# Assignment 2 – Vehicle dynamics

Matteo Gravagnone s319634, Danilo Guglielmi s318083

### Introduction

In this report, we are going to analyse the implementation and simulation of a rear-wheel-drive electric passenger car in MATLAB and Simulink. The target of the work is to develop a model of a vehicle motion and carry out tests of different cases, related to the main elements that are involved in the longitudinal dynamics (acceleration, friction braking, energy consumption, regenerative braking)

As a case study vehicle was given in the assignment, a set of parameters was already provided Fig/table and, whenever required, reasonable assumptions were made in order to obtain a complete and consistent description of the vehicle.

### Description of the model

The overall model consists of multiple subsystems, each one of them modelling different aspects of the vehicle.

#### Wheels

4 independent wheels, which share a common referenced model *wheel\_model.slx*, take as inputs the vertical load, friction coefficient, velocity of the vehicle and applied motor and braking (due to dissipative brakes) torques.

According to the Pacejka 96 tyre model, forces are computed and, in our case study, only the longitudinal component is used in next steps.

A second subsystem, regarding wheel dynamics, applies the moment balance equation, which considers torque given from the motor, friction brakes, rolling resistance and longitudinal force, in order to compute the angular acceleration and speed.

Tyre longitudinal slip is computed as a function of the vehicle speed and the obtained angular speed, it can fed either directly to the Pacejka function or to a subsystem which models tyre relaxation and computes a delayed slip ratio based on a mass-damper first order dynamics similarity, based on values set in the script.

ABS??

#### Longitudinal dynamics

Acceleration and speed at the center of gravity are computed based on the force balance equation, which considers the total longitudinal force, aerodynamic drag, gravity (due to possible inclination, but always set to 0 in our examination) and rolling forces.

As some of the formulas hold only for non-negative speeds, a saturation block is used.

#### Power losses

In the model some power losses are firstly computed and then integrated to obtain the energy losses of rolling resistance, aerodynamic drag, powertrain, transmission and longitudinal slip.

#### Vertical load distribution

The computation of vertical force *Fz* for each wheel takes into account also load transfers due to aerodynamic drag and non-zero acceleration while it neglects the aerodynamic downforce and the rolling resistance parameter Dx which shifts the application point of Fzr and Fzf as it was considered negligible in prior simulations.

#### Electric machine

In the first part of the subsystem, we computed the motor speed as a function of the vehicle speed, gear ratio and wheel radius. Then, a MATLAB function computes the maximum available torque that can be produced at a given motor speed. This default behaviour can be modified by test cases in which we’re interested in specific torque profiles, while preserving the torque/speed characteristic of the electric motor.

Furthermore, a transfer function was added in order to model a realistic motor with a given torque generation time constant.

#### Battery

To manage the current available energy, we decided to work in terms of Wh. The available energy from the battery is computed starting from an initial State of Charge (SoC) and the nominal capacity. It is then modified during the simulations based on the energy requested or given from the Electric Motor, after taking into account the efficiency of motor and inverter. It has to be noticed that the implemented mechanism can also manage recovered energy obtained during regenerative braking in a consistent way. The SoC is updated at each time step.

#### Braking System

Our friction braking system consists of 4 individually controlled brakes with a fixed front-to-rear brake torque distribution.

To compute the total braking force, we first compute the force for the front wheels by taking into account the vertical force of the front Fzf, a specific friction coefficient μ and the current position of the brake pedal, that we normalized in [0;1] interval. The rear braking force is then derived from the front one and the fixed distribution. The output of the subsystem is a braking torque Tb (defined as positive) which goes to the corresponding wheel model.

A first-order transfer function, also in this case, was added in order to simulate a realistic behaviour of the braking system.

### Tests