quad a_{req} \end{cases} \tag{5.3}$$

Starting from this relationship and considering the same wheel speed, it is clear how proportional the torque required from the wheels is with the power required. Considering the efficiencies present in the drivelines as constant and unitary as a first approximation, it is possible to obtain a total proportionality between the power required from the wheels and from the power unity in the car.
The optimization of the profile therefore requires the maximization of what the efficiencies are. In this case one of the main points in which the optimization comes into play is that of the electric motor and thermal engine maps in the vehicle.
The problem posed therefore becomes the evaluation of an equivalent fuel of all the work points in question present on the engine. A specific loss function is therefore characterized (5.4):

$$m_{eq} = m_f(gear, T_{req}, w) + m_{EM}(gear, T_{req}, w) \tag{5.4}$$

This formulation would guarantee the overall minimization of the instantaneous working points of the machines. The function is then obtained for the entire profiling horizon. The function to be minimized therefore becomes (5.5):

$$Cost = \sum_{k=1}^{N} (m_{eq}(k) + \omega \cdot T_{fric}(k)) \cdot \Delta t(k) \tag{5.5}$$

This sum is therefore entirely equivalent to the energy that each working point in the overall speed profiling provides to the final obtainable result. A diagram representing the possible algorithm for obtaining this approach is shown below Figure 5.1:

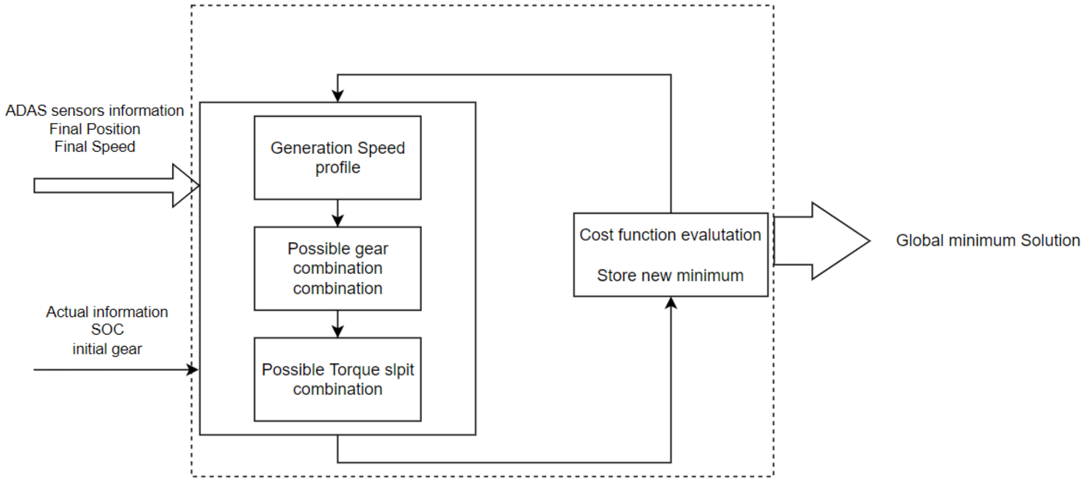

Figure 5.1: Global speed profiling optimization problem

The second approach, for the creation of an optimized driving profile, is that of minimizing the second part of the equation (5.5). This part of the equation is formed by specific parameters of the car that characterize its performance.
For the development of this problem the main point is that of minimizing the energy dissipated by these general frictional forces.
The characterization of the frictions is therefore of fundamental importance for approaching this method, therefore general characterizing parameters of the vehicle are obtained, i.e. the Coast Down parameters, which characterize constant linear and quadratic friction under certain conditions. Obviously, these parameters are an approximation of the real frictions of the vehicle.
Generally, the frictions of the car can be modeled as (5.6):

$$T_{fric} = K \cdot (a_0 + a_1 \cdot v(t) + a_2 \cdot v(t)^2) \tag{5.6}$$

Such frictions are difficult to model as many of them are highly non-linear, therefore the approximation previously shown is evaluated as a compromise.

The Coast Down factors foresee the approximation by means of 3 characterizing parameters:

- constant;
- linear;
- quadratic.

These parameters represent a second degree curve quadratically proportional to the speed and this indicates how much friction force corresponds to a specific speed.
The second method therefore seeks to minimize this energy dissipated by friction.
This approach is an improvement only if the car in question has at least one electric more capable of recovering energy through braking, exploiting their intrinsic qualities of reversibility.
This approach cannot therefore be applied to a vehicle powered by the thermal engine alone.
The problem as mentioned, deals with the minimization of the energy dissipated by the three frictions. By imposing the equation indicating the friction force, the following equation is obtained for the power computation (5.7):

$$P_{fric} = \omega \cdot T_{fric}(k)) \tag{5.7}$$

in order to obtain the energy, is therefore exploited (5.8):

$$E_{fric} = \int_{t_0}^{t_f} F_{fric} \cdot ds \tag{5.8}$$

evaluating the displacement calculated as (5.9) .;

$$ds = v \cdot dt \tag{5.9}$$

The final formulation is then obtained by incorporating the previously cited relations. Particular emphasis is placed on the degree of the characteristic polynomial. The formulation thus obtained of the problem, which as explained wants to find the optimization in a horizon k is the following (5.10):

$$Cost = \sum_{k=1}^{N} \omega \cdot T_{fric}(k)) \cdot \Delta t(k) \tag{5.10}$$

A figure of the algorithm used to solve the problem in question is then shown Figure 5.2.
The difficulty in adapting the logic to the car in which it is implanted represents a major limitation to the type of logic that can be implemented in the car for speed

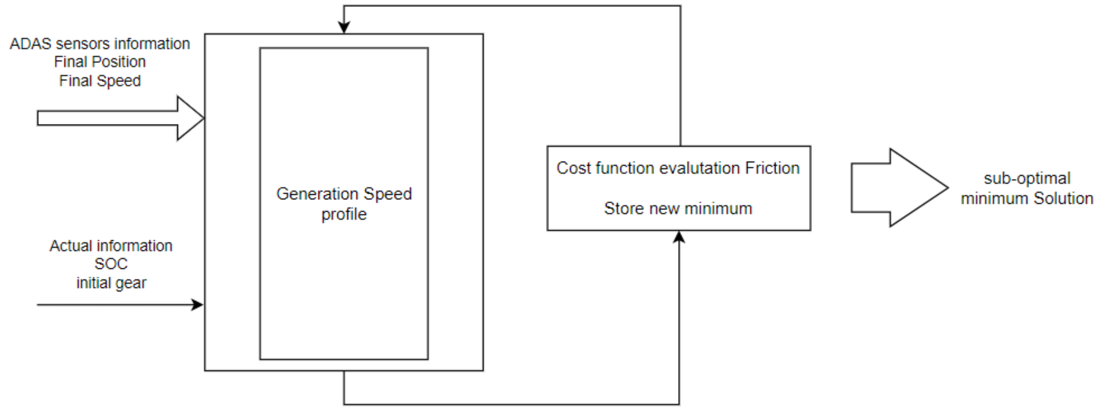

Figure 5.2: Sub-optimal speed profiling optimization problem

profiling. In the text in question we want to evaluate an approach that is not going to be influenced by the hardware present, above all due to the transmission shifting logic, which are usually difficult to obtain.
For this reason, the main point of optimization is to be carried out by minimizing the frictions at a given speed.

5.1 State of art

The implementation of this algorithm collides with many limitations and many different types of approaches. Since it is always difficult to define a single approach for obtaining the overall minimum, it is necessary to consider the formulation obtained to evaluate which type of approach is more correct to compute the desired minimum result. It is clear that the minimum in question can be of two types:

- a solution that guarantees the optimal minimum, ie. The absolute minimum of the function in question in that given interval;
- a solution that guarantees only a minimum sub-optimal, this type of approach gives as a solution a function that is minimized only in part and not completely.

The first of the two approaches is the more complex and requires a meticulous and analytical study of the problem. It is usually solved numerically due to its high computational effort. This type of solutions are usually used as a benchmark for the evaluation of approaches that minimize the function in a sub-optimal way.
The two global minimization approaches considered in many approaches are of two types:

- Dynamic Programming;
- Solver minimization.

The first of the two is a very generic type of algorithm and commonly used in many types of numerical problems. It can be used in both minimization and maximization problems in a fairly equivalent way.
Solver minimization, on the other hand, includes quite different subcategories within it. It provides for the formulation of the problem very explicitly. Based on pre-made algorithms [29], it is able to find a minimum or a maximum of the inserted cost function relative to a determined output vector. The main solver configurations include:

- Linear Programming (LP);
- Quadratic Programming (QP);
- Conic Programming (ConicProg/ConeProg).

Among the application solutions currently in use, attempts are made to optimize the computational effect given by the types of numerical solutions that guarantee the global minimum.

[30] In this test we refer to a method for suboptimal minimization called Pontryagin's Maximum Principle (PMP). This approach seeks to minimize the fuel of the vehicle into consideration. This algorithm in addition to being much less energy-intensive in computational terms also guarantees the possible insertion of various types of limitations due to the imposed path, such as:

- speed limits;
- integration within the argument of the limits due to the gear change;
- integration of optimization taking into account the slope of the road.

[31] This typology also poses the computational problem as the main challenge for obtaining an efficient driving profile, showing how methods capable of providing the absolute best are not capable of being feasible in hardware for real-time applications. It is therefore shown how sub-optimal solutions still manage to guarantee an excellent minimum point in relation to the previous one.
Specifically, the text proceeds with profiling using a complex approach. It provides a Bi-level programming capable of calculating an ofline solution and then an online solution for obtaining the optimal sub. For the development of this algorithm,

PMP conditions are also used here for the reduction of the non-linear states. As a consequence the final complexity of the problem also decreases.
An MPC capable of implementing the following is therefore implemented for the optimal solution:

- Partial update of the convergence of states to lighten computationally;
- Resolve a single QP (SQP) instead of waiting for complete convergence.

5.1.1 Dynamic Programming method

Dynamic Programming is a programming method that is used in the most diverse scientific application fields. Its wide implementation is due to the accuracy. It is able to adapt to the problem to which it has been associated.
In the specific case it allows to calculate the absolute minimum of a given cost function in a given range of application. Specifically, the previously cited formulation is mentioned (5.10).

[32] shows in detail the schematic of the algorithm to be implemented for the study of the velocity minimum cost trajectory problem.

The algorithm is then summarized in the following Figure 5.3: As shown, this type of approach allows the calculation of the 'road' which allows to obtain the absolute minimum. However, this approach involves the calculation of every possible (non-Causal) option, saving each of the possible paths at the same time. This type of approach is then carried out in Back-propagation, that is, starting from the end of the horizon of which the result is to be found, and going back to the initial state. Obviously, for this formulation, limitations imposed by the system taken into consideration must be respected, such as:

- Initial states;
- Final states;
- Maximum to minimum variation of the states.

The states in question represent the variables involved in the system under consideration. The diagram of a problem in DP of possible application is shown below Figure 5.4, showing what type of output the algorithm would give as a solution.

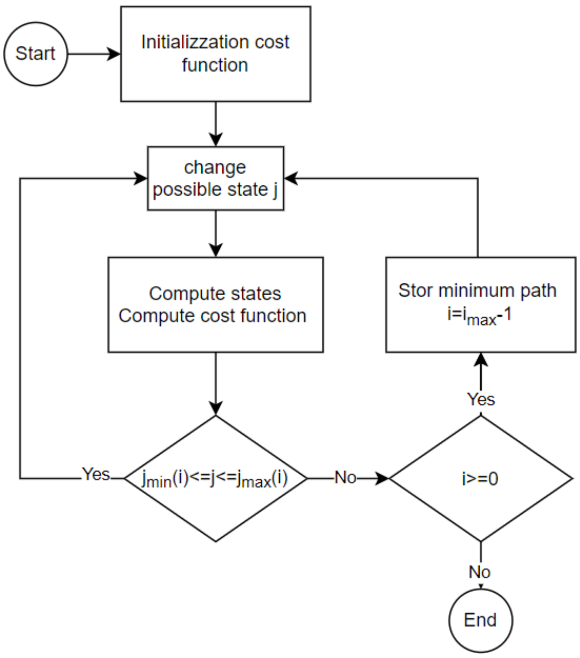

Figure 5.3: Dynamic Programming Algorithm

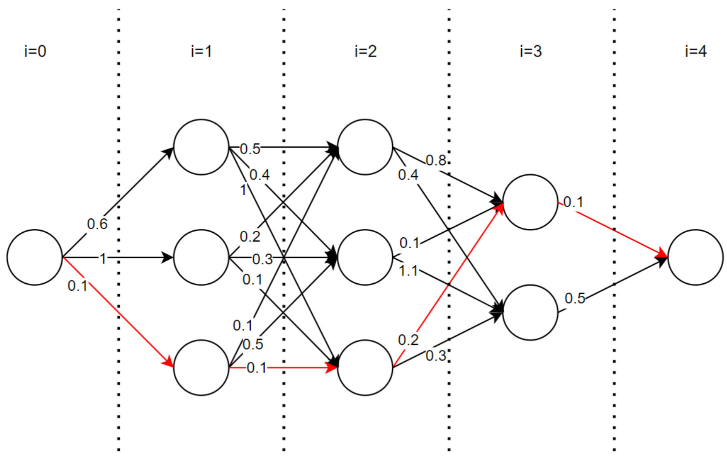

Figure 5.4: Dynamic Programming possible solution for a given minimization task [3]

5.1.2 Optimization problem solution using solver

The second approach in question is that of solving the problem through the approach of an algorithm created specifically for the computation of that particular problem.

In this case we refer to specific solvers. Considering the formulation of the problem (5.11), the solver gives the solution:

$$\min_x (f^T x) \tag{5.11}$$

where x represents the variable to be minimized.In the case study in question it represents the speed vector to be found for the correct pursuit of the speed profiling. In relation to the specific case it is possible to insert particular constraints that limit the behavior. These conditions must be entered correctly within the wording. They are specifically dependent on the type of problem and the type of solver used.

Variable states output settings

To obtain the solution, the setting of the problem involves the imposition of the possible variations that the state variables may have.
Imposing kinematic conditions, starting from an approch given in [29], the definition of what acceleration is makes possible to impose the states of velocity and position as a consequence.
In the specific case study, the formulation taken into consideration provides that the acceleration is defined by a constant value for a certain determined time interval, which in particular will be at least an order of magnitude less than the total time. The following (5.12):

$$a(\Delta t(i)) = a_i = constant \tag{5.12}$$

using the kinematic bond is obtain the velocity (5.13):

$$v(\Delta t(i)) = \int_{t_0}^{t_0+\Delta t} a_i(t) \cdot dt = v(t_0) + a_i \cdot (\Delta t(i)) \tag{5.13}$$

Then imposing the equation (5.6), it is possible to obtain the following relationship with the position (5.14):

$$x(\Delta t(i)) = x(t_0) + v(t_0) \cdot \Delta t(i) + \frac{a_i}{2} \cdot (\Delta t(i))^2 \tag{5.14}$$

The bond in question for the considered Δt is entirely comparable to a uniformly accelerated motion. In the overall it will therefore be a succession of the latter.

5.2 Minimization solution methods

5.2.1 Global problem optimization solution using solver

The formulation of the problem is only the first step for the implementation of a solution obtainable through a solver.
One of the first steps to be carried out is the drafting of this formulation in a manner that is congruent with the solver used [33], remembering how different formulations may not be easily solved with numerical methods even if they represent the same formulation.

Minimization problem definition

The definition of the problem depends on the considered application. Having to find the speed profile obtainable from the car, one of the main points is the definition of the states that the architecture wants to control, limit and implement.
The implementation, as mentioned, will be the speed profile obtained. This provides the state defined as 'speed' having well-defined limitations. These conditions must be integrated with the boundary of the considered problem The problem is then implemented with the following limitations:

- initial state definition;
- final state definition.

Initial conditions

These limitations are intended as the initial points where the car is when the solver is activated. Therefore, they define:

- Initial position, variable not of much importance but fundamental for obtaining the final position seen as relative distance;
- Initial speed, that defines the most important state where the car is. It influences many of the aspects related to the profiling obtained.

Finial conditions

The final conditions represent the states that the car must categorically have at the end of the speed profile implementation. They are of particular interest especially for safety reasons as these points are often imposed by:

- Stop or in any situations in which the vehicle must necessarily stop;
- Minimum distance to be maintained from the car followed, even in the case of final speed other than zero.

The final state are defined as:

- Final position, which can be interpreted as a relative distance with respect to the initial positioning of the vehicle. It defines one of the degrees of freedom of the system;
- Final speed, that represents at what velocity the vehicle must be at the end of the horizon for which the problem is defined.

Starting from the kinematic analysis previously exposed during the formulation of the (5.1) problem, it is possible to deduce how the variable that defines the distance actually implemented is the acceleration.
Graphically, the initial states are interpreted as starting and arrival point that absolutely cannot be transgressed. The Figure 5.5 gives a possible graphical interpretation of the trajectory problem of the speed in question.

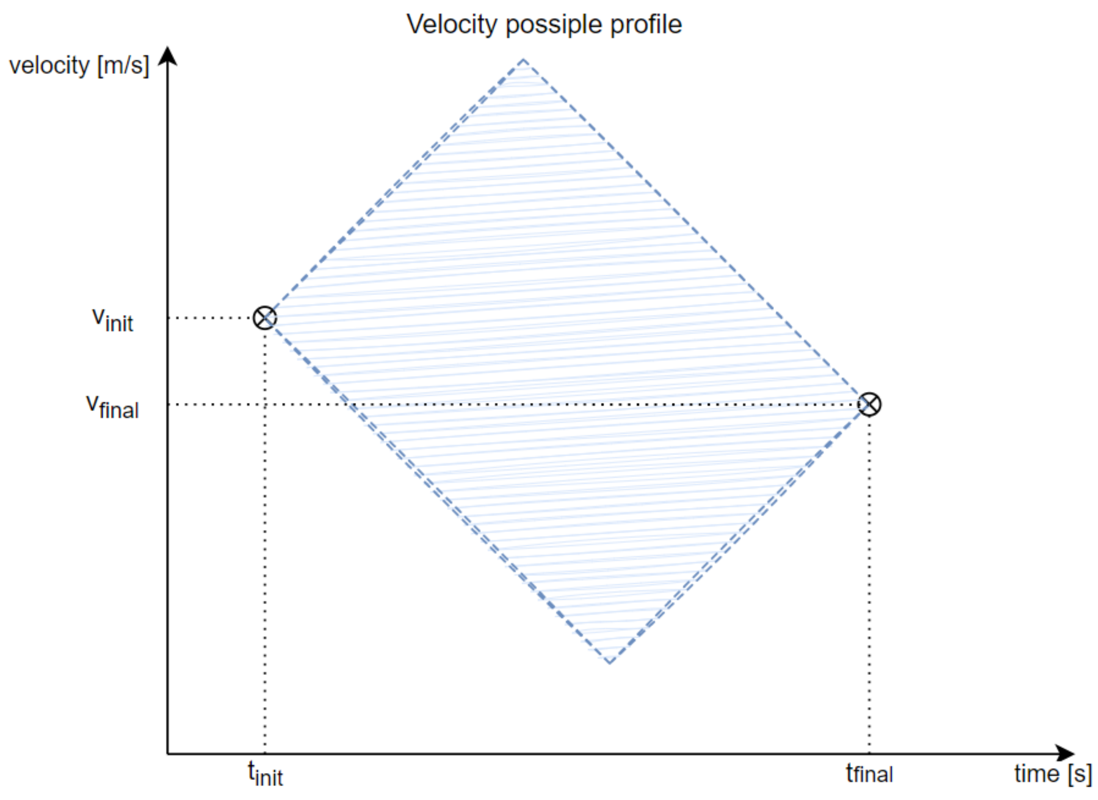

Figure 5.5: Cloud of all possible speed combination in relation to the speed, time and acceleration constraints

The graph represents the cloud of possible speeds that can be implemented from the initial time to the final problem optimization time. However, the final position must coincide with the imposed by the boundary conditions. This leads to obtain a possible cloud characterizing the possible combinations of the position patterns Figure 5.6.

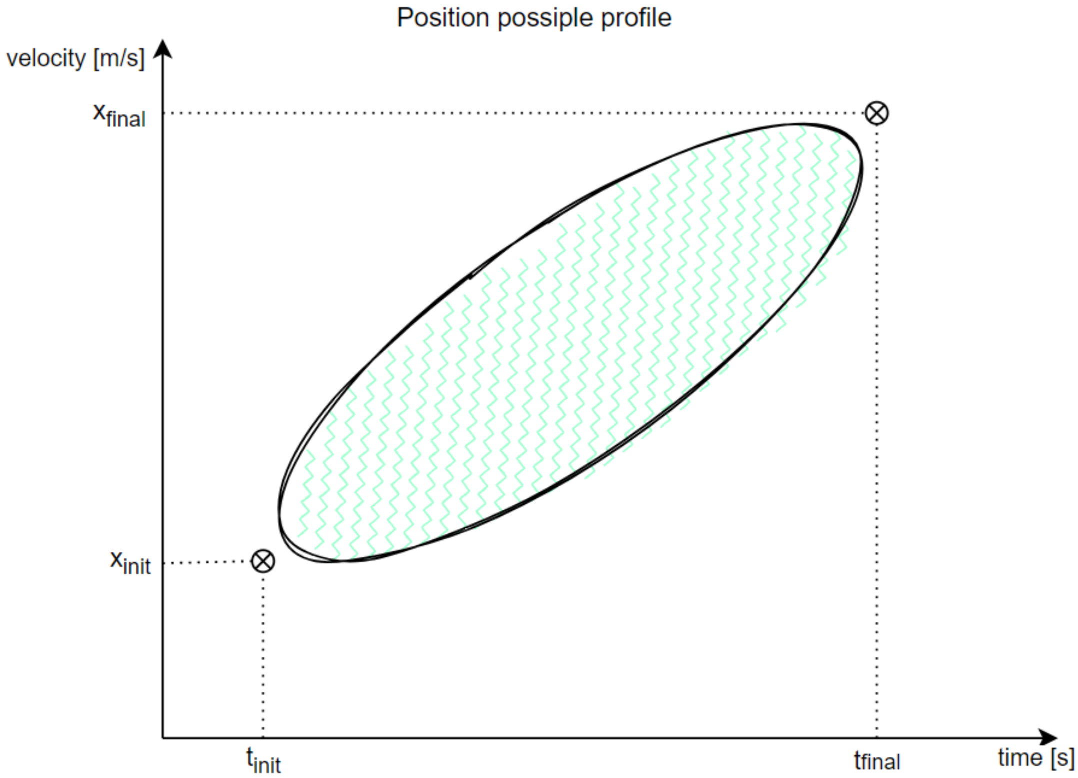

Figure 5.6: Cloud of all possible position combination in relation to the speed, time, acceleration and position constraints

Speed, position and time constraints

By imposing initial and final conditions, there are 2 degrees of freedom to which one can particularly refer:

- maximum and minimum accelerations;
- t_{final} profiling execution time.

Acceleration

The first of these concerns the actuation that allows the speed variation. In this context, the definition of a maximum and minimum acceleration is of obvious importance as it defines the maximum possible variation of speed in a given Δt. These boundary conditions are to be defined before the problem and are limited by the torque performance of the power unity collocated in the car.
Graphically, the maximum and minimum acceleration are seen respectively as the maximum slope and the minimum slope of the speed.

Final time

The second degree of system freedom comes to be the time in which the speed profiling must conclude. This is because generally by defining initial speed, final speed and final position the time during which this operation is carried out can have more solutions.
The timing of execution greatly influences the type of profiling produced by imposing this limit. The implementation of the profile under consideration is more or less aggressive. This variable in the problem formulation is inserted as input.
This assumption leads to obtain a value for the cost function which depends on what time it is entered. It is then also taken into account to find the final run time for which the cost function is minimal.

State variable settings

The problem is then imposed in such a way to find the acceleration for each time interval to be maintained in order to obtain the required speed profile, keeping the limitations imposed by the boundary conditions as mandatory. An initial speed value is imposed at time t_0 in order to compute a closed solution to the problem.
One of the main advantages of this formulation is that the speed state variable is continuous in each of its definition points, thus guaranteeing speed oscillations with less harmonics, effectively avoiding any possibility of discontinuous variation. This greatly improves what the response of the overall system.
By imposing a constant acceleration at any Δt, the main problem is that of the possible variation of this step state which, although to a lesser extent than discontinuity in the speed test, leads to the presence of harmonics during actuation. This type of problem is solved by inserting a first order low pass filter with a dynamics suitably chosen to cut such behaviors.
The formulation adopted therefore allows the following type of curves Figure 5.7 for speed profiling. As explained, it should be noted that the velocity profile is continuous in its domain and how the same cannot be stated for the acceleration profile.

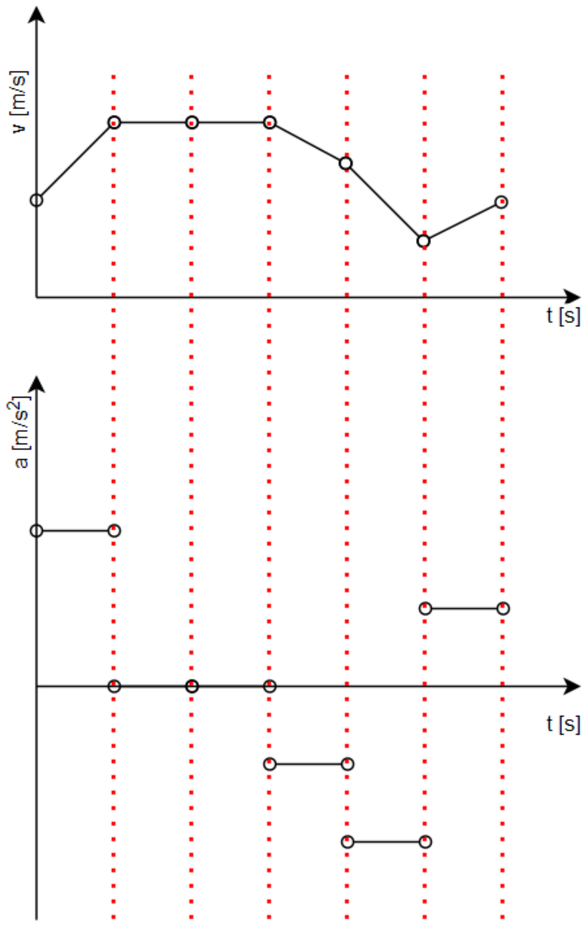

Figure 5.7: Acceleration and consequent speed possible patterns in a solution computed using a solver minimization problem

However, it should be noted that by decreasing the time intervals, a lower stepped acceleration pattern can be obtained, which therefore becomes more reliable. This subdivision of the time, however, makes the solution of the case under examination even more burdensome at the computational level.

Cost function definition for the speed profiling minimization problem

As previously explained in the text, the main point for obtaining the speed profiling is considered to be the minimization of the generic frictions of the vehicle. Since they are dependent on the speed of the car, they are therefore strongly influenced by profiling. This kind of approach has been developed for the definition of a braking profile, in which the minimization of friction is able to increase the recovery of electrical energy obtainable from the electric car during the braking phase.

Therefore, taking advantage of the coast down factors, approximating the generic frictions of the car as the speed varies (5.6).
This cost function, however, is modified to improve the possible profiles obtained, these adjustments require the insertion of a weight to advise against speed profiles that keep the speed constant.
A term dependent on acceleration is then inserted into the cost function. This term represents the car's inertia force (5.15):

$$F_{inertial} = M \cdot a \tag{5.15}$$

The resulting cost function then becomes the following (5.16):

$$COST_{traction} = \sum_{t_{initial}}^{t_{final}} ((a_0 + a_1 \cdot v(t) + a_2 \cdot v(t)^2 - a(t) \cdot M) \tag{5.16}$$

Specifically, this formulation minimizes the traction energy that the given speed profile has in the interval considered. This therefore strongly disadvantages high accelerations. Taking into consideration a case in which the velocity delta between initial and final (5.17):

$$\Delta v = v_{final} - v_{initial} < 0 \tag{5.17}$$

This formulation disfavors constant accelerations or high decelerations followed by short final accelerations. This represents a better formulation as it is evident that for the maintenance of a constant speed the actual wheels torque demand required is evidently different from 0 (5.18):

$$v(t) > 0 \quad v(t) = constant \tag{5.18}$$

The speed profiles obtained will therefore tend to have, in the landslide scenario, an acceleration which for each time interval considered is lower than the acceleration value obtained from the coast down term. This area of implementation can therefore be considered as the following Figure 5.8.

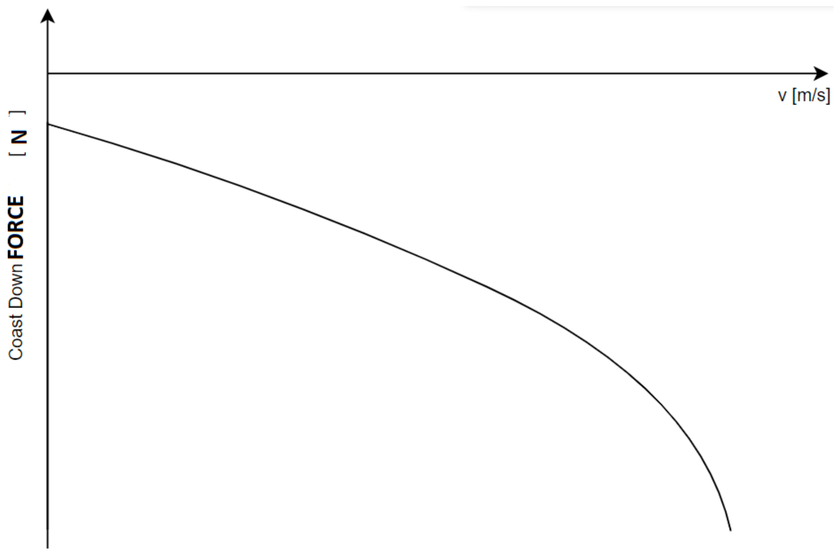

Figure 5.8: Coast down force trend as function of the vehicle speed

Physically, this type of trend is indicative of the fact that the Coast Down curve obtained represents the deceleration force that the car has when no brake command is implemented, therefore making a deceleration less than the same as that of coast down allows not to implement any positive throttle command type, thus requiring only a brake actuation.
This implementation in the vehicle provides the recovery of energy, as mentioned, through the use of the reversibility of the electric machine.

Solver's computational time for processing the global minimum solution

To obtain the optimal speed profile, the solution by means of a solver is better than a possible approximation. This certainly allows to reach an optimal than other types of implementation. However, the computational level of the solver used is taken into consideration. As the computational demand increases the level of time taken by any hardware to complete the solution increases.
For the evaluation of this aspect through the use of time-counting commands present in the Matlab environment, an initial evaluation of the timing of the code used for the resolution by solver was carried out. Among the main factors that influence the increase of the computational level there are:

- The number of time sub-intervals present within the horizon in which the profiling wants to be carried out;
- Number of iterations that the solver in question performs to obtain the result.

The more iterations are present, the closer the numerical result obtained will be to the true minimum (calculable in the case of infinite iterations);

- Type of solver used (LP, QP, ConeProg).

The number of sub-intervals chosen N is used to make the profiling solution obtained approximately defined. This improves more important aspects of the architecture. A more defined and precise pattern of speed and acceleration to follow an acceleration pattern that has fewer jump-type discontinuities. This guarantees a lower presence of unwanted harmonics within the control architecture.
The number of iterations instead is proportional to the time that the algorithm inside the solver performs the calculation to obtain the desired pattern vector. It iteratively reaches a new result and stops when the difference between one iteration and the next has a difference below a certain threshold. Lowering the threshold and increasing the number of iterations improves the final result.
The type of solver used instead represents the lowest level algorithm implemented by the function for obtaining the requested result. It therefore affects the method by which the solution is calculated during an iteration.

Solver global minimization method integration in the overall control logic

Taking into consideration the execution time of the solution explained above, it is necessary to distinguish what type of application can be implemented. This method, having very long computation times, is not possible to be applied it in a Real time control. This solution can therefore be used in 2 distinct ways:

- as a controller capable of providing offline solution;
- as a benchmark for other types of solutions that only give a sub optimal result in order to quantize the difference between the two solutions.

The previous code method is executed and a speed profile is generated. Only afterwards the optimized profile will be imposed by the car model and the results will be evaluated.
Considering the type of method, one of its strengths is the suitable precision with which an absolute minimum is reached in the interval under consideration of the cost function considered.
This result, however, is in contrast with what is the real implementation of the vehicle. It requires that the time when the profile is generated is lower than the real time with which the microcontroller present in the car control unit implements,

otherwise problems would be obtained of overrun. These effects are very dangerous as the control unit is unable to carry out the required task in the required time.

5.2.2 Sub-optimal minimum optimization method

To obtain the absolute minimum, a numerical solution with high computational effort was created, but the process of solving that minimum problem weighs in the overall calculation of the task present in a possible microcontroller.
The problem of recreating a logic that is able to provide a driving speed profile in a defined scenario is therefore highlighted. At the same time an algorithm manages to obtain this profiling in a calculation time short enough to be carried out through a temporal task of the hardware that may be present.
For the application of this method, an algorithm is then evaluated. That minimizes the same cost function previously defined in (5.16).
This approach allows to obtain a sub-optimal speed profile which, however, unlike the one previously obtained, can be applied completely without any delay thanks to its shorter computational time.
The approach used to realize this problem is to implement only a subset of all possible speed combinations, evaluating for each of the profiles obtained the cost that is obtained with the specific cost function. Consider the following points as a fair compromise:

- Similarity with the signal obtained by implementing the problem using the solver;
- Implementation of possible profiles able to obtain a non-discontinuous speed trend;
- Guarantee freedom in obtaining profiling.

These three characteristics were then used for the choice of the curve during this type of optimization.

Sub-optimal problem definition

The generated profile must also comply with the additional conditions imposed to obtain an appropriate signal:

- Initial speed condition;
- Final speed condition;

- Continuity of the generated profile.

To comply with these imposed conditions, a profile characterized by the presence of two constant accelerations in sections, defined on two possible time intervals, is then evaluated.
Considering then this type of acceleration, continuity is imposed on the generated function by matching the speed values at the point where the passage between the two takes place.
The kinematic equations that define the motion are then obtained and implemented.
It is possible to define the complete profiling and implement it. Thus imposing two constant accelerations (5.19):

$$a_1 = constant \quad a_2 = constant \tag{5.19}$$

The motion can be divided into the succession of 2 uniformly accelerated motions. It is then possible to calculate the relationship that binds speed to the two accelerations as the following system (5.20) (5.21):

$$v_1(t) = v_{init} + a_1 \cdot t \tag{5.20}$$

$$v_2(t) = v_p + a_2 \cdot (t - t_p) \tag{5.21}$$

where t_p is the instant of time in which there is the change between the two accelerations.
v_p instead represents the speed obtained at that point. This is of relevance since it imposes continuity to the profiling speed. This represents an important factor since a variation with infinite speed derivative cannot be possible.
Applying kinematic reports the evaluation of the position will be (5.22) (5.23):

$$x_1(t) = x_{init} + v_{init} \cdot t + \frac{a_1}{2} \cdot t^2 \tag{5.22}$$

$$x_2(t) = x_p + v_p \cdot (t - t_p) + \frac{a_2}{2} \cdot (t - t_p)^2 \tag{5.23}$$

x_{init} is the initial position. As the problem is set up, this magnitude does not affect the solution obtained so it can be neglected and placed equal to 0.
x_p is the space traveled by the vehicle in the instant of time t_p. It is calculated as (5.24):

$$x_1(t_p) = x_{init} + v_{init} \cdot t_p + \frac{a_1}{2} \cdot t_p^2 = x_p \tag{5.24}$$

Another important imposition as a boundary condition is the insertion of a value defined as the final velocity (5.25). It coincides with the value imposed in input to the algorithm. That speed is then set using (5.21):

$$v_2(t_f) = v_p + a_2 \cdot (t_f - t_p) = v_f \tag{5.25}$$

t_f is the time when the generated profiling ends.
This consideration made on the speed must also be carried out on the final position to which the car will arrive when the profiling is completed. This must comply with the conditions at the boundary imposed, using (5.23). It is obtained that the final position therefore to be (5.26):

$$x_2(t_f) = x_p + v_p \cdot (t_f - t_p) + \frac{a_2}{2} \cdot (t_f - t_p)^2 = x_f \tag{5.26}$$

by having defined these characteristic equations, evaluating the type of input that will be used of the profiling logic in charge is the next step for the definition of the problem. This input is then considered as:

- Final position x_f;
- Initial speed v_{init};
- Final speed v_f.

By setting in this way the kinematic equations previously shown, it is evident that there are more unknowns variables than the equations. For the resolution it is therefore mandatory to impose one of the two cases:

- Accelerations of the two sections a_1 and a_2;
- Timing at which the profiling is carried out t_p and/or t_f.

The choice taken into consideration is to impose a possible range of constant accelerations and consequently calculating the complete speed profile.
The problem is reduced and the next step is finding an equation of the second order in t_p or t_f. Using appropriate solvers of systems of equations, the results are then cyclically obtained associate to the chosen accelerations. Among these results, there are present impossible kinds of results:

- Negative time values;
- Speed profiles that have values lower than 0, since they must be strictly positive speed profiles;
- $tp > tf$.

By imposing these conditions, it is therefore possible to obtain a complete and feasible speed profile in the case under consideration.
For this type of problem taken into consideration certainly, the solution obtained will be only sub-optimal compared to that obtained through types of approaches with higher computational effort. It is possible then to modify the cost function so as to be able to calibrate the weights that each individual contribution gives for obtaining a result that tries to approximate the solution previously obtained by the solver.
The modification of the cost function therefore leads to the choice of some weights. They were chosen by testing the types of solution obtained and comparing the results by evaluating which of the possible choices was more convenient.
The function previously used for minimization with solver (5.16) is then modified as follows (5.27):

$$COST_{traction} = \sum_{t_{initial}}^{t_{final}} (\alpha \cdot (a_0 + a_1 \cdot v(t) + a_2 \cdot v(t)^2 - \beta \cdot a(t) \cdot M) \tag{5.27}$$

- α represents the weight relative to the inertial part present in the equation;

- β represents the weight relative to the coast down part.

Their variation involves the tendency to weigh approximately high accelerations of the Coast Down value.
The combination of these two is relative to their variation the minimum profiling obtained will be different. An obvious difference in this formulation of the problem is that the final time of the profile is not determined and/or chosen a priori, but only a consequence of the chosen accelerations. This implies that a determined choice of these variables leads to obtain different times. For this reason, a condition has been implemented in the algorithm for which the generated profile must have a maximum final time.
The algorithm created specifically defines a range of accelerations to be tested. Each iteration tests a different combination of accelerations consequently calculating the results of the second degree equations that result in the times of the specific speed profile. Knowing these profiles, the cost function is evaluated for all the possible patterns generated. This process is then iterated and is evaluated the lowest cost function. The scheme of the algorithm developed to obtain sub-optimal profiling is shown below Figure 5.9:

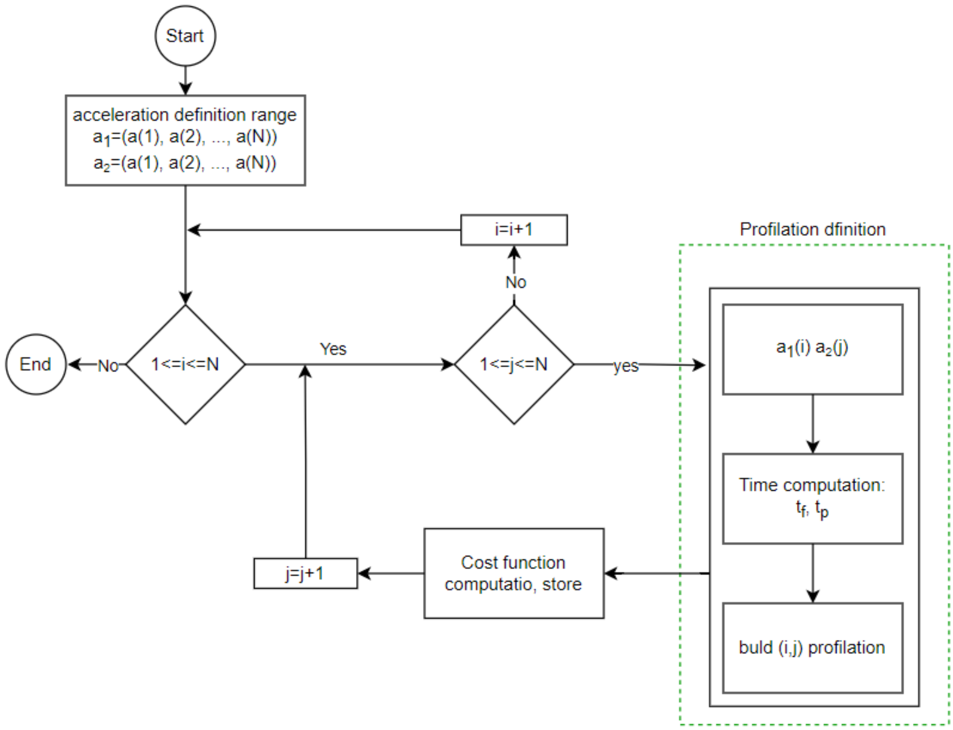

Figure 5.9: Online speed profiling computation method flow chart

This algorithm then provides N*N possible types of profiling types. These patterns represent a subset of all possible combinations. Taking into consideration a single cycle of profiling obtained, assuming a solution among those admissible, the type of profile will be as follows Figure 5.10:

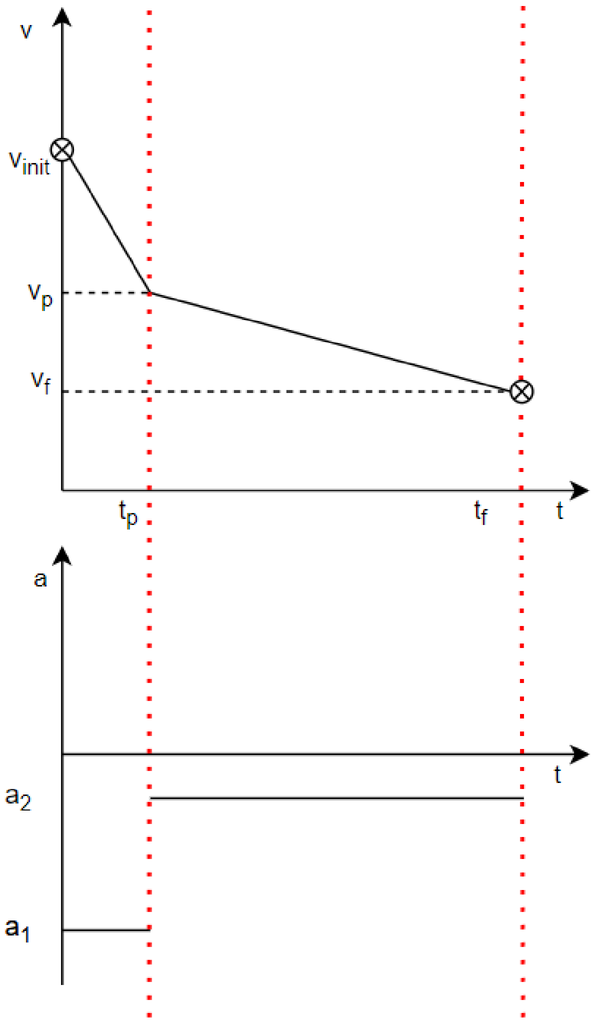

Figure 5.10: Possible speed pattern for the fast profiling method

This figure shows the particular case examined as a scenario, i.e. a slowdown from the initial speed.
The algorithm will then continue to generate possible types of driving profiles until a very dense cloud is obtained in which all the allowed profiles will be present.
A final step is defined by imposing a certain maximum time, after which to neglect the possible results obtained. This condition can be imposed even more stringently to impose a certain profile with a precise final time. The possible cloud of combinations achievable using this algorithm is then shown below Figure 5.11:

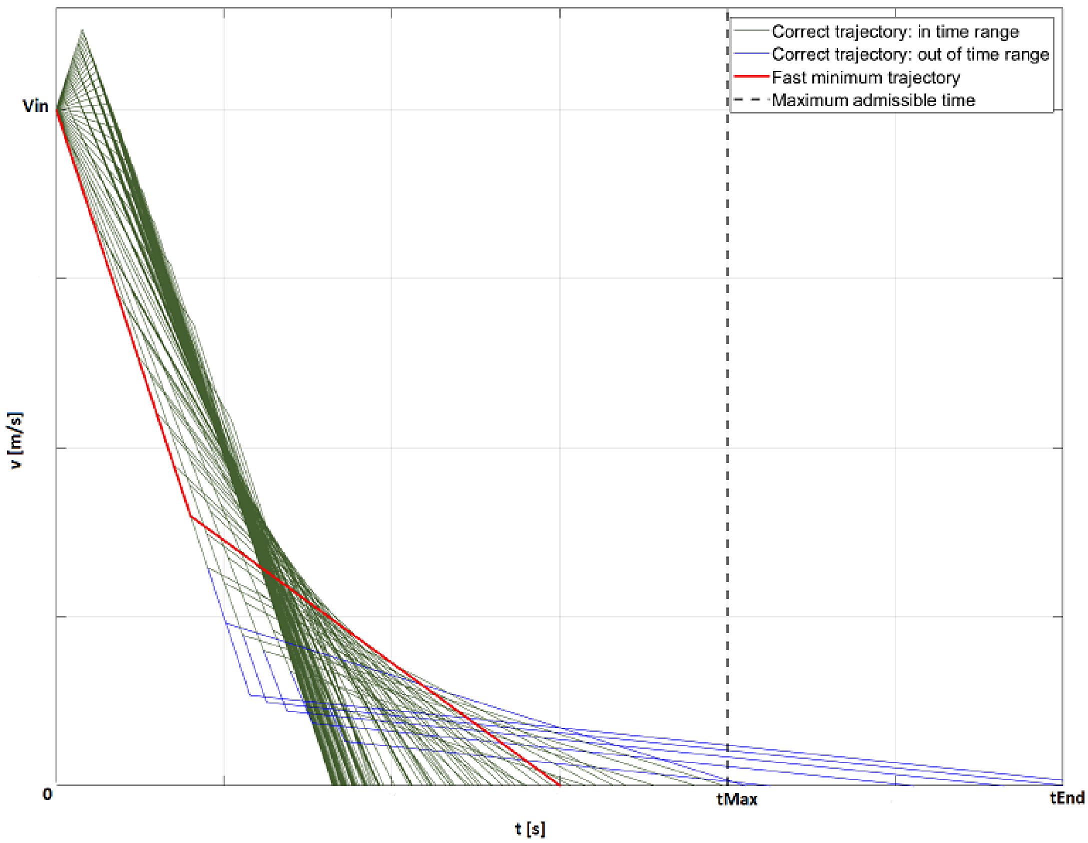

Figure 5.11: Possible speed cloud for the fast profiling method

Computational efficiency

Highlighting again how the solution in charge is only a subset among the possible combinations, the results in computational times are much less than the general solution shown previously through solver. This algorithm therefore allows to calculate the total speed profile in a very short time and feasible in a time task not too high.

Integration of the sub-optimal fast algorithm in the overall control logics

This developed algorithm is integrated into a profiling logic triggered by the recognition of driving situations through ADAS. The speed trajectory is then profiled to obtain a pattern that tries to maximize the energy recovered from the electric machine, this taking into consideration as mentioned a slowdown scenario. The following is a scheme of the overall system integrated Figure 5.12:

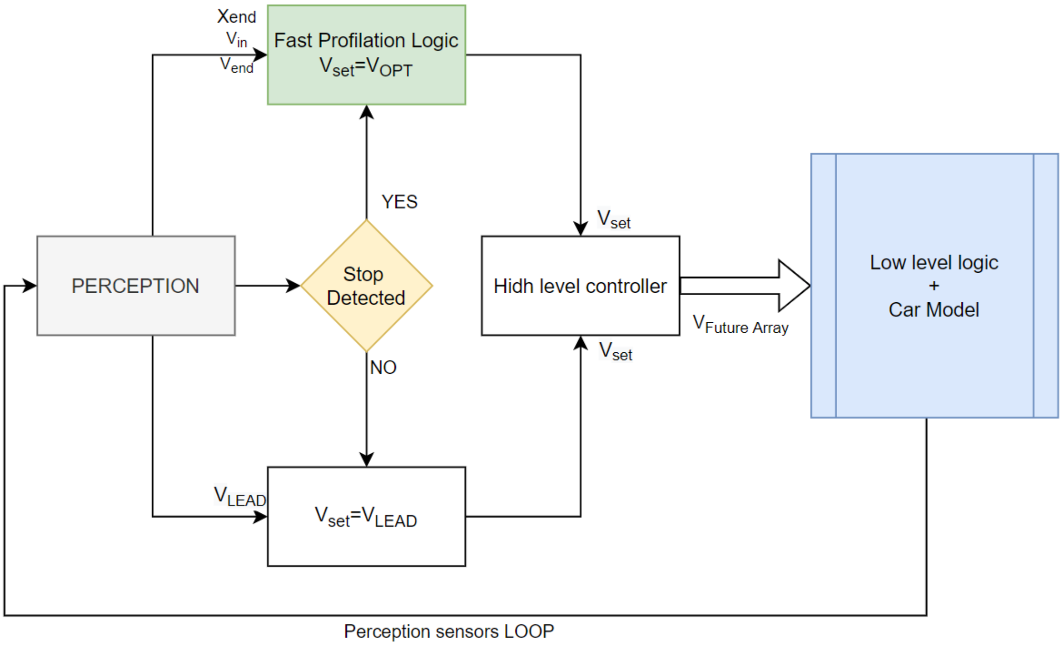

Figure 5.12: Schematic integration speed profiling logic for implementation in a tracking scenario

5.3 Comparison among solver global optimal, sub optimal and standard speed profiles in deceleration scenario

The solutions in charge therefore represent two methods for obtaining the same result applicable for driving profiling. The first, being a computationally very inefficient and difficult to implement solution, is considered as a benchmark for the validation of the second developed. Taking into consideration a case belonging to the scenario previously shown, it is then defined Table 5.1:

	v_{init}	v_{final}	x_{final}	time horizon
Parameters	20	0	100	13 sec

Table 5.1: Initial and final conditions for the speed profiling constraint definition

For the definition of the problem through the second approach, a time range is chosen between (5.28):

$$t - \delta <= t <= t + \delta \quad \delta = 0.1sec \tag{5.28}$$

For the comparison, the speed and acceleration profiles obtained in the two algorithms examined are then shown. The main difference is related to the acceleration that, being definable differently in several intervals, leads to obtain a complex profile but still similar to the one generated by the sub-optimal algorithm. Another point is related to how the second method is a subset of the previous case. The acceleration given by the solver provides the computation in N different intervals, with therefore many types of N^N combinations. In the second method instead the intervals examined are only 2 so the possible combinations are minor. There is only one distinction that, while previously the sub-intervals of time were constant and defined a priori, in the second it is defined dynamically in relation to the acceleration examined at that particular moment. This represents a greater degree of freedom and turns out to be a more elastic algorithm for obtaining a feasible profile. To complete the comparison, a speed trajectory considered 'basic' is chosen by an average driver during a normal driving route for deceleration. This profile is then imposed to compute a result that is considered as a standard in order to understand how optimal the profiling obtained in the two previous cases is in relation. The comparison between the three profiles is then shown Figure 5.13.

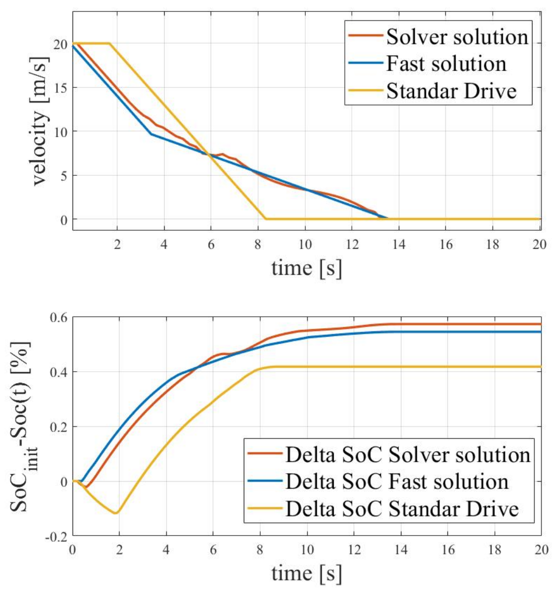

Figure 5.13: Comparison among optimal, sub-optimal minimization and standard driver speed profiles

For the evaluation of improvements the final state of charge of the battery is examined. The results obtained by imposing the three different types of speed are tabulated below Table 5.2:

———	ΔSOC_{final}	Improvements
Solver solution	0.5721	37%
Fast algorithm solution	0.5440	31%
Normal driver trend	0.4169	Baseline

Table 5.2: Velocity profiling results collector and improvements evaluation

Analyzing these results, it is evident how the driving profile obtained by the solver increases the recharge energy more than the others. The influence given by friction during the charging phase is therefore demonstrated.
Taking into account also the actual computational difference between the two solutions obtained, through the solver and the fast algorithm it is highlighted how this solution can guarantee an excellent compromise between performance and possible real implementation in the car.

Chapter 6

Conclusion

In this final chapter are summarized the objectives that are covered within the paper, the methods with the relative results obtained and some types of future improvements that can give an improvement in the results.

The goal of the work carried out is to find control methodologies capable of reducing the actual consumption of the vehicle. The vehicle examined for this purpose is a Full-Hybrid Electric Vehicle with the parallel hybrid architecture and an electric motor in the P2 position. The work was carried out in a Matlab/Simulink simulation environment.

In this context, the first approach was to compare the performance between the previous transmission located in the car and a Dual-Clutch Transmission (DCT) in which different control logics are implemented during gear change. The control provides for two different types of implementation: the first in which a total disengaging of the two clutches is imposed and the second in which a partial overlap is imposed during the gear change. Testing these configurations shows improvements in the DCT during load start-up. In fact, the energies provided by the source to bring the load into rotation in a given time are computed and it is noted that the DCT is able to obtain 26.6% and 27.35% improvement, respectively for logic with total disengaging and logic with partial engagement. This result is represented in terms of energy in relation to the automatic transmission configuration with Torque Converter previously present in the car, in the phases in which the motion is initialized. This disadvantage has been attributed to the presence of slippage given by the presence of the Torque converter in the initial stages of motion. In addition, a logic has been implemented to work in parallel with the high-level controller that provides the gears to be inserted in the Gearbox. This logic must be able to predict the choice in the gear that will be required in the immediate future. This work was carried out to take advantage of the particularity of the

DCT of pre-engage a gear for reducing gear-shift times. This estimator was created by evaluating the characteristic signals that are used by the high-level logic for the choice of gear and reconstructing the future signal through their derivatives. The results find a good response but are influenced by the high dynamics of the vehicle because they increase the amount of noise obtained by reconstructing the signal through the derivatives.

The second approach aims at creating a logic capable of acting in parallel with the human driver. It tries to discern the type of driving pattern of the driver, categorizing it into different types, disadvantaging all those driving profiles that bring high inefficiencies. This logic for brevity is hereinafter called P-DRIVER (Predictive-DRIVER). This logic is created by exploiting the information coming from the ADAS sensors present in the car. By being able to perceive the external environment, it is able to provide methods to predict what will be the driving profile carried out. The logic created then acts as a supervisor on the demand in power of the car. The car controller related to the throttle, characterized by the presence of a PID, is examined. It goes to have as a reference a speed/acceleration signal and as feedback the corresponding signal of the instantaneous value of the car. The supervisor created acts by dynamically calibrating the gains of the PID in order to require less effective torque with the same error between reference and feedback. The creation of the supervisor is made by the exploitation of a Fuzzy logic through: the correct exploitation of input signals (characterized and obtained considering a future speed vector of the vehicle), the choice of appropriate membership functions to avoid high computational efforts and the choice of rules that could discern between the various driving profiles in relation to the inputs used. The supervisor created will implement by varying 3 possible types of PID gain, respectively: LOW to discourage, MEDIUM to maintain a neutral behavior, and HIGH to increase the response in terms of torque of the car. For the evaluation of this logic, a chase scenario is examined in which the lead vehicle proceeds with a certain driving profile. Considering the dependence of logic on the estimated speed, it is appropriate to make the lead vehicle proceed with a known and regulated profile that can be reproducible and contain different sub-patterns within it. The driving profile of the WLTC related to the vehicle (class 3b) is examined. In this context, the results in terms of emissions and battery consumption of the hybrid vehicle under consideration are collected. In terms of initial and final battery state, the differences are marginal and no differences are noted. In terms of CO_2 emissions, there is an effective improvement in g/km of 1.43 g/km corresponding to an effective improvement in emissions compared to the deactivation of this logic, in percentage terms the improvement is 0.54%.

The third approach involves creating a logic capable of providing an optimized speed profile to the car through the exploitation of the signals provided by the ADAS sensors present in the car. This logic tries to implement an optimization algorithm in which the equation related to the traction energy of the car is minimized. In this equation the coast down parameters of the car are of main importance. They represent the frictions of the vehicle as a function of the speed. To obtain these parameters it is important to carry out experimental tests or simulations bringing the vehicle to a certain fixed speed, disengage the clutch and evaluate how the speed decreases over time. This deceleration is therefore characteristic of the car and does not require any action of the mechanical brake or recovery of the electric motor brake. Minimizing the equation also containing these factors allows to obtain a driving profile that makes even speed sections weigh at zero acceleration or in any case negative but greater than the corresponding deceleration of coast down, since they still require a certain throttle angle. The problem explained was carried out by two approaches: the first one in which the equation is implemented and minimized through the use of a solver. This solver allows to obtain the speed profile optimized in relation to the function to be minimized. The result obtained is close to the solution of global minimum. This implementation, however, clashes with the time of obtaining the profile, it is impossible the real-time implementation. The second approach to the problem instead goes to propose an algorithm capable of being implemented in real-time. This logic obtains a profile in such a way as to evaluate only a possible subset of combinations of accelerations favoring a much lower computational weight than in the previous case. For the evaluation of these logics, a characterizing scenario is chosen. In the specific case, the scenario considered is that of braking, in which in case of an electric or hybrid motorization the result is evaluated in terms of recovered electrical energy. The result of the two logics is then compared with a speed profile considered as 'standard', that is a profile that a normal driver would do in the scenario considered. The results show that compared to the standard speed profile, the two logics provide an improvement in terms of battery recovery of 37% for the solution obtained through the solver solution and 31% from the solution through sub-optimal algorithm. In relation to the shorter computational time, which therefore allows an easy implementation, the algorithm that provides the sub-optimal turns out to be a good compromise compared to the solution obtained through solver.

6.1 Future improvements

The future works that can be carried out are related to the three logics shown.
DCT control logics:

- Improvement of the logic of partial engagement during gear change by implementing low-level controllers capable of precisely controlling the relative sliding of the two clutches;
- Implementation of a logic of estimation of the future gear that is less affected by the dynamics of the vehicle.

P-DRIVER Logic:

- Optimization of the controller for different types of internal controller;
- Improvement and insertion of additional rules for the identification of inefficient profiles, in trade off to the possible real-time implementation.

Optimized speed Logic:

- Improvement and integration of the two profiling logics, taking advantage of the solution obtained through solver to be able to calculate offline profiles and then partially correct them with the sub-optimal algorithm, so as to be able to achieve a better energy recovery;
- Implementation of logics that examine the actual working points of the motorization present in the car in order to extrapolate a driving profile that is as good as possible compared to the inefficiencies of the electric car and the thermal engine.

Bibliography

[1] R. Sacchi, C. Bauer, B. Cox, and C. Mutel. «When, where and how can the electrification of passenger cars reduce greenhouse gas emissions?» In: *Renewable and Sustainable Energy Reviews* 162 (2022), p. 112475. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2022.112475. URL: https://www.sciencedirect.com/science/article/pii/S136403212200380X (cit. on p. 1).

[2] J. Gessat. «Electrically Powered Hydraulic Steering Systems for Light Commercial Vehicles». In: *SAE Technical* Paper 2007-01-4197 (2007). ISSN: 0148-7191. DOI: https://doi.org/10.4271/2007-01-4197 (cit. on p. 1).

[3] Lorenzo Serrao Simona Onori and Giorgio Rizzon. «Hybrid Electric Vehicles Energy Management Strategies». In: *Springer* (2015) (cit. on pp. 2, 8, 10, 69).

[4] P. Pisu and G. Rizzoni. «A supervisory control strategy for series hybrid electric vehicles with two energy storage systems». In: *2005 IEEE Vehicle Power and Propulsion Conference*. 2005, 8 pp.-. DOI: 10.1109/VPPC.2005.1554534 (cit. on p. 2).

[5] Guido Ricardo Guercioni, Enrico Galvagno, Antonio Tota, and Alessandro Vigliani. «Adaptive Equivalent Consumption Minimization Strategy With Rule-Based Gear Selection for the Energy Management of Hybrid Electric Vehicles Equipped With Dual Clutch Transmissions». In: *IEEE Access* 8 (2020), pp. 190017–190038. DOI: 10.1109/ACCESS.2020.3032044 (cit. on p. 2).

[6] Chan-Chiao Lin, Huei Peng, J.W. Grizzle, and Jun-Mo Kang. «Power management strategy for a parallel hybrid electric truck». In: *IEEE Transactions on Control Systems Technology* 11.6 (2003), pp. 839–849. DOI: 10.1109/TCST.2003.815606 (cit. on p. 3).

[7] Chan-Chiao Lin, Huei Peng, and J.W. Grizzle. «A stochastic control strategy for hybrid electric vehicles». In: *Proceedings of the 2004 American Control Conference*. Vol. 5. 2004, 4710–4715 vol.5. DOI: 10.23919/ACC.2004.1384056 (cit. on p. 3).

[8] Regione Piemonte. *Piattaforma tecnologica di Filiera Pi.Te.F.* 2020. URL: https://www.regione.piemonte.it/web/temi/fondi-progetti-europei/fondo-europeo-sviluppo-regionale-fesr/ricerca-sviluppo-tecnologico-innovazione/piattaforma-tecnologica-filiera-pitef (cit. on p. 3).

[9] Politecnico di Torino. *Politecnico di Torino*. 2022. URL: https://www.polito.it/ (cit. on p. 3).

[10] Podium Engineering. *Podium Advanced technologies*. 2022. URL: https://www.podium-tech.com/ (cit. on p. 3).

[11] E. Galvagno, M. Velardocchia, and A. Vigliani. «Dynamic and kinematic model of a dual clutch transmission». In: *Mechanism and Machine Theory* 46.6 (2011), pp. 794–805. ISSN: 0094-114X. DOI: https://doi.org/10.1016/j.mechmachtheory.2011.01.013. URL: https://www.sciencedirect.com/science/article/pii/S0094114X11000231 (cit. on p. 17).

[12] Bernd Matthes. «Dual Clutch Transmissions–Lessons Learned and Future Potential». In: *SAE TECHNICAL PAPER SERIES* (2005). ISSN: 0148-7191. DOI: https://doi.org/10.4271/2005-01-1021 (cit. on p. 17).

[13] N. Cavina, D. Olivi, E. Corti, L. Poggio, and F. Marcigliano. «Development of a Dual Clutch Transmission Model for Real-Time Applications». In: *IFAC Proceedings Volumes* 45.30 (2012). 3rd IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling, pp. 440–447. ISSN: 1474-6670. DOI: https://doi.org/10.3182/20121023-3-FR-4025.00006. URL: https://www.sciencedirect.com/science/article/pii/S1474667015351909 (cit. on pp. 17, 22).

[14] ASNU Transmission Products. *Understanding Torque Converters*. URL: https://asnutp.com/understanding-torque-converters/ (cit. on pp. 17, 18).

[15] Inc. The MathWorks. *Simscape*. Natick, Massachusetts, United State, 2020. URL: https://it.mathworks.com/help/physmod/simscape/ (cit. on p. 19).

[16] Manish Kulkarni, Taehyun Shim, and Yi Zhang. «Shift dynamics and control of dual-clutch transmissions. Mechanism and Machine Theory, 42, 168-182». In: *Mechanism and Machine Theory* 42 (Feb. 2007), pp. 168–182. DOI: 10.1016/j.mechmachtheory.2006.03.002 (cit. on p. 22).

[17] Salvatore Solarino. «Development of logic for pollutant emission reduction of a Hybrid Electric Vehicle». In: *Politecnico di Torino* Master Thesis (2022) (cit. on pp. 25, 28).

[18] Vipin Singh and Sharad Sharma. «Dual Clutch Transmission System». In: *International Journal of Innovative Engineering Research* 7 (May 2017) (cit. on p. 27).

[19] Sergio Lancelotti. «Lezioni di Analisi Matematica I». In: *CELID* (2016) (cit. on p. 30).

[20] Scott Schnelle, Junmin Wang, Richard Jagacinski, and Hai-jun Su. «A feedforward and feedback integrated lateral and longitudinal driver model for personalized advanced driver assistance systems». In: *Mechatronics* 50 (2018), pp. 177–188. ISSN: 0957-4158. DOI: https://doi.org/10.1016/j.mechatronics.2018.02.007. URL: https://www.sciencedirect.com/science/article/pii/S0957415818300291 (cit. on p. 42).

[21] Linzhen Nie, Jiayi Guan, Chihua Lu, Hao Zheng, and Zhishuai Yin. «Longitudinal speed control of autonomous vehicle based on a self-adaptive PID of radial basis function neural network». In: *IET Intelligent Transport Systems* 12.6 (2018), pp. 485–494. DOI: https://doi.org/10.1049/iet-its.2016.0293. eprint: https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/iet-its.2016.0293. URL: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-its.2016.0293 (cit. on p. 43).

[22] Roberto A. Varella, Marta V. Faria, Pablo Mendoza-Villafuerte, Patrícia C. Baptista, Luis Sousa, and Gonçalo O. Duarte. «Assessing the influence of boundary conditions, driving behavior and data analysis methods on real driving CO2 and NOx emissions». In: *Science of The Total Environment* 658 (2019), pp. 879–894. ISSN: 0048-9697. DOI: https://doi.org/10.1016/j.scitotenv.2018.12.053. URL: https://www.sciencedirect.com/science/article/pii/S0048969718348939 (cit. on p. 45).

[23] Eva Ericsson. «Independent driving pattern factors and their influence on fuel-use and exhaust emission factors». In: *Transportation Research Part D: Transport and Environment* 6.5 (2001), pp. 325–345. ISSN: 1361-9209. DOI: https://doi.org/10.1016/S1361-9209(01)00003-7. URL: https://www.sciencedirect.com/science/article/pii/S1361920901000037 (cit. on p. 45).

[24] Qiuyi Guo, Zhiguo Zhao, Peihong Shen, Xiaowen Zhan, and Jingwei Li. «Adaptive optimal control based on driving style recognition for plug-in hybrid electric vehicle». In: *Energy* 186 (2019), p. 115824. ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2019.07.154. URL: https://www.sciencedirect.com/science/article/pii/S0360544219314963 (cit. on p. 45).

[25] L.A. Zadeh. «Fuzzy logic». In: *Computer* 21.4 (1988), pp. 83–93. DOI: 10.1109/2.53 (cit. on p. 46).

[26] C.C. Lee. «Fuzzy logic in control systems: fuzzy logic controller. I». In: *IEEE Transactions on Systems, Man, and Cybernetics* 20.2 (1990), pp. 404–418. DOI: 10.1109/21.52551 (cit. on p. 46).

[27] Didier Dubois and Henri Prade. «What are fuzzy rules and how to use them». In: *Fuzzy Sets and Systems* 84.2 (1996). Dedicated to the Memory of Professor Arnold Kaufmann, pp. 169–185. ISSN: 0165-0114. DOI: https://doi.org/10.1016/0165-0114(96)00066-8. URL: https://www.sciencedirect.com/science/article/pii/0165011496000668 (cit. on p. 49).

[28] Werner Van Leekwijck and Etienne E. Kerre. «Defuzzification: criteria and classification». In: *Fuzzy Sets Syst.* 108 (1999), pp. 159–178 (cit. on pp. 50, 52).

[29] Inc. The MathWorks. *Discretized Optimal Trajectory, Problem-Based*. Natick, Massachusetts, United State, 2022. URL: https://it.mathworks.com/help/optim/ug/discretized-optimal-trajectory-cone-programming.html (cit. on pp. 67, 70).

[30] Dimitar Filev Engin Ozatay Umit Ozguner. «Velocity profile optimization of on road vehicles: Pontryagin's Maximum Principle based approach, Control Engineering Practice». In: *Control Engineering Practice* 61 (2017), pp. 244–254 (cit. on p. 67).

[31] Nalin Kumar Sharma Ahad Hamednia et al. «Computationally Efficient Algorithm for Eco-Driving Over Long Look-Ahead Horizons». In: *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 23 (2022) (cit. on p. 67).

[32] Eric Bideaux Felicitas Mensing Rochdi Trigui1. «Vehicle Trajectory Optimization for Application in Eco-Driving». In: *IEEE* (2011) (cit. on p. 68).

[33] Inc. The MathWorks. *'SOLVE'. Equations and systems solver*. Natick, Massachusetts, United State, 2020. URL: https://it.mathworks.com/help/symbolic/solve.html?overload=solve+false (cit. on p. 71).