

Longitudinal Active Suspension Control in a Half-Car Model with Unsprang Masses

Automatic Control
Electronic Engineering for Intelligent Vehicles
University of Bologna

A.A. 2025-2026

Alessandro Briccoli, Cristian Cecchini, Mario Di Marino

February 18, 2026

Abstract

This report presents the modeling, design, and simulation of an active suspension control system aimed at improving ride comfort and handling performance in passenger vehicles. The study focuses on a half-car model that includes both front and rear suspension dynamics, allowing the analysis of vertical and pitch motion of the vehicle body. Unlike passive suspension systems, which rely solely on spring-damper elements, the active suspension system introduced here incorporates actuators capable of generating controlled forces to counteract road disturbances in real time.

To achieve the desired ride quality, a control strategy based on Proportional-Integral-Derivative (PID) controllers is developed. Additionally, a state estimation technique using a Kalman observer is incorporated to provide real-time estimates of the states, helping to adapt to acceleration and road profile in a faster and more effective way.

The performance of the control system is evaluated through simulations conducted in MATLAB/Simulink. Results show a significant reduction in body acceleration and pitch angle variation when compared to a passive suspension system, demonstrating the effectiveness of the proposed approach in enhancing ride comfort. This project perfectly demonstrates, once again, the massive importance of active control strategies in automotive suspension design.

Contents

1	Introduction	4
1.1	Motivations	4
1.2	List of Symbols	6
1.3	Dynamic Model	7
1.3.1	State vector	7
1.3.2	Control inputs	8
1.3.3	Suspension kinematics	8
1.3.4	Suspension and tire forces	8
1.3.5	Sprung mass dynamics	9
1.3.6	Unsprung mass dynamics	9
1.3.7	State-space representation	9
1.4	Sensor Model	10
1.4.1	Measurement Vector	10
1.4.2	Accelerometer Model	11
1.4.3	Gyroscope Model	11
1.4.4	Suspension Deflection Sensors	11
1.4.5	Noise Modeling	13
1.5	Performance Variables and Control Objectives	13
1.5.1	Control Objectives	13
1.5.2	Output Error Function	14
1.5.3	Observability Considerations	14
1.5.4	Augmented Performance Output for Control Design	14
1.5.5	Definition of Suspension Deflections	15
1.5.6	Physical Meaning of the Augmented Components	15
1.5.7	Relation with the Output Error Function	15
1.5.8	Control-Oriented Interpretation	16
1.5.9	System Linearization	17
1.6	System Linearization	17
1.6.1	Linearization Around the Operating Point	17
1.6.2	Matrix calculus	18
1.6.3	Linear Model Analysis	24
1.7	Control	26
1.7.1	Reachability	26
1.7.2	Integral Action	27

1.7.3	Observability	28
1.7.4	Kalman Filter Design	29
1.7.5	LQR Cost Function and Weight Selection	29
2	MATLAB/Simulink Implementation	31
2.1	Overall Simulation Architecture	31
2.1.1	Symbolic Linearization and Control Synthesis Script . . .	31
2.2	Nonlinear Half-Car Model	43
2.2.1	State Dynamics Block $f(p, x, u, w)$	43
2.2.2	Nonlinear Kinematics	43
2.2.3	Suspension and Tire Forces	44
2.2.4	Actuator Force Distribution	44
2.2.5	Longitudinal Coupling and Load Transfer	45
2.2.6	State Equations	45
2.2.7	Measurement Block $h(p, x, w, n)$	45
2.2.8	Inertial Accelerations	45
2.2.9	Projection to Body Frame	46
2.2.10	Sensor Output Vector	46
2.2.11	Performance Block $g(p, x, r)$	46
2.2.12	Road Profile Generator	47
2.2.13	Longitudinal Wheel Forces Block	48
2.2.14	Linearized Model	49
2.2.15	State Observer	50
2.2.16	Augmented LQR Controller	50
2.2.17	Simulation Management and Visualization	51
2.3	Vehicle Parameters and Numerical Setup	51
2.3.1	Mass and Geometry	52
2.3.2	Suspension and Tire Characteristics	52
2.3.3	Anti-Pitch Geometry	52
3	Control Design	54
3.1	Proposed Control Architecture	54
3.1.1	Implicit Disturbance Rejection vs. Explicit Feed-Forward .	55
3.2	Augmented LQR-PID Control Architecture	55
3.2.1	Augmented System and LQR Synthesis	55
3.2.2	Weight Selection	56
3.3	State Estimation: Optimal Kalman Filter	56
3.3.1	Stochastic System Model	56
3.3.2	Observer Dynamics	56
3.3.3	State Reconstruction and Implicit Disturbance Reaction .	56
3.3.4	Observer Stability	57
3.4	Modeling of Anti-Dive and Anti-Squat Geometry	57
3.4.1	Geometric Force Definition	57
3.4.2	Modified Heave Dynamics	57
3.4.3	Modified Pitch Dynamics	57
3.5	Numerical Control Design Analysis	57

3.5.1	Augmented System Dynamics	57
3.5.2	Optimal LQR Gains	57
3.5.3	Kalman Gain Matrix	58
4	Simulation Experiments and Results	59
4.1	Open-Loop Linear vs Nonlinear Response	59
4.2	Effect of Anti-Dive and Anti-Squat Geometry	60
4.3	Road-Induced Oscillations without Anti-Dive / Anti-Squat	61
4.4	Integral Action and Trade-Offs	62
4.5	Observer Performance and Control Effort Analysis	63
4.5.1	State Estimation Accuracy	63
4.5.2	Control Inputs with Observer Only	64
4.5.3	Control Inputs with Observer and Integral Action	65
4.6	Observer Feasibility and Practical Considerations	65
4.7	Severe Road Bump Scenario	66
4.7.1	Case 5: Linear vs Nonlinear Model under Bump Excitation	66
4.7.2	Case 6: Controlled Linear Model vs Uncontrolled Nonlinear Plant	67
4.7.3	Sensor Interpretation and Measurement Limitations	68
4.8	Trade-Offs	69
4.9	Additional Experimental Scenarios	69
4.9.1	Case 7: Observer + Integral Control with Anti-Dive and Anti-Squat	69
4.9.2	Case 8: Four Acceleration Profiles with Different Force Distributions	71
4.9.3	Overall Insight from Additional Experiments	72
5	Conclusions and Future Developments	73

Chapter 1

Introduction

1.1 Motivations

Suspension systems play a crucial role in ensuring vehicle stability, comfort, and safety. Traditional passive suspensions cannot adapt to changing road conditions, leading to undesired oscillations and reduced performance.

In this project, focus on active suspension control in the longitudinal direction (front and rear), which aims to reduce vertical oscillations and pitch movements when the vehicle passes over bumps or uneven surfaces. The goal is to design a control system that keeps the car body as steady as possible, improving both passenger comfort and vehicle handling.

The half-car model was chosen because it provides a good trade-off between model simplicity and dynamic realism and allows to study the impact of front and rear suspension forces on the vehicle behavior without dealing with the complexity of a full 3D model.

1.2 List of Symbols

Table 1.1: Symbol Table

Symbol	Description	Unit
x	State vector ($x \in \mathbb{R}^8$)	—
p_z	Sprung mass vertical displacement (heave)	m
$v_z = \dot{p}_z$	Sprung mass vertical velocity	m/s
θ	Pitch angle	rad
$\omega = \dot{\theta}$	Pitch rate	rad/s
z_{wf}, z_{wr}	Front and rear unsprung mass positions	m
v_{zf}, v_{zr}	Front and rear unsprung mass velocities	m/s
$u = [u_F \ u_M]^T$	Control input vector	—
u_F	Total vertical control force	N
u_M	Control pitching moment	N m
f_{af}, f_{ar}	Front and rear actuator forces	N
$w \in \mathbb{R}^6$	Disturbance vector	—
$z_{road,f}, z_{road,r}$	Road vertical profiles (front, rear)	m
$\dot{z}_{road,f}, \dot{z}_{road,r}$	Road profile velocities	m/s
F_{xf}, F_{xr}	Longitudinal tire forces	N
s_f, s_r	Front and rear suspension deflections	m
f_{sf}, f_{sr}	Suspension forces	N
f_{tf}, f_{tr}	Tire forces	N
k_f, k_r	Suspension stiffness coefficients	N/m
β_f, β_r	Suspension damping coefficients	N s/m
k_{tf}, k_{tr}	Tire stiffness coefficients	N/m
m	Sprung mass	kg
m_{wf}, m_{wr}	Unsprung masses	kg
J	Pitch moment of inertia	kg m^2
d_f, d_r	CG to axle distances	m
h_{cg}	Height of center of gravity	m
g	Gravitational acceleration	m/s^2
γ_f, γ_r	Anti-dive / Anti-squat coefficients	—
$F_{tot,x}$	Total longitudinal force ($F_{xf} + F_{xr}$)	N
$y \in \mathbb{R}^5$	Measurement vector	—
ν_i	Sensor noise components	—
$z \in \mathbb{R}^6$	Performance output vector	—
r_z	Heave reference	m
r_θ	Pitch reference	rad
η_z, η_θ	Integral states	$\text{m s} / \text{rad s}$
A, B_1, B_2	State matrices	—
C, D_1, D_2	Output matrices	—
C_E, D_{E1}, D_{E2}	Performance matrices	—
Q, R	LQR weighting matrices	—
K_s	State feedback gain	—
K_i	Integral gain	—
K_o	Kalman observer gain	—
Q_{kal}, R_{kal}	Kalman covariance matrices	—

1.3 Dynamic Model

The considered vehicle model is a nonlinear longitudinal half-car representation equipped with two independently actuated suspensions. The model captures the vertical and pitch dynamics of the sprung mass, as well as the vertical dynamics of the front and rear unsprung masses. Tire compliance is explicitly included through linear tire stiffness, allowing the interaction between the wheels and the road profile to be accurately represented.

The sprung mass is assumed to be a rigid body with two degrees of freedom: vertical translation and pitch rotation. Each unsprung mass is modeled as a single vertical degree of freedom. Road excitations are treated as external inputs acting at the tire-road contact points and are not included as dynamic states.

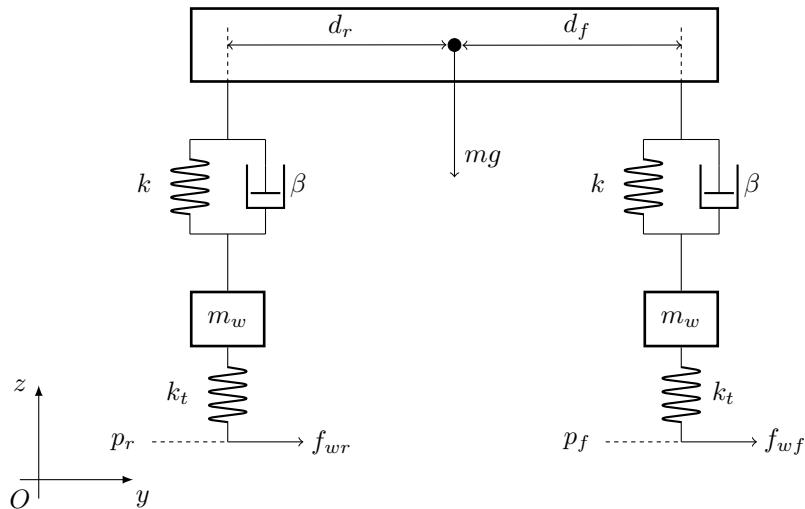


Figure 1.1: Half-car model with sprung and unsprung masses.

1.3.1 State vector

The system state vector is defined as:

$$x = [p_z \quad v_z \quad \theta \quad \omega \quad z_{wf} \quad v_{zf} \quad z_{wr} \quad v_{zr}]^T \in \mathbb{R}^8 \quad (1.1)$$

where p_z and v_z denote the vertical position and velocity of the sprung mass center of gravity, θ and ω are the pitch angle and pitch rate, and z_{wf}, z_{wr} together with their velocities describe the front and rear unsprung mass dynamics.

1.3.2 Control inputs

The active suspension system is driven by two control inputs:

$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}. \quad (1.2)$$

Here u_1 represents the total vertical force applied to the sprung mass, and u_2 is the pitching moment about the center of mass. The corresponding actuator forces at the front and rear suspensions are obtained through the static force distribution:

$$f_{af} = \frac{d_r u_1 + u_2}{d_f + d_r}, \quad (1.3)$$

$$f_{ar} = \frac{d_f u_1 - u_2}{d_f + d_r}, \quad (1.4)$$

with d_f and d_r denoting the distances from the center of mass to the front and rear axles, respectively.

1.3.3 Suspension kinematics

The suspension deflections and relative velocities are given by:

$$s_1 = z_s + d_f \sin \theta - z_{wf}, \quad s_3 = z_s - d_r \sin \theta - z_{wr}, \quad (1.5)$$

$$s_2 = \dot{z}_s + d_f \dot{\theta} \cos \theta - \dot{z}_{wf}, \quad s_4 = \dot{z}_s - d_r \dot{\theta} \cos \theta - \dot{z}_{wr}. \quad (1.6)$$

These expressions account for both the translational motion of the sprung mass and the geometric contribution due to pitch rotation.

1.3.4 Suspension and tire forces

The suspension forces are modeled as linear spring-damper elements:

$$f_{sf} = -k_f s_1 - \beta_f s_2, \quad (1.7)$$

$$f_{sr} = -k_r s_3 - \beta_r s_4, \quad (1.8)$$

where k_f , k_r and β_f , β_r are the stiffness and damping coefficients of the front and rear suspensions.

The tire-road interaction is described using linear tire stiffness:

$$f_{tf} = k_{tf}(z_{rf} - z_{wf}), \quad (1.9)$$

$$f_{tr} = k_{tr}(z_{rr} - z_{wr}), \quad (1.10)$$

with z_{rf} and z_{rr} denoting the vertical road displacements at the front and rear contact points.

1.3.5 Sprung mass dynamics

The vertical acceleration of the sprung mass follows from Newton's second law:

$$\ddot{z}_s = -g + \frac{1}{m} (f_{sf} + f_{sr} + f_{af} + f_{ar}). \quad (1.11)$$

The pitch dynamics about the center of mass are governed by:

$$\ddot{\theta} = \frac{1}{J} (d_f(f_{sf} + f_{af}) - d_r(f_{sr} + f_{ar}) + h_{cg}F_x), \quad (1.12)$$

where J is the pitch moment of inertia, h_{cg} is the height of the center of mass, and

$$F_x = F_{xf} + F_{xr}. \quad (1.13)$$

This term accounts for the coupling between longitudinal forces and pitch dynamics through load transfer effects.

1.3.6 Unsprung mass dynamics

The vertical dynamics of the front and rear unsprung masses are given by:

$$\ddot{z}_{wf} = \frac{1}{m_{wf}} (f_{tf} - f_{sf} - f_{af} - m_{wf}g), \quad (1.14)$$

$$\ddot{z}_{wr} = \frac{1}{m_{wr}} (f_{tr} - f_{sr} - f_{ar} - m_{wr}g). \quad (1.15)$$

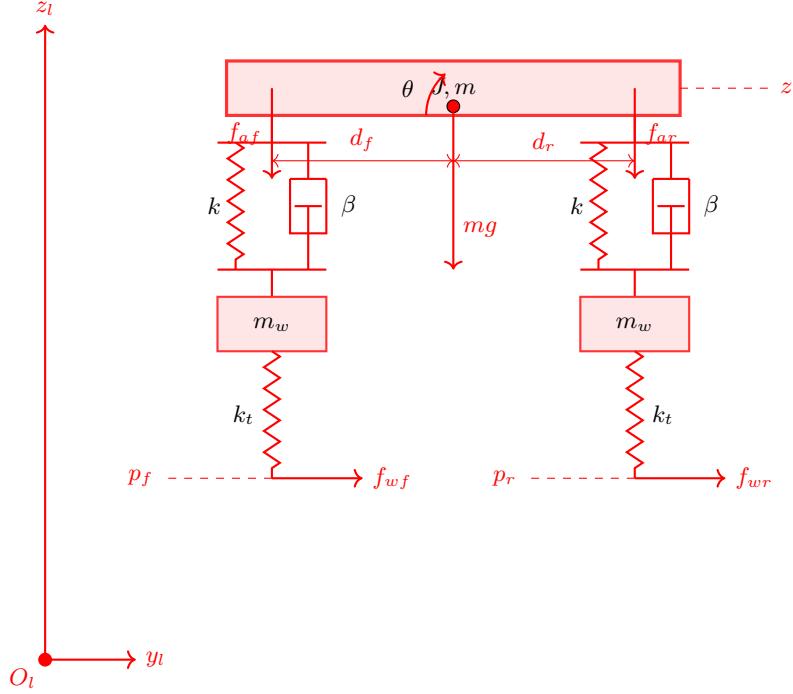
1.3.7 State-space representation

Collecting all terms, the nonlinear vehicle dynamics can be written in first-order state-space form as:

$$\dot{x} = f(x, u, w, r), \quad (1.16)$$

where w represents external longitudinal force disturbances and r collects the road profile inputs at the wheel contact points.

This formulation provides a complete nonlinear description of the vertical and pitch dynamics of the vehicle, forming the basis for linearization, controller synthesis, and observer design.



1.4 Sensor Model

The control architecture relies on a set of onboard sensors that provide real-time measurements of the vehicle vertical and pitch dynamics. The selected sensor suite is designed to ensure observability of the nonlinear half-car model and to provide sufficient information for feedback control in the presence of road disturbances and measurement noise.

All sensor measurements are assumed to be affected by additive noise and are expressed in the vehicle body-fixed reference frame.

1.4.1 Measurement Vector

The complete measurement vector is defined as:

$$y = \begin{bmatrix} y_y \\ y_z \\ y_g \\ y_f \\ y_r \end{bmatrix} = \begin{bmatrix} \sin(\theta)(f_2 + g) + \cos(\theta)\frac{F_x}{m} \\ \cos(\theta)(f_2 + g) - \sin(\theta)\frac{F_x}{m} \\ \dot{\theta} \\ s_1 \\ s_3 \end{bmatrix} + \begin{bmatrix} \nu_y \\ \nu_z \\ \nu_g \\ \nu_f \\ \nu_r \end{bmatrix}. \quad (1.17)$$

This vector includes measurements from two accelerometers, one gyroscope, and two suspension displacement sensors.

1.4.2 Accelerometer Model

The first two outputs y_y and y_z represent the longitudinal and vertical accelerations measured in the body-fixed reference frame. These signals are obtained from MEMS accelerometers mounted near the vehicle center of mass.

The accelerometer outputs are nonlinear functions of the vehicle dynamics and include both inertial and gravitational contributions. The term $f_2 + g$ corresponds to the absolute vertical acceleration of the sprung mass, while F_x/m represents the longitudinal acceleration induced by external forces.

The trigonometric terms $\sin(\theta)$ and $\cos(\theta)$ account for the projection of the acceleration vector onto the body-fixed axes due to the pitch angle, resulting in an intrinsic coupling between vertical and pitch dynamics.

A typical automotive-grade device is the **STMicroelectronics LIS3DH**, providing 12-bit resolution and low noise density.



Figure 1.2: MEMS accelerometer LIS3DH

1.4.3 Gyroscope Model

The third measurement y_g corresponds to the pitch angular rate:

$$y_g = \dot{\theta} + \nu_g. \quad (1.18)$$

This signal is provided by a gyroscope rigidly attached to the vehicle body and delivers high-bandwidth information essential for pitch stabilization and transient response improvement.

A representative sensor is the **Bosch BMI160**, integrating accelerometer and gyroscope within a compact IMU package.

1.4.4 Suspension Deflection Sensors

The last two measurements y_f and y_r correspond to the front and rear suspension deflections:

$$y_f = s_1 + \nu_f, \quad y_r = s_3 + \nu_r. \quad (1.19)$$

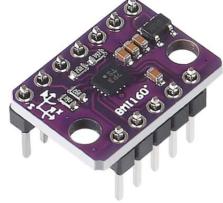


Figure 1.3: Bosch BMI160 IMU

The deflections are defined as:

$$s_1 = (z_s + d_f \sin \theta) - z_{wf}, \quad (1.20)$$

$$s_3 = (z_s - d_r \sin \theta) - z_{wr}, \quad (1.21)$$

and represent the relative displacement between the sprung mass and the unsprung masses. These measurements provide direct information on road excitation and wheel-body interaction.

Linear potentiometers such as the **SLS190** are assumed.



Figure 1.4: Linear suspension potentiometer

1.4.5 Noise Modeling

All sensor measurements are affected by additive noise:

$$\nu = [\nu_y \quad \nu_z \quad \nu_g \quad \nu_f \quad \nu_r]^T. \quad (1.22)$$

Noise components are modeled as zero-mean Gaussian processes with variances derived from typical automotive sensor specifications. Accelerometers and gyroscopes include white noise and bias drift, while suspension sensors are mainly affected by quantization and thermal noise.

1.5 Performance Variables and Control Objectives

To evaluate the effectiveness of the active suspension system and to formulate the optimization problem (e.g., for optimal control or performance monitoring), a specific performance output vector $z \in \mathbb{R}^6$ is defined. Unlike the measured output vector y , which depends on available sensors, the vector z contains the physical variables that strictly define the ride quality and the mechanical constraints of the vehicle.

The performance vector is constructed as a function of the state vector x , the vehicle parameters p , and the reference vector r :

$$z = g(x, p, r). \quad (1.23)$$

1.5.1 Control Objectives

The control objective is defined in terms of vertical position and perceived pitch regulation. An apparent pitch angle θ_a is reconstructed from accelerometer data as:

$$\theta_a = \sin^{-1} \left(\frac{y_y}{\sqrt{y_y^2 + y_z^2}} \right). \quad (1.24)$$

The control error vector is defined as:

$$e = \begin{bmatrix} \frac{(s_1 + \nu_f)d_r + (s_3 + \nu_r)d_f}{d_r + d_f} - r_z \\ \theta_a - r_\theta \end{bmatrix}. \quad (1.25)$$

where r_z and r_θ denote the desired vertical position and pitch angle references. The first component regulates the vehicle vertical displacement through a weighted average of suspension deflections, while the second penalizes deviations from the perceived pitch angle experienced by passengers.

1.5.2 Output Error Function

The complete nonlinear output error function used for control design is:

$$he(x, u, w) = \begin{bmatrix} \frac{(s_1 + \nu_f)d_r + (s_3 + \nu_r)d_f}{d_r + d_f} - r_z \\ \sin^{-1} \left(\frac{h_1 + \nu_y}{\sqrt{(h_1 + \nu_y)^2 + (h_2 + \nu_z)^2}} \right) - r_\theta \end{bmatrix}. \quad (1.26)$$

where h_1 and h_2 represent the ideal accelerometer outputs derived from the system dynamics.

1.5.3 Observability Considerations

The selected sensor configuration ensures observability of the states relevant for control. Accelerometer and gyroscope measurements capture translational and rotational dynamics, while suspension deflection sensors resolve wheel–body interaction and road-induced disturbances, enabling reliable state estimation through standard observers.

1.5.4 Augmented Performance Output for Control Design

In addition to the nonlinear output error function $h_e(x, u, w)$ previously defined for tracking purposes, the performance output vector has been extended to match the complete Simulink control architecture and to explicitly account for dynamic and structural constraints of the half-car model.

Specifically, the performance vector is no longer limited to static regulation of vertical position and perceived pitch angle, but it also includes additional dynamic quantities required to improve transient behavior, damping characteristics, and suspension stroke monitoring.

The resulting performance vector is defined as:

$$z = g(p, x, r) \in \mathbb{R}^6, \quad (1.27)$$

with the following structure:

$$z = \begin{bmatrix} p_z - r_z \\ \theta - r_\theta \\ \omega \\ s_f \\ s_r \\ v_z \end{bmatrix}. \quad (1.28)$$

1.5.5 Definition of Suspension Deflections

The suspension deflections are explicitly reconstructed from the kinematic relations of the half-car model as:

$$s_f = (p_z + d_f \sin \theta) - z_{wf}, \quad (1.29)$$

$$s_r = (p_z - d_r \sin \theta) - z_{wr}, \quad (1.30)$$

where:

- p_z is the vertical position of the center of gravity;
- θ is the pitch angle;
- z_{wf} and z_{wr} are the front and rear wheel hub vertical positions;
- d_f and d_r are the longitudinal distances between the center of gravity and the front and rear axles.

1.5.6 Physical Meaning of the Augmented Components

The augmentation of the performance vector introduces four additional quantities beyond the basic tracking errors:

1. ω (pitch rate): This term is included to provide additional rotational damping. Penalizing ω improves transient response and reduces oscillatory pitch dynamics.
2. s_f and s_r (front and rear suspension deflections): These terms allow the controller to indirectly monitor suspension stroke limits (jounce and rebound). Their inclusion prevents excessive suspension travel and improves ride comfort and safety.
3. v_z (vertical velocity of the center of gravity): This component contributes to heave damping, reducing vertical oscillations and improving passenger comfort.

1.5.7 Relation with the Output Error Function

The first two components of z are consistent with the regulation objectives previously introduced through the nonlinear output error function $h_e(x, u, w)$. In particular:

- The vertical position error corresponds to the regulation of the vehicle heave motion.
- The pitch error enforces alignment with the desired pitch reference.

The additional components do not modify the reference definitions r_z and r_θ , and no previously introduced variable names have been altered. Instead, the performance matrix has been augmented to reflect the complete Simulink block implementation, where tracking, damping, and structural constraints are treated simultaneously within the control synthesis framework.

1.5.8 Control-Oriented Interpretation

From a control design perspective, the augmented vector z defines the full set of regulated outputs used for performance shaping. This formulation allows:

- reference tracking (heave and pitch);
- oscillation damping (through v_z and ω);
- suspension constraint management (through s_f and s_r).

1.5.9 System Linearization

To facilitate the control design of the longitudinal half-car model equipped with active front and rear suspension systems, the initial step involves the linearization of the nonlinear system dynamics. This process is performed by identifying appropriate steady-state operating points (x^*, y^*, w^*) , which characterize representative conditions under which the vehicle is expected to operate. Linearizing the system around these equilibrium points enables the derivation of a time-invariant linear approximation of the vehicle dynamics, thereby simplifying the synthesis and analysis of control strategies.

1.6 System Linearization

To facilitate the control design of the longitudinal half-car model equipped with active front and rear suspension systems, the initial step involves the linearization of the nonlinear system dynamics. This process is performed by identifying appropriate steady-state operating points (x^*, y^*, w^*) , which characterize representative conditions under which the vehicle is expected to operate. Linearizing the system around these equilibrium points enables the derivation of a time-invariant linear approximation of the vehicle dynamics, thereby simplifying the synthesis and analysis of control strategies.

1.6.1 Linearization Around the Operating Point

Consider the nonlinear system model:

$$\begin{aligned}\dot{x} &= f(x, u, w), \quad x(t_0) = x_0 \\ y &= h(x, u, w) \\ e &= h_e(x, u, w)\end{aligned}\tag{1.31}$$

The steady-state operating points (x^*, u^*, w^*) is called *equilibrium triplet* if satisfies the condition:

$$f(x^*, u^*, w^*) = 0\tag{1.32}$$

and defines the equilibrium output and error as:

$$y^* := h(x^*, u^*, w^*), \quad e^* := h_e(x^*, u^*, w^*)\tag{1.33}$$

The variations around the equilibrium point are defined as:

$$\begin{aligned}\tilde{x} &:= x - x^* \\ \tilde{y} &:= y - y^* \\ \tilde{e} &:= e - e^* \\ \tilde{u} &:= u - u^* \\ \tilde{w} &:= w - w^*\end{aligned}\tag{1.34}$$

Using the fact that $\dot{x}^* = 0$, the dynamics of the variations are:

$$\begin{aligned}\dot{\tilde{x}} &= f(x^* + \tilde{x}, u^* + \tilde{u}, w^* + \tilde{w}), \quad \tilde{x}(t_0) = x_0 - x^* \\ \tilde{y} &= h(x^* + \tilde{x}, u^* + \tilde{u}, w^* + \tilde{w}) \\ \tilde{e} &= h_e(x^* + \tilde{x}, u^* + \tilde{u}, w^* + \tilde{w})\end{aligned}\tag{1.35}$$

To obtain a tractable model for controller synthesis, a first-order Taylor expansion is applied around the equilibrium point. The resulting Jacobian matrices are defined as:

$$\begin{aligned}A &:= \frac{\partial f(x, u, w)}{\partial x} \Big|_{\substack{x=x^* \\ u=u^* \\ w=w^*}} & B_1 &:= \frac{\partial f(x, u, w)}{\partial u} \Big|_{\substack{x=x^* \\ u=u^* \\ w=w^*}} & B_2 &:= \frac{\partial f(x, u, w)}{\partial w} \Big|_{\substack{x=x^* \\ u=u^* \\ w=w^*}} \\ C &:= \frac{\partial h(x, u, w)}{\partial x} \Big|_{\substack{x=x^* \\ u=u^* \\ w=w^*}} & D_1 &:= \frac{\partial h(x, u, w)}{\partial u} \Big|_{\substack{x=x^* \\ u=u^* \\ w=w^*}} & D_2 &:= \frac{\partial h(x, u, w)}{\partial w} \Big|_{\substack{x=x^* \\ u=u^* \\ w=w^*}} \\ C_e &:= \frac{\partial h_e(x, u, w)}{\partial x} \Big|_{\substack{x=x^* \\ u=u^* \\ w=w^*}} & D_{e1} &:= \frac{\partial h_e(x, u, w)}{\partial u} \Big|_{\substack{x=x^* \\ u=u^* \\ w=w^*}} & D_{e2} &:= \frac{\partial h_e(x, u, w)}{\partial w} \Big|_{\substack{x=x^* \\ u=u^* \\ w=w^*}}\end{aligned}\tag{1.36}$$

Neglecting second-order terms, the linearized system becomes the so-called *design model*:

$$\begin{cases} \dot{\tilde{x}} = A\tilde{x} + B_1\tilde{u} + B_2\tilde{w}, \quad \tilde{x}(t_0) = x_0 - x^* \\ \tilde{y} = C\tilde{x} + D_1\tilde{u} + D_2\tilde{w} \\ \tilde{e} = C_e\tilde{x} + D_{e1}\tilde{u} + D_{e2}\tilde{w} \end{cases}\tag{1.37}$$

This Linear Time-Invariant (LTI) approximation of the nonlinear model is valid in a neighborhood of the equilibrium point, enabling efficient analysis and controller design under small perturbations.

1.6.2 Matrix calculus

For the matrix calculation, the procedure described in equations (1.36) was followed, obtaining:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{k_f+k_r}{m} & -\frac{\beta_f+\beta_r}{m} & -\frac{d_f k_f - d_r k_r}{m} & -\frac{\beta_f d_f - \beta_r d_r}{m} & \frac{k_f}{m} & \frac{\beta_f}{m} & \frac{k_r}{m} & \frac{\beta_r}{m} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{d_f k_f - d_r k_r}{J} & -\frac{\beta_f d_f - \beta_r d_r}{J} & -\frac{k_f d_f^2 + k_r d_r^2}{J} & -\frac{\beta_f d_f^2 + \beta_r d_r^2}{J} & \frac{d_f k_f}{J} & \frac{\beta_f d_f}{J} & -\frac{d_r k_r}{J} & -\frac{\beta_r d_r}{J} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{k_f}{m_{wf}} & \frac{\beta_f}{m_{wf}} & \frac{d_f k_f}{m_{wf}} & \frac{\beta_f d_f}{m_{wf}} & -\frac{k_f + k_{tf}}{m_{wf}} & -\frac{\beta_f}{m_{wf}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{k_r}{m_{wr}} & \frac{\beta_r}{m_{wr}} & -\frac{d_r k_r}{m_{wr}} & -\frac{\beta_r d_r}{m_{wr}} & 0 & 0 & -\frac{k_r + k_{tr}}{m_{wr}} & -\frac{\beta_r}{m_{wr}} \end{bmatrix} \quad (1.38)$$

$$B_1 = \begin{bmatrix} 0 & 0 \\ \frac{1}{m} & 0 \\ 0 & 0 \\ 0 & \frac{1}{J} \\ 0 & 0 \\ -\frac{d_r}{m_{wf}(d_f + d_r)} & -\frac{1}{m_{wf}(d_f + d_r)} \\ 0 & 0 \\ -\frac{d_f}{m_{wr}(d_f + d_r)} & \frac{1}{m_{wr}(d_f + d_r)} \end{bmatrix} \quad (1.39)$$

$$B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\gamma_f}{m} & \frac{\gamma_r}{m} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{h_{cg} + d_f \gamma_f}{J} & \frac{h_{cg} - d_r \gamma_r}{J} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{k_{tf}}{m_{wf}} & 0 & 0 & 0 & -\frac{\gamma_f}{m_{wf}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{k_{tr}}{m_{wr}} & 0 & 0 & -\frac{\gamma_r}{m_{wr}} \end{bmatrix} \quad (1.40)$$

$$D_1 = \begin{bmatrix} 0 & 0 \\ \frac{1}{m} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (1.41)$$

$$D_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & \frac{1}{m} & \frac{1}{m} \\ 0 & 0 & 0 & 0 & \frac{\gamma_f}{m} & \frac{\gamma_r}{m} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.42)$$

$$C_e = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & d_f & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & -d_r & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.43)$$

$$D_{e1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (1.44)$$

$$D_{e2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & -\frac{k_f(p_z - z_{wf}) + k_r(p_z - z_{wr})}{k_f(p_z - z_{wf}) + k_r(p_z - z_{wr})} & -\frac{k_f(p_z - z_{wf}) + k_r(p_z - z_{wr})}{k_f(p_z - z_{wf}) + k_r(p_z - z_{wr})} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.45)$$

Numerical Matrices

To calculate the numerical values of the matrices, the parameters were replaced with the values of the vehicle chosen for this study: All parameters used to find the numeric matrices are based on the Rolls-Royce Ghost 6.7 V12, a car suitable for this type of active suspensions and with same suspension parameters on front and rear axle, this simplifies slightly our model:



Figure 1.5: Rolls-Royce Ghost 6.7 V12

Symbol	Description	Value	Unit
m	Total mass of the vehicle	2550	kg
g	Gravitational acceleration	9.81	m/s^2
h_{cg}	Height of center of gravity	0.60	m
d_f	Front axle distance from CoM	1.65	m
d_r	Rear axle distance from CoM	1.65	m
L	Wheelbase	3.30	m
J	Pitch moment of inertia	4080	$\text{kg}\cdot\text{m}^2$
k_f	Front suspension stiffness	35000	N/m
k_r	Rear suspension stiffness	35000	N/m
β_f	Front damping coefficient	4200	$\text{N}\cdot\text{s}/\text{m}$
β_r	Rear damping coefficient	4200	$\text{N}\cdot\text{s}/\text{m}$
m_{wf}	Front unsprung mass	48	kg
m_{wr}	Rear unsprung mass	48	kg
k_{tf}	Front tire stiffness	270000	N/m
k_{tr}	Rear tire stiffness	270000	N/m
ℓ_0	Static suspension height	0.55	m

Table 1.2: Dynamic model parameters of a Rolls-Royce vehicle (Half-Car Model)

The substitution of the values in the table 1.2 into the matrices will result in the following outcome:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -27.45 & -3.294 & 0 & 0 & 13.73 & 1.647 & 13.73 & 1.647 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -47.53 & -5.703 & 14.4 & 1.728 & -14.4 & -1.728 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 729.2 & 87.5 & 1203 & 144.4 & -6354 & -87.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 729.2 & 87.5 & -1203 & -144.4 & 0 & 0 & -6354 & -87.5 \end{bmatrix} \quad (1.46)$$

$$B_1 = \begin{bmatrix} 0 & 0 \\ 3.922 \times 10^{-4} & 0 \\ 0 & 0 \\ 0 & 2.494 \times 10^{-4} \\ 0 & 0 \\ -1.042 \times 10^{-2} & -6.313 \times 10^{-3} \\ 0 & 0 \\ -1.042 \times 10^{-2} & 6.313 \times 10^{-3} \end{bmatrix} \quad (1.47)$$

$$B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.961 \times 10^{-5} & 3.137 \times 10^{-5} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.702 \times 10^{-4} & 1.167 \times 10^{-4} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 5625 & 0 & 0 & 0 & -1.042 \times 10^{-3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5625 & 0 & 0 & -1.667 \times 10^{-3} \end{bmatrix} \quad (1.48)$$

$$D_1 = \begin{bmatrix} 0 & 0 \\ 3.922 \times 10^{-4} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (1.49)$$

$$D_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 3.922 \times 10^{-4} & 3.922 \times 10^{-4} \\ 0 & 0 & 0 & 0 & 1.961 \times 10^{-5} & 3.137 \times 10^{-5} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.50)$$

$$C_e = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1.65 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1.65 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.51)$$

$$D_{e1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (1.52)$$

$$D_{e2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3.998 \times 10^{-5} & 3.998 \times 10^{-5} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.53)$$

1.6.3 Linear Model Analysis

To support the development of an effective suspension controller and gain insight into the vehicle dynamic behavior, the complete linearized half-car model is analyzed around the static equilibrium configuration.

The equilibrium corresponds to the vehicle at rest, with zero pitch angle, zero vertical and angular velocities, and suspension deflections balancing gravitational loads.

Starting from the linearized model derived in (1.37), the intrinsic open-loop dynamics of the passive system can be studied by neglecting control inputs and external disturbances (i.e., $\tilde{u} = 0$, $\tilde{w} = 0$).

Under these conditions, the system reduces to the homogeneous form:

$$\dot{\tilde{x}} = A\tilde{x} \quad (1.54)$$

which characterizes the natural dynamic behavior of the linearized half-car model in open loop.

where the state vector of variations is defined as:

$$\tilde{x} = \begin{bmatrix} \tilde{p}_z \\ \tilde{v}_z \\ \tilde{\theta} \\ \tilde{\omega} \\ \tilde{z}_{wf} \\ \tilde{v}_{wf} \\ \tilde{z}_{wr} \\ \tilde{v}_{wr} \end{bmatrix} \quad (1.55)$$

The first four states describe the sprung-mass dynamics (heave and pitch), while the last four correspond to the unsprung masses (front and rear wheel dynamics).

Open-Loop Eigenvalue Analysis

To evaluate stability and dynamic characteristics, the eigenvalues of matrix A are computed by solving:

$$\det(A - \lambda I) = 0 \quad (1.56)$$

The resulting eigenvalues are:

$$\lambda_{1,2} = -44.0652 \pm j 64.6521 \quad (1.57)$$

$$\lambda_{3,4} = -44.2357 \pm j 63.0972 \quad (1.58)$$

$$\lambda_{5,6} = -1.3319 \pm j 4.8425 \quad (1.59)$$

$$\lambda_{7,8} = -2.3658 \pm j 6.2788 \quad (1.60)$$

All eigenvalues have strictly negative real parts, indicating that the passive linearized half-car model is asymptotically stable in open loop.

The spectrum reveals two distinct dynamic regimes:

- **High-frequency modes** (approximately 12 Hz), associated with the wheel-hop dynamics of the unsprung masses. These modes exhibit significant damping due to tire stiffness and suspension damping.
- **Low-frequency modes** (approximately 0.8–1.1 Hz), corresponding to the sprung-mass heave and pitch motions. These modes are lightly damped and dominate ride comfort behavior.

Although the open-loop system is internally stable, the lightly damped body modes can degrade ride comfort and increase settling time. Therefore, active suspension control is introduced not to ensure stability, but to reshape the closed-loop poles, increase damping, and improve overall dynamic performance.

1.7 Control

1.7.1 Reachability

To determine whether the system states can be influenced by the control inputs, a reachability (controllability) analysis is performed on the pair (A, B_1) .

The reachability matrix is defined as:

$$\mathcal{R} = \begin{bmatrix} B_1 & AB_1 & A^2B_1 & \dots & A^7B_1 \end{bmatrix} \quad (1.61)$$

The system is fully reachable if:

$$\text{rank}(\mathcal{R}) = n \quad (1.62)$$

where $n = 8$ is the number of states.

The computation performed in MATLAB:

```
R_full = ctrb(A_N, B1_N);
rank(R_full)
```

gives:

$$\text{rank}(\mathcal{R}) = 8 \quad (1.63)$$

Since the rank equals the state dimension, the pair (A, B_1) is **fully reachable**.

Although the system has only two control inputs, the dynamic coupling between sprung and unsprung masses ensures that the entire state space can be influenced.

Stabilizability

Full reachability directly implies stabilizability. Therefore, it is possible to design a state-feedback control law:

$$\tilde{u} = K\tilde{x} \quad (1.64)$$

such that the closed-loop matrix:

$$A + B_1K \quad (1.65)$$

is Hurwitz (all eigenvalues in the open left-half complex plane), guaranteeing internal stability.

1.7.2 Integral Action

To eliminate steady-state error in the presence of constant disturbances, integral action is introduced in accordance with the internal model principle. Rejection of constant disturbances and tracking of constant references require the controller to embed an internal model of constant signals, which is achieved by adding integrator states.

The regulated output is defined as:

$$\tilde{e} = C_e \tilde{x} + D_{e1} \tilde{u} + D_{e2} \tilde{w} \quad (1.66)$$

The integrator state η is introduced as:

$$\dot{\eta} = \tilde{e} \quad (1.67)$$

The augmented state vector becomes:

$$x_e = \begin{bmatrix} \tilde{x} \\ \eta \end{bmatrix} \quad (1.68)$$

The augmented system is therefore:

$$\dot{x}_e = \begin{bmatrix} A & 0 \\ C_e & 0 \end{bmatrix} x_e + \begin{bmatrix} B_1 \\ D_{e1} \end{bmatrix} \tilde{u} + \begin{bmatrix} B_2 \\ D_{e2} \end{bmatrix} \tilde{w} \quad (1.69)$$

For constant disturbances \tilde{w} , asymptotic regulation $\lim_{t \rightarrow \infty} \tilde{e}(t) = 0$ is guaranteed if the augmented pair (A_{aug}, B_{aug}) is stabilizable.

A necessary structural condition for perfect steady-state rejection is:

$$\text{rank} \begin{bmatrix} A & B_1 \\ C_e & D_{e1} \end{bmatrix} = n + m_e \quad (1.70)$$

where n is the number of plant states and m_e the number of integrated outputs.

The resulting state-feedback control law is:

$$\tilde{u} = -K_s \tilde{x} - K_i \eta \quad (1.71)$$

where the total gain $[K_s \ K_i]$ is obtained via LQR design on the augmented system.

Since the original pair (A, C) is observable and the augmented pair (A_{aug}, B_{aug}) is stabilizable, the separation principle still applies. Therefore, the observer and the integral state-feedback controller can be designed independently without compromising closed-loop stability.

1.7.3 Observability

In practice, the full state vector is not directly measurable. Therefore, observability of the pair (A, C) must be verified to ensure that the internal states can be reconstructed from the available sensors.

The observability matrix is defined as:

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^7 \end{bmatrix} \quad (1.72)$$

The system is fully observable if:

$$\text{rank}(\mathcal{O}) = n \quad (1.73)$$

where $n = 8$ is the state dimension.

The computation performed in MATLAB:

```
0_full = obsv(A_N, C_N);
rank(0_full)
```

yields:

$$\text{rank}(\mathcal{O}) = 8 \quad (1.74)$$

Since the rank equals the number of states, the pair (A, C) is **fully observable**. This implies that no internal mode of the system evolves independently of the measured outputs.

From a physical perspective, the dynamic coupling between sprung and unsprung masses, combined with the availability of accelerometer, gyroscope and suspension deflection measurements, provides sufficient information to reconstruct both positional and velocity states.

Detectability

Full observability directly implies detectability. Therefore, all unstable modes of A are observable, and it is possible to design an observer such that the estimation error dynamics

$$\dot{e}_x = (A - LC)e_x \quad (1.75)$$

are asymptotically stable.

This guarantees that the state estimate \hat{x} converges to the true state x , provided the pair (A, C) is observable and L is properly selected.

1.7.4 Kalman Filter Design

Since the pair (A, C) is fully observable, a full-order steady-state Kalman observer is designed.

The observer dynamics are:

$$\dot{\hat{x}} = A\hat{x} + B_1\tilde{u} + K_f(y - C\hat{x}) \quad (1.76)$$

The estimation error $e_x = x - \hat{x}$ evolves as:

$$\dot{e}_x = (A - K_f C)e_x \quad (1.77)$$

The steady-state Kalman gain K_f is obtained by solving the dual algebraic Riccati equation:

$$A^\top P + PA - PC^\top R_{obs}^{-1} CP + Q_{obs} = 0 \quad (1.78)$$

with

$$K_f = PC^\top R_{obs}^{-1} \quad (1.79)$$

In practice, the gain is computed exploiting the LQR duality:

```
K_f = lqr(A_N', C_N', Q_obs, R_obs)';
```

where:

- Q_{obs} represents the assumed process uncertainty,
- R_{obs} models the measurement noise covariance.

The tuning of Q_{obs} and R_{obs} reflects the relative confidence in model dynamics and sensors. In particular, higher weights in R_{obs} reduce the influence of noisy measurements, while lower values increase reliance on selected sensors.

Since (A, C) is observable and $R_{obs} > 0$, the solution P is positive definite and the matrix $(A - K_f C)$ is Hurwitz. Therefore, the estimation error converges asymptotically to zero.

1.7.5 LQR Cost Function and Weight Selection

The state-feedback gain is obtained by solving the infinite-horizon quadratic optimization problem:

$$J = \int_0^\infty (\tilde{x}_e^\top Q_{aug} \tilde{x}_e + \tilde{u}^\top R_{aug} \tilde{u}) dt \quad (1.80)$$

where $\tilde{x}_e \in \mathbb{R}^{10}$ includes the physical states and the integral states.

The weighting matrices are structured as:

$$Q_{aug} = \text{diag}(q_{pz}, q_{vz}, q_\theta, q_\omega, q_{wf}, q_{vwf}, q_{wr}, q_{vwr}, q_{Ih}, q_{I\theta}) \quad (1.81)$$

$$R_{aug} = r_u I_2 \quad (1.82)$$

The adopted numerical values are:

$$q_{pz} = 5 \cdot 10^7 \cdot w_P = 9 \cdot 10^9 \quad (1.83)$$

$$q_\theta = 8 \cdot 10^7 \cdot w_P = 1.44 \cdot 10^{10} \quad (1.84)$$

$$q_{vz} = 5 \cdot 10^7 \cdot w_D = 3 \cdot 10^9 \quad (1.85)$$

$$q_\omega = 8 \cdot 10^7 \cdot w_D = 4.8 \cdot 10^9 \quad (1.86)$$

$$q_{Ih} = 10^9 \cdot w_I = 8 \cdot 10^{10} \quad (1.87)$$

$$q_{I\theta} = 10^9 \cdot w_I = 8 \cdot 10^{10} \quad (1.88)$$

$$r_u = 5 \cdot 10^{-3} \quad (1.89)$$

with tuning factors:

$$w_P = 180, \quad w_D = 60, \quad w_I = 80 \quad (1.90)$$

The weighting strategy reflects the control objectives:

- Large weights on heave and pitch states enforce comfort improvement.
- Significant weights on vertical and angular velocities increase damping.
- Very large integral weights guarantee zero steady-state error.
- A relatively small r_u allows sufficiently aggressive actuation.

The large numerical spread of the weights (spanning several orders of magnitude) compensates for the different physical units of the state variables and implicitly performs state scaling.

From a control perspective, the ratio between Q_{aug} and R_{aug} directly determines the aggressiveness of the feedback law, shaping the trade-off between ride comfort and actuator effort.

Chapter 2

MATLAB/Simulink Implementation

2.1 Overall Simulation Architecture

The complete control framework has been implemented in MATLAB/Simulink using a modular structure that separates the nonlinear plant, the linearized model used for control synthesis, the state observer, the performance evaluation block, and the disturbance generators.

The simulation architecture is organized into the following main subsystems:

- Nonlinear Half-Car Plant (blocks f , h , and e),
- Road Profile Generator,
- Longitudinal Wheel Forces Generator,
- Linearized Model (for control design and validation),
- State Observer,
- Augmented LQR Controller with integral action,
- Enable logic and scopes for scenario-based simulations.

This structure ensures coherence between the analytical derivations, the symbolic linearization procedure, and the numerical simulations.

2.1.1 Symbolic Linearization and Control Synthesis Script

The core of the control-oriented implementation is the MATLAB script dedicated to symbolic linearization and numerical evaluation of the half-car model.

This file plays a fundamental role in the overall workflow, since it:

- Defines the full nonlinear model in symbolic form,
- Computes all Jacobian matrices required for linearization,
- Evaluates the model at a physically consistent equilibrium point,
- Generates the numerical state-space matrices,
- Verifies observability,
- Designs both the stabilizing LQR controller and the state observer,
- Builds the augmented controller with integral action.

The script therefore acts as the analytical bridge between the nonlinear Simulink plant and the linear control architecture implemented in the simulation model.

1. Symbolic Parameters

In the first section, all physical parameters of the half-car model are declared symbolically:

```

6 %% 1. SYMBOLIC PARAMETERS
7 syms m kf kr betaf betar ell0 g df dr J mwf mwr ktf ktr hcg gammaf gammaf real
8 vars = [m kf kr betaf betar ell0 g df dr J mwf mwr ktf ktr hcg gammaf gammaf];
9
10

```

This step ensures that the subsequent Jacobian computation remains fully analytical.

Keeping parameters symbolic allows:

- Exact differentiation of nonlinear expressions,
- Clear separation between model structure and numerical values,
- Reusability of the script for different vehicle configurations.

2. State Variables

The full nonlinear state vector is defined as:

```

11 %% 2. STATE VARIABLES (8x1 column)
12 syms pz vz theta omega z_wf vz_wf z_wr vz_wr real
13 states = [pz; vz; theta; omega; z_wf; vz_wf; z_wr; vz_wr];
14

```

This 8-state formulation includes:

- Sprung mass vertical dynamics (p_z, v_z),
- Pitch dynamics (θ, ω),

- Front and rear unsprung mass dynamics.

Defining the states explicitly in symbolic form ensures consistency between:

- Nonlinear dynamics,
- Measurement equations,
- Performance outputs,
- Linearized state-space matrices.

3. Control Inputs

The control input vector is defined as:

```

15      %% 3. CONTROL INPUTS (2x1 column)
16      syms u_F u_M real
17      inputs_u = [u_F; u_M];
18

```

These represent:

- u_F : total vertical control force,
- u_M : control pitching moment.

The symbolic formulation allows correct front/rear force redistribution according to the geometric distances d_f and d_r .

4. Disturbances

The disturbance vector includes:

```

19      %% 4. DISTURBANCES (6x1 Column)
20      syms zroad_f dzroad_f zroad_r dzroad_r Fxf Fxr real
21      inputs_w = [zroad_f; dzroad_f; zroad_r; dzroad_r; Fxf; Fxr];
22

```

This formulation is particularly important because it separates:

- Vertical road excitations,
- Longitudinal wheel forces.

Such separation enables independent simulation of:

- Road-induced vibrations,
- Acceleration and braking scenarios,
- Combined vertical and longitudinal disturbances.

5. Sensor Noise and References

Sensor noise variables n_i are introduced symbolically in the measurement equations.

Reference signals r_z and r_θ are defined separately to ensure that the linearization process is performed around equilibrium without embedding reference offsets directly into the Jacobians.

```

23 %% 5. SENSOR NOISE & REFS
24 syms n1 n2 n3 n4 n5 rz_ref rtheta_ref real
25 noise = [n1; n2; n3; n4; n5];
26 refs = [rz_ref; rtheta_ref];
27

```

This separation is crucial for correct controller synthesis, since references are handled as exogenous inputs in Simulink rather than as part of the plant dynamics.

6. Nonlinear Kinematics and Dynamics

This section constructs the core nonlinear dynamics of the half-car model.

```

28 %% 6. KINEMATICS & DYNAMICS
29 s1 = (pz + df * sin(theta)) - z_wf;
30 s3 = (pz - dr * sin(theta)) - z_wr;
31 s2 = vz + df * omega * cos(theta) - vz_wf;
32 s4 = vz - dr * omega * cos(theta) - vz_wr;
33
34 fsf = -kf*s1 - betaf*s2;
35 fsr = -kr*s3 - betar*s4;
36 F_v_long_f = gammaf * Fxf;
37 F_v_long_r = gammar * Fxr;
38 F_tot_x = Fxf + Fxr;
39
40 ftf = ktf*(zroad_f - z_wf);
41 ftr = ktr*(zroad_r - z_wr);
42
43 fas = (u_M + dr*u_F)/(df + dr);
44 far = (df*u_F - u_M)/(df + dr);
45
46 % Equazioni di stato (dxdt)
47 f2 = -g + (fsf + fsr + fas + far + F_v_long_f + F_v_long_r)/m;
48 f4 = (df*(fsf + fas + F_v_long_f) - dr*(fsr + far + F_v_long_r) + F_tot_x*hcg)/J;
49 f_wf = (ftf - fsf - fas - F_v_long_f - mwf*g)/mwf;
50 f_wr = (ftr - fsr - far - F_v_long_r - mwr*g)/mwr;
51
52 f_sys = [vz; f2; omega; f4; vz_wf; f_wf; vz_wr; f_wr];
53

```

Relative velocities s_2 and s_4 include the nonlinear coupling term $\omega \cos \theta$, ensuring correct pitch-to-heave interaction.

Spring-damper forces are then defined, followed by:

- Tire stiffness forces,
- Active force redistribution,

- Longitudinal load transfer contributions.

Finally, the full nonlinear state equation:

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

is assembled in vector form.

This formulation is identical to the one implemented in the Simulink nonlinear plant block, guaranteeing coherence between symbolic derivation and simulation model.

7. Output Definition: Sensors and Performance Channels

After defining the nonlinear dynamics, the script introduces two distinct output vectors:

- The measurement vector $y(x, u, w)$,
- The performance vector $z(x, u, w)$.

```

54 %% 7. OUTPUTS (Sensori y e Performance z)
55 ax_i = F_tot_x/m;
56 az_i = f2;
57 az_r = az_i + g;
58
59 y_sys = [sin(theta)*az_r + cos(theta)*ax_i + n1;
60           cos(theta)*az_r - sin(theta)*ax_i + n2;
61           omega + n3;
62           s1 + n4;
63           s3 + n5];
64
65 z_sys = [pz; theta; omega; s1; s3; vz];
66

```

Measurement Vector The sensor output is defined as:

$$y = \begin{bmatrix} a_y \\ a_z \\ \omega \\ s_1 \\ s_3 \end{bmatrix},$$

where the accelerometer signals are reconstructed from inertial accelerations:

$$a_x = \frac{F_{xf} + F_{xr}}{m}, \quad a_z = f_2 + g.$$

These accelerations are then projected into the body reference frame using $\sin \theta$ and $\cos \theta$, reproducing the nonlinear coupling between pitch angle and measured vertical acceleration.

Sensor noise terms n_i are explicitly included in symbolic form. This is essential for the subsequent observer design, since the noise covariance matrix R_{obs} directly reflects this structure.

Performance Vector The performance vector is defined as:

$$z = \begin{bmatrix} p_z \\ \theta \\ \omega \\ s_1 \\ s_3 \\ v_z \end{bmatrix}.$$

Importantly, reference subtraction is not embedded in this symbolic definition. References enter the Simulink model externally as offsets, ensuring that the Jacobian matrices are computed with respect to the true physical states rather than tracking errors.

This separation preserves correctness of the linearization process.

8. Jacobian Computation

The core purpose of the script is the computation of all required Jacobian matrices.

```

67 %% 8. JACOBIANS (calcolo di tutte le matrici)
68 A_sym = jacobian(f_sys, states);
69 B1_sym = jacobian(f_sys, inputs_u);
70 B2_sym = jacobian(f_sys, inputs_w);
71
72 C_sym = jacobian(y_sys, states);
73 D1_sym = jacobian(y_sys, inputs_u);
74 D2_sym = jacobian(y_sys, inputs_w);
75
76 CE_sym = jacobian(z_sys, states);
77 DE1_sym = jacobian(z_sys, inputs_u);
78 DE2_sym = jacobian(z_sys, inputs_w);
79

```

The following symbolic derivatives are computed:

$$\begin{aligned} A &= \frac{\partial f}{\partial x}, & B_1 &= \frac{\partial f}{\partial u}, & B_2 &= \frac{\partial f}{\partial w}, \\ C &= \frac{\partial y}{\partial x}, & D_1 &= \frac{\partial y}{\partial u}, & D_2 &= \frac{\partial y}{\partial w}, \end{aligned}$$

$$C_E = \frac{\partial z}{\partial x}, \quad D_{E1} = \frac{\partial z}{\partial u}, \quad D_{E2} = \frac{\partial z}{\partial w}.$$

This step guarantees that the linearized model is obtained through exact analytical differentiation, avoiding numerical approximation errors.

The resulting structure is fully consistent with the generalized plant formulation:

$$\begin{aligned}\dot{x} &= Ax + B_1u + B_2w, \\ y &= Cx + D_1u + D_2w, \\ z &= C_Ex + D_{E1}u + D_{E2}w.\end{aligned}$$

9. Numerical Evaluation at the Equilibrium Point

After symbolic derivation, all expressions are numerically evaluated at a physically consistent static equilibrium.

```

80 %% 9. NUMERICAL EVALUATION
81 m_n = 2550; kf_n = 35000; kr_n = 35000; betaf_n = 4200; betar_n = 4200;
82 ell0_n = 0.55; g_n = 9.81; df_n = 1.65; dr_n = 1.65; J_n = m_n*(0.38*(df_n+dr_n))^2;
83 mwf_n = 48; mwr_n = 48; ktf_n = 270000; ktr_n = 270000; hcg_n = 0.60;
84 gammaf_n = 0.00; gammar_n = 0.00;
85
86 p_num = [m_n, kf_n, kr_n, betaf_n, betar_n, ell0_n, g_n, df_n, dr_n, J_n, mwf_n,
87           mwr_n, ktf_n, ktr_n, hcg_n, gammaf_n, gammar_n];
88
89 %% Calcolo Punto di Equilibrio (Vettore COLONNA)
90 pz_static = -(m_n * g_n) / (kf_n + kr_n);
91 h_static = -((m_n + mwf_n + mwr_n) * g_n) / (2 * ktf_n);
92
93 pzo = pz_static + h_static;
94 th0 = 0;
95
96 x0 = [pz_static + h_static; 0; 0; 0; h_static; 0; h_static; 0];
97
98 x0_lin_en = 0;
99
100
101 %% Sostituzione numerica
102 sub_map = [states; inputs_u; inputs_w; noise; refs];
103 sub_val = [x0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0];
104
105 clean_sub = @(S) double(vpa(subs(subs(S, vars, p_num), sub_map, sub_val), 10));

```

The equilibrium configuration includes:

- Gravitational preload of the suspension,
- Static tire deflection,
- Zero pitch angle,
- Zero velocities.

The vertical equilibrium position is computed and tire compression is determined from total static load distribution.

10. Numerical matrices

Substituting both parameter values and equilibrium state into the symbolic Jacobians produces the numerical matrices:

These matrices are exported to the MATLAB workspace and used directly in the Simulink linear control block.

```
107 %% 10. MATRICI FINALI
108 A_N = clean_sub(A_sym);
109 B1_N = clean_sub(B1_sym);
110 B2_N = clean_sub(B2_sym);
111 C_N = clean_sub(C_sym);
112 D1_N = clean_sub(D1_sym);
113 D2_N = clean_sub(D2_sym);
114 CE_N = clean_sub(CE_sym);
115 DE1_N = clean_sub(DE1_sym);
116 DE2_N = clean_sub(DE2_sym);
117
118 % Calcolo dei valori di equilibrio per i sensori e le performance
119 y_eq = clean_sub(y_sys);
120 z_eq = clean_sub(z_sys);
121
122 disp('Linearizzazione completa eseguita con successo.');
123
124
```

11. Observability Analysis

Before designing the observer, the script verifies structural observability.

```
124 %% 11. VERIFICA OSSERVABILITÀ
125 % Calcolo della matrice di osservabilità per il sistema (A_N, C_N)
126 Ob = obsv(A_N, C_N);
127
128 % Verifica del rango
129 n_stati = size(A_N, 1);
130 rango_Ob = rank(Ob);
131
132 fprintf('--- Analisi Osservabilità ---\n');
133 fprintf('Numero di stati: %d\n', n_stati);
134 fprintf('Rango della matrice di osservabilità: %d\n', rango_Ob);
135
136 if rango_Ob == n_stati
137     disp('RISULTATO: Il sistema è COMPLETAMENTE OSSERVABILE. Possiamo progettare l''Observer.');
138 else
139     disp('ATTENZIONE: Il sistema NON è completamente osservabile. L''Observer non potrà stimare tutti gli stati.');
140     % Analisi dei sottospazi non osservabili (opzionale)
141     [~, ~, ~, ex] = obsvf(A_N, B1_N, C_N);
142 end
143
```

The observability matrix is computed as:

$$\mathcal{O} = \begin{bmatrix} C_N \\ C_N A_N \\ C_N A_N^2 \\ \vdots \end{bmatrix}.$$

Its rank is compared with the number of states.

Full rank confirms that the selected sensor configuration (accelerometers, gyroscope, suspension deflections) allows reconstruction of all eight states.

This analytical verification ensures that the observer design is theoretically admissible.

12. LQR State Feedback Design

A stabilizing LQR controller is first designed using:

$$K_s = \text{lqr}(A_N, B_{1N}, Q_{lqr}, R_{lqr}).$$

The weighting matrix Q_{lqr} assigns:

- High weight to vertical displacement p_z ,
- Even higher weight to pitch angle θ ,
- Moderate weight to velocities v_z and ω ,
- Low weight to unsprung mass dynamics.

The matrix R_{lqr} limits control aggressiveness and prevents unrealistic actuator demand.

This step provides a baseline stabilizing controller.

```

144 %% 12. TUNING STABILIZZATORE (LQR)
145 % Torniamo a valori più umani e bilanciamo meglio
146
147 q_pz    = 1e6;
148 q_theta = 5e6;    % Teniamo il pitch 5 volte più pesante del bounce
149 q_vz    = 1e2;    % Smorzamento (fondamentale per non farlo esplodere)
150 q_omega = 1e2;    % Smorzamento rotazionale
151
152 % Ricostruiamo Q
153 Q_lqr = diag([q_pz, q_vz, q_theta, q_omega, 10, 10, 10, 10]);
154
155 % Alziamo R: non sotto 1e-3.
156 % R funge da "freno" per evitare che il controllo diventi troppo nervoso.
157 r_u = 1e-3;
158 R_lqr = diag([r_u, r_u]);
159
160 [Ks, ~, ~] = lqr(A_N, B1_N, Q_lqr, R_lqr);
161
162

```

13. Observer Design via Dual LQR

The state observer gain K_o is computed using the dual LQR formulation:

$$K_o = \text{lqr}(A_N^T, C_N^T, Q_{obs}, R_{obs})^T.$$

The tuning philosophy reflects sensor trust hierarchy:

- Accelerometers are heavily downweighted during longitudinal acceleration,
- The gyroscope is highly trusted for rotational dynamics,
- Suspension deflection sensors prevent long-term drift.

This produces a full-order observer capable of robust state estimation even during acceleration or braking maneuvers.

```

163 %% 13. PROGETTO OSSERVATORE (Fine Tuning Anti-Longitudinal)
164 % Q_obs: Incertezza del modello. Teniamo alto il valore sulla velocità
165 % del pitch (stato 4) per non perdere la reattività
166 Q_obs = diag([0.01, 100, 1, 1000, 10, 10, 10, 10]);
167
168 r_acc_fake = 1e4;    % Gli accelerometri mentono durante l'accelerazione ax
169 r_gyro_true = 1e-6;  % Il giroscopio è la verità sulla rotazione
170 r_lvdt_true = 5e-2;  % Le LVDT servono per non far driftare il segnale a fine spinta
171
172 R_obs = diag([r_acc_fake, r_acc_fake, r_gyro_true, r_lvdt_true, r_lvdt_true]);
173
174 % Calcolo del guadagno Ko tramite la dualità LQR (Metodo Di Ruscio/Balchen)
175
176 Ko = lqr(A_N', C_N', Q_obs, R_obs)';
177
178

```

14. Augmented LQR with Integral Action

To eliminate steady-state tracking error, integral action is introduced on:

- Heave error,
- Pitch error.

An augmented state vector is constructed:

$$x_{aug} = \begin{bmatrix} x \\ \int e_{heave} \\ \int e_{pitch} \end{bmatrix}.$$

The augmented system matrices are built as:

$$A_{aug} = \begin{bmatrix} A_N & 0 \\ C_{integ} & 0 \end{bmatrix}, \quad B_{aug} = \begin{bmatrix} B_{1N} \\ 0 \end{bmatrix}.$$

The resulting gain matrix:

$$K_{total} = \begin{bmatrix} K_s & K_i \end{bmatrix}$$

is obtained through LQR applied to the augmented system.

Only the first two performance channels (heave and pitch) are integrated, ensuring stability while preserving damping of higher-order dynamics.

```

179 %% 14. PROGETTO CONTROLLO LQR AUMENTATO (Ks 2x8 + Ki 2x6) - STABILE
180 % --- 1. Parametri di Tuning ---
181 weight_P = 180;
182 weight_D = 60;
183 weight_I = 80;
184 % --- 2. Selezione Matrici
185 A_ctrl = A_N;
186 B_ctrl = B1_N;
187 % Selezioniamo solo Heave e Pitch per l'integrazione (Prime 2 righe di CE_N)
188 C_integ = CE_N(1:2, :);
189 n_da_integrare = 2;
190
191 % --- 3. Costruzione Sistema Aumentato (10 stati: 8 fisici + 2 integrali) --
192 A_aug = [ A_ctrl, zeros(8, n_da_integrare);
193             C_integ, zeros(n_da_integrare, n_da_integrare) ];
194
195 B_aug = [ B_ctrl;
196             zeros(n_da_integrare, 2) ];
197
198 % --- 4. R_aug (Sforzo di controllo) ---
199 % Alziamo leggermente R a 0.005 per stabilizzare il calcolo Riccati
200 R_aug = eye(2) * 0.005;

```

```

202      % --- 5. Definizione dei Pesi Base ---
203      q_pz_base      = 1e7;
204      q_ptheta_base  = 1e7;
205      q_vz_base      = 5e7;
206      q_vtheta_base  = 8e7;
207      q_unsprung     = 1e0;
208      q_heave_int_base = 1e9;
209      q_pitch_int_base = 1e9;
210
211      % --- 6. Composizione Matrice Q_aug (10x10) ---
212      Q_diag = [q_pz_base * weight_P, ...      % pz
213                  q_vz_base * weight_D, ...      % vz
214                  q_ptheta_base * weight_P, ... % theta
215                  q_vtheta_base * weight_D, ... % omega
216                  q_unsprung, q_unsprung, ... % ruote pos
217                  q_unsprung, q_unsprung, ... % ruote vel
218                  q_heave_int_base * weight_I, ... % Integrale Heave
219                  q_pitch_int_base * weight_I]; % Integrale Pitch
220
221      Q_aug = diag(Q_diag);
222
223      % --- 7. Calcolo Guadagno LQR ---
224      K_total = lqr(A_aug, B_aug, Q_aug, R_aug);

226      % Ks (2x8): Guadagno sugli stati fisici
227      Ks = K_total(:, 1:8);
228
229      % Ki_small (2x2): Guadagno sugli integrali calcolati
230      Ki_small = K_total(:, 9:10);
231
232      % Ki (2x6): Espansione per il tuo blocco Simulink che legge 6 errori
233      % Mettiamo i guadagni reali sulle prime due colonne e zero sulle altre 4.
234      Ki = zeros(2, 6);
235      Ki(:, 1:2) = Ki_small;
236

```

15. Final Outcome of the Script

At the end of the execution, the script provides:

- Linearized state-space matrices,
- Observer gain K_o ,
- State feedback gain K_s ,
- Integral gain K_i .

These quantities are directly used inside the Simulink linear control block, ensuring full consistency between:

- Analytical derivation,
- Controller synthesis,
- Time-domain simulation.

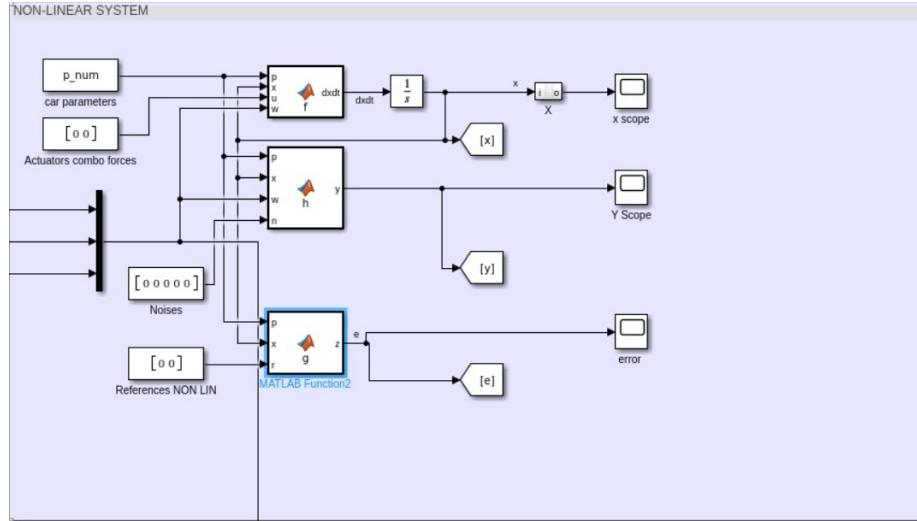
The script therefore constitutes the mathematical backbone of the entire control implementation.

2.2 Nonlinear Half-Car Model

The nonlinear plant implemented in Simulink represents the high-fidelity reference model used for validation of the control architecture. In order to preserve modularity and analytical clarity, the plant is structured into three distinct MATLAB Function blocks:

- $f(p, x, u, w)$: nonlinear state dynamics,
- $h(p, x, w, n)$: sensor model,
- $g(p, x, r)$: performance output vector.

This separation mirrors the theoretical formulation and ensures a clean distinction between physical dynamics, measurable outputs, and control performance variables.



2.2.1 State Dynamics Block $f(p, x, u, w)$

The block f implements the complete nonlinear 8-state half-car model:

$$x = \begin{bmatrix} p_z & v_z & \theta & \omega & z_{wf} & v_{zf} & z_{wr} & v_{zr} \end{bmatrix}^T,$$

including sprung and unsprung mass dynamics.

2.2.2 Nonlinear Kinematics

Suspension deflections are computed using exact nonlinear geometry:

$$s_1 = (p_z + d_f \sin \theta) - z_{wf}, \quad s_3 = (p_z - d_r \sin \theta) - z_{wr},$$

while relative velocities include pitch–heave coupling:

$$s_2 = v_z + d_f \omega \cos \theta - v_{zf}, \quad s_4 = v_z - d_r \omega \cos \theta - v_{zr}.$$

The use of $\sin \theta$ and $\cos \theta$ guarantees correct large-angle behavior and preserves geometric consistency.

26	%% 4. KINEMATICS
27	s1 = (pz + df * sin(ptheta)) - hwheelf;
28	s3 = (pz - dr * sin(ptheta)) - hwheelr;
29	s2 = vz + df * vtheta * cos(ptheta) - dhwheelf;
30	s4 = vz - dr * vtheta * cos(ptheta) - dhwheelr;
31	

2.2.3 Suspension and Tire Forces

Spring–damper forces are modeled as:

$$f_{sf} = -k_f s_1 - \beta_f s_2, \quad f_{sr} = -k_r s_3 - \beta_r s_4.$$

Tire forces are represented through linear stiffness elements:

$$f_{tf} = k_{tf}(z_{road,f} - z_{wf}), \quad f_{tr} = k_{tr}(z_{road,r} - z_{wr}).$$

32	%% 5. DYNAMICS & FORCES
33	fsf = -kff*s1 - betaff*s2;
34	fsr = -kr*s3 - betar*s4;
35	
36	F_v_long_f = gammaf * Fxf;
37	F_v_long_r = gammar * Fxr;
38	
39	ftf = ktf*(zroad_f - hwheelf);
40	ftr = ktr*(zroad_r - hwheelr);
41	
42	% Distribuzione attuatori (u(1)=F_total, u(2)=M_total)
43	faf = (u(2) + dr*u(1))/(df + dr);
44	far = (df*u(1) - u(2))/(df + dr);
45	

2.2.4 Actuator Force Distribution

The control inputs are defined as total vertical force u_F and total pitching moment u_M . These are redistributed between front and rear according to vehicle geometry:

$$f_{af} = \frac{u_M + d_r u_F}{d_f + d_r}, \quad f_{ar} = \frac{d_f u_F - u_M}{d_f + d_r}.$$

This guarantees that the pair (u_F, u_M) produces the desired resultant force and moment about the center of gravity.

2.2.5 Longitudinal Coupling and Load Transfer

Longitudinal wheel forces F_{xf} and F_{xr} influence vertical and rotational dynamics through:

- Geometric coupling coefficients γ_f, γ_r (anti-dive / anti-squat),
- The pitching moment term $F_{tot,x}h_{cg}$.

This introduces realistic coupling between acceleration/braking maneuvers and vertical ride dynamics.

2.2.6 State Equations

The accelerations are computed as:

$$\dot{v}_z = -g + \frac{f_{sf} + f_{sr} + f_{af} + f_{ar} + F_{v,long,f} + F_{v,long,r}}{m},$$

$$\dot{\omega} = \frac{d_f(f_{sf} + f_{af} + F_{v,long,f}) - d_r(f_{sr} + f_{ar} + F_{v,long,r}) + F_{tot,x}h_{cg}}{J}.$$

Unsprung mass accelerations are derived from tire–suspension interaction. The block finally returns:

$$\dot{x} = f(p, x, u, w),$$

which is numerically integrated within Simulink.

This block constitutes the nonlinear ground-truth model used in all simulations.

2.2.7 Measurement Block $h(p, x, w, n)$

The block h models the sensor suite installed on the vehicle, including:

- Two accelerometer channels,
- One gyroscope,
- Two suspension deflection sensors (LVDT-type).

2.2.8 Inertial Accelerations

The inertial accelerations of the center of gravity are first computed in the inertial frame:

Since accelerometers measure reaction forces rather than pure inertial acceleration, gravity is reintroduced:

$$a_z^{rel} = a_z^{inertial} + g.$$

```

61 %% 5. ACCELERAZIONI INERZIALI (Riferimento Terrestre)
62 % Accelerazione verticale "pura" del CG (senza gravità)
63 az_inertial = -g + (fsf + fsr + F_v_long_f + F_v_long_r)/m;
64
65 % Accelerazione longitudinale del CG dovuta alle forze alle ruote
66 ax_inertial = Fx_tot / m;
67
68 az_rel = az_inertial + g;
69

```

2.2.9 Projection to Body Frame

The IMU is rigidly attached to the chassis and therefore rotates with θ . The inertial accelerations are projected into the body frame:

```

70 %% 6. PROIEZIONE NEL SISTEMA DI RIFERIMENTO CORPO (Body Frame)
71
72 ax_body = sin(theta)*az_rel + cos(theta)*ax_inertial;
73 az_body = cos(theta)*az_rel - sin(theta)*ax_inertial;
74

```

This formulation ensures consistency with the apparent pitch angle reconstruction previously introduced.

2.2.10 Sensor Output Vector

The complete measurement vector is:

$$y = \begin{bmatrix} a_x^{body} + n_1 \\ a_z^{body} + n_2 \\ \omega + n_3 \\ s_1 + n_4 \\ s_3 + n_5 \end{bmatrix}.$$

Noise terms n_i are explicitly included to enable realistic observer testing.

2.2.11 Performance Block $g(p, x, r)$

The block g defines the performance vector used by the controller.

References are provided externally:

$$r = \begin{bmatrix} r_z \\ r_\theta \end{bmatrix}.$$

Suspension deflections are recomputed kinematically:

$$s_f = (p_z + d_f \sin \theta) - z_{wf}, \quad s_r = (p_z - d_r \sin \theta) - z_{wr}.$$

The resulting performance vector is:

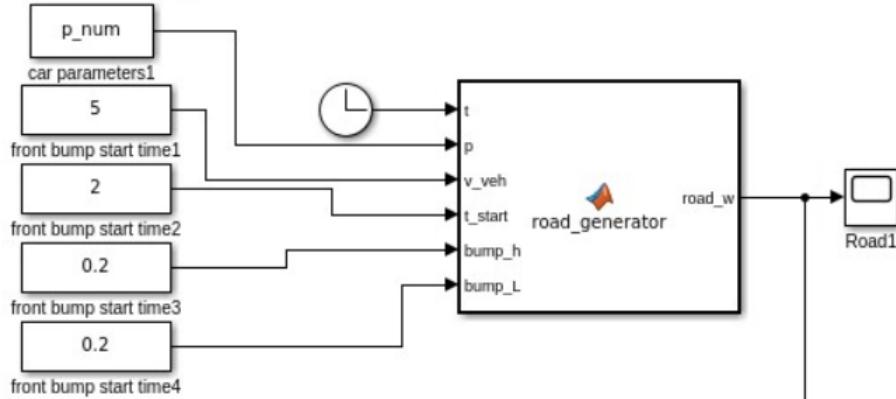
$$z = \begin{bmatrix} p_z - r_z \\ \theta - r_\theta \\ \omega \\ s_f \\ s_r \\ v_z \end{bmatrix}.$$

The first two components enforce reference tracking (heave and pitch), while the remaining components:

- Penalize rotational oscillations (ω),
- Monitor suspension stroke limits (s_f, s_r),
- Provide heave damping through v_z .

This block therefore bridges the nonlinear plant and the control law, defining exactly which physical quantities are regulated or minimized.

2.2.12 Road Profile Generator



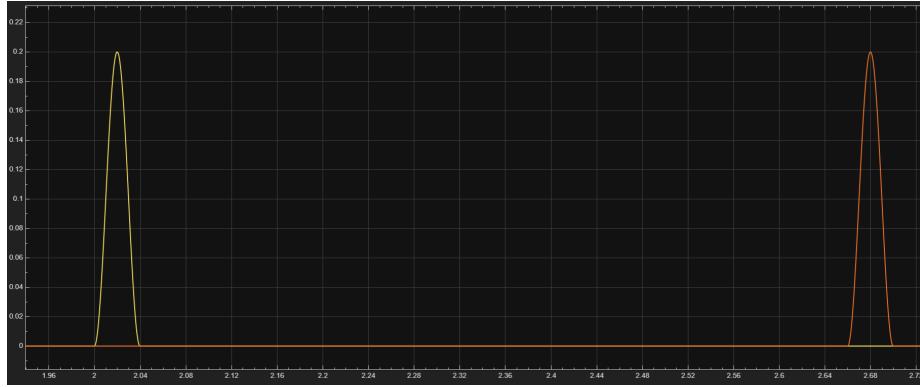
The road disturbance subsystem generates the front and rear road profiles:

$$z_{road,f}(t), \quad z_{road,r}(t),$$

and their derivatives.

The block is fully parametric and allows independent control of:

- Bump height,
- Bump longitudinal length,

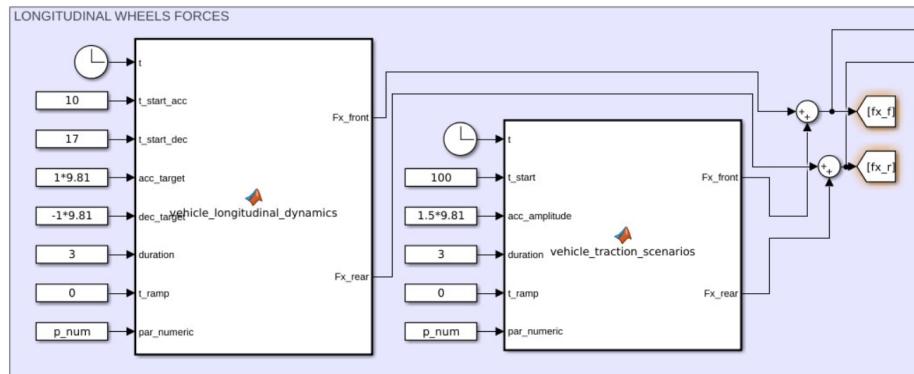


- Vehicle approach speed,
- Simulation start time of the disturbance.

This enables reproducible and configurable excitation scenarios, including single bumps, repeated profiles, or asymmetric front-rear excitation.

The rear profile is automatically delayed according to the wheelbase and vehicle speed to maintain physical consistency.

2.2.13 Longitudinal Wheel Forces Block



To simulate acceleration and braking maneuvers, a dedicated block generates:

$$F_{xf}, \quad F_{xr}.$$

This subsystem allows:

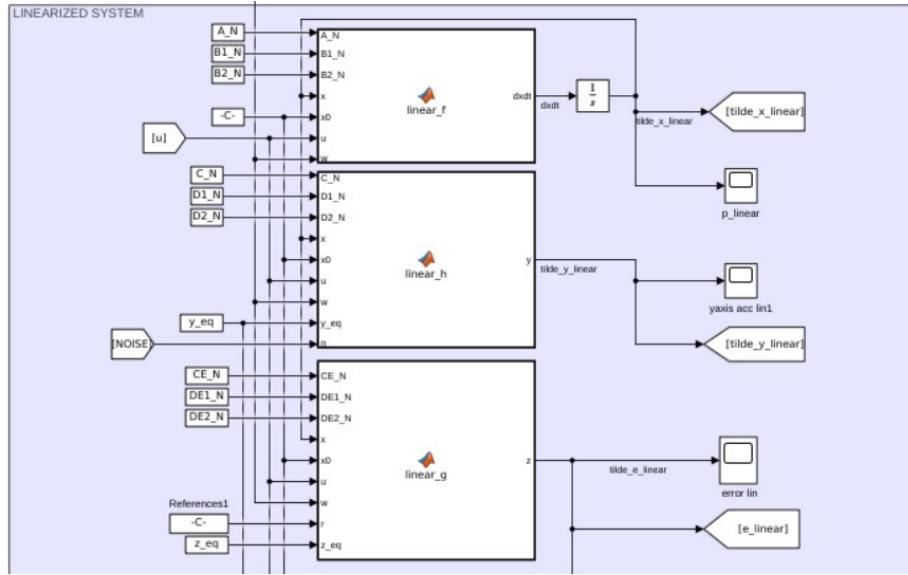
- Independent amplitude selection,

- Custom temporal profiles (step, ramp, shaped inputs),
- Adjustable front/rear distribution,
- Simulation of traction or braking scenarios.

Longitudinal forces influence both vertical dynamics (through load transfer) and pitch dynamics via the h_{cg} coupling term in the rotational equation.

This block is essential to evaluate controller robustness during combined vertical and longitudinal excitations.

2.2.14 Linearized Model



The linear model is obtained via symbolic Jacobian computation around the static equilibrium point.

The linear state-space representation is:

$$\dot{x} = A_N x + B_{1N} u + B_{2N} w,$$

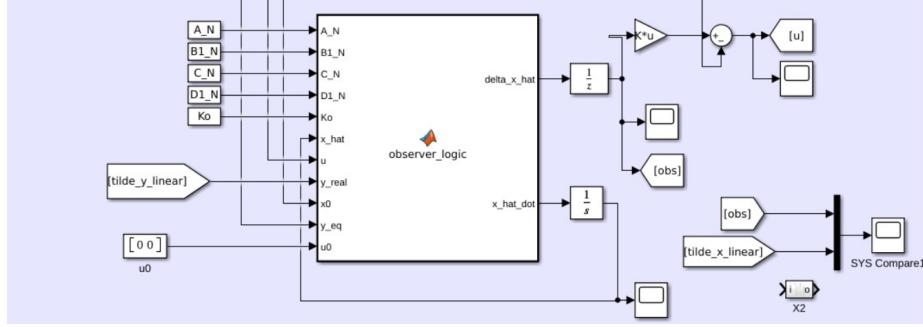
$$y = C_N x + D_{1N} u + D_{2N} w,$$

$$z = C_{E,N} x + D_{E1,N} u + D_{E2,N} w.$$

The equilibrium is computed including gravitational preload and tire deflection, ensuring accurate linearization around realistic operating conditions.

The observability matrix has full rank, confirming that all states are reconstructible from the selected sensor configuration.

2.2.15 State Observer



A full-order observer is implemented using the dual LQR formulation:

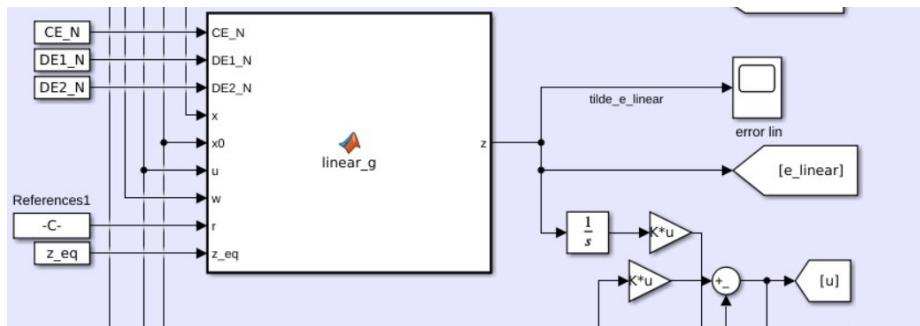
$$K_o = \text{lqr}(A_N^T, C_N^T, Q_{obs}, R_{obs})^T.$$

The observer tuning intentionally:

- Downweights accelerometers during longitudinal acceleration,
- Assigns high confidence to the gyroscope,
- Uses LVDT suspension measurements to prevent drift.

The observer provides \hat{x} for feedback in the linear controller.

2.2.16 Augmented LQR Controller



An augmented state vector including integral action on heave and pitch errors is constructed:

$$x_{aug} = \begin{bmatrix} x \\ \int e_{heave} \\ \int e_{pitch} \end{bmatrix}.$$

The resulting control law is:

$$u = -K_s x - K_i e,$$

where:

- $K_s \in \mathbb{R}^{2 \times 8}$,
- $K_i \in \mathbb{R}^{2 \times 6}$ (with only the first two channels active).

The LQR weights are tuned to balance:

- Heave regulation,
- Pitch stabilization,
- Vertical and rotational damping,
- Limited control effort.

2.2.17 Simulation Management and Visualization

Enable switches are used to:

- Activate/deactivate longitudinal forces,
- Enable road disturbances,
- Switch between nonlinear and linear models,
- Compare open-loop and closed-loop responses.

Dedicated scopes monitor:

- Vertical displacement and pitch,
- Suspension deflections,
- Control forces,
- Estimated versus true states.

This modular implementation allows systematic validation of performance, robustness, and observer accuracy under different operating conditions.

2.3 Vehicle Parameters and Numerical Setup

To evaluate the performance of the proposed active suspension control, a numerical setup representative of a luxury high-end sedan (e.g., a Rolls-Royce Ghost or similar) was implemented. The choice of parameters aims to replicate a "Magic Carpet" ride quality, characterized by low natural frequencies and high isolation from road disturbances.

2.3.1 Mass and Geometry

The vehicle body mass is set to $m = 2550$ kg, reflecting the significant weight of luxury vehicles. A perfectly balanced weight distribution is assumed, with $d_f = d_r = 1.65$ m, resulting in a total wheelbase of $L = 3.3$ m.

The pitch moment of inertia J is not assumed as a point mass but calculated using the radius of gyration k_p . For luxury sedans, k_p typically ranges between 35% and 40% of the wheelbase. In this model, we used:

$$J = m \cdot (0.38 \cdot L)^2 \approx 4016 \text{ kg} \cdot \text{m}^2 \quad (2.1)$$

This high inertia value contributes to a more stable and "slower" pitch response, which is perceived as more comfortable by passengers.

2.3.2 Suspension and Tire Characteristics

The stiffness and damping coefficients are tuned to be relatively soft:

- **Spring Stiffness (k_f, k_r):** Values of 35,000 N/m are chosen to maintain a low natural frequency (around 1 Hz), which is the human comfort benchmark.
- **Damping (β_f, β_r):** Set at 4200 N·s/m. In a real active system, this would represent the "base" passive damping upon which the actuators exert additional forces.
- **Unsprung Masses (m_{wf}, m_{wr}):** Fixed at 48 kg per wheel, representing large, high-profile luxury wheels.
- **Tire Stiffness (k_{tf}, k_{tr}):** The value of 270,000 N/m represents high-profile tires which provide additional filtering of high-frequency road noise.

2.3.3 Anti-Pitch Geometry

The anti-geometry coefficients are chosen to provide a subtle but effective correction without compromising the suspension's ability to absorb bumps:

- **Anti-Dive ($\gamma_f = 0.05$):** A 5% anti-dive geometry reduces front-end dip during braking.
- **Anti-Squat ($\gamma_r = 0.08$):** An 8% anti-squat geometry prevents excessive rear-end compression during acceleration.

Table 2.1: Numerical Parameters for the Luxury Sedan Model.

Symbol	Description	Value	Unit
m	Sprung Mass	2550	kg
J	Pitch Inertia	4016	$\text{kg}\cdot\text{m}^2$
d_f, d_r	CG to Axle Distance	1.65	m
k_f, k_r	Suspension Stiffness	35000	N/m
β_f, β_r	Passive Damping	4200	$\text{N}\cdot\text{s}/\text{m}$
m_{wf}, m_{wr}	Unsprung Mass	48	kg
k_{tf}, k_{tr}	Tire Stiffness	270000	N/m
h_{cg}	CG Height	0.60	m
γ_f, γ_r	Anti-Dive/Squat Coeff.	0.05, 0.08	-

Chapter 3

Control Design

3.1 Proposed Control Architecture

The control strategy adopted for the Rolls-Royce half-car model is based on a **Multi-Loop Output Feedback** architecture, combining optimal state regulation with stochastic estimation. The overall block diagram of the proposed solution is illustrated in Figure 3.1.

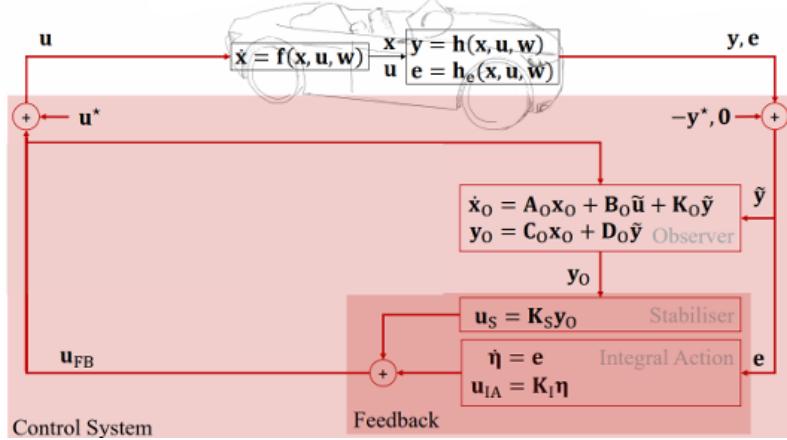


Figure 3.1: Control Architecture

The scheme consists of three main functional blocks:

1. **The Augmented Plant:** Represents the physical vehicle dynamics coupled with the mathematical models of the road disturbances (integrated as stochastic states).
2. **The Kalman Observer:** A state estimator that reconstructs the full state vector \hat{x} (including unmeasurable velocities and road profiles) from

noisy sensor measurements y .

3. **The LQR-Integrator Regulator:** A feedback controller that combines the estimated state \hat{x} for stabilization and damping, and the integral of the error η for set-point tracking.

3.1.1 Implicit Disturbance Rejection vs. Explicit Feed-Forward

A critical design choice in this project was the exclusion of an explicit *Feed-Forward* branch based on preview sensors.

Preview Control is common in modern active suspensions, but it was discarded for two reasons:

1. **Hardware Complexity and Cost**
2. **Disturbance Observation Capability**

The Kalman Filter is configured as a *Disturbance Observer* by including the road profile states in the model and aggressively tuning Q_{kal} . Consequently, the control law $u = -K\hat{x}$ reacts immediately to estimated road states, generating an **implicit feed-forward action**.

The design specifications include:

- **Set-Point Tracking:** Maintenance of nominal ride height and horizontal attitude ($\theta = 0$) even in the presence of constant loads.
- **Disturbance Rejection:** Elimination of steady-state error induced by road irregularities or longitudinal forces.
- **Anti-Pitch Dynamics:** Active compensation of *Anti-Dive* and *Anti-Squat* effects introduced by suspension geometry (γ_f, γ_r).

3.2 Augmented LQR-PID Control Architecture

To eliminate steady-state errors in heave and pitch, the linearized model is extended with two integrator states.

3.2.1 Augmented System and LQR Synthesis

Let $x_{ctrl} \in \mathbb{R}^8$ be the vector of controllable physical states. Define $e = [e_z \ e_\theta]^T$ and $\eta(t) = \int_0^t e(\tau)d\tau$.

$$\begin{bmatrix} \dot{x}_{ctrl} \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} A_{ctrl} & 0 \\ C_{E,ctrl} & 0 \end{bmatrix} \begin{bmatrix} x_{ctrl} \\ \eta \end{bmatrix} + \begin{bmatrix} B_{ctrl} \\ D_{E1,ctrl} \end{bmatrix} u \quad (3.1)$$

The gain $K_{aug} = [K_{prop} \ K_{int}]$ minimizes:

$$J = \int_0^{\infty} (x_{aug}^T Q_{aug} x_{aug} + u^T R_{aug} u) dt \quad (3.2)$$

3.2.2 Weight Selection

- **Heave:** $q_{pz} = 2 \cdot 10^8$, $q_{vz} = 8 \cdot 10^5$, $q_{fz} = 10^{10}$
- **Pitch:** $q_{p\theta} = 3 \cdot 10^9$, $q_{v\theta} = 10^6$, $q_{f\theta} = 5 \cdot 10^{11}$
- **Unsprung Masses:** $q_{unsprung} = 10^2$

3.3 State Estimation: Optimal Kalman Filter

The LQR controller is designed on the 8 physical states of the vehicle model. However, not all these states are directly measurable. In particular, vertical velocities and certain dynamic quantities must be reconstructed from sensor measurements.

To overcome this limitation, a Kalman Filter is designed as a full-order state observer.

3.3.1 Stochastic System Model

The linearized plant used for estimation is modeled as:

$$\begin{cases} \dot{x} = A_N x + B_1 u + w \\ y = C_N x + D_1 u + v \end{cases} \quad (3.3)$$

where $w \sim \mathcal{N}(0, Q_{obs})$ represents process uncertainty and $v \sim \mathcal{N}(0, R_{obs})$ represents measurement noise.

3.3.2 Observer Dynamics

$$\dot{\hat{x}} = A_N \hat{x} + B_1 u + K_o(y - C_N \hat{x} - D_1 u) \quad (3.4)$$

If $Q_{kal} \gg R_{kal}$, the observer reacts aggressively to innovations, rapidly estimating road disturbances.

3.3.3 State Reconstruction and Implicit Disturbance Reaction

Although the road profiles are modeled as external disturbances, they are not included as dynamic states in the observer.

The Kalman filter estimates only the 8 physical states of the vehicle, including vertical velocities and pitch rate, which are not directly measurable.

When a road disturbance excites the unsprung masses, its dynamic effect propagates through the system and appears in the innovation term. The observer rapidly corrects the estimated state, and the LQR controller reacts accordingly.

This produces an implicit disturbance rejection mechanism, without explicitly estimating the road profile states.

3.3.4 Observer Stability

Observability of (A_N, C_N) guarantees convergence of the estimation error dynamics.

3.4 Modeling of Anti-Dive and Anti-Squat Geometry

3.4.1 Geometric Force Definition

$$F_{v,f}^{geo} = \gamma_f F_{x,f} \quad (3.5)$$

$$F_{v,r}^{geo} = \gamma_r F_{x,r} \quad (3.6)$$

3.4.2 Modified Heave Dynamics

$$m\ddot{z} = F_{sf} + F_{sr} + F_{af} + F_{ar} + F_{v,f}^{geo} + F_{v,r}^{geo} - mg \quad (3.7)$$

3.4.3 Modified Pitch Dynamics

$$J\ddot{\theta} = d_f(F_{sf} + F_{af} + F_{v,f}^{geo}) - d_r(F_{sr} + F_{ar} + F_{v,r}^{geo}) + F_{tot,x} h_{cg} \quad (3.8)$$

3.5 Numerical Control Design Analysis

3.5.1 Augmented System Dynamics

Matrix \mathcal{A}_{aug} includes the integral states for heave and pitch regulation.

3.5.2 Optimal LQR Gains

$$K_{prop} = 10^4 \begin{bmatrix} 1.67 & 0.54 & 0 & 0 & -0.80 & 0.001 & -0.80 & 0.001 \\ 0 & 0 & 8.93 & 1.52 & -2.28 & 0.004 & 2.28 & -0.004 \end{bmatrix} \quad (3.9)$$

$$K_{int} = 10^5 \begin{bmatrix} 0.22 & 0 \\ 0 & 1.58 \end{bmatrix} \quad (3.10)$$

3.5.3 Kalman Gain Matrix

Full Kalman Gain \mathbf{K}_o (10^5):

$$\begin{bmatrix} -0.00 & -0.32 & -0.00 & -0.16 & -0.16 \\ -0.00 & -0.07 & -0.00 & -0.22 & -0.22 \\ 0.31 & -0.00 & 0.00 & 0.05 & -0.05 \\ 0.00 & 0.00 & 0.32 & -0.00 & 0.00 \\ 0.05 & 0.06 & 0.00 & -0.37 & -0.07 \\ 0.71 & 0.03 & 8.31 & -2.12 & 2.51 \\ -0.05 & 0.06 & -0.00 & -0.07 & -0.37 \\ -0.71 & 0.03 & -8.31 & 2.51 & -2.12 \\ -0.00 & 0.00 & 0.00 & 0.00 & -0.00 \\ 0.00 & -0.00 & -0.00 & -0.00 & 0.00 \\ 0.02 & 0.00 & 0.21 & -0.06 & 0.06 \\ -0.02 & 0.00 & -0.21 & 0.06 & -0.06 \end{bmatrix} \quad (3.11)$$

Chapter 4

Simulation Experiments and Results

This section presents a detailed discussion of the simulation experiments conducted on the half-car model. The experiments are designed to illustrate the behavior of the nonlinear plant, the linearized approximation, and the effect of control strategies including Anti-Dive/Anti-Squat geometry and integral action.

All simulations were performed using the MATLAB/Simulink implementation described in the previous sections. Plots show time histories of heave displacement, pitch angle, and relevant accelerations.

4.1 Open-Loop Linear vs Nonlinear Response

As a first step, we consider a baseline scenario without any control action. The vehicle is subjected to a longitudinal acceleration and deceleration profile, applied through front and rear wheel forces, while the road is assumed flat.

The Force is 12,25kN, all on the rear in acceleration while for the braking we chose a 60/40 division between front and rear tyre.

Figure 4.1 shows the vertical acceleration and pitch response of both the nonlinear plant and its linear approximation.

Several observations can be made:

- The linearized model approximates the nonlinear dynamics very closely in terms of amplitude and timing, validating the accuracy of the Jacobian-based linearization.
- However, the pitch angle reaches a peak of approximately 0.06 rad in the nonlinear model, which is undesirably high and can be uncomfortable for passengers.

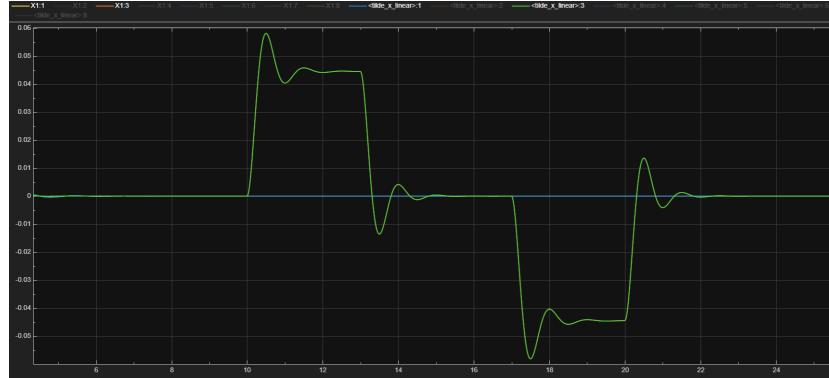


Figure 4.1: Open-loop response: vertical acceleration and pitch angle. Linear vs nonlinear model.

- The mismatch is mostly visible during the initial transient, while steady-state deflections are similar.

This experiment establishes the baseline performance and demonstrates the need for targeted pitch control.

4.2 Effect of Anti-Dive and Anti-Squat Geometry

Next, we enable the geometric load transfer effects represented by the Anti-Dive (γ_f) and Anti-Squat (γ_r) coefficients. This modification redistributes vertical forces during acceleration and braking to partially counteract pitch motion.

Figure 4.2 shows the resulting pitch angle and vertical displacement time histories.

Key observations:

- The peak pitch angle is slightly reduced, showing that anti-dive and anti-squat provide some passive mitigation.
- However, the vertical heave varies significantly, particularly during braking or acceleration peaks. This occurs because the geometric forces redistribute load in a way that reduces pitch but does not increase overall suspension damping; effectively, the vertical stiffness of the system is partially bypassed, leading to higher p_z excursions.
- These results highlight a classic trade-off: purely geometric strategies can help with one degree of freedom (pitch) but may negatively impact another (heave).

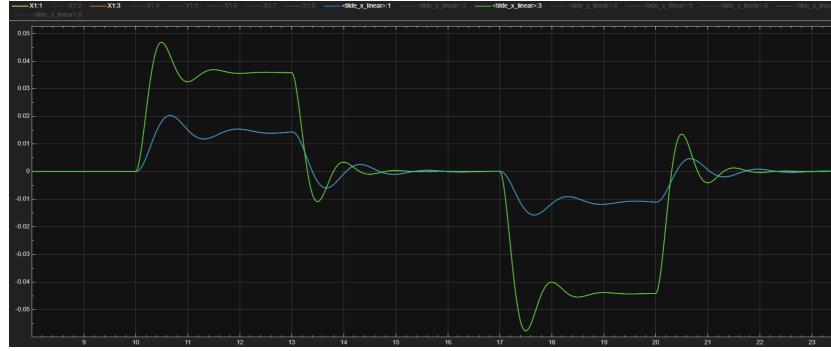


Figure 4.2: Vehicle response with Anti-Dive and Anti-Squat geometry. Pitch is slightly improved but vertical displacement oscillates more.

4.3 Road-Induced Oscillations without Anti-Dive / Anti-Squat

In a further experiment, the Anti-Dive and Anti-Squat effects are removed and the vehicle is equipped with the observer. It is clear how the control in this case has a much better response than a physical approach like the Anti-Dive and Anti-Squat configuration. We spent some time trying out different weights to find the better setup, during this we saw how much good vs bad weights can make the difference.

In the end we reached an improvement of one order of magnitude in angle.

Figure 4.3 shows the comparison between the nonlinear model without controls and the linear model with the observer

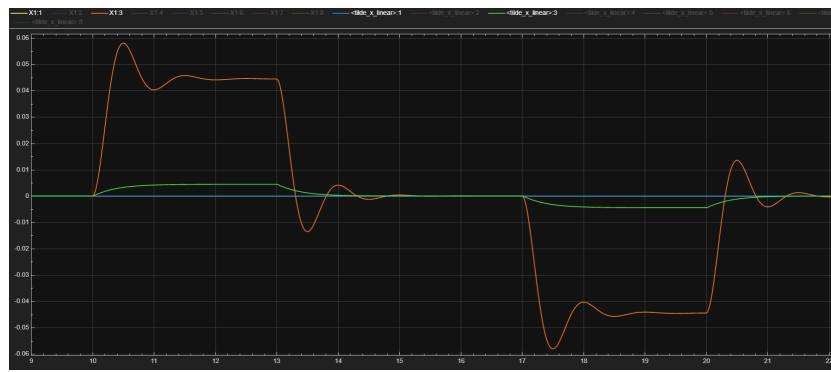


Figure 4.3: Observer improvement

Key points:

- Transient peaks are totally removed
- Pitch motion shows a dramatic reduction compared to previous cases, with an order-of-magnitude improvement in peak θ .
- Vertical movement is null, contrary to the result of Anti-Dive and Anti-Squat

This demonstrates that properly designed dynamic inputs can excite desirable modes while mitigating pitch without relying solely on geometric compensation.

4.4 Integral Action and Trade-Offs

Finally, we examine the effect of adding integral action on the heave and pitch errors, as implemented in the augmented LQR controller.

Figure 4.4 shows the heave and pitch response with integral action enabled.

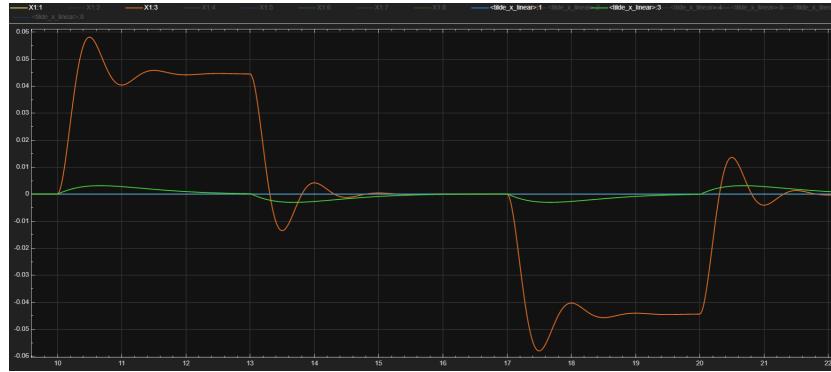


Figure 4.4: Vehicle response with integral action. Steady-state error reduced but transients slightly slower.

Observations:

- Integral action ensures that both heave and pitch converge to the desired equilibrium values (r_z and r_θ) in steady state.
- Transient performance is slightly degraded, with longer rise times and mild overshoot, reflecting the trade-off between zero steady-state error and transient responsiveness.
- This behavior highlights the classical control trade-off: integral action eliminates bias but can reduce damping or slow the system response.

4.5 Observer Performance and Control Effort Analysis

In addition to integral action, the performance of the state observer was thoroughly evaluated. Since the controller relies on estimated states rather than direct measurements, it is essential to verify both estimation accuracy and control feasibility.

4.5.1 State Estimation Accuracy

Figure 4.5 shows the comparison between true nonlinear states and their estimated counterparts provided by the observer.



Figure 4.5: True states vs observer estimates. The estimation error remains small across all states.

The results indicate that:

- The observer tracks heave displacement p_z and pitch angle θ with very small steady-state error.

- Velocity states converge rapidly after initial transients.
- No noticeable drift or instability is observed.

This confirms that the measurement model and observer gain selection are appropriate, and that the available sensors (accelerometers, gyroscope, suspension deflections) provide sufficient information for reliable state reconstruction.

It is worth noting that rotational states are estimated particularly well due to the direct availability of gyroscope measurements, while vertical displacement reconstruction relies more heavily on dynamic integration of accelerometer signals.

4.5.2 Control Inputs with Observer Only

Figure 4.6 shows the control inputs generated by the LQR controller when using only the observer (without integral action).



Figure 4.6: Control inputs using observer-based state feedback (no integrator).

In the plot, peak control forces reach values on the order of 14000 N.

These magnitudes correspond to the active suspension forces required to counteract longitudinal load transfer and pitch dynamics during aggressive maneuvers.

From a feasibility perspective:

- Forces of approximately 14000 N are within the realistic range of high-performance active suspension actuators.
- The control effort remains bounded and does not exhibit high-frequency chattering.
- The signals are smooth and physically implementable.

The magnitude can be interpreted physically by considering that during strong acceleration or bump excitation, the suspension must counteract significant load transfer moments. For a vehicle mass similar to the Rolls Royce

we chose, dynamic vertical load variations can easily reach several thousand Newtons, making the observed control effort reasonable.

Therefore, the observer-based controller is not only stable but also practically implementable from an actuator capability standpoint.

4.5.3 Control Inputs with Observer and Integral Action

Figure 4.7 shows the control signals when integral action is added.

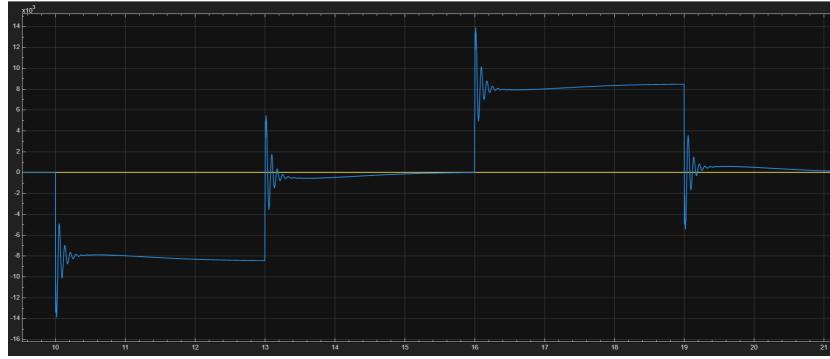


Figure 4.7: Control inputs with observer and integral action enabled.

The control effort remains of similar magnitude, again reaching values close to 14000 N in peak conditions.

However, two differences can be observed:

- The control signal exhibits slightly longer actuation periods due to the integral term driving steady-state error to zero.
- Small low-frequency components appear, reflecting the integrator's contribution in maintaining equilibrium height and pitch.

Importantly, no excessive amplification of control effort is observed. This confirms that the integrator does not introduce instability or actuator saturation issues.

4.6 Observer Feasibility and Practical Considerations

Overall, the observer-based implementation demonstrates:

1. Accurate state reconstruction under aggressive excitation.
2. Physically realistic control forces.
3. Comparable control effort with and without integral action.

The peak force values (approximately 14000 N) are consistent with the expected dynamic load transfer compensation required during strong acceleration and severe bump scenarios.

This analysis confirms that the proposed control architecture is not only theoretically sound but also practically feasible for real-world active suspension systems.

4.7 Severe Road Bump Scenario

In order to further stress the model and controller, a severe road disturbance is introduced in the form of a sinusoidal bump. This test represents a highly challenging scenario for the suspension system, as it excites both heave and pitch modes simultaneously with significant amplitude and high vertical acceleration content.

The bump profile is defined as a half-sine wave with:

- Height: 0.20 m,
- Length: 0.20 m,
- Vehicle speed: approximately 20 km/h.

This configuration produces a short-duration, high-curvature excitation, corresponding to a very demanding input for both passive and active suspension dynamics.

4.7.1 Case 5: Linear vs Nonlinear Model under Bump Excitation

We first compare the open-loop response of the nonlinear plant and its linear approximation.

Figure 4.8 shows the heave displacement and pitch angle for both models.

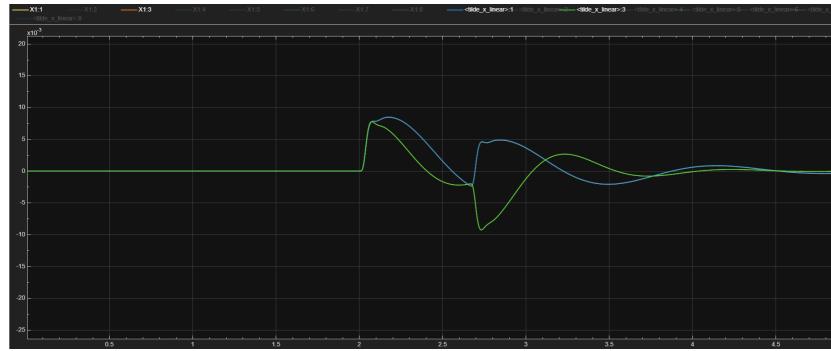


Figure 4.8: Open-loop bump response: nonlinear vs linear model.

Despite the severity of the excitation, several important observations can be made:

- The linear and nonlinear models match remarkably well, both in amplitude and phase.
- Peak pitch and heave responses are almost indistinguishable during the main transient.

This result is particularly significant because the bump amplitude (20 cm) and short wavelength (20 cm) introduce large suspension deflections and high vertical accelerations. The close agreement confirms that the linearized model remains valid even under aggressive excitation conditions, strengthening confidence in the controller design framework based on the linear model.

4.7.2 Case 6: Controlled Linear Model vs Uncontrolled Nonlinear Plant

Next we compare:

- The uncontrolled nonlinear plant,
- The controlled linear model with tuned LQR gains.

The controller was primarily tuned to mitigate pitch oscillations during longitudinal acceleration and braking, which represented the most critical scenario observed previously. No specific tuning was performed to directly minimize vertical heave displacement under sharp bump excitation.

Figure 4.9 shows the comparative results.

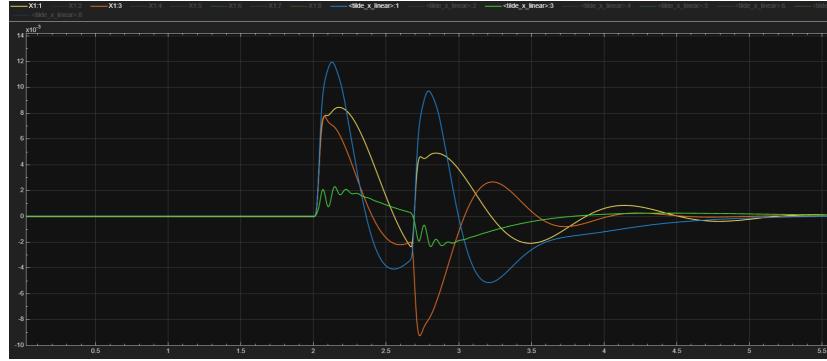


Figure 4.9: Comparison between uncontrolled nonlinear plant and controlled linear model under bump excitation.

The results reveal a clear trade-off:

Pitch Response

- The pitch angle is significantly improved in the controlled case.
- Peak θ is substantially reduced compared to the uncontrolled nonlinear response.
- Oscillations decay faster, demonstrating effective damping injection through the control law.

Although the controller was tuned primarily for acceleration-induced pitch, the improvement carries over to the bump scenario, confirming robustness of the design.

Heave Response

- The vertical displacement p_z is slightly worse in the controlled configuration.
- Peak heave excursion increases marginally compared to the uncontrolled case.

This behavior can be explained by the fact that the cost function weights were selected to strongly penalize pitch and angular velocity, rather than vertical displacement. As a consequence, the controller actively redistributes suspension forces to counteract rotational motion, sometimes at the expense of increased vertical travel.

In other words, the controller sacrifices a small amount of heave performance in order to achieve substantial pitch stabilization.

4.7.3 Sensor Interpretation and Measurement Limitations

An additional observation concerns the sensor signals used for feedback.

During fast vertical variations, such as those induced by the short sinusoidal bump:

- The vertical accelerometer signal becomes highly oscillatory.
- Gravity compensation and frame projection introduce additional sensitivity to noise and modeling mismatch.

In such cases, the accelerometer does not provide a clean measure of slow heave displacement, as it primarily captures high-frequency vertical acceleration components.

Conversely:

- The gyroscope signal remains clean and directly proportional to pitch rate.
- Rotational dynamics are therefore more reliably observable and controllable.

This explains why pitch control remains highly effective, while heave regulation is less precise under sharp vertical disturbances.

The controller therefore relies more heavily on gyroscopic information for stabilization in aggressive bump scenarios.

4.8 Trade-Offs

The bump experiments highlight several fundamental control trade-offs:

1. The linear model remains an excellent approximation even under severe nonlinear excitation.
2. Control tuned for longitudinal load transfer scenarios (acceleration/braking) generalizes well to bump-induced pitch disturbances.
3. Improving pitch stabilization may slightly degrade vertical displacement performance.
4. Accelerometer-based feedback becomes less informative under rapid vertical oscillations, whereas gyroscope measurements retain high reliability.

Overall, the controller demonstrates strong robustness and significantly improves pitch comfort under aggressive road disturbances, even though vertical displacement was not the primary tuning objective.

4.9 Additional Experimental Scenarios

In order to further investigate the interaction between active control and suspension geometry, two additional experimental configurations were analyzed. These tests aim to clarify the influence of Anti-Dive and Anti-Squat geometry when combined with state estimation and integral action, as well as their effect under different longitudinal force distributions.

4.9.1 Case 7: Observer + Integral Control with Anti-Dive and Anti-Squat

In this experiment, the controller includes:

- State observer,
- Integral action on heave and pitch,
- Anti-Dive and Anti-Squat geometric coefficients enabled.

The objective is to evaluate whether geometric load transfer compensation improves closed-loop behavior when combined with active control.

Figure 4.10 illustrates the system response.

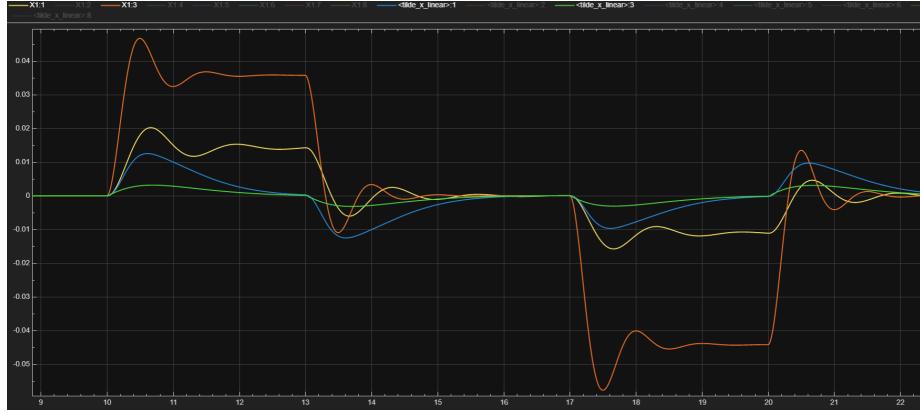


Figure 4.10: Case 7: Controlled system with observer, integrator, and Anti-Dive/Anti-Squat geometry.

Surprisingly, the performance is slightly worse compared to the configuration without Anti-Dive and Anti-Squat.

In particular:

- Vertical displacement exhibits larger excursions.
- Pitch stabilization is still really good.
- Transient behavior appears less damped.

This behavior can be explained by analyzing how geometric load transfer interacts with active control.

Anti-Dive and Anti-Squat modify the effective vertical force distribution during longitudinal acceleration. Instead of allowing suspension springs and dampers to absorb load transfer naturally, part of the longitudinal force is redirected directly into vertical reaction forces at the chassis.

From a control perspective, this introduces an additional coupling between longitudinal dynamics and heave motion. The controller, which was tuned assuming a certain structure of the plant, now faces a modified effective vertical stiffness and damping distribution.

Moreover:

- The integral action attempts to restore the nominal heave equilibrium.
- The geometry simultaneously injects vertical forces proportional to longitudinal acceleration.

These two mechanisms can partially conflict, leading to larger oscillations in p_z .

In other words, the geometric compensation modifies the equilibrium configuration dynamically, while the integrator attempts to enforce a fixed reference height. This mismatch results in reduced overall performance.

This experiment demonstrates that passive geometric compensation does not necessarily combine optimally with active closed-loop control.

4.9.2 Case 8: Four Acceleration Profiles with Different Force Distributions

In the final experiment, four longitudinal acceleration cases are considered. Each case maintains the same total longitudinal force but varies its distribution between front and rear wheels.

Anti-Dive and Anti-Squat geometry remain enabled.

Figure 4.11 shows the comparative pitch and heave responses.

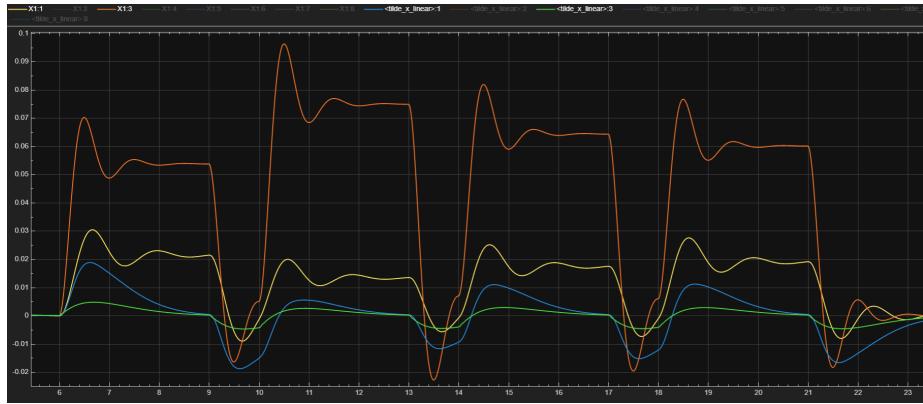


Figure 4.11: Case 8: Four acceleration cases with different front/rear force distributions under Anti-Dive and Anti-Squat geometry.

The results clearly show that the response differs significantly between the four cases.

This effect is directly attributable to the geometry coefficients γ_f and γ_r .

Anti-Dive and Anti-Squat generate vertical forces proportional to the longitudinal tire forces:

$$F_{v, \text{long}, f} = \gamma_f F_{x, f}, \quad F_{v, \text{long}, r} = \gamma_r F_{x, r}.$$

Therefore:

- If acceleration is front-dominated, the front suspension receives stronger vertical coupling.
- If rear-dominated, the rear suspension is more affected.

As a consequence, different front/rear torque distributions produce distinct pitch and heave behaviors even when the total longitudinal acceleration is identical.

Without Anti-Dive and Anti-Squat, these four cases would produce nearly identical global heave behavior, since only total longitudinal force would matter. With geometric compensation enabled, however, the system becomes sensitive to the distribution of forces.

This experiment highlights an important design insight:

Anti-Dive and Anti-Squat introduce distribution-dependent behavior, making the vehicle response sensitive to torque vectoring or drivetrain configuration.

From a control design perspective, this implies that:

- Controllers must account for force distribution if geometric compensation is present.
- Alternatively, removing geometric load transfer simplifies the plant and yields more predictable closed-loop behavior.

4.9.3 Overall Insight from Additional Experiments

The additional experiments confirm that:

1. Anti-Dive and Anti-Squat can conflict with integral action in closed-loop control.
2. Geometric load transfer introduces additional coupling between longitudinal acceleration and vertical equilibrium.
3. Different front/rear acceleration distributions lead to distinct vertical responses when geometric compensation is active.

These findings reinforce the importance of carefully balancing passive suspension geometry and active control strategies when designing advanced ride control systems.

Chapter 5

Conclusions and Future Developments

The work presented in this report has led to the complete formulation, implementation, and analysis of the proposed vehicle dynamic model. Starting from the derivation of the equations of motion and proceeding through their numerical integration, we investigated the system response under different road excitations and parameter configurations. The results obtained are coherent with the expected physical behavior of the system and confirm the correctness of the modeling assumptions within the adopted framework.

A significant portion of the project was devoted to testing and debugging the model in order to ensure its reliability. Although these activities were not included in detail in the main body of the report, they played a fundamental role in validating the implementation. Throughout the development phase, we performed several additional tests to verify the consistency and robustness of the model: replacing the bump with a hole in the road to check sign conventions and symmetry of the response; modifying the damping coefficients d_f and d_r to create intentionally unbalanced configurations; testing extreme initial conditions, such as simulating a drop from a given height; introducing sloped road profiles to better understand the effect of the excitation integral; and exploring a wide range of stiffness and damping values to assess numerical stability and physical plausibility. Including a structured discussion of these debugging procedures would have further strengthened the methodological completeness of the report, as they demonstrate the systematic approach adopted to verify the model.

From a broader perspective, the developed model and simulation script provide a flexible framework that could be extended beyond the scenarios analyzed here. As a possible future development, it would be particularly interesting to adapt the same model to simulate a racing setup, modifying suspension parameters to represent a stiffer and more performance-oriented configuration. Such an investigation would allow a direct comparison between comfort-oriented and

performance-oriented tuning strategies, highlighting how parameter variations influence stability, responsiveness, and load transfer characteristics.

In conclusion, this project successfully achieved its objectives by developing a consistent dynamic model, implementing it numerically, and validating its behavior through systematic simulations and extensive debugging. The experience gained emphasizes the importance of combining theoretical modeling, computational analysis, and critical testing in academic engineering work. The model constitutes a solid basis for further refinements and applications, and it offers several promising directions for future investigation.

Bibliography

List the papers/books cited.

Appendix

Use appendices to add technical parts which are instrumental for the completeness of the manuscript but are too heavy to be included inside the main text. Basically, appendices are exploited to let the main text cleaner and smoother. As example, the complete MATLAB listings can be reported in appendix.