



Budapest University of Technology and Economics
Faculty of Transportation Engineering and Vehicle Engineering
Department of Control for Transportation and Vehicle Systems

MASTER'S THESIS

**LMI Based LPV/ H_∞ Gain Scheduled Control of
Semi-Active Suspension with Magnetorheological
Damper**

Author:

Sergen Güler

2020

Contents

1. Introduction	5
2. Control-Oriented Model of MR Damper	6
2.1. Passivity Constraint Passive Suspensions	6
2.2. Damper Model	8
2.2.1. Static Guo-Model	8
2.2.2. Control Oriented Guo Model	9
2.2.3. Model Constraints.....	9
3. Suspension System Effects on Vehicle Dynamics	11
3.1. Quarter Car Model for Suspension System Analysis.....	11
3.2. Sprung Mass Vertical Acceleration	13
3.3. Unsprung Mass Displacement	14
3.4. Comfort/Road Holding Trade-Off	14
4. Performance Objectives for Semi-Active Suspension Control	15
4.1. Comfort Control Goal.....	15
4.2. Road Holding Control Goal.....	16
5. Linear Parameter Varying (LPV) Systems.....	17
5.1. Affine Parameter Dependent Model	18
5.2. Polytopic Models.....	20
5.3. LPV Model for Semi-Active Suspension	21
5.3.1. Selection of Scheduling Parameters, Measurement Outputs and Control Outputs	21
5.3.2. State Space Model	22
5.3.3. Dissipativity and Saturation Constraint	24
5.3.4. Reformulation of State Space Model with Constraints	26
5.3.5. Creating Semi-Active Suspension Polytopic Model with Matlab	28
6. Nominal LPV model of Semi-Active Suspension	34
6.1. Frequency Response of Nominal Damper Force	36
7. LMI Based Gain Scheduled LPV/ H^∞ Control of Semi-Active Suspension.....	38
7.1. H^∞ Control.....	38

7.2.	Riccatti and LMI Based Approaches	39
7.3.	Quadratic Lyapunov Functions	41
7.4.	LMI Formulation.....	42
7.5.	Robust Gain Scheduled Control	44
7.6.	H^∞ Problem and Control Strategy of Semi-Active Suspension with MR Damper 47	
7.7.	Selection of Weighting Functions with Genetic Algorithms	51
7.7.1.	Genetic Algorithms	52
7.7.2.	Implementation of Genetic Algorithms	53
7.7.3.	Implementation of LPV/ H^∞ Gain Scheduled Controller in MATLAB/ Simulink62	
8.	Frequency Domain Results of LPV/ H^∞ Semi Active Suspension.....	71
9.	Time Domain Results of LPV/ H^∞ Semi Active Suspension.....	72
9.1.	Bump Test	72
9.2.	Sinus Wave Source Test	78
9.3.	ISO Road Level Test	83
10.	Conclusion.....	91
	References	92

1. Introduction

Automobiles are exposed to a broad spectrum of vibrations due to road roughness, engine, power transmission equipments, tire/wheel connection when they travel. Road roughness influences the vehicle dynamics higher than other vibration sources. Two main vehicle dynamics that affect ride comfort and road holding are vertical acceleration and tire deflection respectively. Suspension systems are used for vibration isolation and maintain or improve vehicle performance along with travel time. Suspension systems consist of an elastic element and damping element to dissipate the energy comes from vibration sources. There are 3 types of suspension system

- Passive suspension: The property of this type of suspension system is time-invariant and can dissipate only energy.
- Semi-active suspension: Their damping coefficient can be adjusted by external signals and they are different from passive suspension from this point. However, they can only dissipate the energy like passive suspensions.
- Active suspension: They can both dissipate and generate the energy via external actuator can supply energy into the system.

The main difference between semi-active and active suspension is that semi-active suspension damping coefficients can be changed and control without putting any external energy in system. They consist of controllable dampers such as MR and ER. These dampers include special fluid whose viscosity characteristic can be change in magnetic and electric field respectively by applying electric current. This property makes them controllable. Main objective of suspension design is improving ride comfort and road holding. Passive dampers only achieve compromise between ride comfort and road holding while active and semi-active dampers can improve both performance objectives but latter suspension is better in terms of cost and performance compromise and widely used in automotive industry.

In this study, semi-active suspension with MR damper is used to improve ride comfort and road holding performance. MR dampers have non-linear character due to their bi-viscous and hysteric behaviours. Static-Guo model is used by considering their non-linear character and transform to control oriented formulation. This model gives more realistic approach than other proposed models. Non-linear control oriented model is recasted in the LPV system and the problem of dissipation in the control architecture is taken into consideration with the input saturation. LMI based H_∞ gain scheduled controller is applied on LPV system in order to improve ride comfort and road holding. Selection of weighting functions is the key part of implementation this type of controller. Weighting functions are found by using Genetic algorithms for multi-objective problem of H_∞ controller to achieve demanded performance objectives of quarter car equipped with semi-active suspension.

2. Control-Oriented Model of MR Damper

Magneto-Rheological (MR) dampers have become one of the most researched devices in many studies on semi-active suspension control. They use MR fluids that can differ depending on the magnetic field generated by electric current. MR dampers have many advantages than correspond semi-active dampers such as ER or friction dampers. This form of damping offers a fast response and reacts hysterically over a wide range of temperatures as well as low battery voltage usage. They are used in many applications like damping devices, actuators, breaks, shock absorbers in automotive industry.

Below figure shows the MR damper working principle. $B=0$ means that there is not any magnetic field on fluid while $B>0$ shows that magnetic field on fluid. [1]

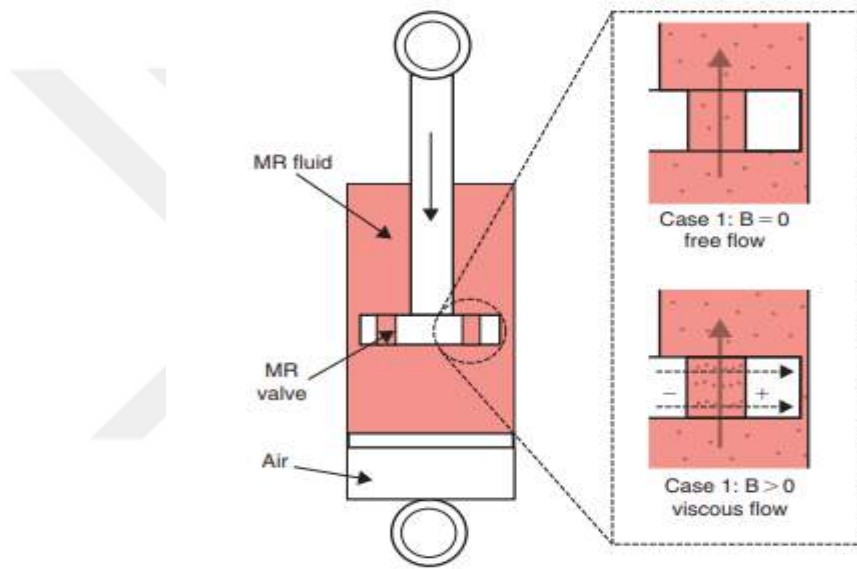


Figure 1: MR Damper Layout (e.g. [1])

2.1. Passivity Constraint Passive Suspensions

Passivity constraint defines that the elements in the suspension system can not introduce energy into the system. Elastic element coil spring can only store the energy getting from the road profile and damper dissipates it.

According to quarter car model Figure 2, the passivity constraint can be defined by

$$\begin{aligned} F_k x &\geq 0 \\ F_d \dot{x} &\geq 0 \\ x &= -(z - z_t - L) \end{aligned} \quad (2.1)$$

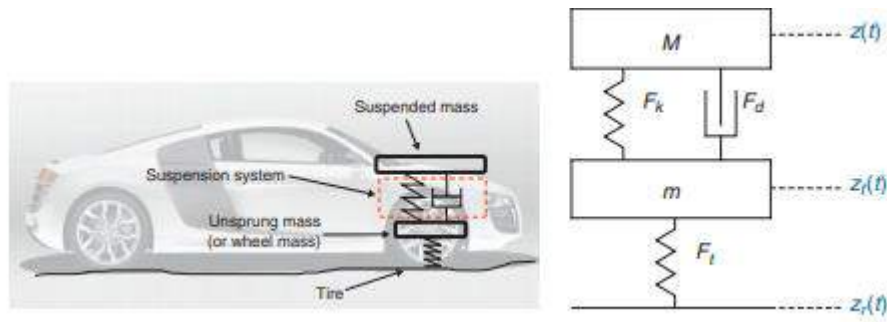


Figure 2: Quarter-car Model of Suspension System In a Vehicle (e.g. [1])

where x is the suspension deflection which is negative in case of suspension extension and is positive in case the suspension is compressed. L is the length of elastic element like coil spring in the suspension. When $x=0$, $F_k(t)=0$.

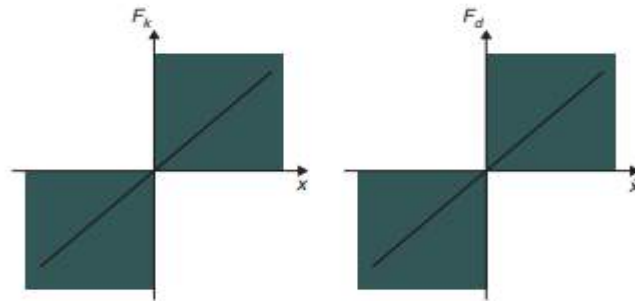


Figure 3: Passivity Constraint of Passive Damper (e.g. [1])

Semi-active dampers can work in two quadrants of force- deflection velocity plane like passive suspension systems while active suspension is able to work on four quadrants. It defines their dissipativity constraint. They change their viscosity depending on magnetic field is occurred on damper fluid. They do not take any external force from any source. Force is naturally changed in their system depending on the viscosity while active suspensions take external force to provide needed conditions of system. It is the main difference between active and semi-active suspensions.

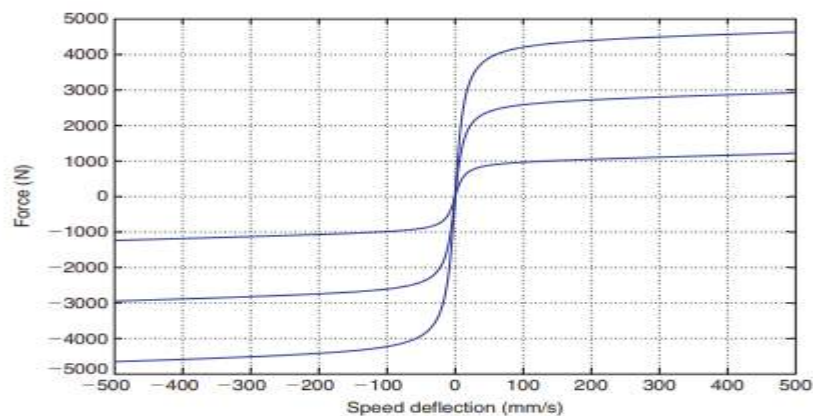


Figure 4: Ideal Damping Characters of MR Shock Absorber (e.g. [1])

2.2. Damper Model

Modelling of semi-active dampers is hard problem due to their nonlinear like bi-viscosity, hysteresis and temperature dependency. There are some most known and common mathematical model to define semi-active dampers.

- Dynamic Bouc-Wen Model
- Static Model with Coulomb friction
- Static Guo Model
- Non-Linear black box model

In this study, Static Guo-Model is taken to implement control method on it.

2.2.1. Static Guo-Model

Non-linear equation model of the semi-active damper is given by

$$(2.2) \quad F_{damper} = a_2 \left(\dot{x}_{mr} + \frac{v_0}{x_0} \right) x_{mr} + a_1 \tanh \left(a_3 \left(\dot{x}_{mr} + \frac{v_0}{x_0} x_{mr} \right) \right)$$

In this equation, a_1 represents dynamical yield force of the MR fluid, a_2 and a_3 are post and pre yield viscous damping coefficients respectively and x_0 denote the absolute value of hysteretic critical velocity $\dot{x}_0$ and hysteretic critical deflection x_0 , where $\dot{x}_0$ and x_0 are defined suspension deflection velocity and deflection resoectively when the applied MR damper force is zero. The model provides dissipativity constraint and works in only 2 quadrant. Model is formulated depending on a tangent hyperbolic function and gives realistically the hysteresis and bi-viscous characteristic of a damper. [2]

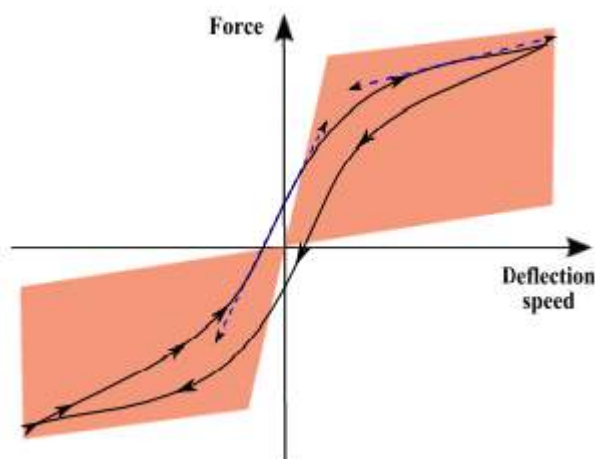


Figure 5: Realistic MR Damper Force (e.g. [2])

2.2.2. Control Oriented Guo Model

Model (2.2) provides the real MR damper behaviour very well and satisfy passivity (e.g. Figure 4). constraint of damper. On the other hand, it is unnecessarily complex for control design. Simpler control-oriented model can be implment depends on the input signal. According to authors [3] and [4], control-oriented model for semi-active can be given by

$$(2.3) \quad F_{damper} = c_0 \dot{x}_{mr} + k_0 x_{mr} + f_I \tanh(c_1 \dot{x}_{mr} + k_1 x_{mr})$$

Where f_I is the controllable force coefficient which varies according to electrical current I in the coil and f_I is simply described by [5]

$$(2.4) \quad f_I = y_{mr} I$$

2.2.3. Model Constraints

Semi-active dampers have some limit to satisfy their passivity constrain. Semi-Active dampers only work on 2 quadrant in force-deflection velocity plane as showing in previous chapter. Guo-Model satisfy this limitation and keep damper force in these 2 quadrant (e.g. Figure 5). According to model and suspension limitations, model has 2 basic constraints:

- Dissipativity Constraint
- Saturation Constraint

According to equation (2.2), F_{damper} has constant c_0, c_1, k_0, k_1 parameters which are damping coefficients and elastic coefficients while f_I is the controllable force coefficient which is varying according to electrical current I in coil. Therefore, f_I has positivity constraint to satisfy semi-activeness of damper

$$(2.5) \quad 0 \leq f_{Imin} \leq f_I \leq f_{Imax}$$

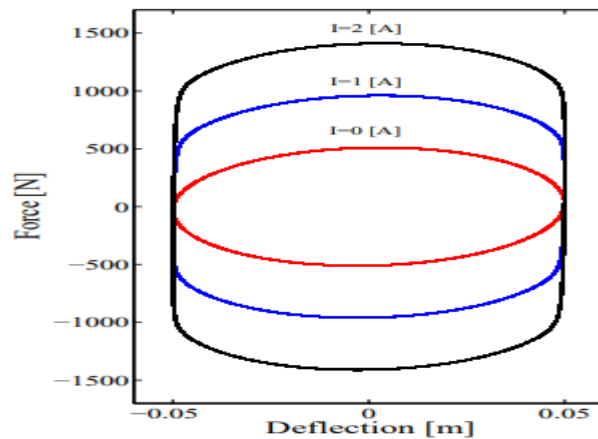


Figure 6: Damper Force Deflection (e.g. [6])

The second constraint is saturation. Damper force must be in peak values which can be produced by changing damper coefficient depends on input current when enters into damper. Thanks to these constraints, model gives realistic damper behaviour by introducing a control output f_I . Limitation of the model is MR dampers' hysteresis which is explained previous chapter is invariant with respect to current I in this assumption. (e.g. Figure 5 and 6) shows the MR damper bi-viscous and the hysteresis in line with when electric current applied in system. [6]

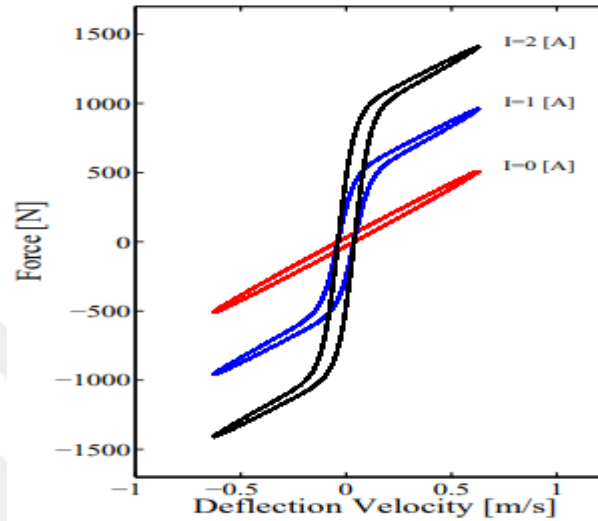


Figure 7: Damper Force-Deflection Velocity (e.g. [6])

Model parameters are taken from the study [7] which shows experimental parameters for damper coefficients.

Table 1: MR Damper Parameters (Source: [7])

MR Damper	Values	Unit
c_0	800.78	[Ns/m]
k_0	620.79	[N/m]
c_1	13.76	[s/m]
k_1	10.54	[1/m]
f_{Imin}	0	[N]
f_{Imax}	800	[N]
γ_{mr}	457.04	[N/A]

3. Suspension System Effects on Vehicle Dynamics

Suspension systems are used for providing comfort and road holding in automotive industry. Suspension system contains 2 main equipment:

- Elastic element
- Damping element

Elastic element is typically a coil spring which delivers a force proportional and opposite to suspension elongation. It carries all the static load and store energy which is occurred by car loads and road disturbance.

Damping element is typically hydraulic shock absorber which dissipates energy is stored by elastic element and produce dissipative force proportional opposite to elongation speed.

Set of these elements links with the sprung body and unsprung body which are respectively car body (chassis) and wheel.

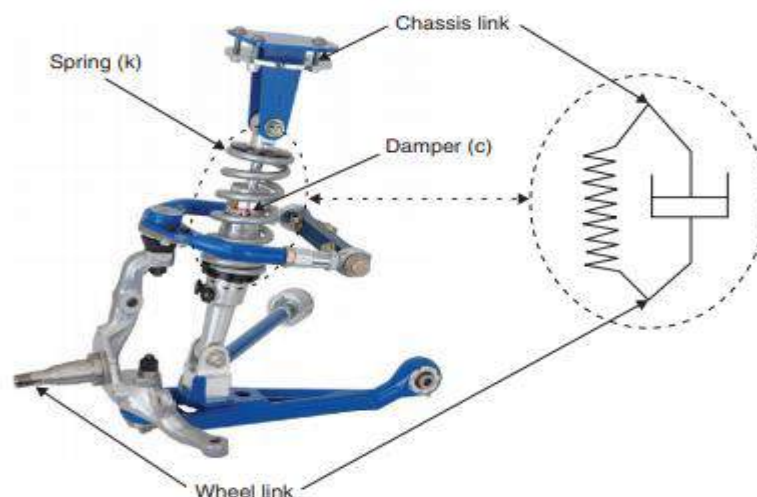


Figure 8: Wheel to Chassis Suspension (e.g. [1])

3.1. Quarter Car Model for Suspension System Analysis

This model is well known as 2-DOF quarter vehicle model (e.g. Figure 9) in order to study vertical dynamics of car which is equipped with suspension systems. The model is suitable and simple for preliminary design. It represents car over only one tire and chassis connection. Parameters that are used in this model for analysing:

- m_{us} : Sprung mass
- m_s : Unsprung mass
- k_s : Spring stiffness coefficient

- k_t : Tire stiffness coefficient
- z_s : Displacement equilibrium point of sprung mass (Car Body)
- z_{us} : Displacement equilibrium point of unsprung mass (Tire)
- z_r : Road profile

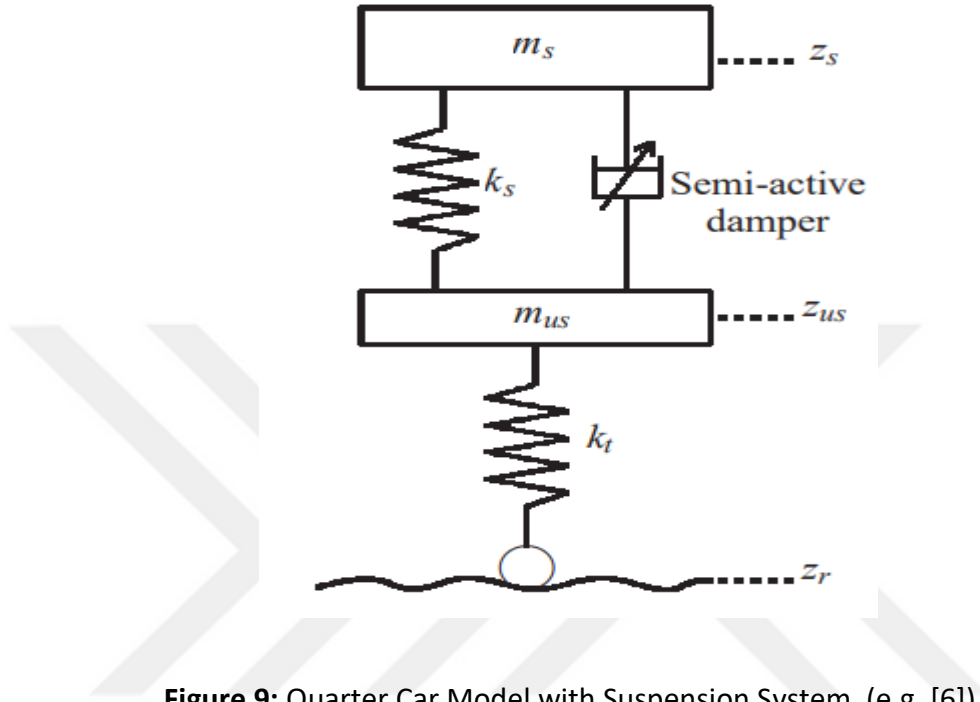


Figure 9: Quarter Car Model with Suspension System (e.g. [6])

Damper equipment can be passive or active depends on the type of suspension. In this model, damper is given as semi-active damper in respect to concept of this study. Second law of Newton is applied on system and dynamical equations of a quarter car are given by [8]

$$(3.1) \quad \begin{aligned} m_s \ddot{z}_s &= -F_{spring} - F_{damper} \\ m_{us} \ddot{z}_{us} &= F_{spring} + F_{damper} - k_t(z_{us} - z_r) \end{aligned}$$

Where $F_{spring} = k_s(z_s - z_{us})$ is the spring force and F_{damper} is damper force.

Table 2: Parameter Values of Quarter Car Model (Source: [8])

Parameters	Value	Unit
m_s	315	[kg]
m_{us}	37.5	[kg]
k_s	29500	[N/m]
k_t	210000	[N/m]

3.2. Sprung Mass Vertical Acceleration

Sprung mass or body vertical acceleration defines comfort level feature on vehicles. Suspension system aims to reduce body vertical acceleration to provide comfortable and smooth travel for passengers. It is one of the important characteristic of the vehicles in terms of passenger health. Details will explain in next chapters.

Vertical acceleration has a direct relation with suspension system. To provide appropriate vertical acceleration, spring stiffness and damping coefficient must be chosen depends on the demanded operation conditions over travel. Vertical acceleration acts different in particular frequency ranges. Vertical acceleration decreases between 0-12 Hz when damping is increased as long as spring stiffness is considered as constant. After 12 Hz, damping and vertical acceleration has a opposite relation. While damping increase, vertical acceleration starts increasing.

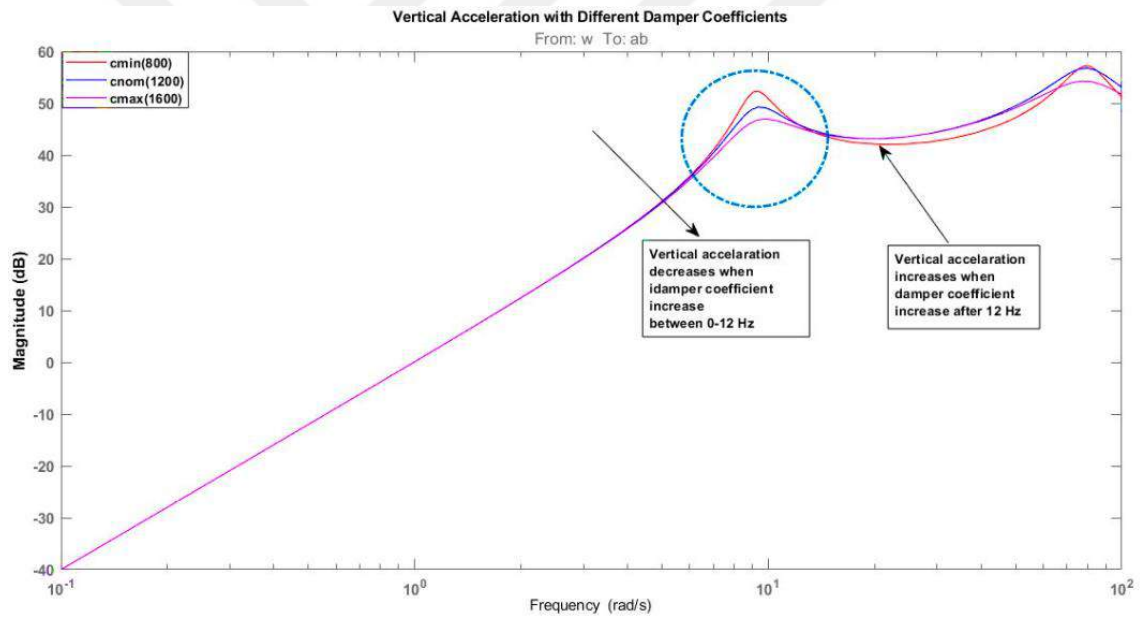


Figure 10: Vertical Acceleration Changes with Damper Coefficients

According to quarter car parameters (e.g. Table 2) and simulate the car model in MATLAB/Simulink (e.g. Figure 10), when damping ratio increased from 800 [Ns/m] to 1600 [Ns/m] body vertical acceleration gain as dB unit decreases between 0-12 Hz. After 12 Hz, vertical body acceleration gain starts increasing as dB unit. In consequence of increasing acceleration gain after 12 Hz, comfort fall into decay while it gives better performance between 0-12 Hz. Note that body vertical acceleration and road disturbance is taken “ab” and “w” in MATLAB/Simulink simulation.

3.3. Unsprung Mass Displacement

Unsprung mass (wheel) displacement is another performance criteria for choosing suspension parameters. Wheel vertical displacement changes tire deflection which is connection between tire and road and defines road holding. To provide better road holding, tire deflection must be minimized over all driving conditions. While tire displacement gain decreases 2-30 Hz and converge to 0 dB gain, when damping ratio increases. It means that tire force will be minimized to maintain road holding.

$$(3.2) \quad Tire_{deflection} = z_{us} - z_r$$

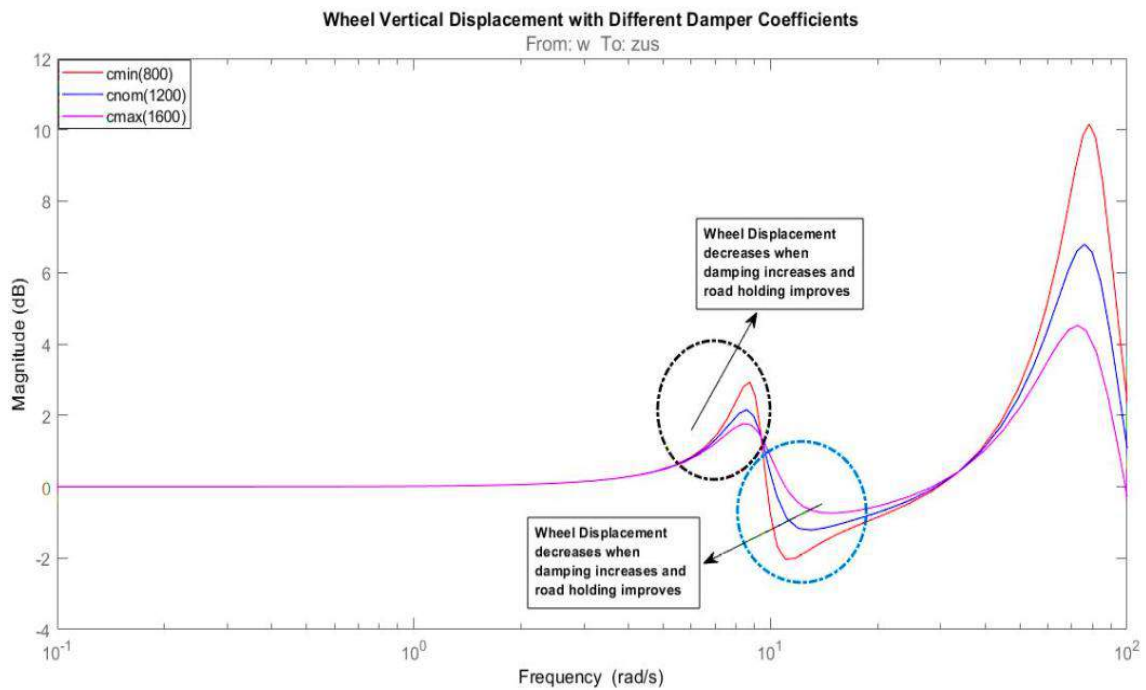


Figure 11: Wheel Displacement with Different Damper Coefficients

According to quarter car parameters (e.g. Table 2) and simulate the car model in MATLAB/Simulink (e.g. Figure 11) when damping ratio increased from 800 [Ns/m] to 1600 [Ns/m] body wheel displacement gain as dB unit decreases between 2-30 Hz and converge to 0 dB gain.

3.4. Comfort/Road Holding Trade-Off

As mentioning about previous chapter, vehicle vertical acceleration and wheel displacement decreases until 12 Hz when damping ratio increases. According to (e.g. Figure 10 and 11), it is clearly shown that there is an unavoidable trade-off between comfort and road holding after around 12 Hz.

Table 3: Comfort/Road Holding Changes When Damping Increases

Performance Features	Comfort ↑ / Road Holding ↑				Trade-Off	Comfort ↓ / Road Holding ↑		
	4Hz	8Hz	10 Hz	11hHz	12 Hz	13 Hz	17Hz	20 Hz
Vertical Acceleration	↓	↓	↓	↓	→	↑	↑	↑
Wheel Displacement	↓	↓	↓	↓	↓	↓	↓	↓
Suspension Deflection	↓	↓	↓	↓	↓	↓	↓	↓
Tire Deflection	↓	↓	↓	↓	↓	↓	↓	↓
Body Displacement	↓	↓	↓	↓	→	↑	↑	↑

Using quarter car model parameters over damping ratio range from 800 [Ns/m] to 1600 [Ns/m], MATLAB/Simulink simulations give the results in (e.g. Table 3) which proves the assumptions from the previous chapter. Simulation results shows that there is a trade-off between comfort and road holding around 12 Hz.

4. Performance Objectives for Semi-Active Suspension Control

As mentioned previous chapter, two main objectives in semi-active suspension control are ride comfort and road holding. Road holding can be calculated and measured by considering dynamic tire force that is the key parameter for road holding. On the other hand, ride comfort is more difficult to measure due to it can be felt or interpreted differently by different passengers. Human sensivity to vibration depends on frequency. In addition, different parts of the human body experience different sensations at the same frequency. There are some criteria that exist in literature to answer the question of which one is the best measure for evaluating driving comfort. [9]

- Janeways's comfort criteria
- Steffens comfort criteria
- Vibration dose (VD)
- ISO 2631

In this study, ISO 2631 criteria is taken to measure driving comfort.

4.1. Comfort Control Goal

The most common criterion is the ISO 2631 standard. It is general standard that can be applied not only to vehicles. It can be measure for all vibrating environments. Standard defines that exposure limits of body is in the range 1-80 Hz and defines the limits for reduced comfort and health protection. This criteria suggest that range of 4-8 Hz for vertical acceleration can be felt more and more sensitive than other frequency ranges for human beings.

$$(4.1) \quad W_{ISO2631} = \frac{81.89s^3 + 796.6s^2 + 1937s + 0.1446}{s^4 + 80.00s^3 + 2264s^2 + 7172s + 21196}$$

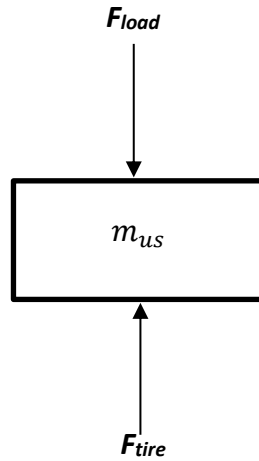
As mentioned previous chapter, when damping increases body vertical acceleration decreases between 0-12 Hz and after start increasing. The main goal in terms of control objective and related to ISO 2631 standard, gain of transfer function for body vertical acceleration which can be denoted $\ddot{z}_s/z_r$ should be minimize all over frequency range to provide ride comfort. Ride comfort objective given by

$$(4.2) \quad Comfort = \min \int_0^{20} \ddot{z}_s/z_r(f)df$$

Equation (4.2) defines to minimize body vertical acceleration frequency range between 0-20 Hz. As mentioned previous chapter, around 12 Hz there is a conflict between road holding and ride comfort. Therefore, objective will be focused on specifically 4-8 Hz according to ISO 2631 standard which shows that this frequency range is more crucial than other frequency domains in terms of comfort and human health.

4.2. Road Holding Control Goal

Another performance criteria is road holding to maintain and keep safe driving. To maintain contact wheel and road dynamic tire force should be smaller than car load along with travel.



$$(4.3) \quad \begin{aligned} F_{tire} &< F_{load} \\ k_t(z_{us} - z_r) &< g(m_s + m_{us}) \end{aligned}$$

Tire force has a direct relation with tire deflection $(z_{us} - z_r)$ which decreases when damping ratio increases between 2-30 Hz according to simulation results in chapter 3.3. The gain of transfer function for tire deflection which can be denoted $(z_{us} - z_r)/z_r$ should be minimized all over frequency range to provide road holding.

Road holding objective given by

$$(4.4) \quad \text{Road Holding} = \min \int_0^{30} (z_{us} - z_r)/z_r(f) df$$

As mentioned previous chapter, around 12 Hz there is a trade-off between ride comfort and road holding. Therefore, objective will be focused on specifically 10-20 Hz where wheel displacement converge to 0 dB gain while vertical acceleration gain increases in case of increased damping ratio. This trade-off will be compromised with Genetic Algorithms in next chapters.

5. Linear Parameter Varying (LPV) Systems

Linear Parameter Varying (LPV) is a method that provides control of the linear parameter variable system which according to its state, has a class of nonlinear systems that can be modeled as parameterized linear systems. These systems are ideal for use with significant uncertainties in the more general case. Multivariable controllers are used when designing feedback controllers for dynamic systems. These controllers are primarily built at different operating points and can linearize systems and models according to parameters. System variables detected and evaluated in real time can be used as planning parameters to linearize the system at various stages. Suspension deflection that can be observed in real time with displacement sensors can be a good example of scheduling parameters. Nonlinear systems can be linearised at scheduling points and can be installed using interpolation methods with the necessary controller for each operating point. This method is called gain scheduling. [10,11]

As mentioned above paragraph, since parameter evolution is measured and observable in real time and if the dynamic system is linear in time, LPV systems can be written similar to LTI form in time variant parameter. LPV system is mathematically given by

$$(5.1) \quad \begin{aligned} dx(t) &= A(p)x(t) + B(p)u(t) \\ y(t) &= C(p)x(t) + D(p)u(t) \end{aligned}$$

- $u(t)$: Inputs
- $y(t)$: Outputs
- $x(t)$: Model states with initial values $x(0) = x_0$
- $Dx(t)$: State derivative vector $\dot{x}$ for continuous-time systems
- $A(p), B(p), C(p), D(p)$ are the state-space matrices parameterized with the vector p scheduling parameters

- The parameters $p = p(t)$ are measurable input functions and model status. The quantity or a variable of many parameters may be explained. The set of scheduling parameters determines the scheduling space for the LPV model.[12]

There are 3 different methodologies to design LPV controller:

1. Linear fractional transformations based on the limited gain theorem for robustness and performance
2. Single Quadratic Lyapunov Functions
3. Parameter Dependent Quadratic Lyapunov Function is to limit the achievable performance level

In this study will be focus on Quadratic Lyapunov Functions to achieve robust performance and stability by solving convex feasibility problem regarding semi-active suspension goals.

5.1. Affine Parameter Dependent Model

Many physical systems may be described by linearized equations in descriptor form

$$(5.2) \quad \begin{aligned} E(p)\dot{x} &= A(p)x + B(p)u \\ y &= C(p)x + D(p)u \end{aligned}$$

Where $A(\cdot), \dots, E(\cdot)$ are known functions of some parameter vector $p = (p_1, \dots, p_s)$. The parameters $p_1, \dots, p_s$ may vary slowly with the time within some prescribed boundaries. In the case of affine parameter-dependent systems the matrices of the state-space description are affine functions of the parameters, which means that these matrices may be represented as

$$(4.3) \quad \begin{aligned} A(p) &= A_0 + p_1 A_1 + \dots + p_s A_s \\ B(p) &= B_0 + p_1 B_1 + \dots + p_s B_s \\ C(p) &= C_0 + p_1 C_1 + \dots + p_s C_s \\ D(p) &= D_0 + p_1 D_1 + \dots + p_s D_s \\ E(p) &= E_0 + p_1 E_1 + \dots + p_s E_s \end{aligned}$$

where $A_0, \dots, A_s, B_0, \dots, B_s$ and so on are constant matrices that do not depend on the parameters $p_1, \dots, p_s$. Note that for a particular system some of these matrices may be zero matrices.[13]

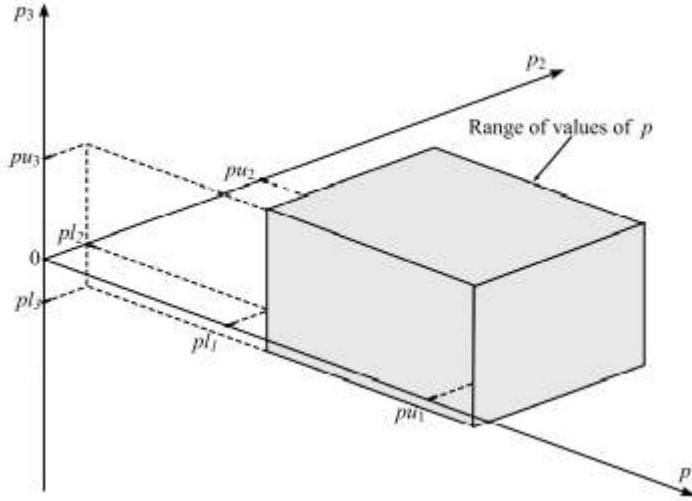


Figure 12: Parameter Box (e.g. [11])

Parameter box consist of all possible paramaters inside of it and varying system depending on the parameters which are limited from corners of box. The advantage of such a system model is the possibility to derive easily several results concerning closed-loop stability and controller design. Parameter dependent system matrix given by

$$S(p) = \begin{bmatrix} A(p) + jE(p) & B(p) \\ C(p) & D(p) \end{bmatrix}$$

$$(5.4) \quad S_i = \begin{bmatrix} A_i + jE_i & B_i \\ C_i & D_i \end{bmatrix}$$

$$i = 0, \dots, s$$

The affine parameter-dependent system is written compactly in

$$(5.5) \quad S(p) = S_0 + p_1 S_1 + \dots + p_s S_s$$

The constants matrices, . . . S_s may be considered as system coefficients. First matrix S_0 is written without any parameter dependent constant. Parameter dependent coefficients of system is taken zero in this matrix. Rest of constant matrices $S_1, \dots, S_s$ contains only parameter dependent coefficients of system are scheduled by the parameter constant.

Parameter range can be described ias a box parameter space and p_i ranges between two known bounds

$$(5.6) \quad p_i \in [p_{lowi}, p_{upi}], \quad i = 1, \dots, s$$

Affine parameter method will applied on semi-active suspension system to convert polytopic model to achive closed loop control targets in next chapters.

5.2. Polytopic Models

Polytopic system is a linear time-varying system

$$(5.7) \quad \begin{aligned} E(t)\dot{x} &= A(t)x + B(t)u \\ y &= C(t)x + D(t)u \end{aligned}$$

Whose SYSTEM matrix

$$(5.8) \quad P = \left\{ \sum_{i=1}^q a_i S_i : a_i \geq 0, \sum_{i=1}^q a_i = 1 \right\}$$

In the above expression $S_1, S_2, \dots, S_q$ are given vertex systems

$$(5.9) \quad S_i = \begin{bmatrix} A_i + jE_i & B_i \\ C_i & D_i \end{bmatrix}, \quad i = 1, \dots, q$$

Corresoiding to the vertices of $S(t)$ [13]

Polytopic system $S(t)$ is a convex combination of system matrices $S_1, S_2, \dots, S_q$ is given by

$$(5.10) \quad S(t) = a_1 S_1 + \dots + a_q S_q$$

The nonnegative numver $a_1, \dots, a_q$ are called polytopic coordinates of $S(t)$

Affine systems depending on the parameter can easily be transformed into polytopic systems. If any parameter p_i differs within a certain interval $[p_{lowi}, p_{upl}]$, $i = 1, \dots, s$, then the parameter vector $p = (p_1, \dots, p_s)$ takes values in a parameter box with $q = 2^s$ corners $\Pi_1, \Pi_2, \dots, \Pi_q$. According to the results of functional analysis, if the function $S(p)$ is affine in p , then this parameter box is mapped into a certain polytope of system matrices. In particular, this polytope is the convex linear image combination $S_1 = S(\Pi_1), S(\Pi_2), \dots, S(\Pi_q)$ of the parameter box corners $\Pi_1, \Pi_2, \dots, \Pi_q$. These images are the vertex systems of the polytopic model. [14]

5.3. LPV Model for Semi-Active Suspension

Quarter car model and damper equation according to Guo-Model is respectively given equation(3.1) and (2.3) in the second and third chapter.

The dynamical equations of quarter car model are rewritten as follows

$$(5.11) \quad \begin{cases} m_s \ddot{z}_s = -k_s(z_s - z_{us}) - c_0(\dot{z}_s - \dot{z}_{us}) - k_0(z_s - z_{us}) \\ \quad - f_I \tanh(c_1(\dot{z}_s - \dot{z}_{us}) + k_1(z_s - z_{us})) \\ m_{us} \ddot{z}_{us} = k_s(z_s - z_{us}) + c_0(\dot{z}_s - \dot{z}_{us}) + k_0(z_s - z_{us}) \\ \quad + f_I \tanh(c_1(\dot{z}_s - \dot{z}_{us}) + k_1(z_s - z_{us})) - k_t(z_{us} - z_r) \end{cases}$$

According to defined semi-active suspension system, scheduling parameters and control objectives in previous chapters, LPV model formulation can be defined by the parameters

- $c_p = c_0$
- $k_p = k_s + k_0$
- $z_{def} = z_s - z_{us}$
- $\dot{z}_{def} = (\dot{z}_s - \dot{z}_{us})$
- $p = \tanh(c_1(\dot{z}_s - \dot{z}_{us}) + k_1(z_s - z_{us}))$

5.3.1. Selection of Scheduling Parameters, Measurement Outputs and Control Outputs

Scheduling parameters should be observable and measured in real time to create LPV structure. Suspension deflection $z_{def} = z_s - z_{us}$ is one of the important characteristic to define performance of system and which can be measured and observed in real time by displacement sensors.

$$(5.12) \quad F_{mr} = f_I \tanh(c_1(\dot{z}_s - \dot{z}_{us}) + k_1(z_s - z_{us}))$$

$$(5.13) \quad F_{damper} = F_{passive} + F_{mr}$$

According to Guo-Model which is mentioned in chapter 1 , current that enters damper can change its viscosity and change the potential force which can be produced by damper. Viscosity changes and its affect on damper force is represented by F_{mr} and positive or negative F_{mr} which can be produced by current determine total damper force in travel conditions by applying electric current I in damper. F_{mr} force depends on $\tanh(c_1(\dot{z}_s - \dot{z}_{us}) + k_1(z_s - z_{us})) - k_t(z_{us} - z_r)$ and f_I which is controllable force coefficient according to Guo-Model.

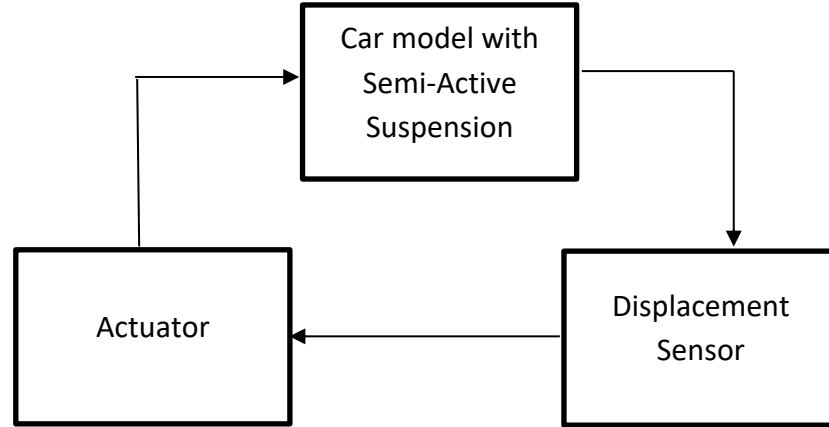


Figure 13: Uncontrolled Suspension Model

Function $\tanh(c_1(\dot{z}_s - \dot{z}_{us}) + k_1(z_s - z_{us}))$ is naturally a scheduling parameter in LPV design since suspension deflection and its derivative can be measured in real time by displacement sensor. Therefore, $z_{def} = z_s - z_{us}$ and $\dot{z}_{def} = (\dot{z}_s - \dot{z}_{us})$ will be select as measurement outputs to build LPV design.

According to model, $\tanh(c_1(\dot{z}_s - \dot{z}_{us}) + k_1(z_s - z_{us}))$ function is bounded $[-1;1]$ for any value of suspension deflection and its derivative. This model satisfy dissipivity and saturation constarint of semi-active damper which will be explain after building state space model.

As mentioned chapter 3, performance objectives of semi-active suspensions are comfort and road holding. Comfort is related to body vertical acceleration while road holding is connected to tire deflection. Tire deflection is a function of tire force and can be measured by tire displacement. According to dymanic equations and control objectives of system, vertical acceleration $\ddot{z}_s$ and z_{us} can be choosen as control outputs to fulfill performance objectives with chosen control method which will be explain in next chapters.

5.3.2. State Space Model

From equation (5.11), the state-space representation of the quarter vehicle model can be deduced as follows:

- $x_s = \text{States}$
- $w = \text{Disturbance}$
- $y = \text{Measurement Outputs}$
- $z = \text{Control Outputs}$
- $A_s = \text{State Matrix}$
- $B_{sw} = \text{Disturbance Matrix}$
- $B_s = \text{Control Input Matrix}$

- C_s = Measurement Output Matrix
- C_{sz} = Control Output Matrix
- D_{sz} = Feedthrough Matrix

$$(5.14) \quad x_s = \begin{bmatrix} z_s \\ \dot{z}_s \\ z_{us} \\ \dot{z}_{us} \end{bmatrix}$$

$$(5.15) \quad w = z_r$$

$$(5.16) \quad y = \begin{bmatrix} z_s - z_{us} \\ \dot{z}_s - \dot{z}_{us} \end{bmatrix}$$

$$(5.17) \quad \begin{cases} \dot{x}_s = A_s x_s + B_{sw} w + B_s p f_I \\ z = C_{sz} x_s + D_{sz} p f_I \\ y = C_s x_s \end{cases}$$

$$(5.18) \quad A_s = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{k_p}{m_s} & -\frac{c_p}{m_s} & \frac{k_p}{m_s} & \frac{c_p}{m_s} \\ 0 & 0 & 0 & 1 \\ \frac{k_p}{m_{us}} & \frac{c_p}{m_{us}} & -\frac{k_p + k_t}{m_{us}} & -\frac{c_p}{m_{us}} \end{bmatrix}$$

$$(5.19) \quad B_{sw} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{k_t}{m_{us}} \end{bmatrix}$$

$$(5.20) \quad B_s = \begin{bmatrix} 0 \\ 1 \\ -\frac{1}{m_s} \\ 0 \\ 1 \\ \frac{1}{m_{us}} \end{bmatrix}$$

$$(5.21) \quad C_s = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}$$

$$(5.22) \quad C_{sz} = \begin{bmatrix} -\frac{k_p}{m_s} & -\frac{c_p}{m_s} & \frac{k_p}{m_s} & \frac{c_p}{m_s} \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$(5.23) \quad D_{sz} = \begin{bmatrix} -\frac{1}{m_s} \\ 0 \end{bmatrix}$$

Suspension deflection and suspension deflection velocity are considered measurement outputs which allows scheduling parameter p can be known in real time.

Performance objectives are ride comfort and road holding can be represented respectively vehicle body acceleration $\ddot{z}_s$ and wheel displacement or bounce z_{us} .

5.3.3. Dissipativity and Saturation Constraint

As explained in the first chapter, the control signal f_I must fulfill the constraint to ensure the dissipation of an MR damper in equation (2.5). Since f_I control signal can take only positive values. Therefore, control input must recast to provide control objectives and semi-activeness of system.

1. First Step

Determine nominal control signal which must be the mean value (nominal force) of maximum and minimum force can be produced by MR damper.

$$(5.24) \quad f_0 = (f_{I_{max}} + f_{I_{min}})/2$$

As MR damper parameters given in (Table 1), control signal and nominal force can take values below

- $f_{I_{max}} = 800 [N]$
- $f_{I_{min}} = 0 [N]$
- $f_0 = 400 [N]$

2. Second Step

Define the control input in order to make control signal f_I positive in all applied current point depending on measurement outputs are provided by displacement sensor to provide dissipativity constraint of MR damper. Control signal equation is recast as control input and its values changes depending on control input values which will be calculated by controller. Control method and sythesis will be explain in next chapters.

$$(5.25) \quad u_I = f_I - f_0$$

$$(5.26) \quad -400 [N] \leq u_I \leq 400 [N]$$

Table 4: Control Signal and Current Input Change Depending on Control Output

Control Signal [N]	Nominal Force [N]	Control Output [N]	Current Coefficient [A/N]	Current [A]
f_I	f_0	u_I	y_{mr}	I
800	400	400	457.04	1.750
400	400	0	457.04	0.875
0	400	-400	457.04	0

Table 4 shows the limit current input and forces (F_{mr}) which can be produced additionally by semi active damper. As written in previous part, total damper force is given by (5.13)

$$(5.13) \quad F_{damper} = F_{passive} + F_{mr}$$

Table 5: Semi-Active Damper Force Relation with Control Signal

Control Signal	Force [N]	Scheduling	F_{mr} [N]	F_{damper} [N]
f_{Imax}	800	-1	-800	$F_{passive} - 800$
		0	0	$F_{passive} + 0$
		1	800	$F_{passive} + 800$
f_{Inom}	400	-1	-400	$F_{passive} - 400$
		0	0	$F_{passive} + 0$
		1	400	$F_{passive} + 400$
f_{Imin}	0	-1	0	$F_{passive} + 0$
		0	0	$F_{passive} + 0$
		1	0	$F_{passive} + 0$

Total damper force F_{damper} changes in scheduling parameter range can be shown in (e.g. Table 5) when the control signal takes the minimum, maximum or nominal force value. Since scheduling parameter can take values $[-1;1]$, F_{mr} can increase and decrease semi-active damper force 800 N. Therefore, control output can take the values $[-400;400]$ to satisfy saturation constraint.

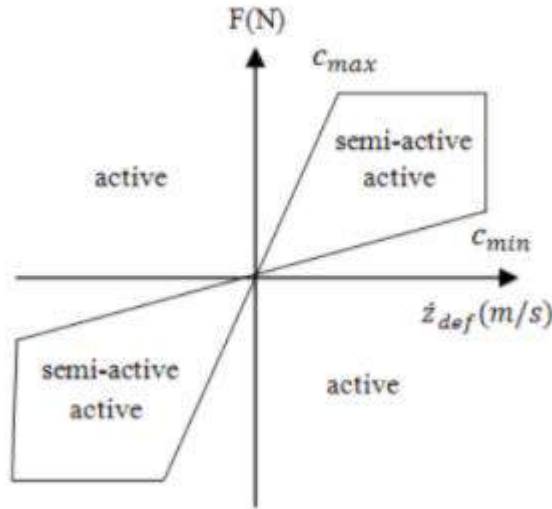


Figure 14: Dissipativity and Saturation Constraint (e.g. [15])

5.3.4. Reformulation of State Space Model with Constraints

According to constraints in previous chapter, state space model should be recast depending on dissipativity and saturation constraint of MR damper. The state-space representation of the quarter car model as follows with this change :

$$(5.27) \quad P: \begin{cases} \dot{x}_s = (A_s + B_{s1} \frac{p}{C_{s1} x_s} C_{s1}) x_s + B_{sw} w + B_s p u_I \\ z = (C_{sz} + D_{s1} \frac{p}{C_{s1} x_s} C_{s1}) x_s + D_{sz} p u_I \\ y = C_s x_s \end{cases}$$

Where B_{s1} , C_{s1} and D_{s1} matrices

$$(5.28) \quad B_{s1} = \begin{bmatrix} 0 \\ -\frac{f_0}{m_s} \\ 0 \\ \frac{f_0}{m_{us}} \end{bmatrix}$$

$$(5.29) \quad C_{s1} = [k_1 \quad c_1 \quad -k_1 \quad -c_1]$$

$$(5.30) \quad D_{s1} = \begin{bmatrix} -\frac{f_0}{m_s} \\ 0 \end{bmatrix}$$

The LPV model (5.27) can be used in this study to design an LPV controller. However the close loop stability and efficiency may not be guaranteed by such a control because the saturation limit is not yet considered in the design. Alternative approaches have been suggested for this issue. For example, [16] defines a scheduling parameter as the difference between the actual controller-controlled damper force and the necessary controller force. However the limitation of dissipation is not fulfilled theoretically. Another alternative is to incorporate an AWBT in the closed loop system [17], [18], [19] to minimize the detrimental impacts of the saturated control performance on the closed loop efficiency. The input saturation as a scheduling parameter is an easy way to fix the problem. In reality, this approach is linked to [20]. Defines saturation as follows: $sat(\cdot)$

$$(5.31) \quad sat(u_I) = \begin{cases} 400 & \text{if } u_I \geq 400 \\ u_I & \text{if } -400 \leq u_I \leq 400 \\ -400 & \text{if } u_I \leq -400 \end{cases}$$

The state-space representation of the system (5.27) subject to input saturation constraint (5.26) is rewritten as

$$(5.32) \quad P: \begin{cases} \dot{x}_s = (A_s + B_{s1} \frac{p}{C_{s1}x_s} C_{s1})x_s + B_{sw}w + B_s p \frac{sat(u_I)}{u_I} u_I \\ z = (C_{sz} + D_{s1} \frac{p}{C_{s1}x_s} C_{s1})x_s + D_{sz} p \frac{sat(u_I)}{u_I} u_I \\ y = C_s x_s \end{cases}$$

After applying input saturation constraint, scheduling parameter p is taken depending on suspension deflection and suspension deflection velocity is converted two scheduling parameter to satisfy dissipativity and saturation constraints of semi-active damper. Scheduling parameters represents as

$$(5.33) \quad p_1 = p \frac{sat(u_I)}{u_I}$$

$$(5.34) \quad p_2 = \frac{p}{C_{s1}x_s}$$

Recasted scheduling parameters are applied in state space representation and LPV system is obtained from (5.32)

$$(5.35) \quad P: \begin{cases} \dot{x}_s = (A_s + B_{s1}C_{s1}p_2)x_s + B_{sw}w + B_s p_1 u_I \\ z = (C_{sz} + D_{s1}C_{s1}p_2)x_s + D_{sz}p_1 u_I \\ y = C_s x_s \end{cases}$$

5.3.5. Creating Semi-Active Suspension Polytopic Model with Matlab

As mentioned in previous chapters, parameter dependent model can be obtained by using affine and polytopic model methods. Semi-active suspension system can be created as polytopic model by using Matlab tools. Following strategy can be follow to obtain polytopic model of system

1. Define scheduling parameters range
2. Create affine parameter dependent model
3. Convert affine parameter dependent model to polytopic model

Scheduling parameters range can be determined from (5.25) to fulfill the dissipativity and saturation constraints which are explained previous chapter and parameter dependent plant change can be shown with limit parameters below table

Table 6: Control Signal Values with Scheduling Parameters

Nominal Force	P2	Control Input	P1	Control Signal
400	0	u_I	-1	$-u_I$
			1	u_I
	0.5		-1	$200 - u_I$
			1	$200 + u_I$
	1		-1	$400 - u_I$
			1	$400 + u_I$

(e.g. Table 6) shows that the limit forces can be obtained by semi-active damper when parameters scheduled in range. This values fulfill the semi-active suspension constraints to create parameter dependent plant. Scheduling parameter range is given by

- $p_1 \in [-1; 1]$

$$(5.36) \quad p_1 = \tanh(C_{s1}x_s) \frac{\text{sat}(u_I)}{u_I}$$

- $p_2 \in [0; 1]$

$$(5.37) \quad p_2 = \frac{\tanh(Cs_1 x_s)}{Cs_1 x_s}$$

From (5.32) affine parameter dependent model can be obtained

$$(5.38) \quad S(p) = S_0 + p_1 S_1 + p_2 S_2$$

$$(5.39) \quad S_0: \begin{cases} \dot{x}_s = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_p}{m_s} & -\frac{c_p}{m_s} & \frac{k_p}{m_s} & \frac{c_p}{m_s} \\ 0 & 0 & 0 & 1 \\ \frac{k_p}{m_{us}} & \frac{c_p}{m_{us}} & -\frac{k_p + k_t}{m_{us}} & -\frac{c_p}{m_{us}} \end{bmatrix} x_s + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{k_t}{m_{us}} \end{bmatrix} w + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_I \\ C_{sz} = \begin{bmatrix} -\frac{k_p}{m_s} & -\frac{c_p}{m_s} & \frac{k_p}{m_s} & \frac{c_p}{m_s} \\ 0 & 0 & 1 & 0 \end{bmatrix} x_s + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u_I \\ C_s = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} x_s + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u_I \end{cases}$$

$$(5.40) \quad S_1: \begin{cases} \dot{x}_s = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x_s + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} w + \begin{bmatrix} 0 \\ -1 \\ \frac{1}{m_s} \\ 0 \\ 1 \\ \frac{1}{m_{us}} \end{bmatrix} u_I \\ C_{sz} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x_s + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u_I \\ C_s = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x_s + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u_I \end{cases}$$

$$(5.41) \quad S_2: \begin{cases} \dot{x}_s = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -\frac{f_0}{m_s} k_1 & -\frac{f_0}{m_s} c_1 & \frac{f_0}{m_s} k_1 & \frac{f_0}{m_s} c_1 \\ 0 & 0 & 0 & 0 \\ \frac{f_0}{m_{us}} k_1 & \frac{f_0}{m_{us}} c_1 & -\frac{f_0}{m_{us}} k_1 & -\frac{f_0}{m_{us}} c_1 \end{bmatrix} x_s + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} w + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_I \\ C_{sz} = \begin{bmatrix} -\frac{f_0}{m_s} k_1 & -\frac{f_0}{m_s} c_1 & \frac{f_0}{m_s} k_1 & \frac{f_0}{m_s} c_1 \\ 0 & 0 & 0 & 0 \end{bmatrix} x_s + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u_I \\ C_s = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x_s + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u_I \end{cases}$$

$$(5.42) \quad E_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

For LTI system set, S_0 should take E_0 as a identity matrix.

$$(5.43) \quad E_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$(5.44) \quad E_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

For LTI system set, E_1 and E_2 should be parameter independent and zero matrix.

In this step, parameter box should be implemented for chosen parameter range by MATLAB tool with "pvec".

```

##### Affine Parameter Box #####

plmin=-1; plmax=1; p2min=0; p2max=1;

pv=pvec('box',[plmin plmax; p2min p2max]);

```

Figure 15: Affine Parameter box with MATLAB

After creation of parameter range and box, LTI models of system can be obtained by MATLAB tool with "ltisys" for each parameter dependent system matrix.

```

%Specify parameter-dependent model
s0=ltisys(Alpv0,[Bwlpv0 Bulpv0],[Czlpv0;Cylpv0],[ Dzwlpv0 Dzulpv0 ;Dywlpv0 Dyulpv0],E0);
s1=ltisys(Alpv1,[Bwlpv1 Bulpv1],[Czlpv1;Cylpv1],[ Dzwlpv1 Dzulpv1 ;Dywlpv1 Dyulpv1],E1);
s2=ltisys(Alpv2,[Bwlpv2 Bulpv2],[Czlpv2;Cylpv2],[ Dzwlpv2 Dzulpv2 ;Dywlpv2 Dyulpv2],E2);

```

Figure 16: LTI System Model with MATLAB

In last step, affine parameter dependent car model with semi-active suspension can be obtained by using "psys" code.

```

#####Affine LPV Model#####

pdPa = psys(pv,[s0 s1 s2 ]);

```

Figure 17: Affine Parameter Dependent Model with MATLAB

Affine parameter system properties can be checked by using "psys" code in MATLAB. Affine parameter dependent semi-active suspension model has 2 parameters with 3 systems (S_0, S_1, S_2) and 4 states , 2 inputs (one of them is disturbance), 4 outputs ($\ddot{z}_s, z_{us}, z_{def}, \dot{z}_{def}$).

```
psinfo(pdPa)
Affine parameter-dependent model with 2 parameters (3 systems)
Each system has 4 state(s), 2 input(s), and 4 output(s)
```

Figure 18: Affine Parameter Dependent Model Result

Parameter trajectory can be defined for demanded parameter range with matlab. In this study, difference between each scheduling parameter point is considered 0.1.

- $p_1 = -1, -0.9, -0.8, \dots, 1$
- $p_2 = 0, 0.1, 0.2, \dots, 1$

In consequence of considered parameter range, p_1 takes 21 points while p_2 takes 11 points. In total, 231 couple of (p_1, p_2) point is created to estimate possible trajectory of system model.

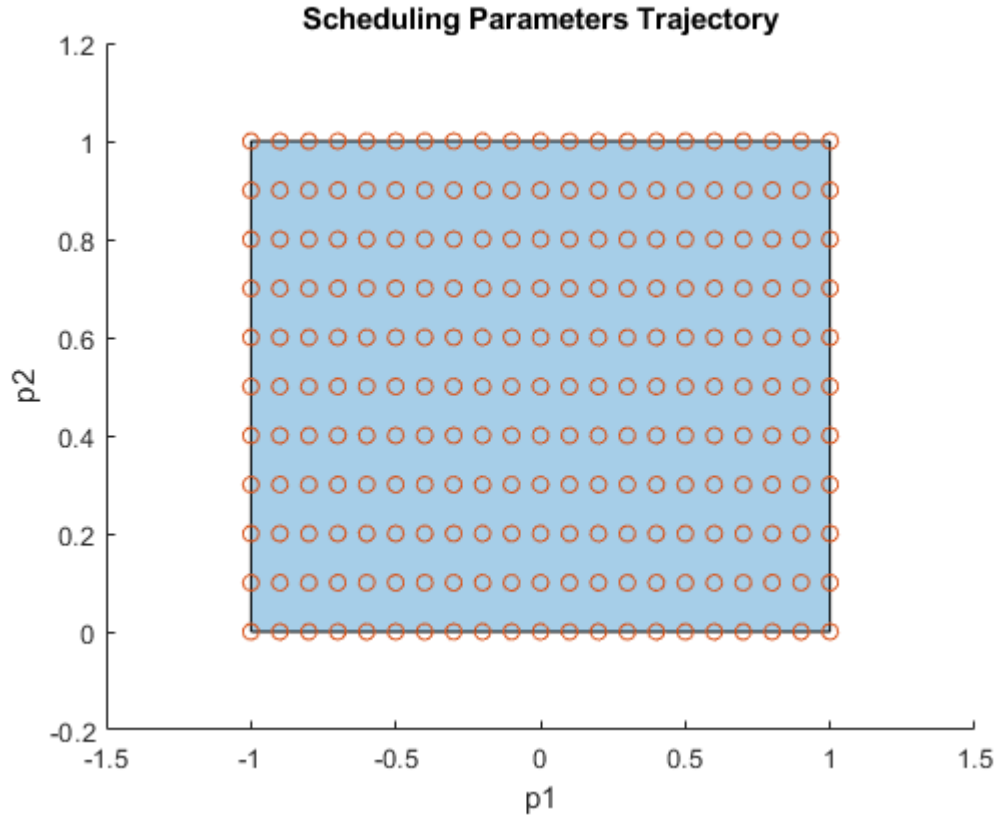


Figure 19: Parameter Trajectory

Affine parameter-dependent systems can be easily converted to polytopic systems. If each parameter p_i varies in some interval $[p_{lowi}, p_{upi}]$, $i = 1, \dots, s$, then the parameter vector $p = (p_1, \dots, p_s)$ takes values in a parameter box with $q = 2^s$ corners $\Pi_1, \Pi_2, \dots, \Pi_q$ as mentioned previous chapter.

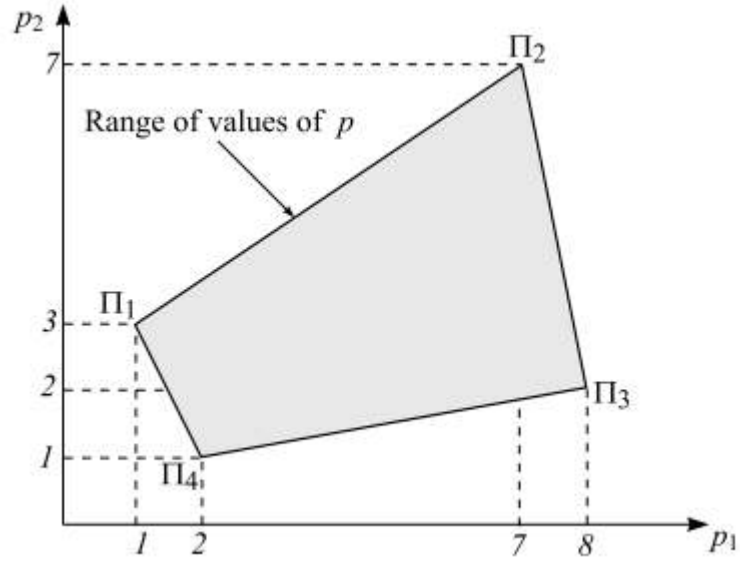


Figure 20: Sample of Polytopic Model Representation (e.g. [13])

In this study, semi-active suspension system model has two parameter (p_1, p_2) and it takes corner is given by

$$(5.45) \quad q = 2^2$$

In this way, the method transforms the current parameter vector $p(t)$ to convex decomposition. The controller state-space matrices are obtained at operating point $p(t)$ which are a control technique, and described in the next chapter by convex interpolation of the linear time-invariant vertex controllers.

MATLAB tool provides easily transform from affine parameter dependent system to polytopic system with "*aff2pol*" code. Polytopic semi-active suspension model has 4 vertex respectively $(-1,0)$, $(-1,1)$, $(1,0)$, $(1,1)$. All other parameters which are calculated above will be inside of these vertices.

```
>> psinfo(pdP)
Polytopic model with 4 vertex systems
Each system has 4 state(s), 2 input(s), and 4 output(s)
```

Figure 21: Semi-Active Suspension Model Polytopic Result

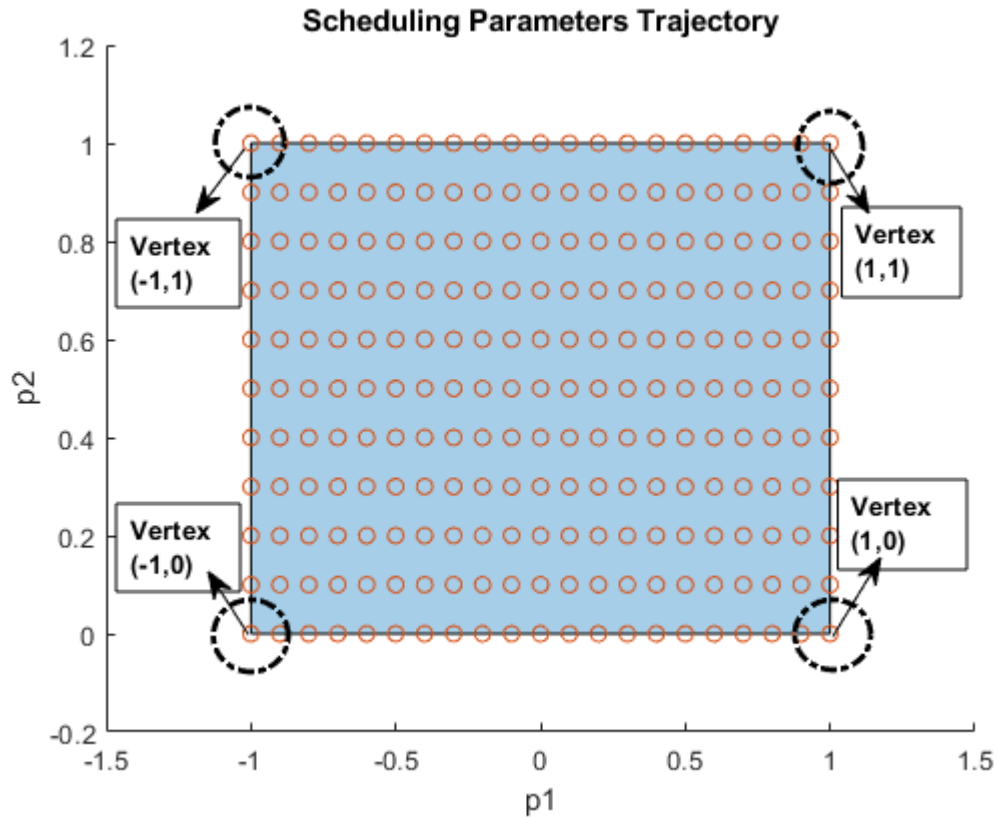


Figure 22: Vertices of Semi-Active Suspension Polytopic Model

Polytopic model has 4 vertices which are corners of parameter box as considered to build model. However, this model only includes these 4 points and there are still many inside of box. Therefore, polytopic coordinate vector should be defined to achieve state space models of each point for simulink model. Thanks to MATLAB tool, which is called "polydec" code, polytopic coordinates can be obtained over all trajectory (e.g. Figure 23).

```

#####polytopic coordinates and trajectory#####
#####Scheduling Parameter Range###

p1=-1:0.1:1;
p2=0:0.1:1;

%%%P(p1,p2) All Parameters Point Vector%%%
p=[];

for i = 1:length(p1)
    for j=1:length(p2)
        po =[ p1(i),p2(j)];
        p=[p;po];
    end
end
%%%Polytopic Coordinates%%%

alpha=[];
for i =1: length(p)
    alphap = polydec(pv,[ p(i,1) p(i,2)]);
    alpha=[alpha;alphap];
end

```

Figure 23: Polytopic Coordinates MATLAB code of Semi-Active Suspension Model

In algorithm (e.g. Figure 23), polytopic coordinates of system is calculated as "*alpha*". In the next step, each polytopic coordinate can be obtained by using the "psinfo" code, which is a multipurpose function for queries on a polytopic and/or parameter-dependent framework. "*psinfo(ps,'eval',p)*" instantiates a vector *p* system of values or polytopic coordinates for the provided vector. For polytopic systems with system matrix ranging in (S_1, S_2, S_3, S_4) for 2 parameter dependent plant and the entries of *p* should be polytopic coordinates $p_1, \dots, p_n$ satisfying $p_j \geq 0$ and the result is the interpolated LTI system of system matrix

$$(4.43) \quad S = \frac{p_1 S_1 + \dots + p_n S_n}{p_1 + \dots + p_n}$$

```

#####Nominal polytopic LPV system for control purpose#####
for j = 1 : length(alpha)
    Pdg = psinfo(pdP, 'eval', alpha(j,:)) ;
    [adg,bdg,cdg,ddg] = ltiss(Pdg) ;
    LPVsysnominal = ss(adg,bdg,cdg,ddg) ;
    gridsys(:, :, j) = ss(adg,bdg,cdg,ddg, 'StateName', {'body travel (m)' 'body vel (m/s)'...
        'wheel travel (m)' 'wheel vel (m/s)'},...
        'InputName', {'w', 'fs_filter' }, 'Outputname', {'ab' 'zus' 'zdef' 'zdotdef' });
end

```

Figure 24: State Space Model for Each Polytopic Coordinate

6. Nominal LPV model of Semi-Active Suspension

As mentioned in previous chapter, nominal damper force f_0 is added in system model in order to satisfy dissipativity constraint of MR damper according to Static-Guo model. This approach is suitable for control as it is explained mathematically in previous chapters. When suspension control signal f_l is 400 [N], it means that f_0 is 400 [N] and control output u_l is 0 [N]. This model can be considered uncontrolled nominal damper model with current input 0.875 [A] can produce 400 [N] additional force inside of damper by changing its viscosity (e.g. Table 4). This model will be used as uncontrolled plant which is created in MATLAB (e.g. Figure 24).

```

val(:, :, 1, 1) [p1=-1, p2=0] =

A =
    body travel    body travel    body vel (m/    wheel travel    wheel vel (m
    body travel      0              1              0              0
    body vel (m/    -95.62         -2.542         95.62         2.542
    wheel travel     0              0              0              1
    wheel vel (m     803.2         21.35         -6403         -21.35

B =
    w    fs_filter
    body travel      0      0
    body vel (m/     0    0.003175
    wheel travel     0      0
    wheel vel (m     5600   -0.02667

```

Figure 25: First Vertex A and B State Space Matrix of Model

```

val(:, :, 231, 1) [p1=1, p2=1] =

A =
    body travel    body vel (m/    wheel travel    wheel vel (m
body travel         0             1             0             0
body vel (m/       -109          -20.02         109          20.02
wheel travel        0             0             0             1
wheel vel (m       915.6          168.1        -6516         -168.1

B =
    w    fs_filter
body travel         0             0
body vel (m/        0    -0.003175
wheel travel        0             0
wheel vel (m       5600          0.02667

```

Figure 26: Last Vertex A and B State Space Matrix of Model

Above figures (e.g. Figure 25) and (e.g. Figure 26) shows that A and B state-space matrices change of polytopic semi-active suspension along with all parameter trajectory according to parameter dependent plant state space representation (5.32) . Note that first and last couple point or coordinates is taken to prove plant changes in scheduling parameter range.

```

C =
    body travel    body vel (m/    wheel travel    wheel vel (m
ab          -95.62         -2.542         95.62         2.542
zus           0             0             1             0
zdef          1             0            -1             0
zdotdef       0             1             0            -1

D =
    w    fs_filter
ab         0    0.003175
zus         0             0
zdef         0             0
zdotdef       0             0

```

Figure 27: First Vertex C and D State Space Matrix of Model

```

C =
    body travel    body vel (m/    wheel travel    wheel vel (m
ab          -109          -20.02         109          20.02
zus           0             0             1             0
zdef          1             0            -1             0
zdotdef       0             1             0            -1

D =
    w    fs_filter
ab         0    -0.003175
zus         0             0
zdef         0             0
zdotdef       0             0

```

Figure 28: Last Vertex C and D State Space Matrix of Model

(e.g. Figure 25) and (e.g. Figure 26) shows that C and D state-space matrices change of polytopic semi-active suspension along with all parameter trajectory according to parameter dependent plant state space representation (5.32) . Note that first and last couple point or coordinates is taken to prove plant changes in scheduling parameter range.

6.1. Frequency Response of Nominal Damper Force

When semi-active suspension system is uncontrolled, control output u_I is not part of model and system is parametrized by only constant MR force f_I which is result of the values f_0 and u_I . System acts like passive model and scheduled as working under 400 [N] with constant 0.815 [A].

- $u_I = 0$ [N]
- $f_0 = 400$ [N]
- $f_I = 400$ [N]
- $I = 0.815$ [A]

SemiActive LPV Nominal Force vs Passive Body Vertical Acceleration

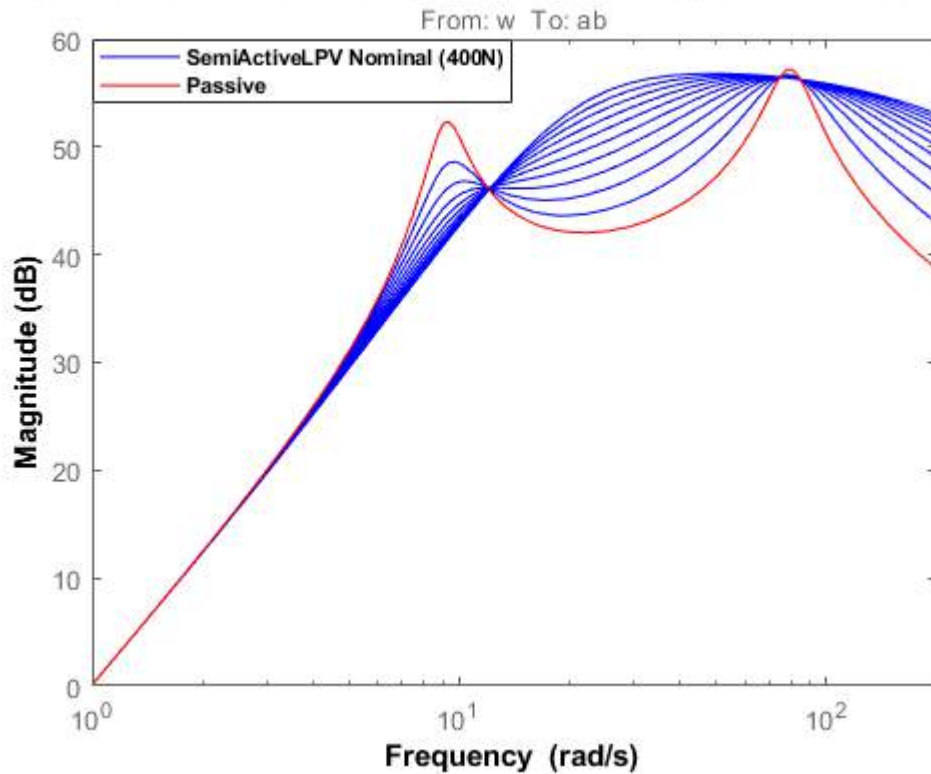


Figure 29: Nominal Damper Body Vertical Acceleration in Frequency Domain

Nominal damper applies additionally constant 400 [N] MR force in system. Body vertical acceleration gain of semi-active suspension system gives result like (e.g. Figure 29) over all scheduling parameter range (p1,p2) when applied constant 400 [N] MR force. Body vertical acceleration gain is decreased between 0-12 Hz as expected while it starts

increasing after 12 Hz when MR force or damping ratio maintain higher than passive suspension system. Comfort improves between 0 -12 Hz but after it starts decreasing since body vertical acceleration gain increases in this frequency range because of damping ratio is higher than passive suspension system.

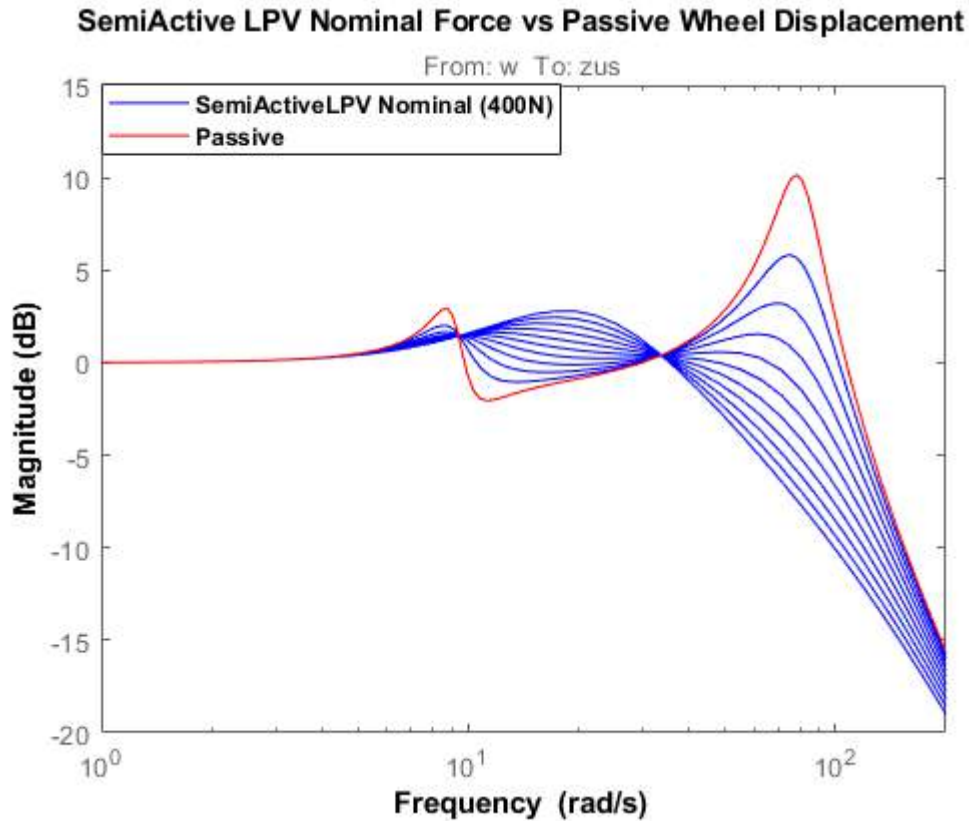


Figure 30: Nominal Damper Body Vertical Acceleration in Frequency Domain

Body wheel displacement gain is decreased between 0-10 Hz together with body vertical acceleration gain when MR force or damping ratio maintain higher than passive suspension system. Wheel displacement gain converge to 0 dB gain as damping ratio or MR force increases. Ride comfort and road holding improves together until 12 Hz. After 10 Hz, higher damping ratio provides better road holding than lower damping ratio until the limit force (e.g. Figure 30). After 12 Hz, trade-off starts between comfort and road holding as mentioned chapter 3.4.

Next chapter will be focus on solving this problem and achieving best compromised comfort – road holding objective by using LMI based robust gain scheduling control strategy with Genetic Algorithms.

7. LMI Based Gain Scheduled LPV/H ∞ Control of Semi-Active Suspension

The entire LPV model is designed for an H ∞ controller with semi-active magnetic rheological damper. This control system aims, depending on the necessary requirements, to improve driving comfort and road holding. The H ∞ method is an effective tool for improving the efficiency of a closed-loop system in the predefined frequency range. Rejection of disturbances, decoupling input/output and general stable or performance goals may be formulated in the face of complex or parametric instability as gain attenuation problems. Control for this type of problem is the key of design techniques.[17]

Three variants of output-feedback H ∞ synthesis

- Control for continuous-time systems with Standard H ∞ controller. The Riccati and LMI solutions are introduced for this problem
- Multi-objective H ∞ design (mixed H₂/H ∞ synthesis combined with LMI regions closed-loop pole positioning)
- Discrete-time H ∞ control Riccati- or LMI-based approaches

The goal of this synthesis is to find an LPV controller that is quadratically stable with the closed loop system for an optimum positive, optimum and real γ optimal quadratic RMS performance. A convex combination of these local controllers would be the LPV/H ∞ controller by using Lyapunov functions and solving LMIs.

7.1. H ∞ Control

The H ∞ standard for a stable transfer function $G(s)$ is its most significant RMS input/output gain.

$$(7.1) \quad \|G\|_{\infty} = \sup \frac{\|y\|_{L_2}}{\|u\|_{L_2}} \quad u \in L_2 \text{ and } u \neq 0$$

where L_2 is the space of signals with finite energy and $y(t)$ is the output of the system G for a certain input $u(t)$. This norm also corresponds to the maximum gain of the frequency response($G(jw)$)[14]

$$(7.2) \quad \|G\|_{\infty} = \sup \frac{\|y\|_{L_2}}{\|u\|_{L_2}} \quad \sigma_{\max}(G(jw))$$

The problem of H ∞ control is one of disturbance rejection in its abstract standard formulation. In particular, the control loop of Figure 31 minimizes the closed-loop RMS gain from w to z . This can be interpreted to minimize the effect on the out performance z in the worst-case disturbance w . [18]

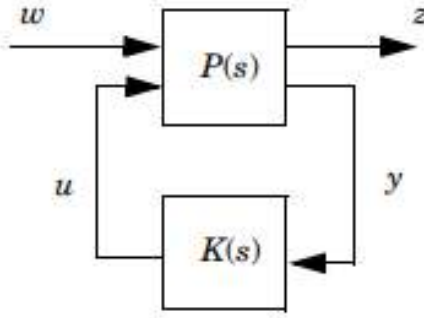


Figure 31: H^∞ Control (e.g. [14])

Partitioning the plant $P(s)$ as

$$(7.3) \quad \begin{bmatrix} Z(s) \\ Y(s) \end{bmatrix} = \begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix} \begin{bmatrix} W(s) \\ U(s) \end{bmatrix}$$

A linear-fractional expression is given by the closed-loop transfer function from w to z

$$(7.4) \quad F(P, K) = P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21}$$

Therefore the optimal H^∞ control aims to minimize the $\|F(P, K)\|_\infty$ over all stabilizing LTI controllers $K(s)$. Alternatively, for closed-loop RMS, we may specify the maximum value of γ and ask the question that there is any current $K(s)$ stability controller which ensures $\|F(P, K)\|_\infty < \gamma$. This is called the suboptimal H^∞ control problem and γ is called the prescribed H^∞ performance. In this traditional formulation, a range of control problems can be recast. [19,20]

7.2. Riccati and LMI Based Approaches

Although the LMI method is much more computerized for major problems, it has the merit of removing the regularity constraints attached to the Riccati solution. The plant $P(s)$ are given in state-space form since both approaches are state-space methods

$$(7.5) \quad \begin{aligned} \dot{x} &= Ax + B_1 + B_2u \\ z &= C_1x + D_{11} + D_{12}u \\ y &= C_3x + D_{21} + D_{22}u \end{aligned}$$

The Riccati-based approach is applicable to plants P satisfying

- D_{12} and D_{21} have full rank (A1)
- P_{12} and P_{21} have no zeros on the $j\omega$ -axis (2)

Provided $\gamma > 0$, the conditions for the existence of internally stabilizing controls $K(s)$ are necessary and sufficient as follows

$$(7.6) \quad \|F(P, K)\|_\infty < \gamma$$

In particular, the H^∞ performance γ is achievable only if and when

$$(7.7) \quad \gamma > \max(\sigma_{\max}(I - D_{12}D_{12}^+)D_{11}), \sigma_{\max}(D_{11}(I - D_{21}D_{21}^+))$$

$$(7.8) \quad H - \lambda \in = \left[\begin{array}{cc|cc} A & 0 & 0 & \gamma^{-1}B_1 & B_2 \\ 0 & -A^T & -C_1^T & 0 & 0 \\ \hline C_1 & 0 & -I & \gamma^{-1}D_{11} & D_{12} \\ 0 & \gamma^{-1}B_1^T & \gamma^{-1}D_{11}^T & -I & 0 \\ 0 & B_2^T & D_{12}^T & 0 & 0 \end{array} \right] - \lambda \left[\begin{array}{c|c} I & 0 \\ \hline 0 & 0 \end{array} \right]$$

$$(7.9) \quad J - \lambda \in = \left[\begin{array}{cc|cc} A^T & 0 & 0 & \gamma^{-1}C_1^T & C_2^T \\ 0 & -A & -B_1 & 0 & 0 \\ \hline B_1^T & 0 & -I & \gamma^{-1}D_{11}^T & D_{21}^T \\ 0 & \gamma^{-1}C_1 & \gamma^{-1}D_{11}^T & -I & 0 \\ 0 & C_2 & D_{21} & 0 & 0 \end{array} \right] - \lambda \left[\begin{array}{c|c} I & 0 \\ \hline 0 & 0 \end{array} \right]$$

stabilizing solutions for each X_∞ and Y_∞

- $X_\infty \succeq 0, \quad Y_\infty \succeq 0, \quad \rho(X_\infty Y_\infty) < \gamma^2$

The optimum γ_{opt} performance of H^∞ can be determined by means of this characterisation by bisection is the so-called γ -iterations. An H^∞ controller with performance $\gamma \geq \gamma_{opt}$ is then given in the formulation of explicit formulas, including X_∞ , Y_∞ , and the plant state-space matrices. Singular H^∞ problems in (A1) and (A2) conditions involve a minor disruption to be regularised. A noteworthy exception is that the optimum H^∞ performance is determined directly when D_{12} or D_{21} is unique [22].

The LMI method applies in contrast to any plant and does not require γ -iterations when compare with Riccati method. Instead, the H^∞ performance is directly optimized by solving the following LMI problem[23,24]

Minimize γ over $R = R^T$ and $S = S^T$ such that

$$(7.10) \quad \begin{bmatrix} N_{12} & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} AR + RA^T & RC_1^T & B_1 \\ C_1 R & -\gamma I & D_{11} \\ B_1^T & D_{11}^T & -\gamma I \end{bmatrix} \begin{bmatrix} N_{12} & 0 \\ 0 & I \end{bmatrix} < 0$$

$$(7.11) \quad \begin{bmatrix} N_{21} & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A^T S + SA & SB_1 & C_1^T \\ B_1^T S & -\gamma I & D_{11}^T \\ C_1 & D_{11} & -\gamma I \end{bmatrix} \begin{bmatrix} N_{21} & 0 \\ 0 & I \end{bmatrix} < 0$$

where N_{12} and N_{21} denote bases of the null spaces of and (C_2, D_{21}) , respectively. This is beyond the reach of the LMI. Notice that LMI constraints (7.10) and (7.11) amount to those inequalities that are solved in H^∞ Riccati equations. An explicit formulas for deriving H^∞ controllers from any solution (R, S, γ) are again available (7.10) and (7.11). [25]

7.3. Quadratic Lyapunov Functions

The definition of quadratic stability and quadratic efficiency is useful in the study of linear time-varying systems [14]

$$(7.12) \quad E(t)\dot{x}(t) = A(t)x(t), \quad x(0) = x_0$$

Given these structures, the presence of a positive and definite quadratic Lyapunov function is a necessary condition for asymptotic stability as follows

$$(7.13) \quad V(x) = x^T P x$$

Lyapunov function derivative gives following inequality

$$(7.14) \quad \frac{dV(x(t))}{dt} < 0$$

over all trajectories. With regard to $Q := P^{-1}$, this is always counterpart following inequality at all times t

$$(7.15) \quad A(t)QE(t)^T + E(t)QA(t)^T < 0$$

The quadratic stability evaluation is not necessarily tractable as (7.14) puts an infinite number of constraints on Q . However, (7.14) in the following cases can be reduced to a finite set of LMI constraints. [14]

- $A(t)$ and $E(t)$ are fixed affine functions of some time-varying parameters $p_1(t), \dots, p_n(t)$.

$$(7.16) \quad A(t) = A_0 + p_1(t)A_1 + \dots + p_n(t)A_n$$

$$(7.17) \quad E(t) = E_0 + p_1(t)E_1 + \dots + p_n(t)E_n$$

This is referred to in Chapter 4.1 as an affine parameter based model.

- $A(t) + jE(t)$ ranges in a fixed polytope of matrices is given by

$$(7.18) \quad A(t) = a_1(t)A_1 + \dots + a_n(t)A_n$$

$$(7.19) \quad E(t) = a_1(t)E_1 + \dots + a_n(t)E_n$$

$$(7.20) \quad a_i(t) \geq 0 \text{ and } \sum_{i=1}^n a_i(t) = 1$$

This is referred to in chapter 5.2 as a polytopic model.

The first case refers to systems with state-space equations that are accurately conditioned on various physical parameters and the other to Robustness Analysis time-change systems modeled on an envelope of LTI systems. Notice that Lyapunov's quadratic function guarantees consistency for arbitrarily rapid changes in time, their main source of conservatism.

Quadratic functions of Lyapunov are also useful for robust RMS performance. Specifically, a time-varying system

$$(7.21) \quad \begin{cases} E(t)\dot{x} = A(t)x + B(t)u \\ y = C(t)x + D(t)u \end{cases}$$

Satisfies zero initial states

$$(7.22) \quad \|y\|_{L_2} < \|u\|_{L_2}$$

If there is a positive Lyapunov function $V(x) = x^T P x$, for all bounded $u(t)$ input

$$(7.23) \quad \frac{dV(x(t))}{dt} + y^T y - \gamma^2 u^T u < 0$$

As far as $Q = P^{-1}$ is concerned, it equals the the family of “Bounded Real Lemma” inequalities[23]

$$(7.24) \quad \begin{bmatrix} A(t)QE^T + E(t)QA^T & B(t) & E(t)QC(t)^T \\ B(t)^T & -\gamma I & D(t)^T \\ C(t)QE(t)^T & D(t) & -\gamma I \end{bmatrix} < 0$$

The smallest γ for which Lyapunov exists is known quadratic H^∞ performance. This is an upper limit to the system's worst case RMS gain.

7.4. LMI Formulation

Appropriate quadratically stable LMI conditions are as follows [26,27,28]

- Consider the parameter-dependent affine model

$$(7.25) \quad E(p)\dot{x} = A(p)x$$

$$(7.26) \quad A(p) = A_0 + p_1 A_1 + \dots + p_n A_n$$

$$(7.27) \quad E(t) = E_0 + p_1 E_1 + \dots + p_n E_n$$

Where $p_i(t) \in [\underline{p}_i, \overline{p}_i]$ and let

$$(7.28) \quad V = \{(\omega_1, \dots, \omega_n) : \omega_i \in \{\underline{p}_i, \overline{p}_i\}\}$$

denote the set of corners of the required parameter box. The dynamical system (7.26) is quadratically stable when symmetrical matrices Q and $\{M_i\}_{i=1}^n$ are available

$$(7.29) \quad A(\omega)QE(\omega)^T + E(\omega)QA(\omega)^T + \sum_i \omega_i^2 M_i < 0 \text{ for all } \omega \in V$$

$$(7.30) \quad A_i P E_i^T + E_i P A_i^T + M_i \geq 0 \text{ for } i = 1, \dots, n$$

$$(7.31) \quad M_i \geq 0$$

$$(7.32) \quad Q \geq I$$

- Consider the parameter-dependent polytopic model

$$(7.33) \quad E(t)\dot{x} = A(t)x$$

$$(7.34) \quad A(t)x + jE(t) \in C_0(A_1 + jE_1, \dots, A_n + jE_n)$$

is quadratically stable if there exist a symmetric matrix Q and scalars $t_{ij} = t_{ji}$ such that

$$(7.35) \quad A_i Q E_j^T + E_j Q A_i^T + A_j Q E_i^T + E_i Q A_j^T < 2t_{ij} I \text{ for } i, j \in (1, \dots, n)$$

$$(7.36) \quad Q > I$$

$$(7.37) \quad \begin{pmatrix} t_{11} & \dots & t_{1n} \\ \vdots & \ddots & \vdots \\ t_{1n} & \dots & t_{nn} \end{pmatrix}$$

- No parameter p_i enters both $A(t)$ and $E(t)$ in the affine case, $A_i = 0$ or $E_i = 0$ for all i . The conditions (7.30) and (7.31) can then be omitted and the parameter box may be verified at the corners (7.15).
- Either $A(t)$ or $E(t)$ is constant in the polytopic case. The only needed thing to do is solve (7.35) and (7.36) for $t_{ij} = 0$.

Similar conditions for robust RMS performance are available. For polytopic systems, it is followed that LMI conditions guarantee RMS efficiency γ .

A symmetric matrix Q and scalars $t_{ij} = t_{ji}$ for $i, j \in \{1, \dots, n\}$ exist such that

$$(7.38) \quad \begin{bmatrix} A_i Q E_j^T + E_j Q A_i^T + A_j Q E_i^T + E_i Q A_j^T & H & H \\ B_i^T + B_j^T & -2\gamma I & H \\ C_i Q E_i^T + C_i Q E_j^T & D_i + D_j & -2\gamma I \end{bmatrix} < 2t_{ij}x1$$

$$(7.39) \quad Q > 0$$

$$(7.40) \quad \begin{pmatrix} t_{11} & \cdots & t_{1n} \\ \vdots & \ddots & \vdots \\ t_{1n} & \cdots & t_{nn} \end{pmatrix} < 0$$

7.5. Robust Gain Scheduled Control

Gain scheduling is commonly used method for controlling classes of nonlinear or linear time-varying systems. Gain scheduling consists of an LTI controller for each operating point and a switching controller if working conditions change instead of looking for a single stable LTI controller over the entire operating range.. This section offers systemic tools to develop H^∞ controllers for linear parameter-dependent systems. These tools refer to time-varying and/or non-linear systems whose linear dynamics are relatively well approximated by affine models depending on the parameter. The following synthesis technique refers to parameter-dependent plants with equations.

$$(7.41) \quad P(., p) \begin{cases} \dot{x} = A(p)x + B_1(p)w + B_2u \\ z = C_1(p)x + D_{11}(p)w + D_{12}u \\ y = C_2x + D_{21}w + D_{22}u \end{cases}$$

Where

$$(7.42) \quad p(t) = (p_1(t), \dots, p_n(t))$$

$$(7.43) \quad \underline{p}_i \leq p_i(t) \leq \overline{p}_i$$

is a time-varying vector of physical parameters such as velocity, angle of attack, stiffness, suspension deflection and $A(.), B_1(.), C_1(.), D_{11}(.)$, are affine functions of $p(t)$. This is a simple model of systems that rely on physical coefficients during operation, for their dynamic equations. When these coefficients experience great variations, a single stable LTI controller is often impossible to achieve higher performance across the operating range. If the parameter values are calculated in real time, controllers that implement these measurements can then be used to change the current operating conditions [29]. Such controllers should be designed for calculating parameters. This control technique usually achieves better performance against large variations in operating conditions.. Note that $p(t)$ may contain part of the state vector x itself provided that it is measurable in the respective states.

If the parameter vector $p(t)$ takes values in a box of R^n with corner

$$(7.44) \quad \{\pi_i\}_{i=1}^n \text{ and } N = 2^n$$

The plant system matrix

$$(7.45) \quad S(p) := \left[\begin{array}{c|cc} A(p) & B_1(p) & B_2 \\ \hline C_1(p) & D_{11}(p) & D_{12} \\ C_2 & D_{21} & D_{22} \end{array} \right]$$

ranges in a matrix polytope with vertices $S(\pi_i)$. Specifically, given any convex decomposition

$$(7.46) \quad p(t) = a_1\pi_1 + \dots + a_N\pi_N, \quad a_i \geq 0, \quad \sum_{i=1}^N a_i = 1$$

the system matrix $S(p)$ is given by p over the corners of the parameter box as follows

$$(7.47) \quad S(p) = a_1S(\pi_1) + \dots + a_NS(\pi_N)$$

This suggests looking for parameter-dependent controllers with equations

$$(7.48) \quad K(., p) \begin{cases} \dot{\zeta} = A_K(p)\zeta + B_K(p)y \\ u = C_K(p)\zeta + D_K(p)y \end{cases}$$

and a vertex property. Provided the convex decomposition of the present $p(t) = \sum_{i=1}^N a_i\pi_i$, the values of $A_K(p)$, $B_K(p)$, . . . and the values $A_K(\pi_i)$, $B_K(\pi_i)$, . . . are extracted at the parameter box corners as following

$$(7.49) \quad \begin{bmatrix} A_K(p) & B_K(p) \\ C_K(p) & D_K(p) \end{bmatrix} = \sum_{i=1}^N a_i \begin{bmatrix} A_K(\pi_i) & B_K(\pi_i) \\ C_K(\pi_i) & D_K(\pi_i) \end{bmatrix}$$

The controller state-space matrices at the operating point $p(t)$ are obtained by convex interpolation of the LTI vertex controllers

$$(7.50) \quad K_i = \begin{bmatrix} A_K(\pi_i) & B_K(\pi_i) \\ C_K(\pi_i) & D_K(\pi_i) \end{bmatrix}$$

This approach allows the controller matrices to be yielded smoothly by the parameter measurements $p(t)$.

Consider the following H^∞ synthesis issue with regard to the interconnection of (e.g. Figure 32) for this class of controllers

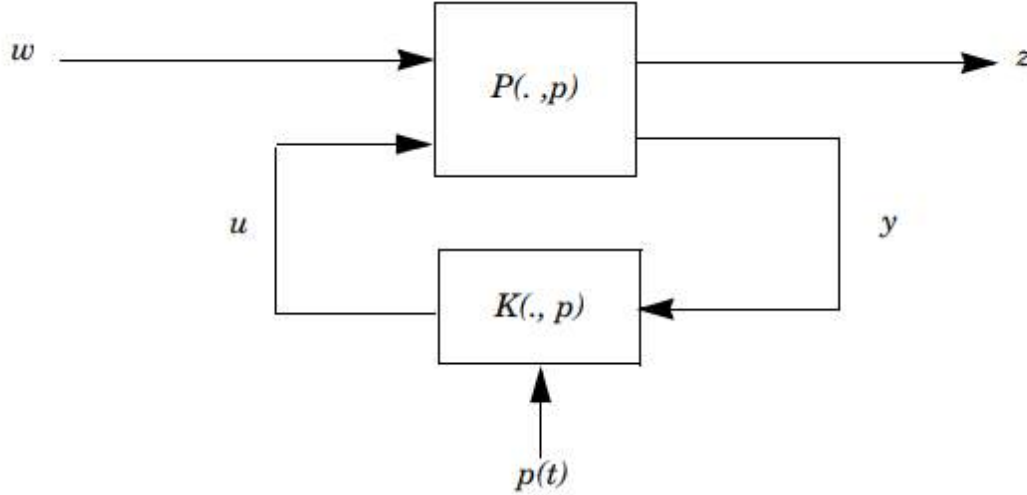


Figure 32: Gain Scheduled H^∞ Problem (e.g. [14])

Develop a gain-scheduled controller $K(., p)$ that meets the the vertex property

- For all permissible parameter path $p(t)$, the closed-loop system is stable
- The worst-case closed-loop RMS gain from w to z is not greater than some level $\gamma > 0$

This synthesis problem can be reduced to the following LMI problem by using the principle of quadratic H^∞ performance to implement the RMS gain limit by observing that the closed-loop system is a polytopic [30, 31]

Find two R and S symmetrical matrixes that

$$(7.51) \quad \begin{bmatrix} N_{12} & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A_i R + R A_i^T & R C_{1i}^T & B_{1i} \\ C_{1i} R & -\gamma I & D_{11i} \\ B_1^T & D_{11i}^T & -\gamma I \end{bmatrix} \begin{bmatrix} N_{12} & 0 \\ 0 & I \end{bmatrix} < 0, i = 1, \dots, N$$

$$(7.52) \quad \begin{bmatrix} N_{21} & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A_i^T S + S A & S B_{1i} & C_{1i}^T \\ B_{1i}^T S & -\gamma I & D_{11i}^T \\ C_{1i} & D_{11i} & -\gamma I \end{bmatrix} \begin{bmatrix} N_{21} & 0 \\ 0 & I \end{bmatrix} < 0, i = 1, \dots, N$$

$$(7.53) \quad \begin{bmatrix} R & I \\ I & S \end{bmatrix} > 0$$

Where

$$(7.54) \quad \begin{bmatrix} A_i & B_{1i} \\ C_{1i} & D_{11i} \end{bmatrix} = \begin{bmatrix} A(\pi_i) & B_1(\pi_i) \\ C_1(\pi_i) & D_{11}(\pi_i) \end{bmatrix}$$

N_{12} and N_{21} are bases of the null spaces of $((B_2^T, D_{12}^T))$ and (C_2, D_{21}) , respectively. Note that requiring B_2, C_2, D_{12}, D_{21} plant matrices to be independent of $p(t)$ can be restrictive in practice. When B_2 and D_{12} rely on $p(t)$ a simple task is to filter u through

a low-pass filter with a sufficiently wide bandwidth to implement these conditions. Parameter dependence of B_2 , and D_{12} can be rejected by adding this filter into A matrix of the augmented plant. Likewise, C_2 and D_{12} parameter dependency can be removed by filtering the output [32].

7.6. H_∞ Problem and Control Strategy of Semi-Active Suspension with MR Damper

The control configuration for semi-active suspensions is given in Fig. 32. The controlled outputs are:

- $\ddot{z}_s = \text{Body vertical acceleration}$
- $z_{us} = \text{Wheel displacement}$

Control outputs are the ride comfort and road holding improvement respectively and they are defined as the performance objectives and control goals in Chapter 2 and 3. The measurement outputs are:

- $z_{def} = \text{Suspension deflection}$
- $\dot{z}_{def} = \text{Suspension deflection velocity}$

They are specified as $z_{def} = z_s - z_{us}$ and $\dot{z}_{def} = (\dot{z}_s - \dot{z}_{us})$ respectively and need to be calculated for the scheduling parameters. To obtain the closed-loop performances in Chapter 3, the weighting functions on controlled outputs are given second order transfer function for control outputs $\{W_{\ddot{z}_s}, W_{z_{us}}\}$ and disturbance input W_{z_r} are given by

$$(7.55) \quad W_{z_r} = 3 \times 10^{-2}$$

$$(7.56) \quad W_{\ddot{z}_s} = \frac{s^2 + 2\zeta_{11}\Omega_{11}s + \Omega_{11}^2}{s^2 + 2\zeta_{12}\Omega_{12}s + \Omega_{12}^2}$$

$$(7.57) \quad W_{z_{us}} = \frac{s^2 + 2\zeta_{21}\Omega_{21}s + \Omega_{21}^2}{s^2 + 2\zeta_{22}\Omega_{22}s + \Omega_{22}^2}$$

As mentioned in previous chapter 6.5, B_2 and D_{12} matrices of plant should parameter independent in order to apply H_∞ gain scheduling strategy by solving LMI problems (7.51). LPV semi-active suspension model from Equation (5.32) whose input B_s and feedthrough matrix D_{sz} are parameter dependent. This problem can be solved by adding low-pass filter control output u_l [32].

$$(7.58) \quad Wf: \begin{pmatrix} \dot{x}_f \\ u_l \end{pmatrix} = \begin{pmatrix} A_f & B_f \\ C_f & 0 \end{pmatrix} \begin{pmatrix} x_f \\ u \end{pmatrix}$$

With

$$(7.59) \quad \|Wf\| \leq 1$$

Where A_f, B_f, C_f are constant matrices.

$$(7.60) \quad W_f = \frac{\Omega_f}{s + \Omega_f}$$

From equation (5.32) and (7.58), LPV system with two scheduling parameters p_1 and p_2 for H_∞ gain scheduled controller is completed with a control oriented closed loop model.

$$(7.61) \quad x = \begin{bmatrix} x_s \\ x_f \end{bmatrix}$$

$$(7.62) \quad P: \begin{cases} \dot{x} = A(p_1, p_2)x + Bu + B_1w \\ z = C_z(p_1, p_2)x \\ y = Cx \end{cases}$$

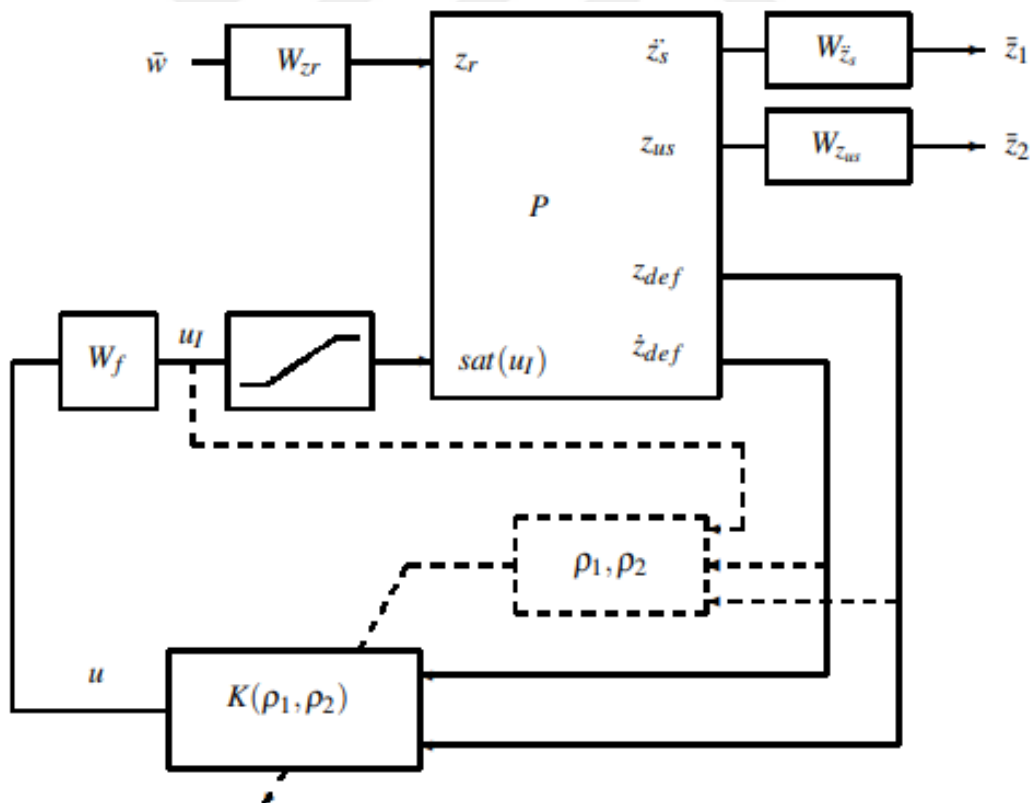


Figure 33: Block Diagram of Semi-Active Suspension Control (e.g. [8])

In this thesis, a polytopic approach for the LPV control design is established. Scheduling parameters p_1 and p_2 are treated as different independent parameters from (5.45) and $\bar{p}$ belongs to a polytope Θ whose vertices (e.g. Figure 22) are

- $\bar{p}_1 = (1,1)$
- $\bar{p}_2 = (-1,1)$
- $\bar{p}_3 = (-1,0)$
- $\bar{p}_4 = (1,0)$

Vector v represents all weighting function parameters which will be used with Genetic Algorithm as a part of decision vector in next chapter to find the best compromise weight parameters and used in augmented system equations with $\bar{p}$ in order to define augmented system

$$(7.63) \quad v = [\Omega_f, \Omega_{11}, \Omega_{12}, \zeta_{11}, \zeta_{12}, \Omega_{21}, \Omega_{22}, \zeta_{21}, \zeta_{22}]$$

Considering the augmented system made of the plant (e.g. Figure 33) and the weighting functions, represented by

$$(7.64) \quad \begin{aligned} \dot{\zeta} &= A^v(\bar{p})\zeta + B_1^v(\bar{p})\bar{w} + B_2^v u \\ \bar{z} &= C_1^v(\bar{p})\zeta + D_{11}^v(\bar{p})\bar{w} + D_{12}^v u \\ y &= C_2\zeta + D_{21}\bar{w} \end{aligned}$$

where x_w being the state vector of weighting functions

$$(7.65) \quad \zeta = \begin{bmatrix} x \\ x_w \end{bmatrix}$$

The LPV controller is defined as follows

$$(7.66) \quad K(\bar{p}): \begin{pmatrix} \dot{x}_c \\ u \end{pmatrix} = \begin{pmatrix} A_c^v(\bar{p}) & B_c^v(\bar{p}) \\ C_c^v(\bar{p}) & D_c^v(\bar{p}) \end{pmatrix} \begin{pmatrix} x_c \\ y \end{pmatrix}$$

where x_c , y and u are respectively the state, the are the input and output of the system-associated controller respectively (6.59). Note that all matrices should have appropriate dimensions.

Since v is the weighting function vector, it is used as the equation (7.64) and (7.66) vector to emphasize the dependency of the general plant on the parameters of the weighting functions, and then of the controller.

H^∞ problem with the LPV whose purpose is to find an LPV controller type $K(p)$ (7.66). The closed-loop system is therefore quadratically stable for a positive real γ . The L_2 induced norm of the operator mapping $\bar{w}$ into $\bar{z}$ is constrained by γ

$$(7.67) \quad \forall \bar{p} \in \Theta, \quad \frac{\|z_w\|_{L_2}}{\|w\|_{L_2}} < \gamma$$

The polytopic approach with a Lyapunov quadratic function is used in this section. For known weighting functions mentioned above, and an appropriate, predefined, positive scalar γ is indicated. Equations (6.68) and (6.69) provides the sufficient condition to solve the LPV/ H^∞ problem where the decision variables are $X, Y, \hat{A}, \hat{B}, \hat{C}$ and $\hat{D}$ [33.34]. The LMI problem for each vertex is solved. Note that the LMI problem involves weighting functions parameter set v to optimize the controller. Optimization will be achieved from Genetic Algorithms by using weightinh function parameter set to solve LMI problem in next part.

$$(7.68) \quad \begin{bmatrix} M_{11}^v(\bar{p}_i) & * & * & * \\ M_{21}^v(\bar{p}_i) & M_{22}^v(\bar{p}_i) & * & * \\ M_{31}^v(\bar{p}_i) & M_{32}^v(\bar{p}_i) & -\gamma I_m & * \\ M_{41}^v(\bar{p}_i) & M_{42}^v(\bar{p}_i) & M_{43}^v(\bar{p}_i) & -\gamma I_p \end{bmatrix} < 0$$

$$(7.69) \quad \begin{bmatrix} X & I \\ I & Y \end{bmatrix} > 0 \text{ for } i = 1:4$$

Where

$$(7.70) \quad \begin{aligned} M_{11}^v(\bar{p}_i) &= A^v(\bar{p}_i)X + XA^v(\bar{p}_i)^T + B_2\hat{C}(\bar{p}_i)X + \hat{C}(\bar{p}_i)^TB_2^T \\ M_{21}^v(\bar{p}_i) &= \hat{A}(\bar{p}_i) + A^v(\bar{p}_i)^TY + C_2^T\hat{D}(\bar{p}_i)^TB_2^T \\ M_{22}^v(\bar{p}_i) &= YA^v(\bar{p}_i) + A^v(\bar{p}_i)^TY + \hat{B}(\bar{p}_i)C_2 + C_2^T\hat{B}(\bar{p}_i)^T \\ M_{31}^v(\bar{p}_i) &= B_1^v(\bar{p}_i)^T + D_{21}^T\hat{D}(\bar{p}_i)^TB_2^T \\ M_{32}^v(\bar{p}_i) &= B_1^v(\bar{p}_i)^TY + D_{21}^T\hat{B}(\bar{p}_i)^T \\ M_{41}^v(\bar{p}_i) &= C_1^v(\bar{p}_i)X + D_{12}\hat{C}(\bar{p}_i) \\ M_{42}^v(\bar{p}_i) &= C_1^v(\bar{p}_i) + D_{12}\hat{D}(\bar{p}_i)C_2 \\ M_{43}^v(\bar{p}_i) &= D_{11}^v(\bar{p}_i) + D_{12}\hat{D}(\bar{p}_i)D_{21} \end{aligned}$$

The controller K_{ci} at vertex i is reconstructed and given by

$$\begin{aligned}
A_c^v(\bar{p}_i) &= N^{-1}(\hat{A}(\bar{p}_i) - Y\hat{A}(\bar{p}_i)X - YB_2D_c^v(\bar{p}_i)C_2X)M^{-T} \\
&\quad - B_c^v(\bar{p}_i)C_2XM^{-T} - M^{-1}YB_2C_c^v(\bar{p}_i) \\
(7.71) \quad B_c^v(\bar{p}_i) &= N^{-1}(\hat{B}(\bar{p}_i) - YB_2D_c^v(\bar{p}_i)) \\
C_c^v(\bar{p}_i) &= (\hat{C}(\bar{p}_i) - D_c^v(\bar{p}_i)C_2X)M^{-T} \\
D_c^v(\bar{p}_i) &= \hat{D}(\bar{p}_i)
\end{aligned}$$

where M, N are defined such that $MN^T = I_n - XY$ that could be resolved by a single decomposition value and a Cholesky factorization. The global H^∞ /LPV controller is the convex combination of these local controller.[32,34]

7.7. Selection of Weighting Functions with Genetic Algorithms

The key part of the H^∞ control design is the selection of weighting functions that rely on engineering expertise and skills. However, even with good engineers, this is not always an easy job. The complexity of choosing weighting functions in many real applications is also greatly increased because of the performance specification. It is essentially the achievement of the best possible outcomes in the optimal design or the optimal improvement of multiple goals for multi-objective design. Therefore it remains an essential problem to optimize weighting functions for the desired output. In the literature, some works concentrate on this reason [35] No matter how complex they are, they suggest considering each system as a combination of first- and second-order sub-systems for which the weighting functions are easy to locate. However in the general case, no simple protocol exists to discover these functions. One common approach is to carry on trial and error.

Recently, [36] and [37] presented the idea of using an optimization function. It seems normal to choose GAs since their formulation is appropriate for this type of problem. In this analysis, optimum weighting functions for the H^∞ /LPV regulation of semi-active suspension systems are achieved by the use of GAs. Suspension system performance requirements as described in Chapters 2 and 3. To achieve the best compression performance via weight functions, a proper objective function is formulated. This objective function is in fact a convex combination of two specifications:

- Passenger comfort
- Road holding

Body vertical acceleration and wheel displacement transfer function gain will be minimized in specified frequencies in order to achieve performance objectives and tune the suspension system via objective function which represent RMS gain of desired performance criteria. The best weighting functions are selected for the H_∞ /LPV controller in the last step.

7.7.1. Genetic Algorithms

Genetic algorithms have been investigated and developed for years. There are several references such as [38,39] in the literature to use genetic algorithms in machine learning and optimization applications. The algorithms are based on the natural selection mechanism and have proved extremely productive in optimizing many realistic applications such as finance and investing techniques, robotics, engineering. In spite of the high computing costs for analyzing and improving the population, they would possibly use global optimization strategies using probabilistic, multi-point searches, random combinations (crossover, mutation) and previous iteration information. Compared with other search techniques, e.g. gradient methods, GAs have a major advantage in searching irrespective of the presence of objective functions and limitations.

As start with a random population. Genetic operation will acquire the new population: selection, crossover and mutation. Using the selection process, the best individuals on the basis of their fitness values are chosen; crossover and mutation are then used to create the new population. Genetic activity in populations persists until the optimization requirements are met or a number of generations completed.

- *Fitness Function*

The fitness of an individual provides advantage in order to select better individuals. An individual that has high fitness value has a great chance to being chosen[36].

- *Selection*

This stage is to arrange and copy individuals by order of fitness satisfaction. The higher the fitness value associated with an individual, the increasing the chances for the individual to take part in the next generation.. [36].

- *Crossover*

This is the main operation of the parent population. It consists of exchanging chain sections between two selected 'parents' and creating two new children.' This exchange could be caused by one or more points. The Fig. 34 is an example of a binary crossover coding.[36]

- *Mutation*

Mutation works on single individual by randomly modifying a portion of it. For binary coding, one or two bits are inverted in a chromosome Fig.34. Other strategies may be used by subtracting or replacing the mutation with a random value picked from a subset of values [36]

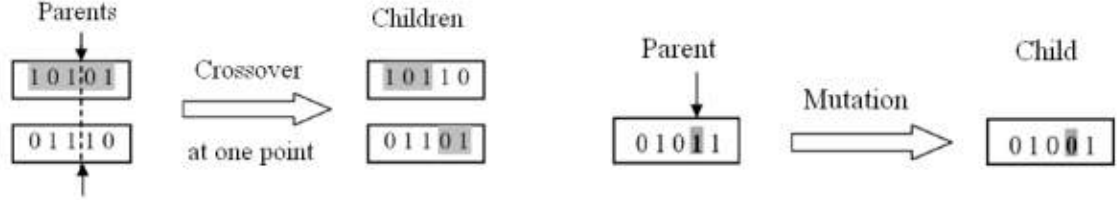


Figure 34: Crossover and Mutation Process (e.g. [36])

One well-known application of the GAs would be to find the optimum solution to the various overlapping objectives multi-objective optimization problem. In practice, this is a common problem and can be defined as follows

$$(7.72) \quad \min F(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_{n_{obj}}(x) \end{bmatrix}, \quad n_{obj} \geq 2, \quad \text{and} \quad x \in C_{set}$$

Where x is defined as the decision vector, the set of possible decision vectors is set by C_{set} and $F(x)$ is the objective vector. The presence of an optimal solution for x is to reduce all objective functions simultaneously $f_1, f_2, f_{n_{obj}}$. Therefore, the "Pareto optimal" definition is generally used to define the solution of the multi-target optimization problem in this case.

7.7.2. Implementation of Genetic Algorithms

The attenuation level γ is to be minimized in a standard H_∞ /LPV problem to fulfill the goals of H_∞ performance objectives. Instead the approach can minimize an objective function that reflects the true performance targets through the optimisation of genetic algorithms. The issue of optimization of the interest is to minimize this objective function rather than to minimize γ .

The problem of optimization for semi-active suspension control is

$$(7.73) \quad \min J(v, \gamma) = \begin{bmatrix} J_{comfort}(v, \gamma) \\ J_{roadholding}(v, \gamma) \end{bmatrix} \quad \{v, \gamma\} \in R^{10}$$

Dimension of the searching space is 10x1 since there are eight parameters for the weighting functions, one from low pass filter and v and attenuation level scalar γ .

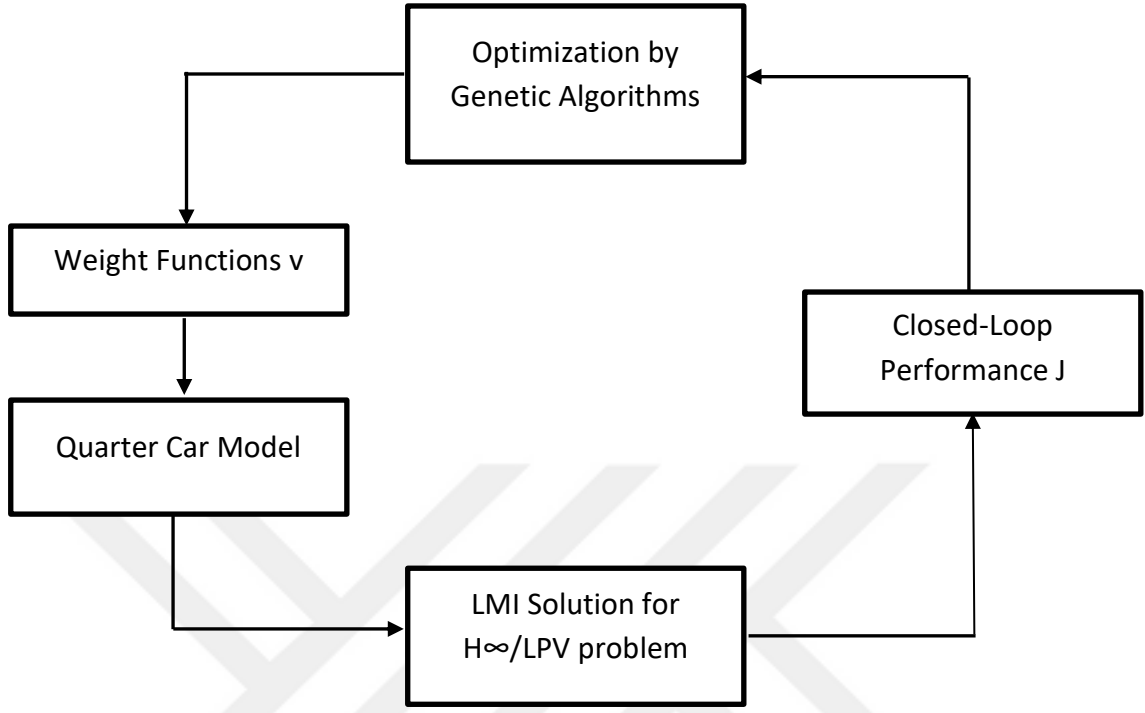


Figure 35: Controller Optimization for Semi-Active Suspension Systems Using GA Optimization

The two aims are focused on the comfort and road handling performances in Chapter 2 and 3 in order to reduce the acceleration of the vehicle's body at low and medium frequencies and the displacement of the the wheel at high speeds for each vertex of the polytope considered. Frequency-based objective functions in equation (7.72) are considered

$$(7.73) \quad J_{comfort} = \sum_{i=1}^4 \int_0^{12} \frac{\ddot{z}_s}{z_r} (f)_i df$$

$$(7.74) \quad J_{roadholding} = \sum_{i=1}^4 \int_{10}^{20} \frac{z_{us}}{z_r} (f)_i df$$

These design objective functions with the ones in chapter 4 and the index “i” stands for the i^{th} vertex of the polytope Θ whose parameters $p(-1,0)$, $p(-1,1)$, $p(1,0)$, $p(1,1)$ are shown (e.g. Figure 22). Therefore, the number of elements in each sum is calculated four. The viability of the LMIs (7.68) and (7.69) can be chosen by the GA-generated "bad decision vectors. Bad vectors of decision can cause infeasible LMI solution. Repeating cross-over or mutation will solve this problem before algorithms can find a potential solution. Objective value J can be allocated as an infinite value for unsuccessful solutions and after some generations the selection process can discard them.

In many applications, a stable stabilizing control unit is needed to maintain the performance of the closed-loop system with input saturation. Stable controllers can be more easily modified and implemented than unstable controllers. Although the sensor feedback fails, the close loop system remains stable. In order to achieve a secure H^∞ /LPV controller for LPV systems, to obtain a stable LPV controller, it is sufficient to ensure that all local controllers [6].

Steps of optimization weighting functions with GA for H^∞ /LPV synthesis

- Step 1

Determine upper and lower bounds of weighting functions in Bode plot in terms of comfort target and road holding by considering chosen vertices.

Table 7: Vertical Acceleration Weight Function bounds

Ω_{11}	Ω_{12}	ζ_{11}	ζ_{12}
20	4	80	4
80	20	105	40

Upper and lower bounds are considered +10 and -10 dB gain from passive suspension vertical acceleration bode plot characteristic.

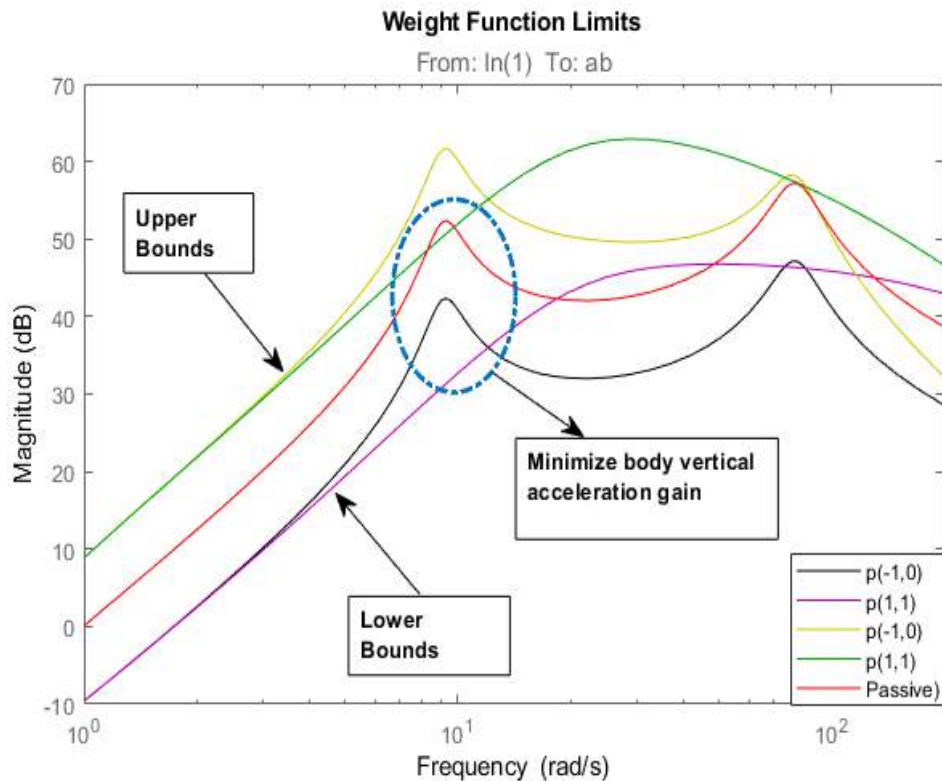


Figure 36: Body Vertical Acceleration Weight Function Bounds

Objective function $J_{comfort}$ is minimized between upper and lower bounds in range of 0-12 Hz to solve multi objective optimization problem by Genetic Algorithms.

Table 8: Wheel displacement weight function bounds

Ω_{21}	Ω_{22}	ζ_{21}	ζ_{22}
70	2	60	100
150	3	110	120

Upper and lower bounds are considered around +5 and -5 dB gain from passive suspension wheel displacement bode plot characteristic.

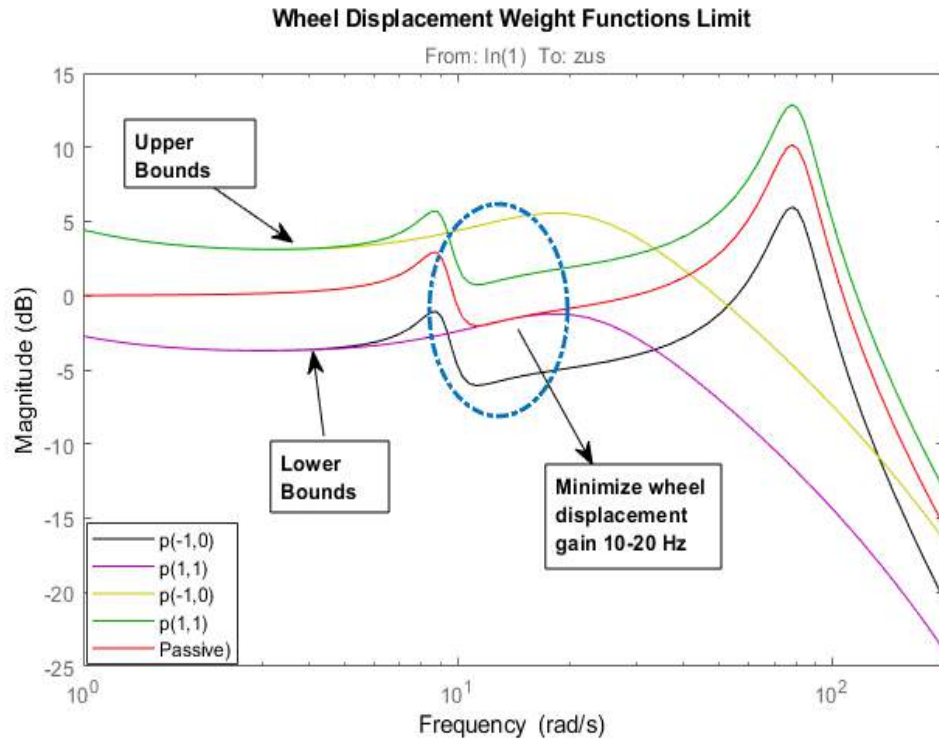


Figure 37: Wheel Displacement Weight Function Bounds

Objective function $J_{roadholding}$ is minimized between upper and lower bounds in range of 10-20 Hz to solve multi objective optimization problem by Genetic Algorithms.

- Step 2

Redefine decision vector v with low-pass filter Ω_f , and γ

$$(7.75) \quad v = [\Omega_{11}, \Omega_{12}, \zeta_{11}, \zeta_{12}, \Omega_{21}, \Omega_{22}, \zeta_{21}, \zeta_{22}, \Omega_f, \gamma]$$

- Step 3

Initiate with random positive weighting functions between these bound to get $v = v^0$ and random positive $\gamma_{ga} = \gamma_{ga}^0$

- Step 4

To determine the minimum real scalar $\gamma_{\min}$, solve the minimization problem of γ subject to LMI (7.68) and (7.69). Solve again a pair of LMIs (7.68) and (7.69) with a couple (v, γ) where $\gamma = \gamma_{\min} + \gamma_{ga}$. Note that this method is used to avoid the infeasibility of the LMIs (7.68) and (7.69). γ will be sum up $\gamma_{\min}$ which solved depending on the bounds and satisfying LMIs with γ_{ga} is calculated by Genetic algorithms. Since $\gamma_{\min}$ satisfies the LMIs, it can ensure the feasibility of LMIs (7.68) and (7.69) for all positive real γ_{ga} . [34]

```
function J=perf(v)

v=[ v(1);v(2);v(3);v(4);v(5);v(6);v(7);v(8);v(9);v(10)];

%%Decision Vector%%
oml1=v(1);
oml2=v(2);
ep11=v(3);
ep12=v(4);
om21=v(5);
om22=v(6);
ep21=v(7);
ep22=v(8);
omf=v(9);
gama=v(10);
```

Figure 38: Decision Vector

As mentioned in step 2 decision vector is built to determine weight functions, low-pass filter and γ_{ga} .

```
%State Space for of Weight Functions
[Azs,Bzs,Czs,Dzs]=tf2ss([1 2*ep11*oml1 oml1^2],[1 2*ep12*oml2 oml2^2]); %Wab
[Azus,Bzus,Czus,Dzus]=tf2ss([1 2*ep21*om21 om21^2],[1 2*ep22*om22 om22^2]); %Wzus

%Weight Functions state-space
Wab=ss(Azs,Bzs,Czs,Dzs);
Wzus=ss(Azus,Bzus,Czus,Dzus);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%LTI WEIGHTS%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Wab=ltisys(Azs,Bzs,Czs,Dzs);
Wzus=ltisys(Azus,Bzus,Czus,Dzus);

Wzr=3e-2;
%%%Low-Pass Filter
Wf = ltisys('tf',omf,[1 omf]) ;
```

Figure 39: Weight Function LTIs

Chosen weight functions from Genetic algorithms convert to LTI models to connect polytopic model in MATLAB environment. MATLAB tool "sconnect" allows to connect polytopic models, affine model and LTIs as can be seen in (e.g. Figure 40)

```

%%%Augmented Plant%%%

input='w';

output='G(1);G(2)';

Kin='K:G(3);G(4)';

[Gcl,r]=sconnect(input,output,Kin,'G:wzr;wf',pdP,'wf:K',Wf,'wzr:w',Wzr);

Gcl = smult(Gcl,sdiag(Wab,Wzus,eye(2)));

[gama_min,K]=hinfgs(Gcl,r,1.9404e3);

gama_new=gama_min+gama;

[gamason,K]=hinfgs(Gcl,r,gama_new);

%%%Close-Loop
pCL=slft(Gcl,K);

```

Figure 40: Augmented Plant from GA Weight Functions

Semi-active suspension polytopic model "pdP" connect with weight functions and create augmented plant thanks to "sconnect" in MATLAB. In this section, $\gamma_{\min}$ is calculated in each section and sum up with "gama" γ_{ga} to achieve "game_new" which represents γ (Step 4) and first $\gamma_{\min}$ is found from $\gamma_{ga}^0 = 1.9404e3$ is calculated from the random weights for first iteration and gives feasible solution (Step3).

- Step 5

Calculate the objective function J to minimize depending on control objectives ride comfort and road holding. Here, feasibility and stability condition is added in algorithm to be avoid infeasible solutions of LMIs.

Condition is that

If (7.68) and (7.69) is feasible and all K_{ci} (7.70) are stable

Calculate J using (7.73) and (7.74)

else

$$J = \infty$$

end

As mentioned before, calculation will be on vertices of $p(-1,0)$, $p(-1,1)$, $p(1,0)$, $p(1,1)$. Vertex polytopic coordinates is given in Fig.41 and is calculated by MATLAB tool "polydec" is explained in previous chapters.

```

%%%Vertices polytopic coordinates%%%

traj=[1 0 0 0;0 1 0 0 ; 0 0 1 0; 0 0 0 1];

Jcomfort=[];
Jroad=[];

for i=1:length(traj);

Pd = psinfo(pCL,'eval',traj(i,:));
[adk,bdk,cdk,ddk] = ltiss(Pd) ;
CLsys = ss(adk,bdk,cdk,ddk) ;

```

Figure 41: Vertices Polytopic Coordinates

Semi active suspension closed-loop polytopic model "pCL" is calculated over all vertices according to their polytopic coordinates.

```

%%%Controller

[gamma,K]=hinfgs(Gcl,r,gama_new);

%%%Close-Loop
pCL=slft(Gcl,K);

tau=quadstab(K);

pgam=quadperf(pCL);

```

Figure 42: Stability and Feasibility Check

MATLAB tool "quadstab" gives quadratic stability of controller and "tau" represents solution of stability investigation. Quadratic stability solution must be lower than 0 (7.68) and (7.69) and closed-loop model "pCL" must give positive scalar value.

```

for k=0.1:0.1:12
Jfre=freqresp(CLsys(1:1,:),k,'Hz');
Jfre=abs(Jfre);
Jcomfort=[Jcomfort;Jfre];
end

for j=10:0.1:20
Jfre=freqresp(CLsys(2:2,:),j,'Hz');
Jfre=abs(Jfre);
Jroad=[Jroad;Jfre];
end

end

Jcomfort=rms(Jcomfort);
Jroad=rms(Jroad);

```

Figure 43: Objective Function Calculation

Objective function J is calculated from (6.68) and (6.69). As mentioned in first chapters, wheel displacement is a measure of road holding and should be minimize between 10-20 Hz according to performance objectives. Wheel displacement is second output of 'CLsys' which represents closed-loop semi-active suspension model and Genetic algorithm objective is that make RMS gain of wheel displacement between 10-20 Hz over all verticies. Other objective is ride comfort and it is a measure of body vertical acceleration gain. According to performance objective, genetic algorithm minimizes the vertical acceleration RMS gain between 0-10 Hz over all verticies to achive performance goal.

```
Jcomfort=rms(Jcomfort);
Jroad=rms(Jroad);

if tau<0 & pgam~inf

    fprintf('Controller is stable over all parameter trajectory')

    J=[Jcomfort ; Jroad]

else

    fprintf('Controller is not stable')

    J=inf

end
```

Figure 44: Stability and Feasibility Condition

Feasibility and stability conditions that mentioned in Step 5 is applied end of the algorithm to be avoid infeasibility and not stable controller. J objective function or fitness function is introduced to algorith which is saved as "perf" MATLAB file.

- Step 6

Implementation of fitness function, bounds and Genetic Algorithm options in MATLAB optimization tool and starting optimization.

The screenshot shows the 'Problem Setup and Results' window in MATLAB. The 'Solver' is set to 'gamultiobj - Multiobjective optimization using Genetic Algorithm'. Under the 'Problem' section, the 'Fitness function' is '@perf' and the 'Number of variables' is '10'. The 'Constraints' section includes fields for 'Linear inequalities' (A: and b:), 'Linear equalities' (Aeq: and beq:), 'Bounds' (Lower: [20 4 80 4 80 70 2 6 100 1 0] and Upper: [80 20 105 40 150 3 110 120 90 10000]), and 'Nonlinear constraint function'. At the bottom, there is a checkbox for 'Use random states from previous run' which is currently unchecked.

Figure 45: Multiobjective Optimiziation Using Genetic Algorithhm

Solver method is multi objective optimization. Fitness function is chosen “perf” which is created in MATLAB as explained previous chapters to determine objective function with LMI solution and control strategy. Decision vector has 10 parameters from weight function, low pass filter and γ . Low pass filter is bounded between 1-90 Hz in large bandwidth of actuator. γ is considered between 0-10000 depends on the choosen random feasible random weight function result.

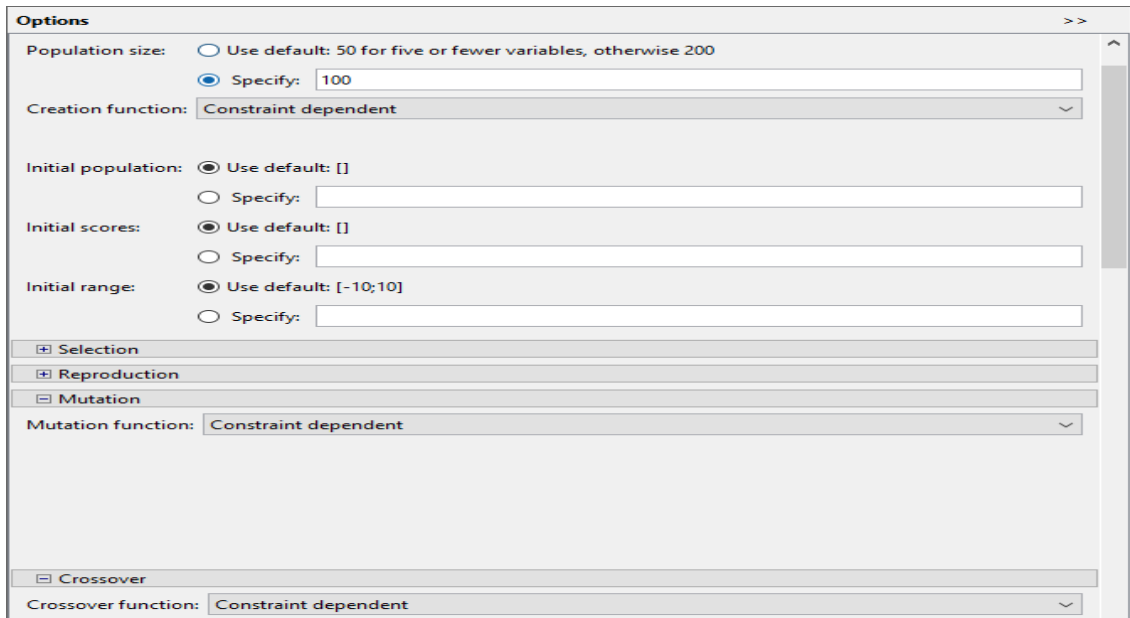


Figure 46: Optimization Tool Options

Population size is chosen 100 which can be taken higher to increase accuracy. Creation function, mutation and crossover function is considered as constraint dependent and selection is taken tournament as default.

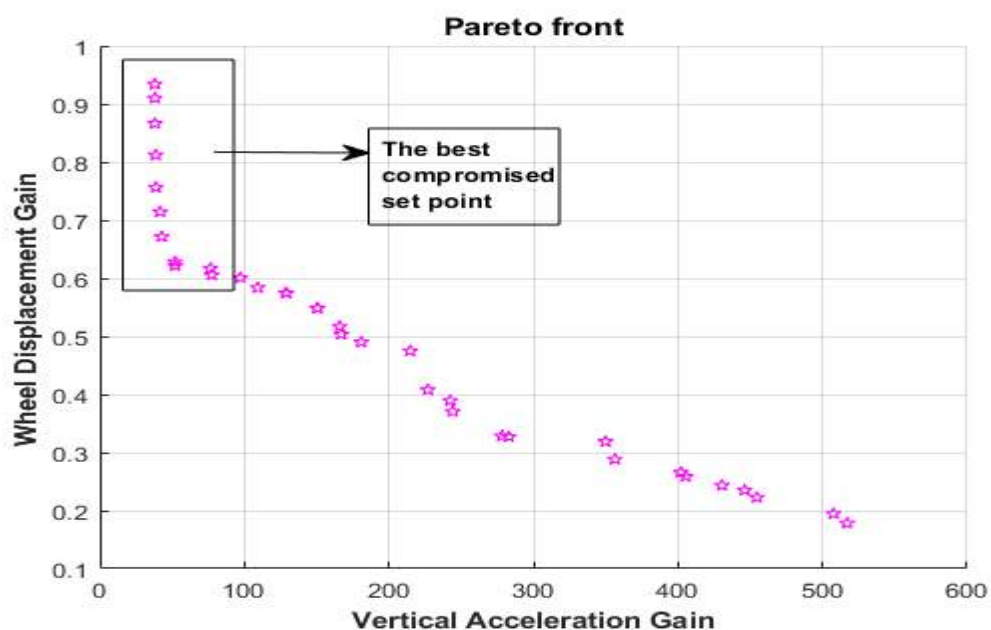


Figure 47: GA Simulation Result

Optimization problem is solved and genetic algorithms gives the values in (e.g. Figure 47). Chosen set point is taken to apply on controller to achieve performance objectives. These points can be the best compromised values in terms of control objectives since vertical acceleration gain seems 0 -100 dB gain and wheel displacement gain is between 0.6 – 0.9 dB gain.

Table 9: Solution of decision vector

Ω_{11}	Ω_{12}	ζ_{11}	ζ_{12}	Ω_{21}	Ω_{22}	ζ_{21}	ζ_{22}	Ω_f	γ
23.05	19.99	87.32	36.36	86.5	2.20	80.20	113	35.29	5.197e3

Optimization gives 35 different decision vector is found by Genetic algorithms. The best compromised weight function in terms of comfort and road holding is taken (e.g. Table 9).

7.7.3. Implementation of LPV/ H^∞ Gain Scheduled Controller in MATLAB/ Simulink

Creation of objective function naturally includes controller implementation in MATLAB since Genetic Algorithm optimization depends on the LMIs solutions of closed-loop system. In this part, weight functions that are found by Genetic algorithms will integrate in system and takes the frequency and time domain responses of semi-active suspension in next parts. In addition, grid method will introduce to apply LPV model in Simulink with " LPV" toolbox.

Creation steps of LPV/ H^∞ LMI based gain scheduled controller in MATLAB

- Step 1

State space creation of LPV semi-active suspension model (see details 5.3.5)

- Step 2

Creation of affine parameter dependent model to convert polytopic model (see details 5.3.5)

- Step 3

Creation of parameter trajectory and polytopic coordinates which includes all possible parameters in parameter box (see details 5.3.5)

- Step 4

Find weight functions by using Genetic algorithms to achieve control goals (see details 7.7.2) and introduce them in MATLAB

```

%%Weights from Genetic Algotihm%%

%Weight Values for Zsdotdot(ab)

oml1=23.05;
oml2=19.99;
ep11=87.32;
ep12=36.36;

%%Weight Values for Zus
om21=86.5;
om22=2.2;
ep21=80.2;
ep22=113;

%State Space Matrix of Weight Functions

[Azs,Bzs,Czs,Dzs]=tf2ss([1 2*ep11*oml1 oml1^2],[1 2*ep12*oml2 oml2^2]); %Wab
[Azus,Bzus,Czus,Dzus]=tf2ss([1 2*ep21*om21 om21^2],[1 2*ep22*om22 om22^2]); %Wzus

```

Figure 48: Weight Functions

Weight functions which are found by Genetic algorithms is introduced the system in MATLAB. In this part, their state space models is taken to convert LTI models in (e.g. Figure 49) to connect the blocks each other.

```

%Weight Functions state-space

Wab=ss(Azs,Bzs,Czs,Dzs);

Wzus=ss(Azus,Bzus,Czus,Dzus);

%%%%%%%%%%Weight Function LTI%%%%%%%%%%

Wab=ltisys(Azs,Bzs,Czs,Dzs);

Wzus=ltisys(Azus,Bzus,Czus,Dzus);

Wzr=3e-2

%%%Low Pass Filter from GA to make input independent for LMI%%

Wf = ltisys('tf',35.29,[1 35.29]);

```

Figure 49: LTIs of Weight Function

- Step 5

Connect the blocks according to block diagram of semi-active suspension control (e.g. Figure 49).

MATLAB tool "sconnect" can connect the affine or polytopic model and their LTIs. Tool computes the system matrix P of the standard H-∞ plant associated $P(s)$ with a given control constructure. It also returns the 1x2 vector R such that

$$R(1) = \text{number of measurements (controller inputs)}$$

$R(2)$ = number of controls (controller outputs)

```

%%%Augmented Plant%%%

input='w';

output='G(1);G(2)';

Kin='K:G(3);G(4)';

%%%Connect Blocks%%%

[Gcl,r]=sconnect(input,output,Kin,'G:wzr;wf',pdP,'wf:K',Wf,'wzr:w',Wzr);

Gaugga = smult(Gcl,sdiag(Wab,Wzus,eye(2)));

figure
omega = logspace( 0,2,200 ) ;
for j = 1 : length(alpha)
Pd = psinfo(Gaugga,'eval',alpha(j,:)) ;
[adk,bdk,cdk,ddk] = ltiss(Pd) ;
sysGaugga = ss(adk,bdk,cdk,ddk) ;
gridGaugga(:, :, j)= ss(adk,bdk,cdk,ddk) ;
[vs] = sigma( sysGaugga, omega ) ;
semilogx( omega, mag2db( vs ) ) ;
title({'Plant with Weights'})
hold on ; grid on ;
end

```

Figure 50: Augmented Plant

Augmented plant is a plant with weights but not with controller. According to block diagram of semi-active suspension control (e.g. Figure 33), input in code represents exogenous input or disturbance Z_r . Output is given as first and second output of polytopic system "G(1); G(2)" which are respectively control outputs $\ddot{z}_s$ body vertical acceleration and z_{us} wheel displacement. Controller inputs "Kin" represent measurement output of the LPV semi-active suspension model which are z_{def} suspension deflection and $\dot{z}_{def}$ suspension deflection velocity respectively. As mentioned in chapter (6.6), control input must be independent and it can be solved by adding suitable low-pass filter. "wf" is connect with control output "K" to get filtered input for LMI solution of closed-loop system. "pdP" is the LPV polytopic semi-active suspension model. "sconnect" give results "Gcl" and "r" which is 1x2 vector with $r(1)$ number of measurement outputs and $r(2)$ number of control outputs. Finally, augmented plant "Gaugga" is delivered by "smult" provides serial interconnection of blocks. First and second outputs of polytopic model $\ddot{z}_s$ and z_{us} turns $W\ddot{z}_s$ and Wz_{us} thanks to weight blocks W_{ab} and Wz_{us} . Measurement outputs directly interconnect with controller block "K" as a input of controller and do not connect with any weight blocks.

```

#####Controller###

[gopt,Kga]=hinfgs (Gaugga,r,5.19e3);

tauga=quadstab (Kga)

figure
omega = logspace( 0,2,200 ) ;
for j = 1 : length(alpha)
Pd = psinfo(Kga,'eval',alpha(j,:)) ;
[adk,bdk,cdk,ddk] = ltiss (Pd) ;
Kgasys = ss(adk,bdk,cdk,ddk) ;
gridKga(:, :, j)= ss(adk,bdk,cdk,ddk);
[vs] = sigma( Kgasys, omega ) ;
semilogx( omega, mag2db( vs ) ) ;
title({'Controller'})
hold on ; grid on ;
end

gridKga.SamplingGrid=struct('p1',p1,'p2',p2);

#####Close Loop###

CLkga=slft (Gaugga,Kga,1,2);

```

Figure 51: Gain Scheduled Controller

Controller is found by MATLAB tool "hinfgs" and this tool will be explained in this part. It forms H-parameter control for parameter-dependent systems with affine dependency on time-varying parameters. These parameters will be calculated in real time. This function implements a quadratic approach to H^∞ performance. This code can be shown as [GOPT,PDK,R,S]=hinfgs(PDS,R,GMIN)

- Inputs

PDS: Parameter dependent plant

R: 1x2 vector defining D_{22} dimensions gives $R(1)$ measurement output number and $R(2)$ control output number.

GMIN: Target value for GOPT. Set GMIN=GAMA (γ) to decide whether GAMMA efficiency is achieved for given γ while set GMIN=0 to obtain optimum achievable GOPT.

- Outputs

GOPT: Gives the achievable performance in case of given $\gamma < \text{GMIN}$ otherwise calculates optimal performance in case of GMIN=0

PDS: Polytopic model PDK=[K1, . . . , Kn] of the gain-scheduled controller. The vertex controller K_j (6.66) is connected to the j -th corner of the parameter box as given by polydec (Fig.23)

R,S : Solutions of the characteristic LMI system

```

19
20
21          3.791719e+04
22          3.066275e+04
23          1.857265e+04
24          8914.875751
25          5162.365636

Result:  reached the target for the objective value
        best objective value:  5162.365636
        f-radius saturation:  0.767% of R =  1.00e+08

Optimal quadratic RMS performance:  5.0843e+03

```

Figure 52 Quadratic RMS Performance

Controller gives positive scalar quadratic RMS performance for linear objective minimization under LMI constraints.

```

Iteration   :   Best value of t so far

1           0.055857
2           1.765518e-03
3           3.585737e-04
4           3.585737e-04
5           2.248931e-04
6           2.248931e-04
7           8.223726e-05
8           2.844555e-05
9           -1.954082e-03

Result:  best value of t: -1.954082e-03
        f-radius saturation:  0.013% of R =  1.00e+08

This system is quadratically stable

tauga =

-0.0020

```

Figure 53: Quadratic Stability

The best objective “tauga” which is measurement of quadratic stability for LMI feasibility problem have to smaller than zero for quadratic stability.

Closed-loop system is formed with LMI based LPV/ H^∞ gain scheduled controller. Controller keeps the needed values of the LPV model in each vertex of $p(-1,0)$, $p(-1,1)$, $p(1,0)$, $p(1,1)$.

MATLAB has "LPV system block" in Simulink framework to built LPV models. This block only accepts grid models of the system. Therefore, semi-active suspension plant and LMI based LPV/ H^∞ gain scheduled controller should be taken as grid model to implement in Simulink (e.g. Figure 55) Semi-active suspension model is taken as a grid model along with all considered parameter range $p(p_1, p_2)$ in "gridsys" (Fig.24) and controller is taken as "gridKga" (e.g. Figure 51)

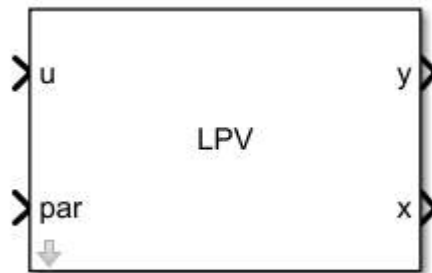


Figure 54: Simulink LPV Block (e.g. [12])

- Accepts input signal $u(t)$. This port accepts a signal of the input dimension in a multi-input situation.
- Signals for variables which define "sampling grid" variables in scheduling space (e.g. Figure 56). Time functions, inputs and states or constants can be the scheduling variables. A scheduling signal can be prepared using clock input (for time), input signal (u) and output signals (x,dx/dt,y) of the LPV block, if needed.
- Model output
- Values of the model states

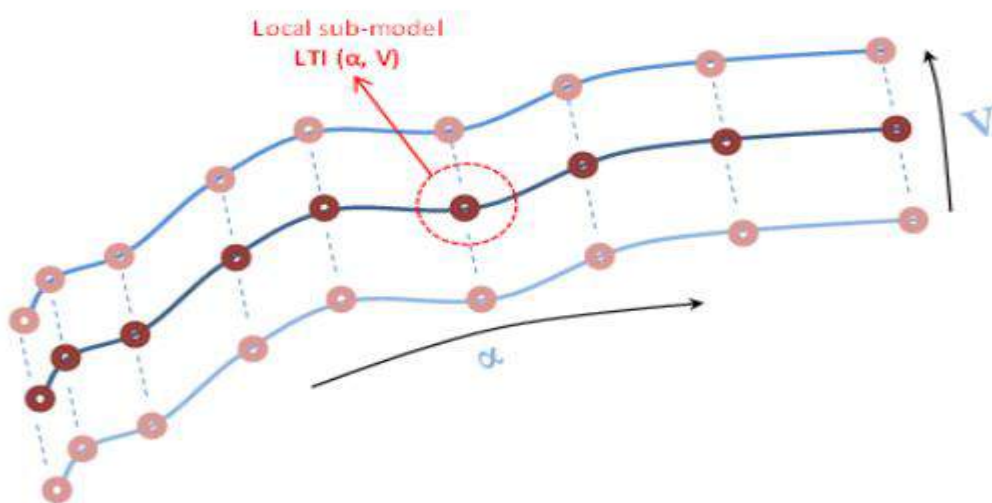


Figure 55: Grid Model (e.g. [12])

LPV system box keeps the system state space arrays in each scheduling parameters point in grid and interpolate them in order to reach the nearest point depends on the changing scheduling parameters in simulation time. Scheduling parameter changes in real time and in this study they are taken as a function of semi-active suspension model states z_s and z_{us} which are the states that determine suspension deflection and suspension deflection velocity.

```
p1=p1';
p2=p2';

gridsys.SamplingGrid=struct('p1',p1,'p2',p2);

%%%Fix and Parameter Dependent Coefficients Logic Matrix for LPV%%

Agridsysfixed=[true(1,4);
               false(1,4);
               true(1,4);
               false(1,4)];

Bgridsysfixed=[true(1,2);
               true(1,1) false(1,1);
               true(1,2);
               true(1,1) false(1,1)];

Cgridsysfixed=[false(1,4);
               true(3,4)];

Dgridsysfixed=[true(1,1) false(1,1);
               true(3,2)];
```

Figure 56:Sampling Grid of the Model

Sampling grid of the model is created according to scheduling parameters p_1 and p_2 (e.g. Figure 19). LPV block in Simulink specifies which status space matrix and delay vector coefficients are set. Fix coefficients of the system model matrices is saved "gridsysfixed" in MATLAB. Scalar Boolean true or false values as logic matrix determine the coefficient fix or parameterized by the scheduling. False provides the change coefficient while true keeps the same coefficient in whole simulation time.

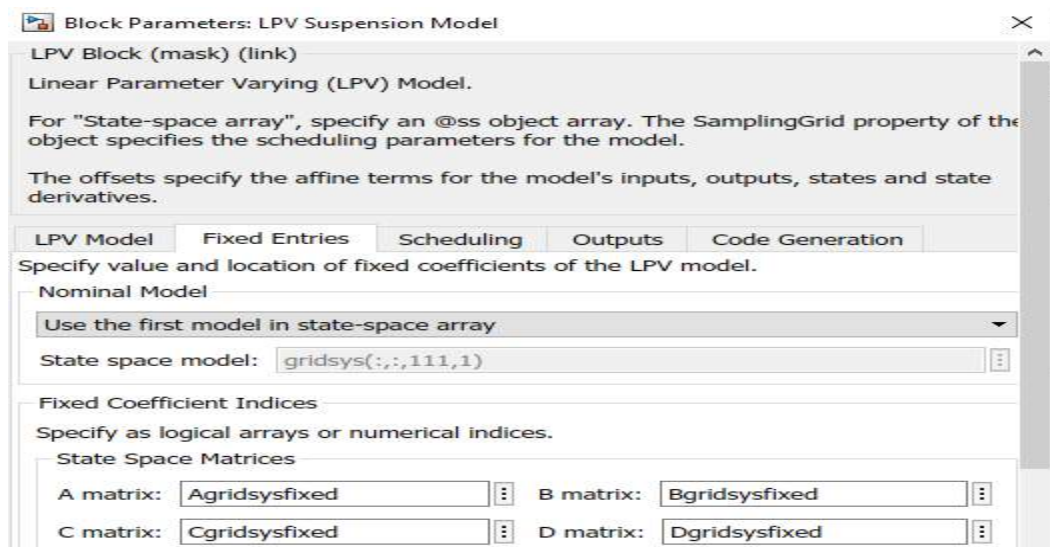


Figure 57: Fix Coefficients of the Plant

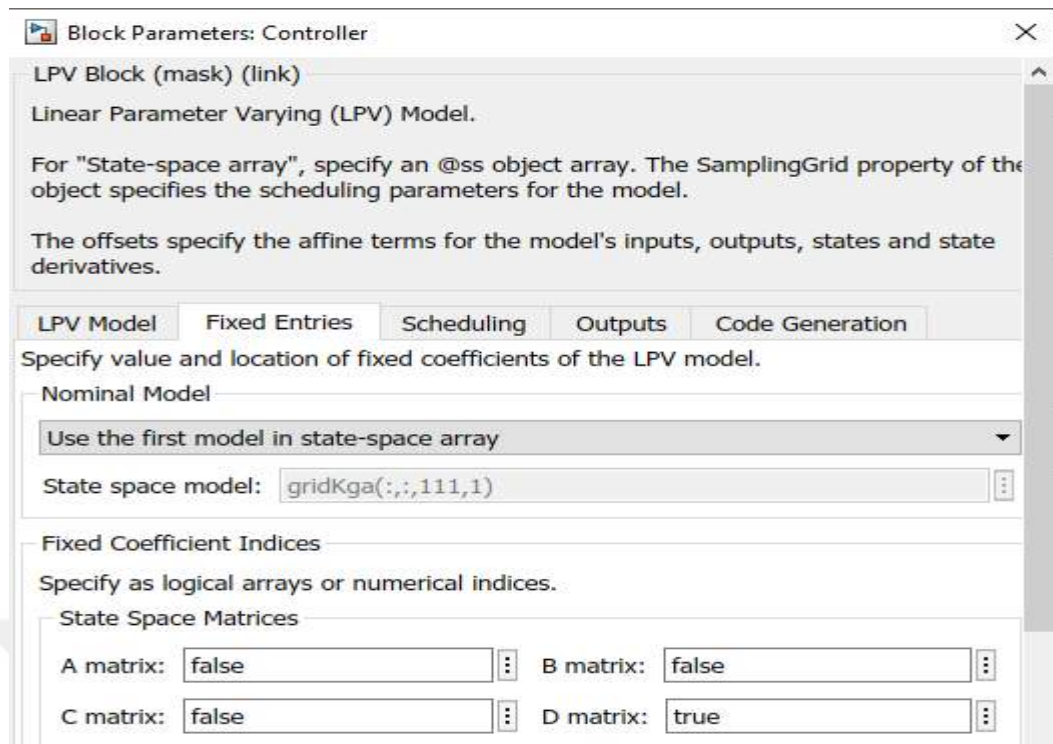


Figure 58: Fix Coefficients of the Controller

System model has fix coefficients while controller state space matrices do not have fix coefficients and should change depending on the semi-active suspension model measurement outputs suspension deflection and suspension velocity with scheduling parameter $p(p_1, p_2)$.

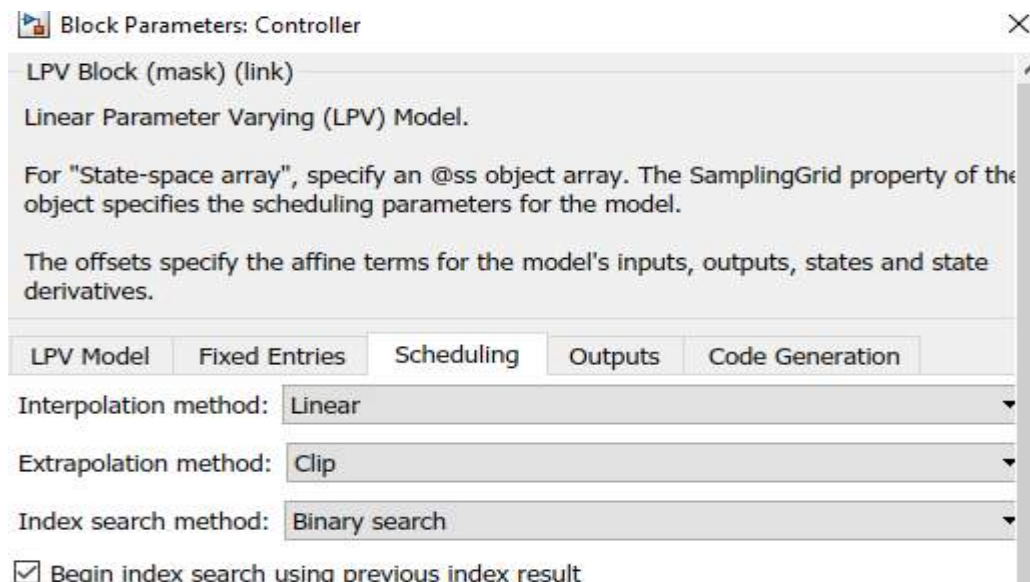


Figure 59: LPV System Box Scheduling

LPV system box keeps state space arrays of system model and controller model in all considered scheduling parameters $p(p_1, p_2)$ and interpolate the model values the

nearest point in calculated new scheduling parameter depending on the simulation time and find the accurate controller for semi-active suspension model at this point and time.

Control model can be generally presented from current input by considering Static-Guo model as well and current input to produce demanded control force can be calculated simply from equation (2.4)

$$(6.71) \quad I = \frac{u_I + f_0}{y_{mr}}$$

where $u_I + f_0 = F_{mr}$

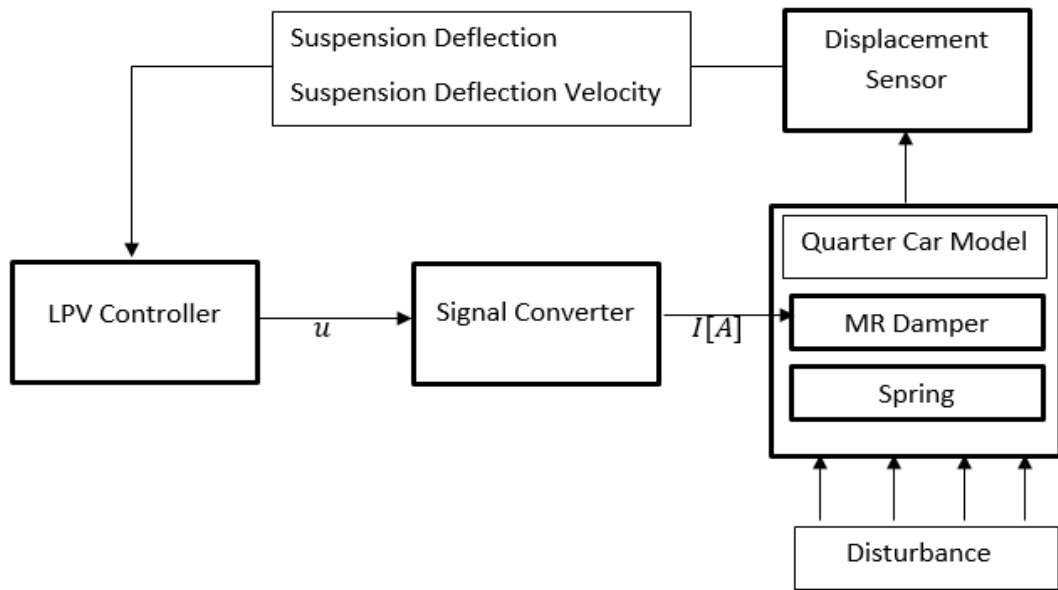


Figure 60: Implementation Scheme

8. Frequency Domain Results of LPV/H ∞ Semi Active Suspension

Performance objectives ride comfort and road holding aims to reduce body vertical acceleration gain between 0-12 Hz and wheel displacement 10-20 Hz to provide safe and comfortable driving. As mentioned in comfort control goal chapter 3.1, human being are more sensitive acceleration between 4-8 Hz according to ISO 2631. Therefore, these frequency range should be focus on to compare ride comfort between passive suspension and semi-active suspension. Since there is a trade-off between ride comfort and road holding after 12 Hz, the best compromised weight functions of $\ddot{z}_s$ and z_s is found and applied on system thanks to multi objective optimization of Genetic algorithms.

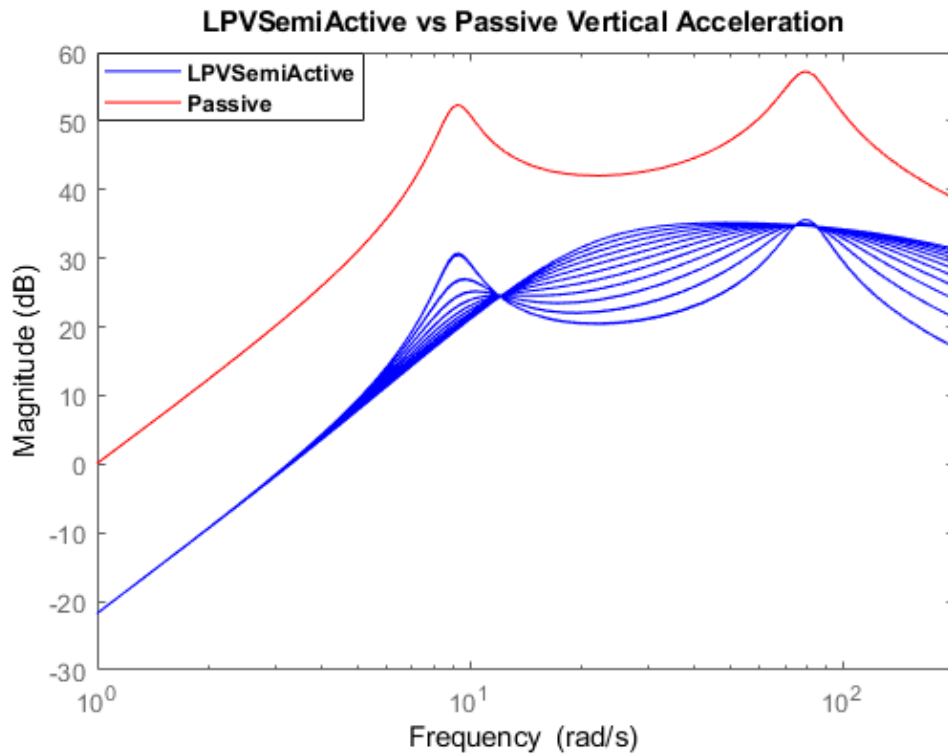


Figure 61: Vertical Acceleration Bode Plot

Frequency result of body vertical acceleration $\frac{\ddot{z}_s}{z_r}$ for passive and LPV/H ∞ semi active suspension systems is given in Fig.60. As mentioned in first paragraph, ride comfort is specifically related to 4-8 Hz vertical acceleration gain according to ISO standard. Bode plot shows that LPV/H ∞ semi active suspension system reduce gain clearly in these range to reduce vertical acceleration gain and provide ride comfort.

Frequency result of body wheel displacement $\frac{z_s}{z_r}$ for passive and LPV/H ∞ semi active suspension systems is given in (e.g. Figure 61). Road holding objective is to reduce wheel displacement gain between 10-20 Hz. After 12 Hz, trade-off between ride comfort and

road holding is unavoidable. According to chosen weight functions by Genetic algorithm, the best compromise wheel displacement result is found for road holding.

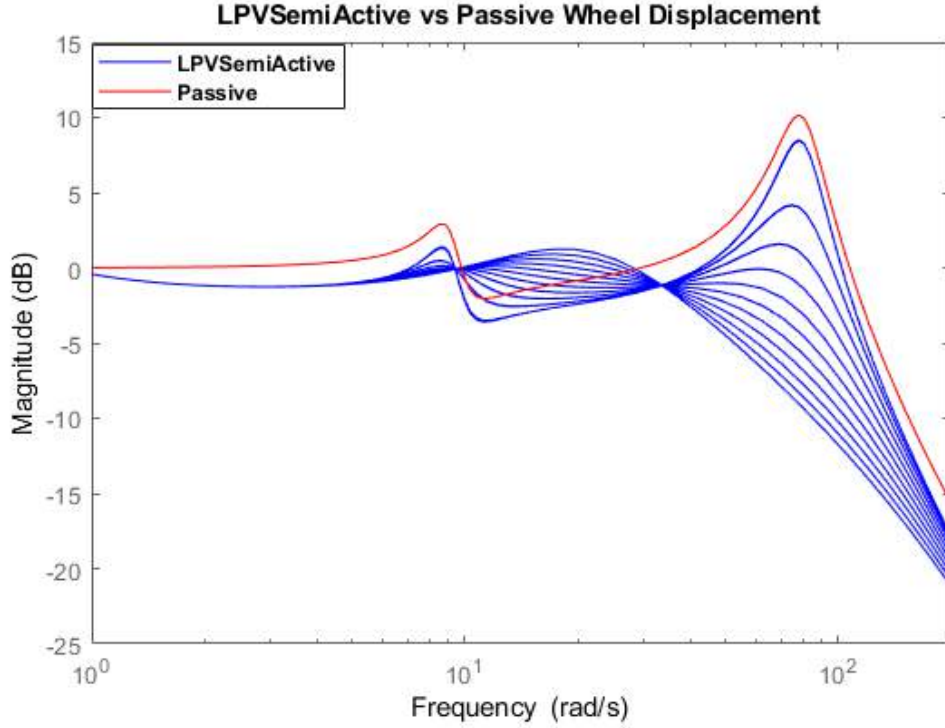


Figure 62: Wheel Displacement Bode Plot

Wheel displacement gain is lower than passive suspension gain in some scheduling parameter locations while is higher in some locations between 10- 20 Hz depending on the compromise with vertical acceleration gain. Figure shows that road holding objective is provided to a large extend in these frequency range.

9. Time Domain Results of LPV/ H_∞ Semi Active Suspension

In this chapter, time domain results of semi active suspension with LMI based LPV/ H_∞ gain scheduled controller is analysed. Bump, sinus wave source and ISO road profiles are taken as disturbance input for analysis.

9.1. Bump Test

Bump disturbance is given by

$$(9.1) \quad z_r(t) = \begin{cases} \frac{h}{2} \left(1 - \cos\left(\frac{2\pi t}{l}\right) \right), & 0 \leq t \leq \frac{l}{v} \\ 0, & t > \frac{l}{v} \end{cases}$$

3 different bump test is made with same bump properties $h = 0.05\text{ m}$ $l = 5\text{ m}$ with different velocity for semi-active suspension. Test with $v = 5\text{ m/s}$ is taken to show in this chapter. Other tests with $v = 10\text{ m/s}$ and $v = 20\text{ m/s}$ are in Appendix.

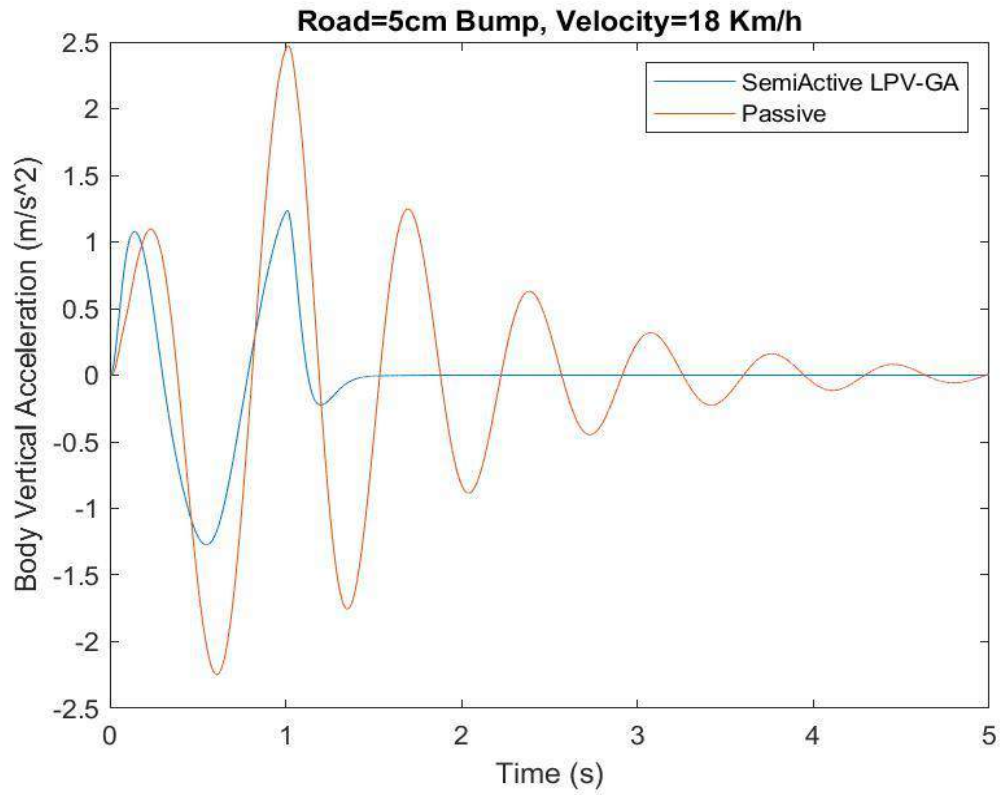


Figure 63: 5cm Bump Vertical Acceleration

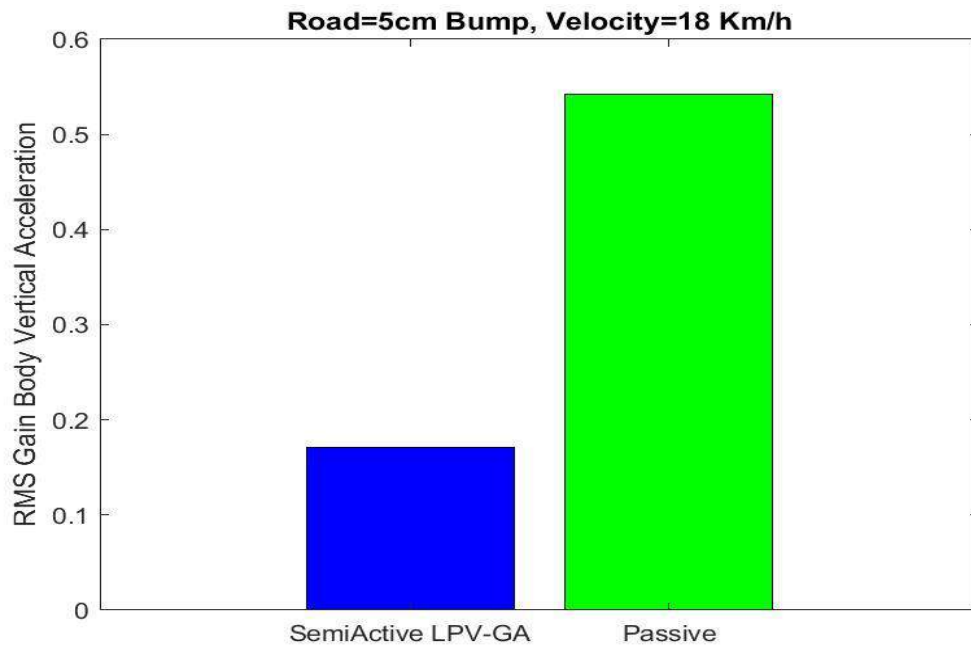


Figure 64: 5cm Bump RMS Gain Vertical Acceleration

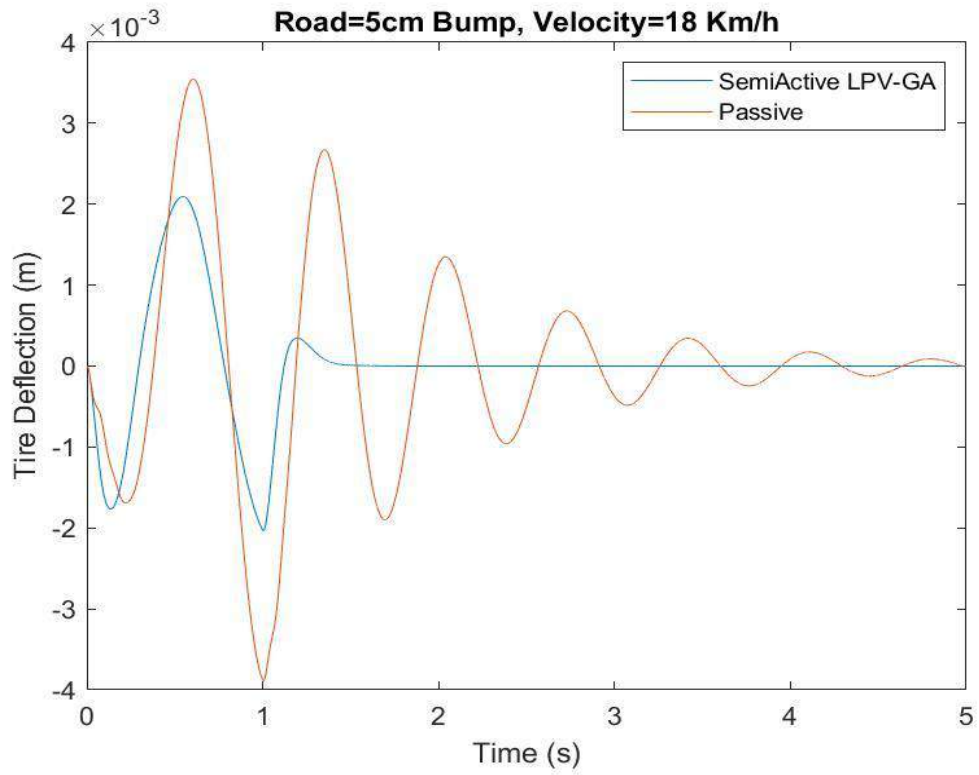


Figure 65: 5cm Bump Tire Deflection

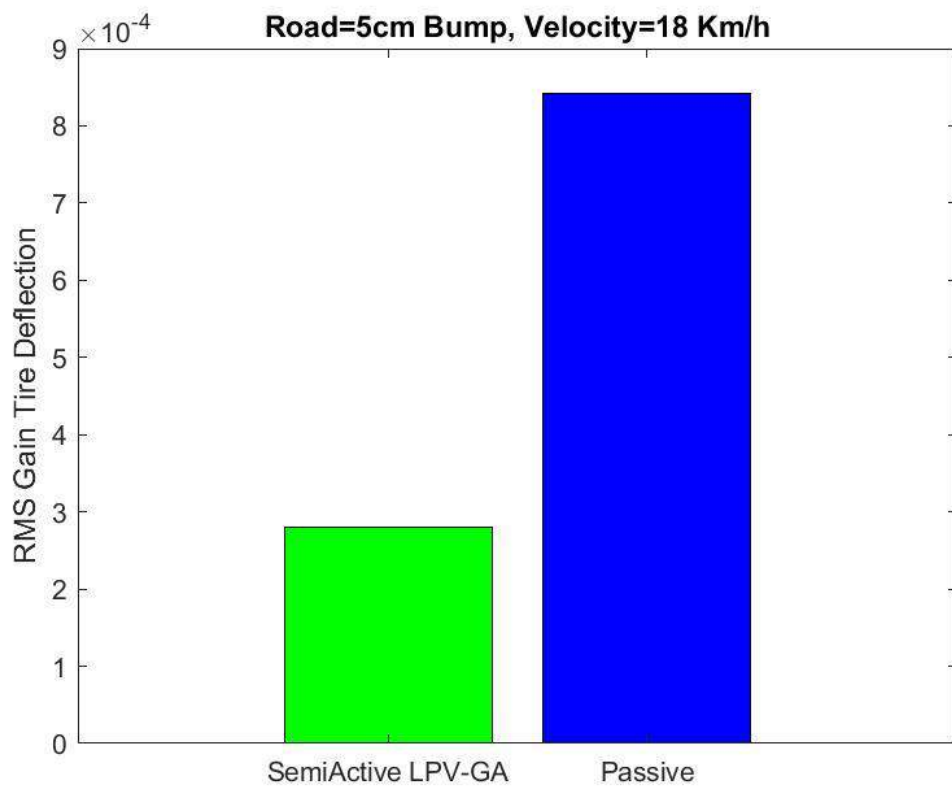


Figure 66 5cm Bump RMS Gain Tire deflection

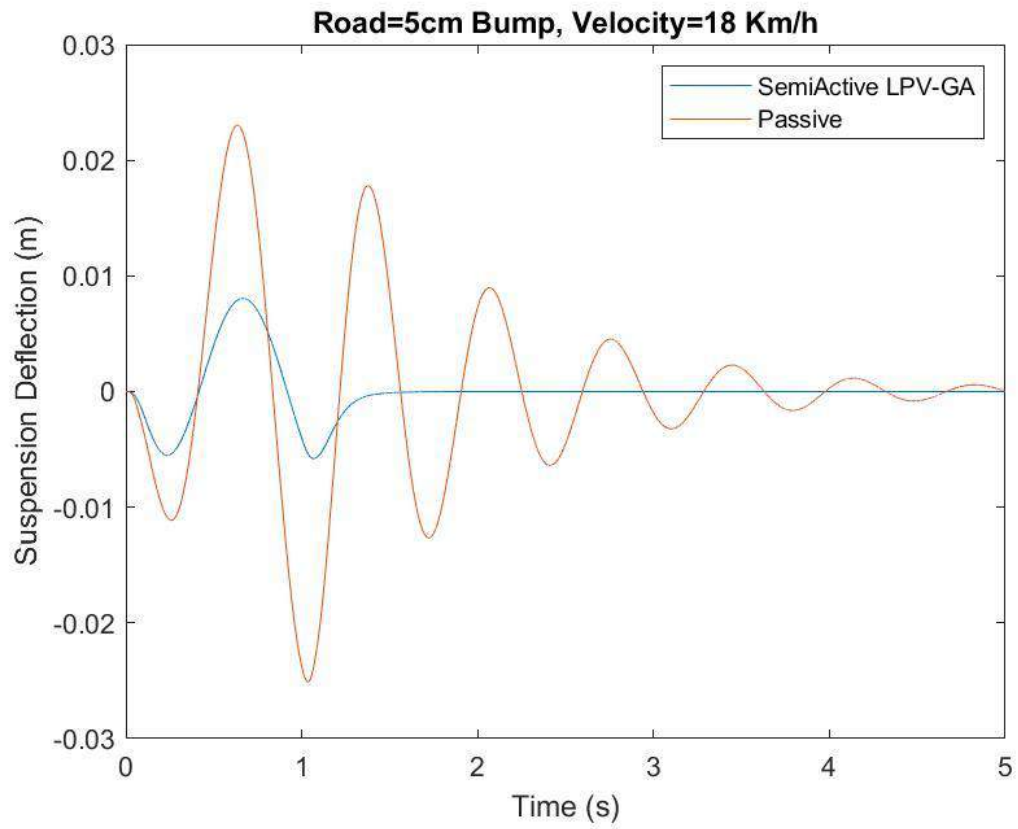


Figure 67: 5cm Bump Suspension Deflection

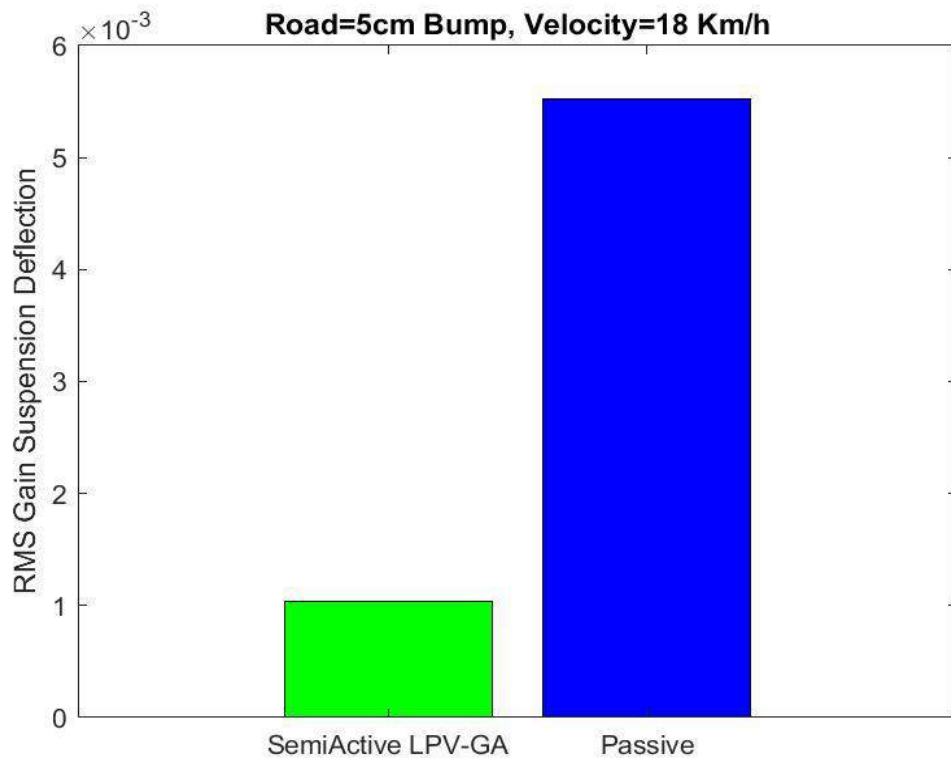


Figure 68: 5cm Bump RMS Gain Suspension Deflection

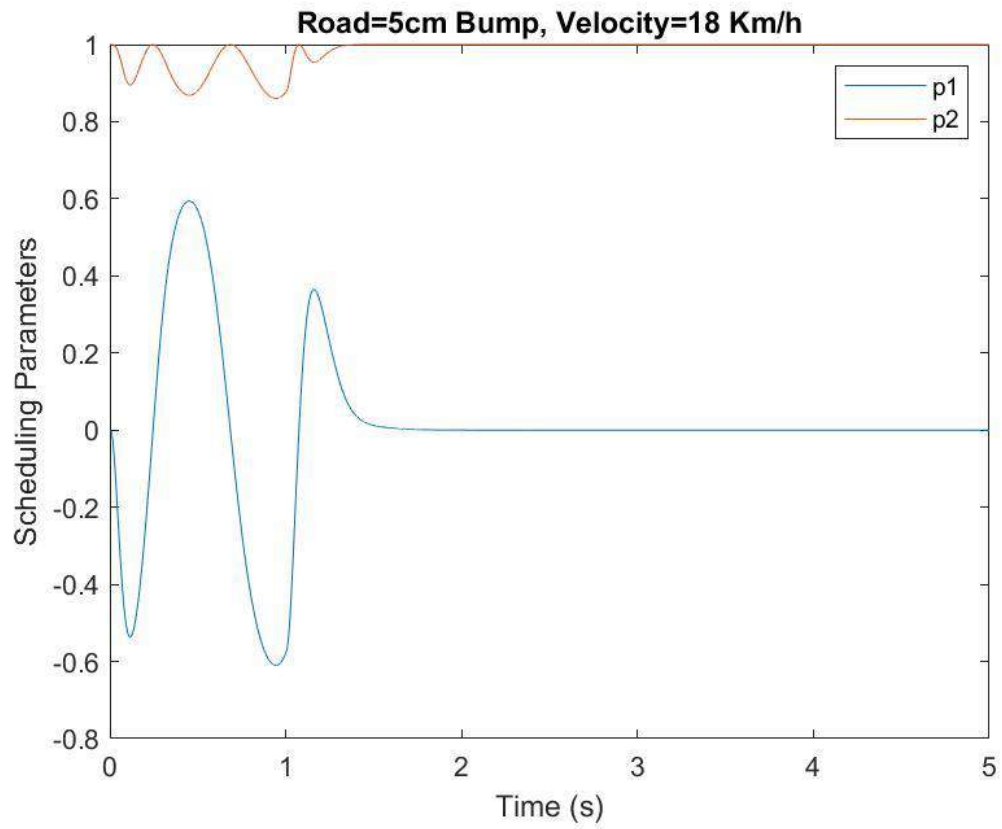


Figure 69: 5 cm Bump Scheduling Parameters

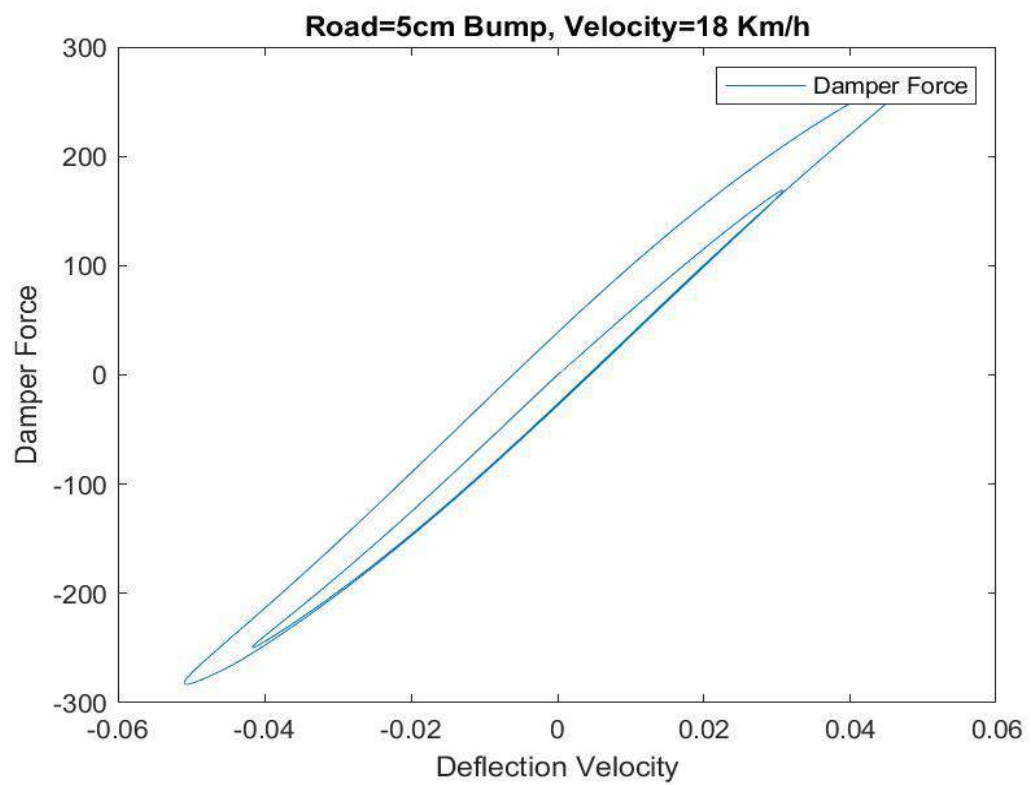


Figure 70: 5 cm Bump Damper Force

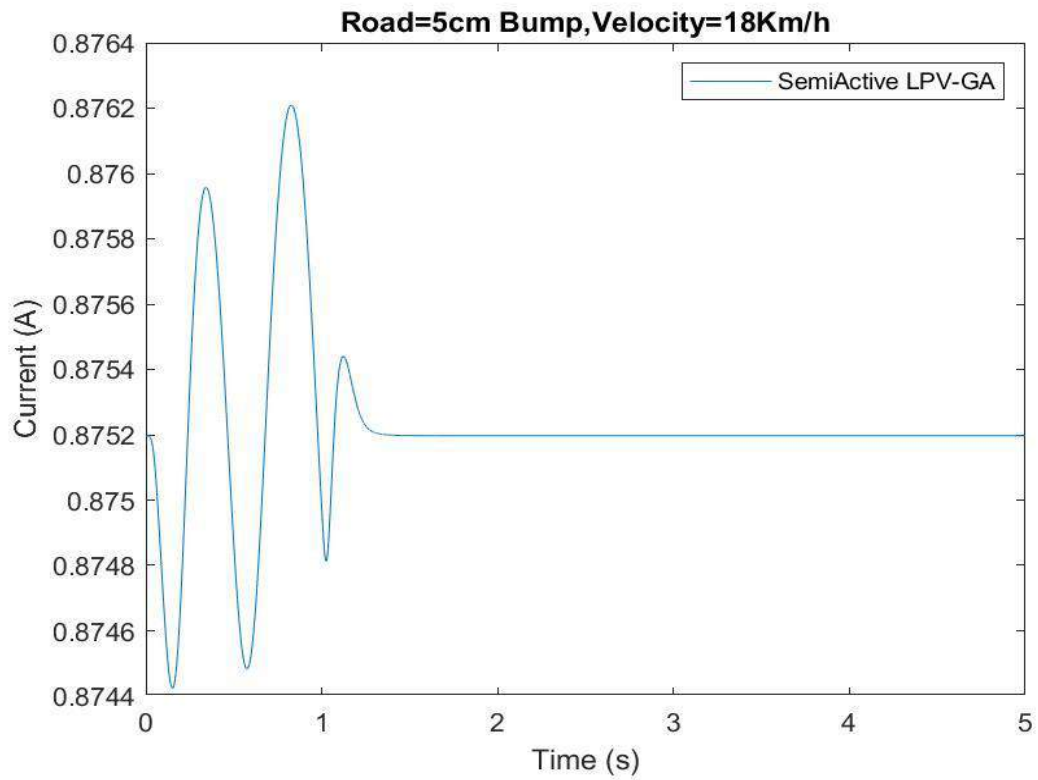


Figure 71: 5 cm Bump Current

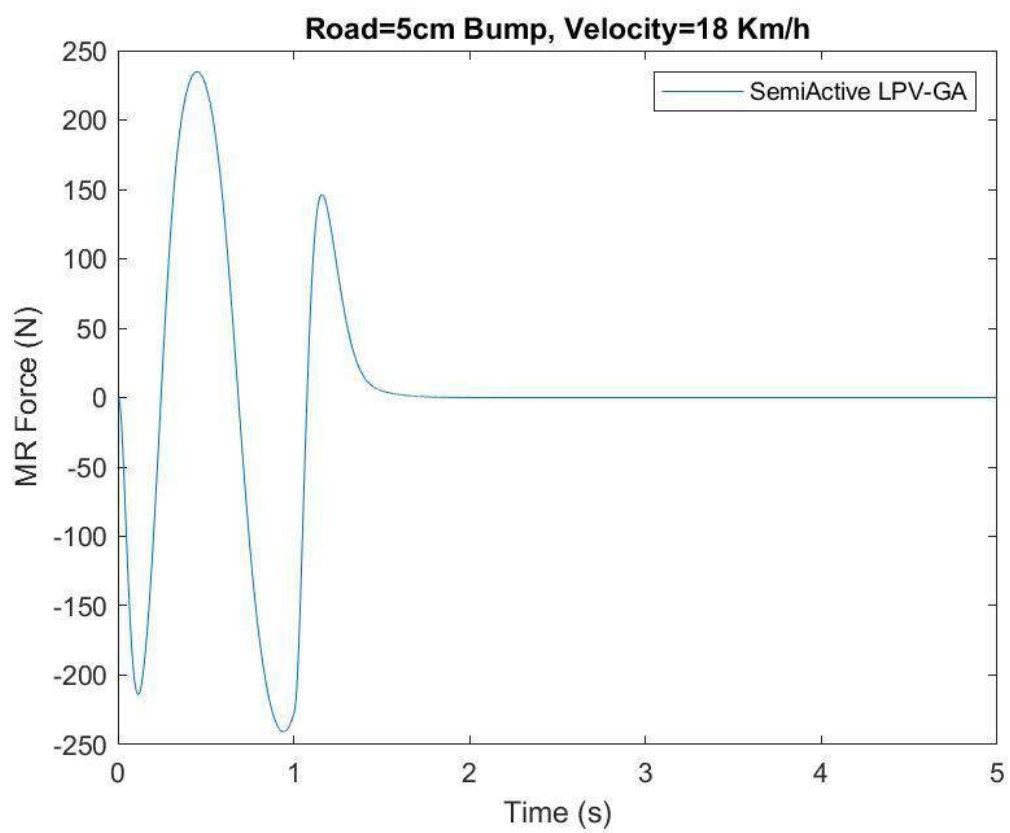


Figure 72 5 cm Bump MR force

9.2. Sinus Wave Source Test

In this section, sinus wave source is taken with 0.05 Amplitude and 8 Hz frequency as disturbance input in order to make time domain analysis.

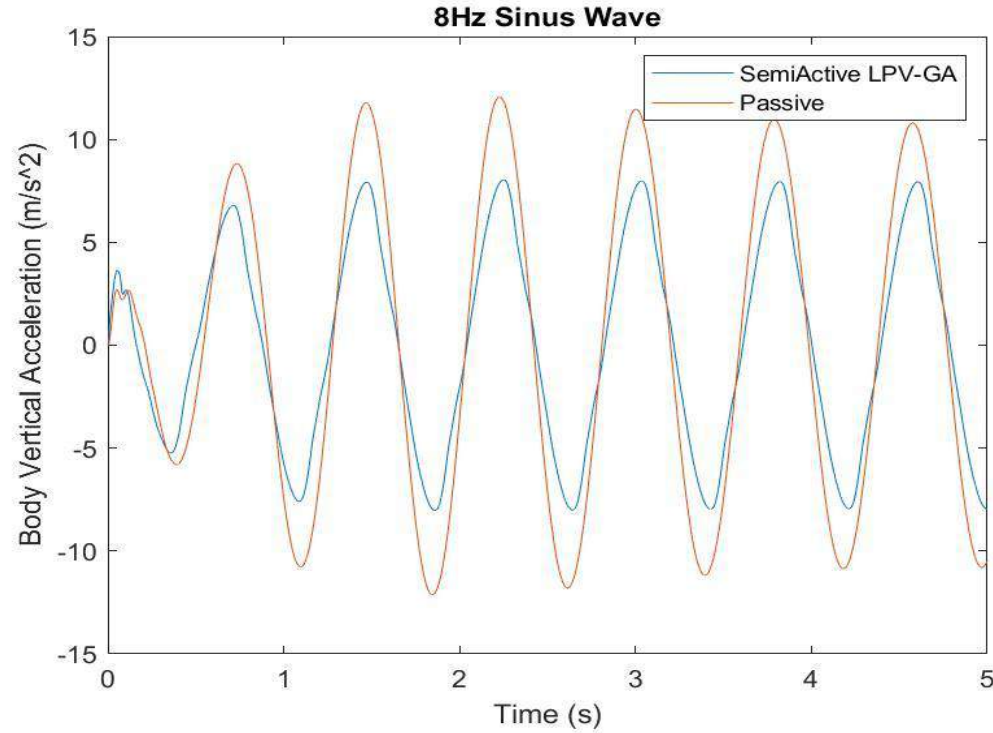


Figure 73: 8Hz Sinus Wave Vertical Acceleration

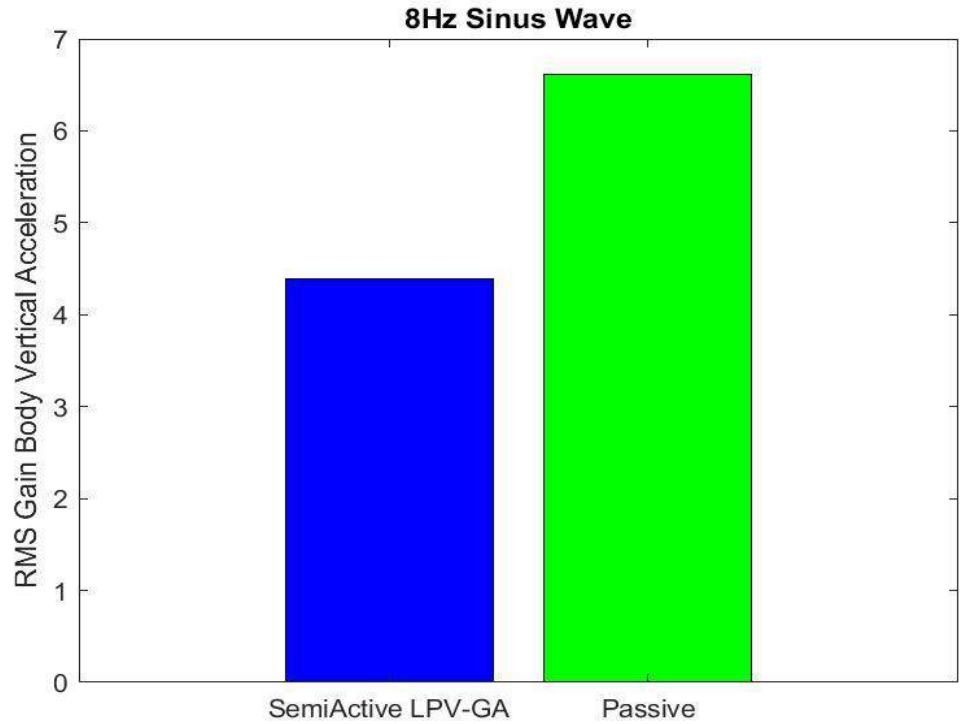


Figure 74: 8Hz Sinus Wave RMS gain Vertical Acceleration

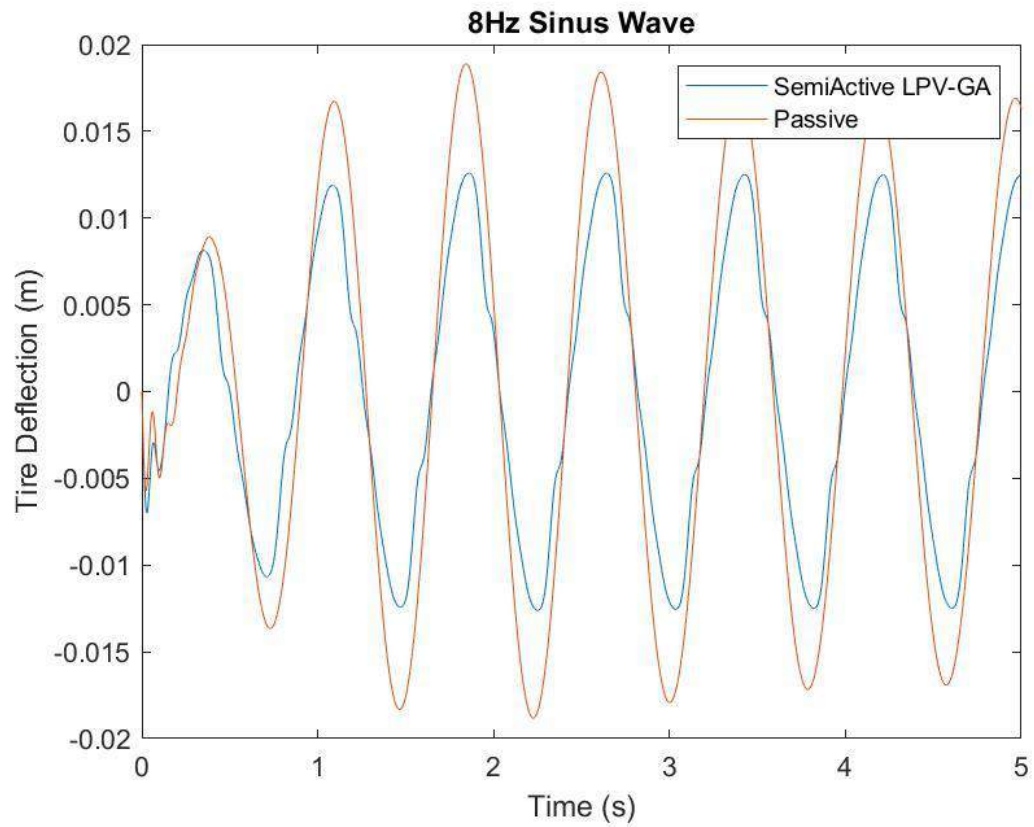


Figure 75: 8Hz Sinus Wave Tire Deflection

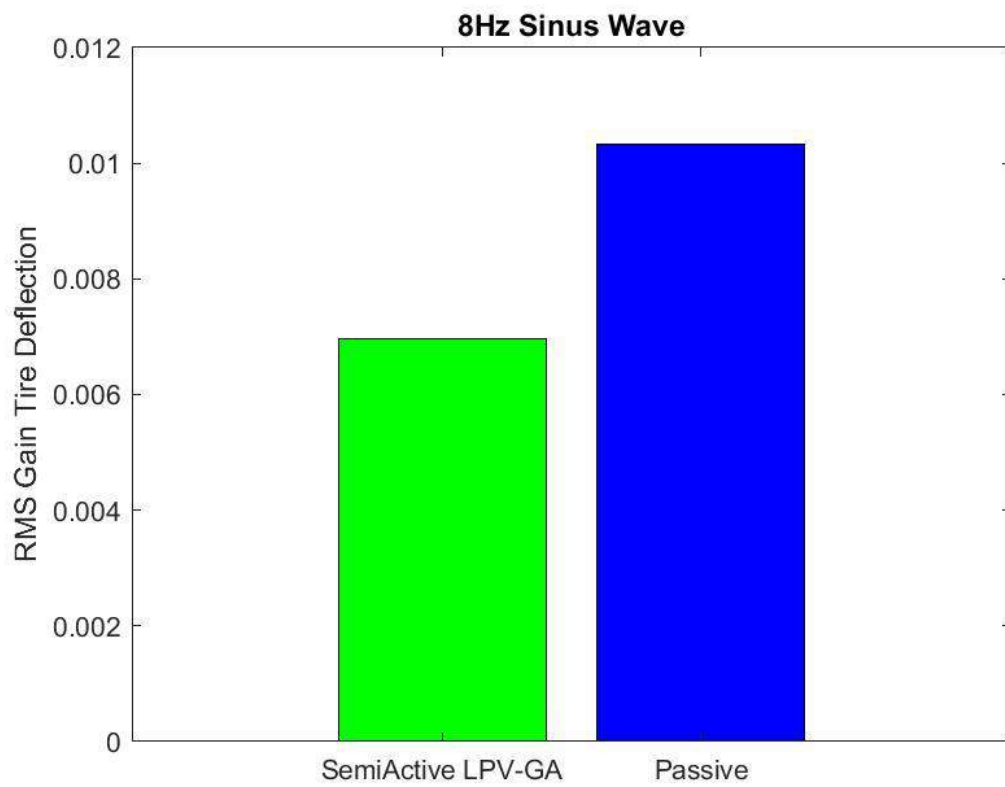


Figure 76: 8Hz Sinus Wave RMS Gain Tire Deflection

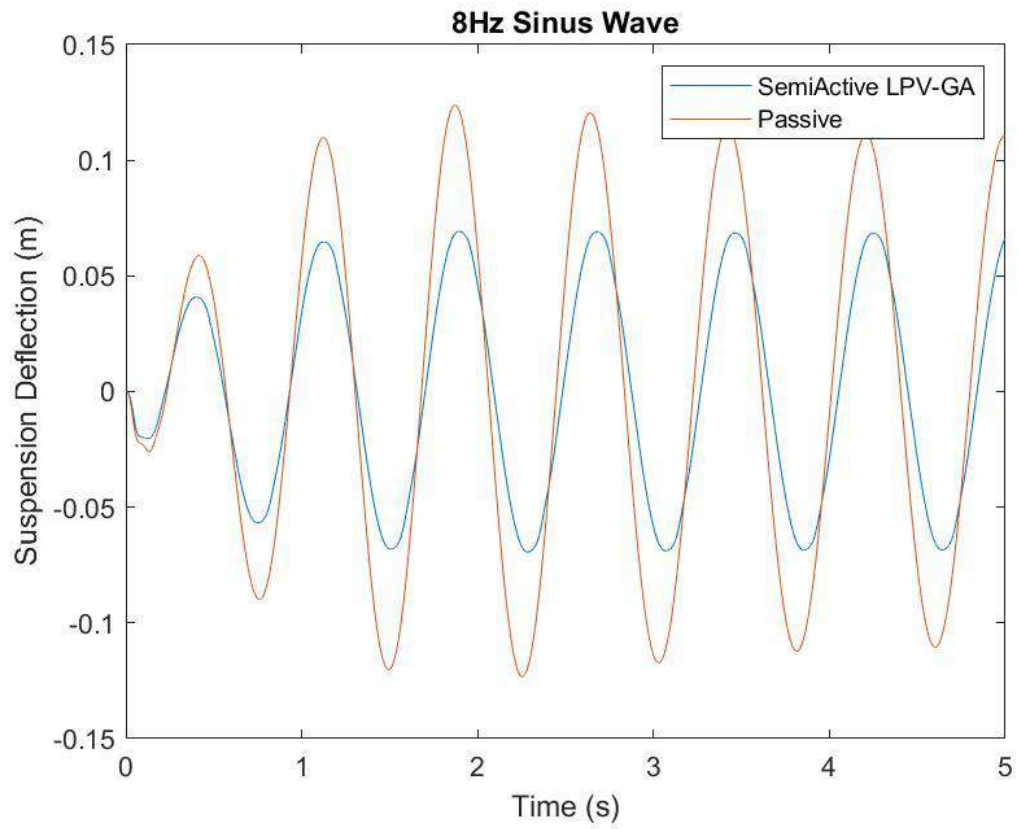


Figure 77: 8Hz Sinus Wave Suspension Deflection

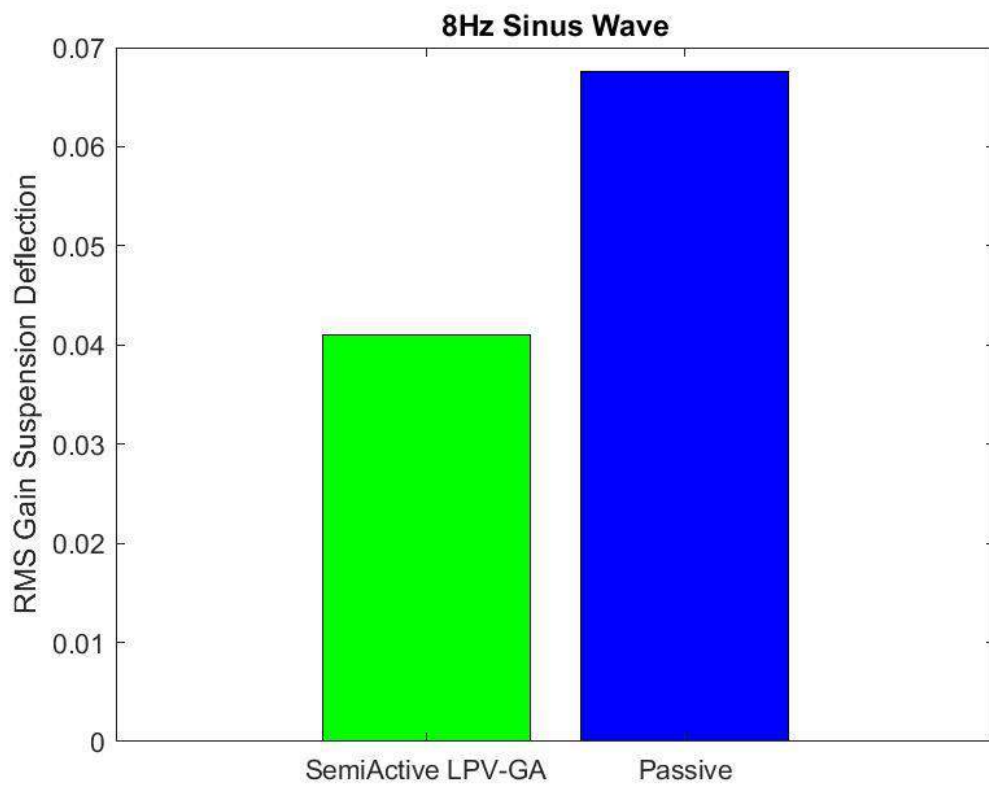


Figure 78: 8Hz Sinus Wave RMS Gain Suspension Deflection

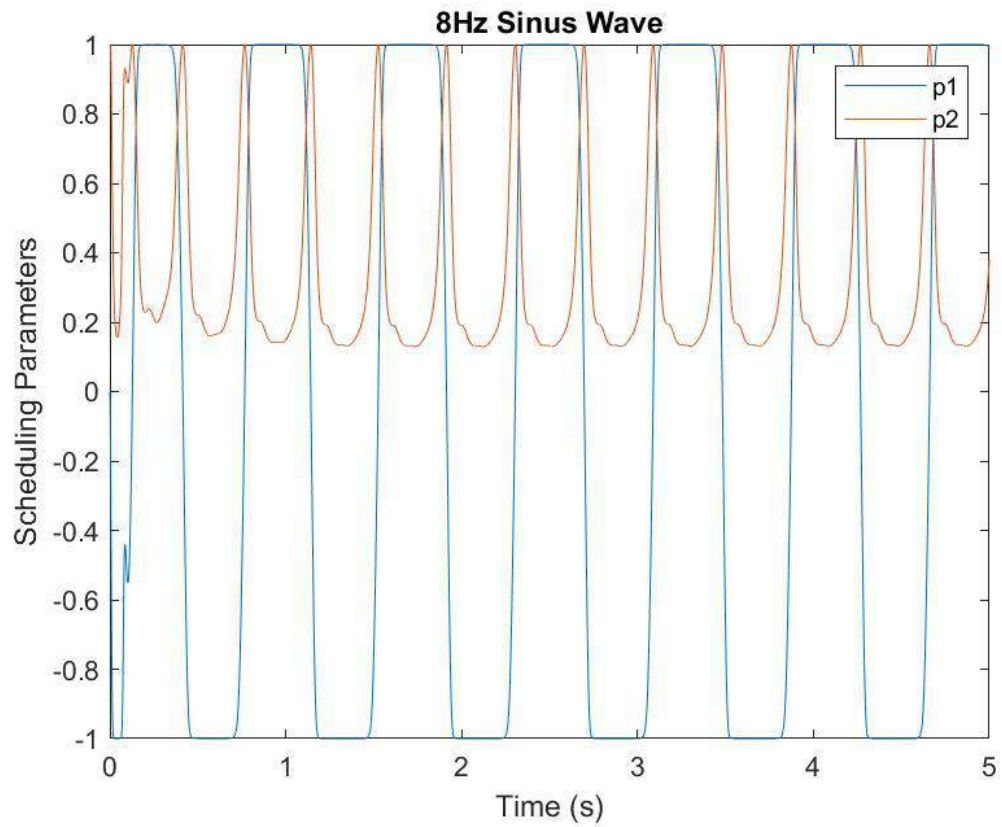


Figure 79: 8Hz Sinus Wave Scheduling Parameters

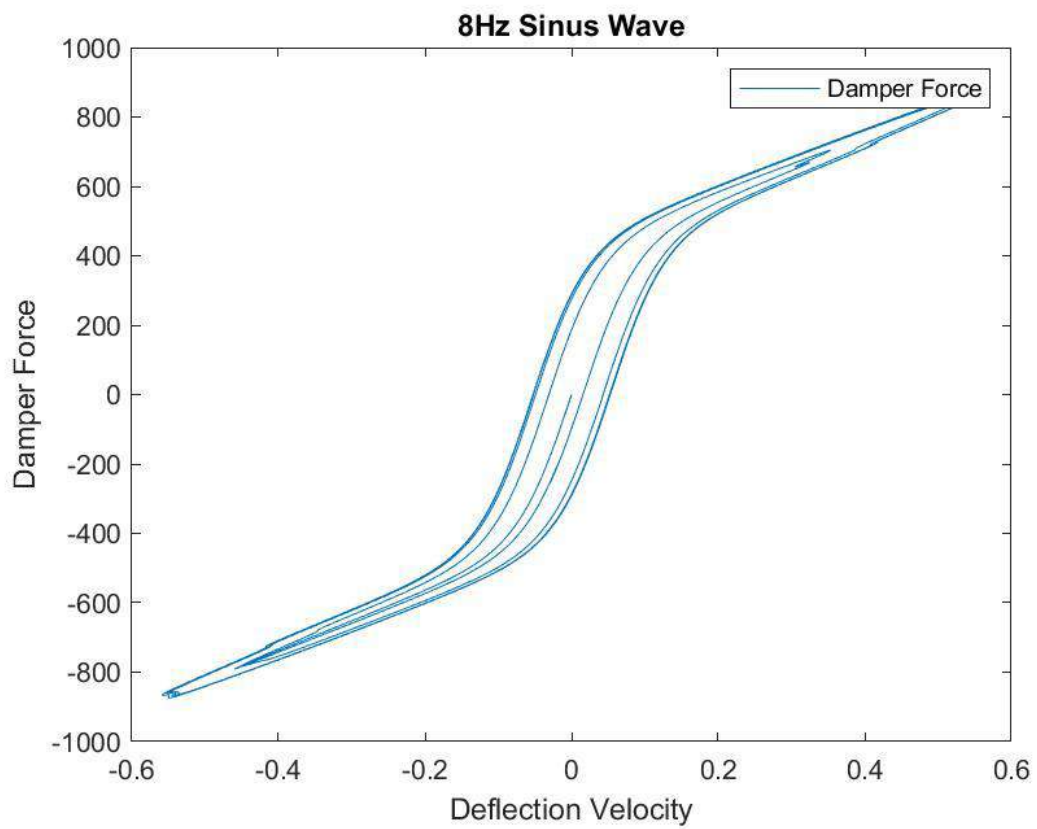


Figure 80: 8Hz Sinus Wave Damper Force

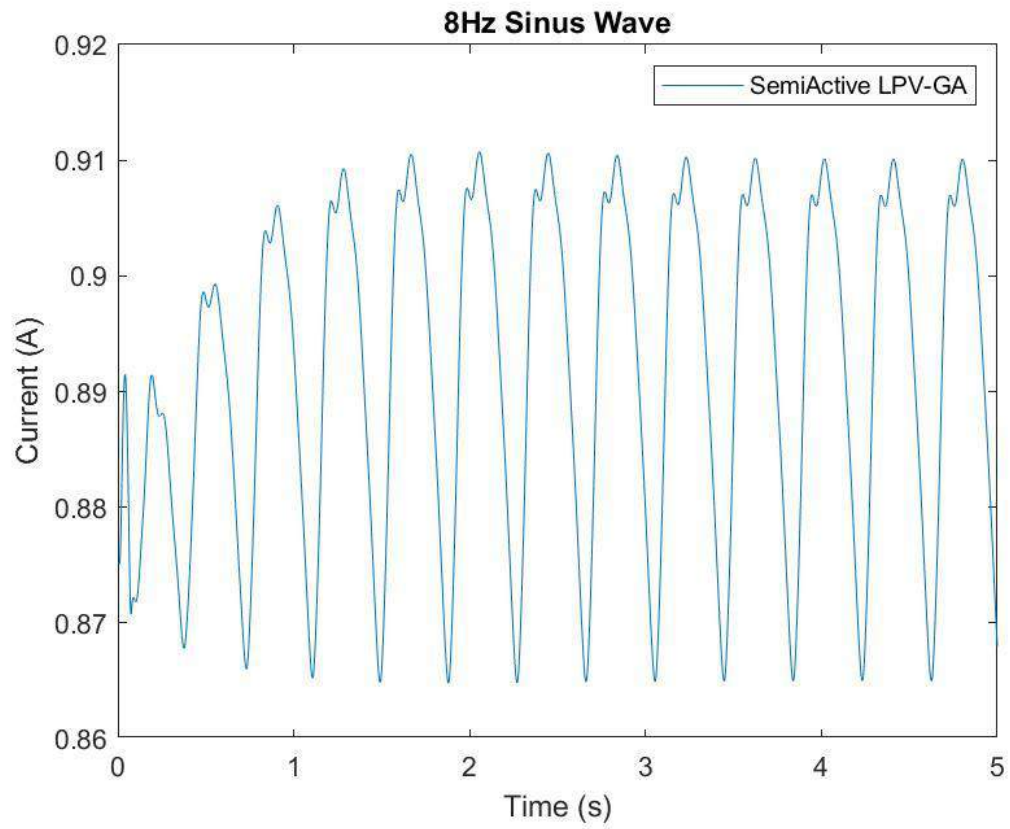


Figure 81: 8Hz Sinus Wave Current

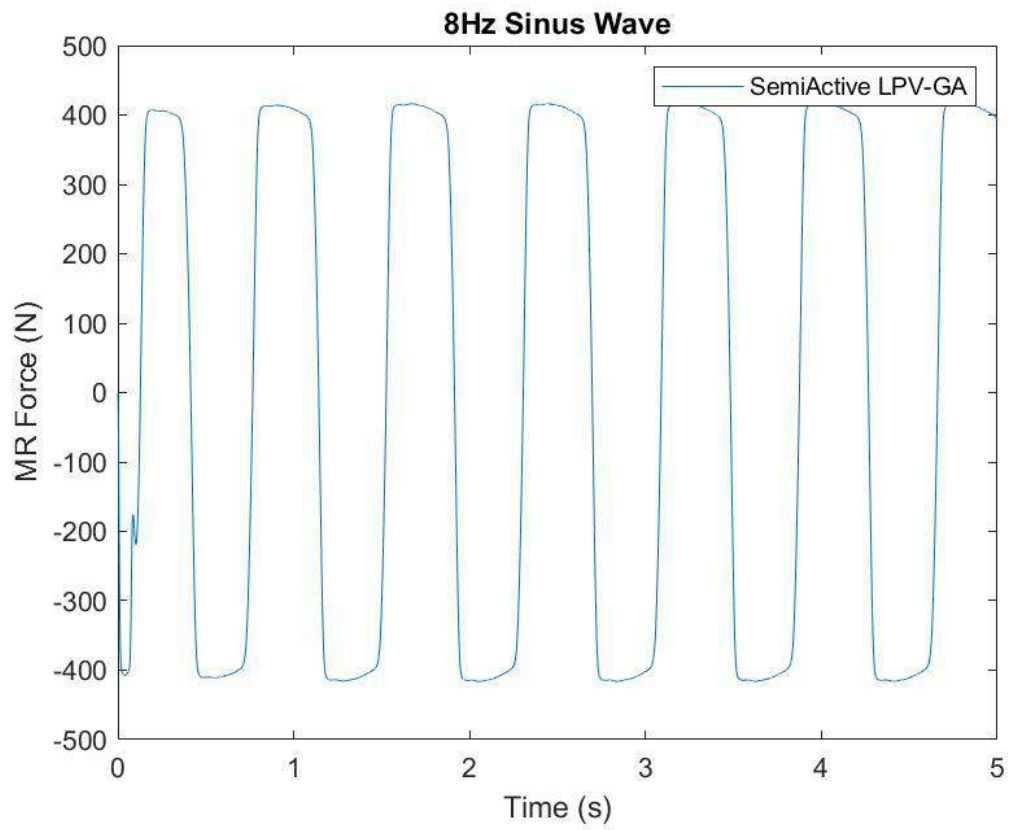


Figure 82: 8Hz Sinus Wave MR Force

9.3. ISO Road Level Test

In general, the distance between the surface of the road and the base plate is referred to as road irregularities. The road profile is often assumed to be a homogenous and isotropical Gaussian random process, and Power Spectral Density (PSD) can be defined as its statistical characteristics [40]. PSD was suggested for the measurement of the road profile by [41] and [42].

$$(9.2) \quad G_q(n) = \frac{\alpha \rho^2}{\pi(\alpha^2 + \rho^2)}$$

where α is the characteristic parameter of the path in m^{-1} and ρ is the variance of the way in m. Both values are shown (e.g. Table 10). [43] The values in Table 1 have been provided in line with the ISO standard road level. As a first order linear method, the road profile in the time domain can be modeled by [42,44]

$$(9.3) \quad \dot{q}(t) = -\alpha v q(t) + \omega(t)$$

where $q(t)$ is unevenness of road in m, v is the speed of the vehicle in km/h, and $\omega(t)$ is a white noise sequence with the following covariance:

$$(9.4) \quad cov[\omega(t)] = E[\omega(t)\omega(t + \tau)] = 2\rho^2 a \delta(\tau)$$

$\alpha=0.15$ and $\rho = 0.03$ has been suggested for asphalt road profile by [45,46]

Table 10: ISO Road Levels (Source: [43])

Road level	α/m	ρ /m
A	0.111	0.0377
B	0.111	0.0754
C	0.111	0.151
D	0.111	0.302
E	0.111	0.603
F	0.111	1.206
G	0.111	2.143
H	0.111	4.825

ISO road profiles is calculated in Simulink according to (8.3) and (8.4) equations and the coefficients are taken from (e.g. Table 10) ISO road level parameters. In this section,

road level B is taken with 72 km/h velocity for simulation. Other test results which are asphalt and C level road with same velocity can be found in Appendix.

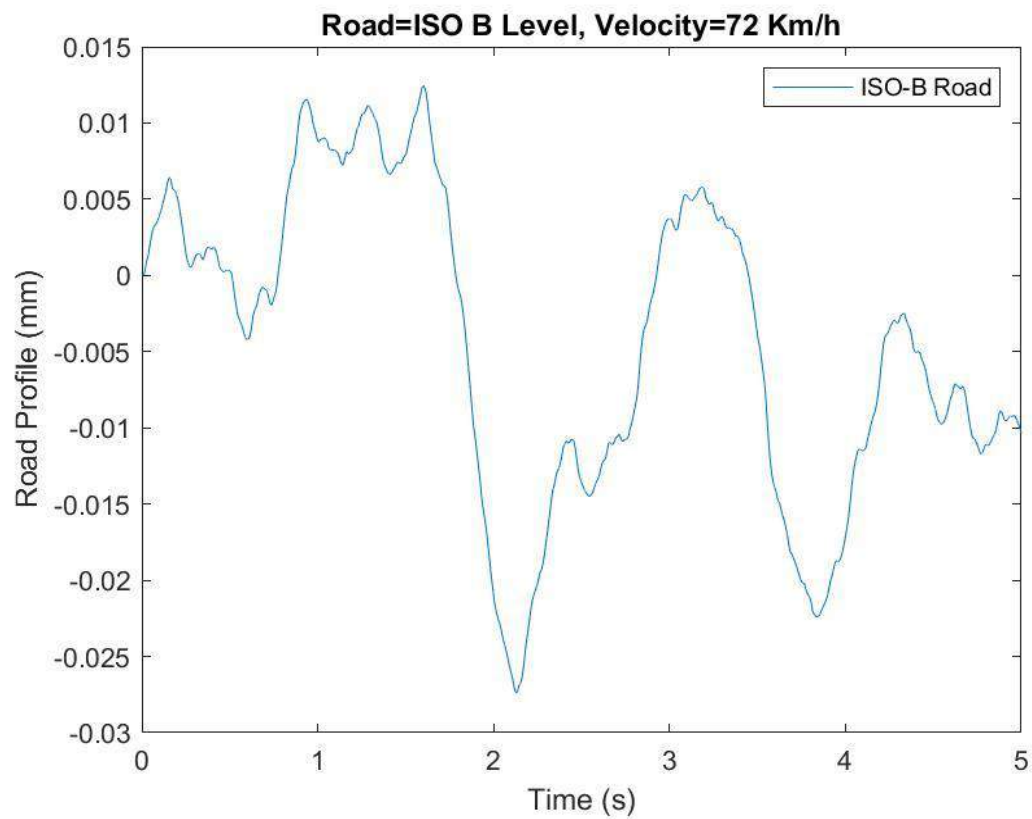


Figure 83: ISO B Road profile

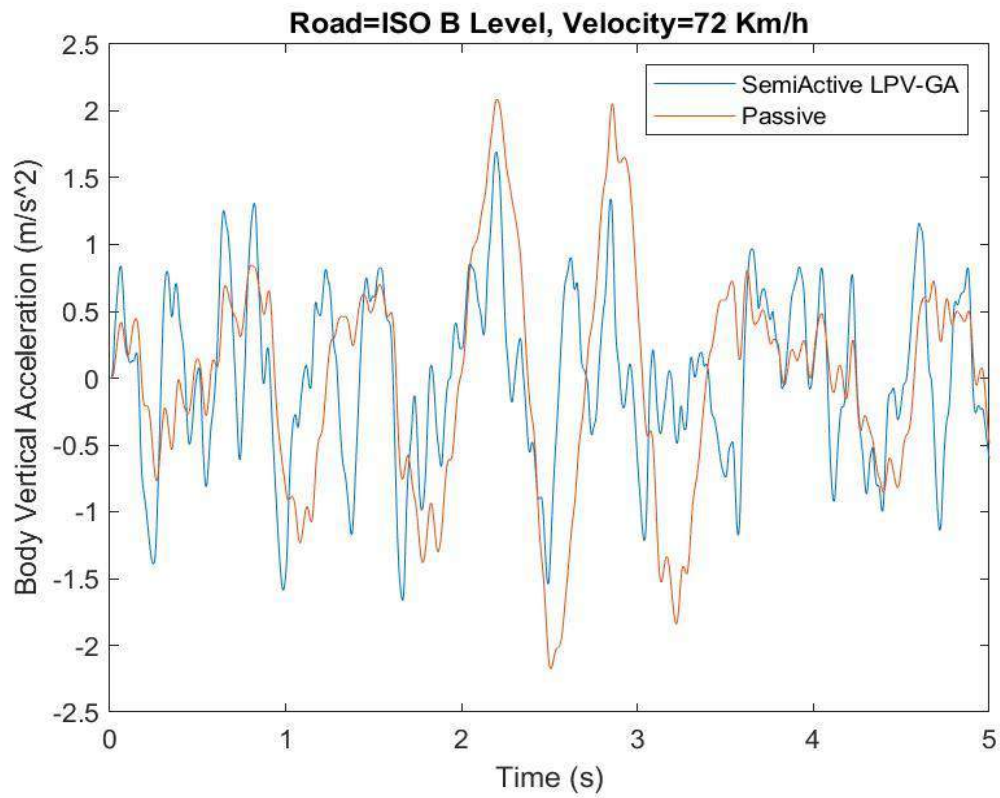


Figure 84 ISO B Road Level Vertical Acceleration

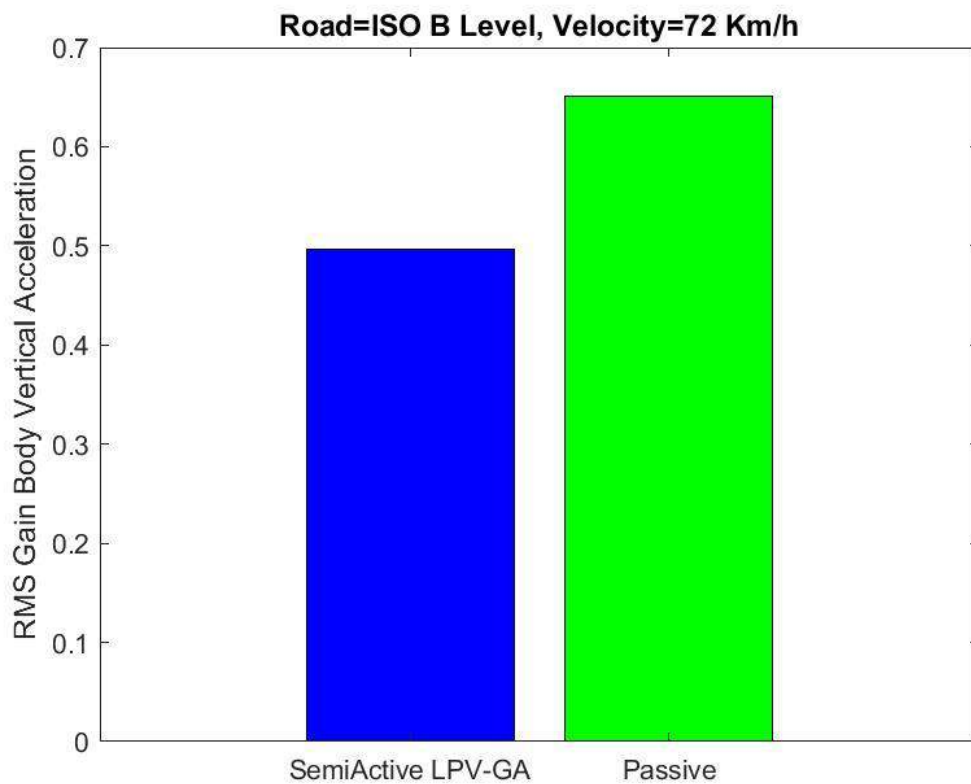


Figure 85: ISO B Road level RMS Gain Vertical Acceleration

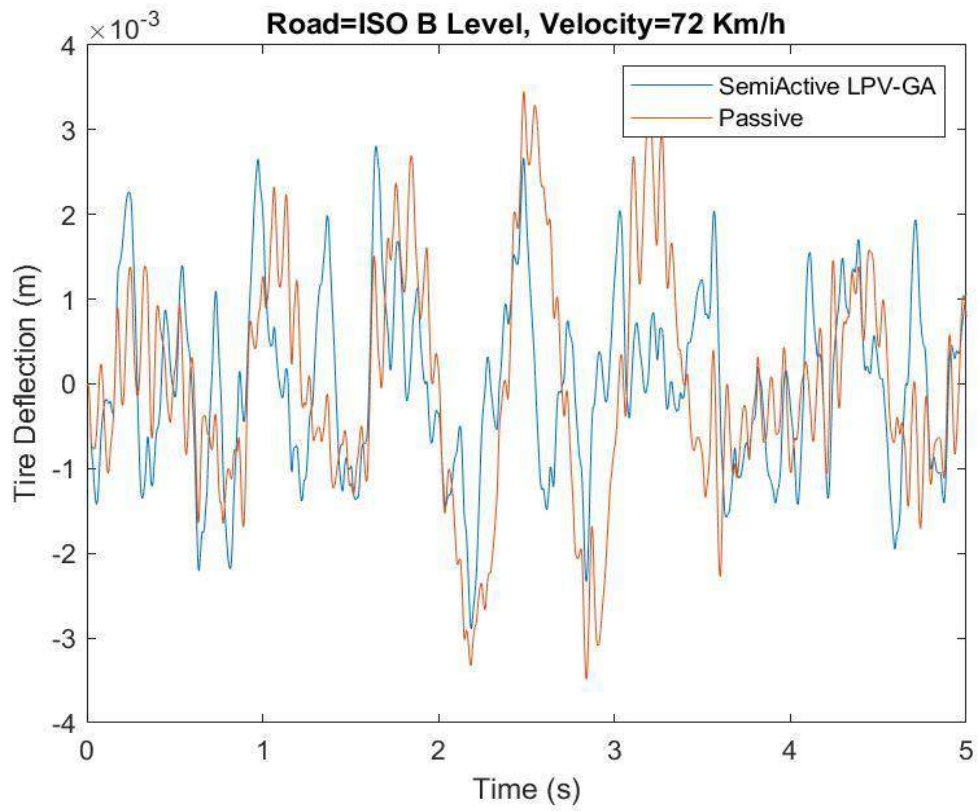


Figure 86: ISO B Road Level Tire Deflection

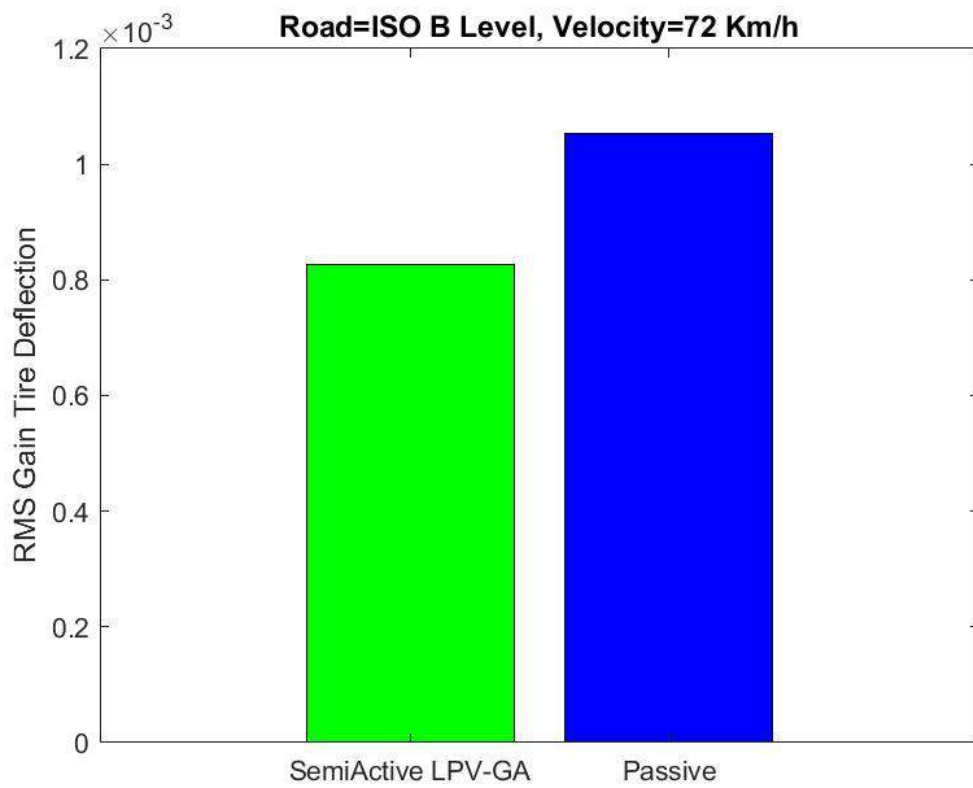


Figure 87 ISO B Road level RMS Gain Tire Deflection

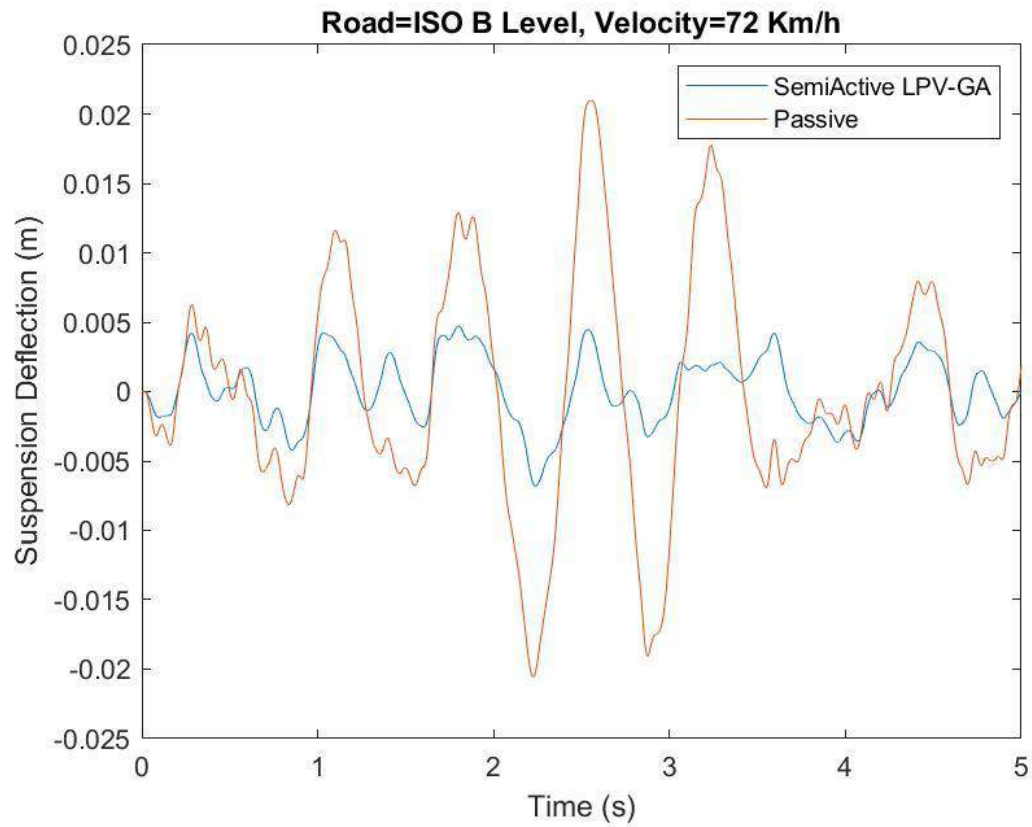


Figure 88: ISO B Road Level Suspension Deflection

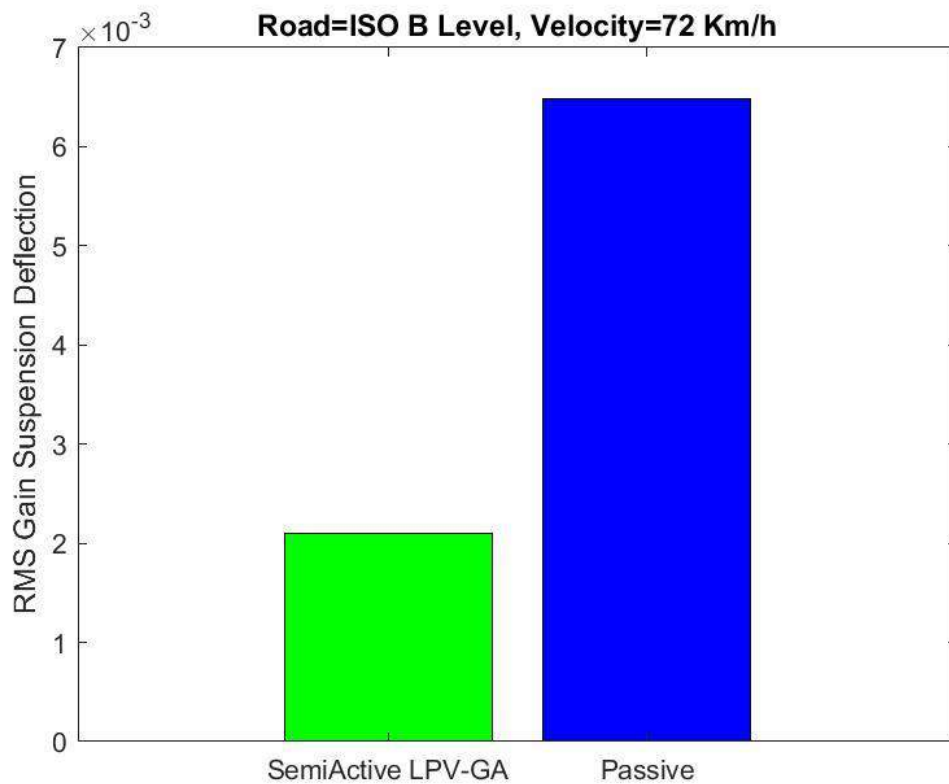


Figure 89: ISO B Road level RMS Gain Suspension Deflection

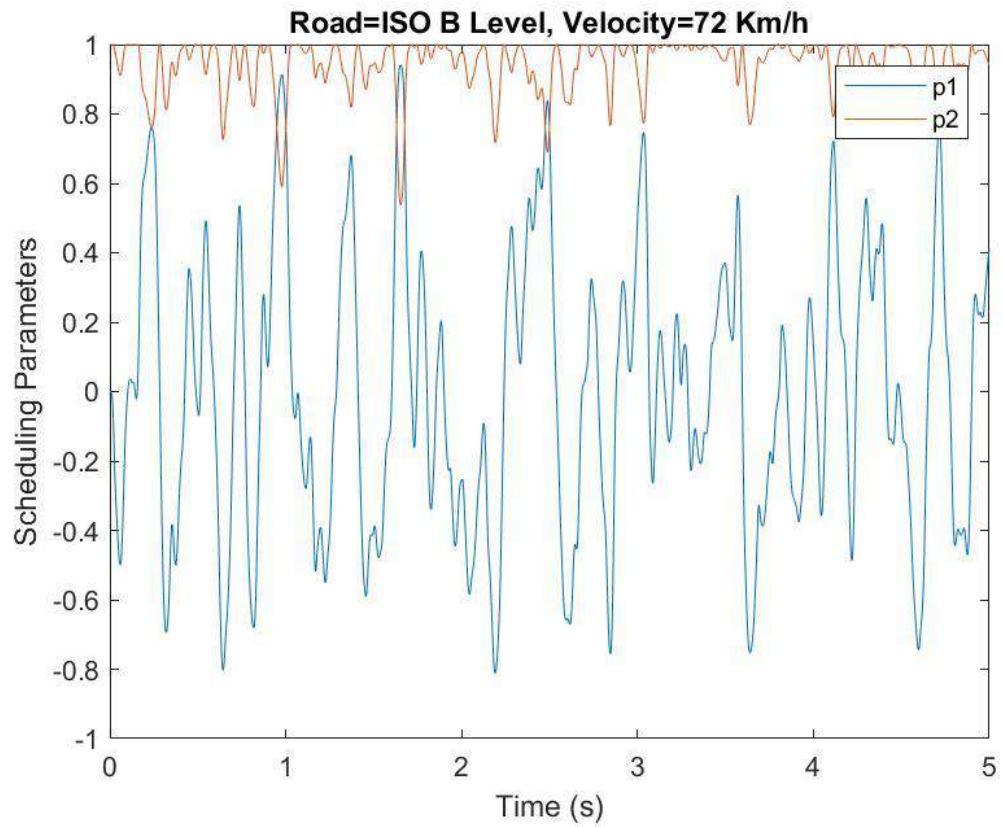


Figure 90: ISO B Road Level Scheduling Parameters

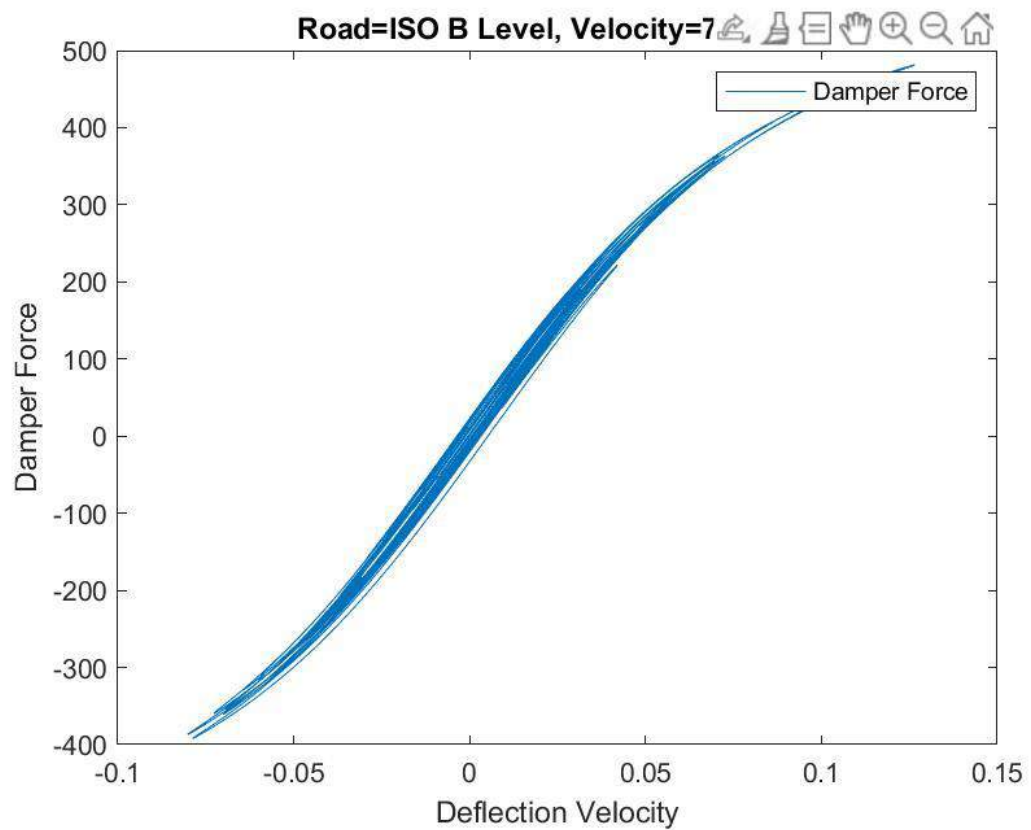


Figure 91: ISO B Road Level Damper Force

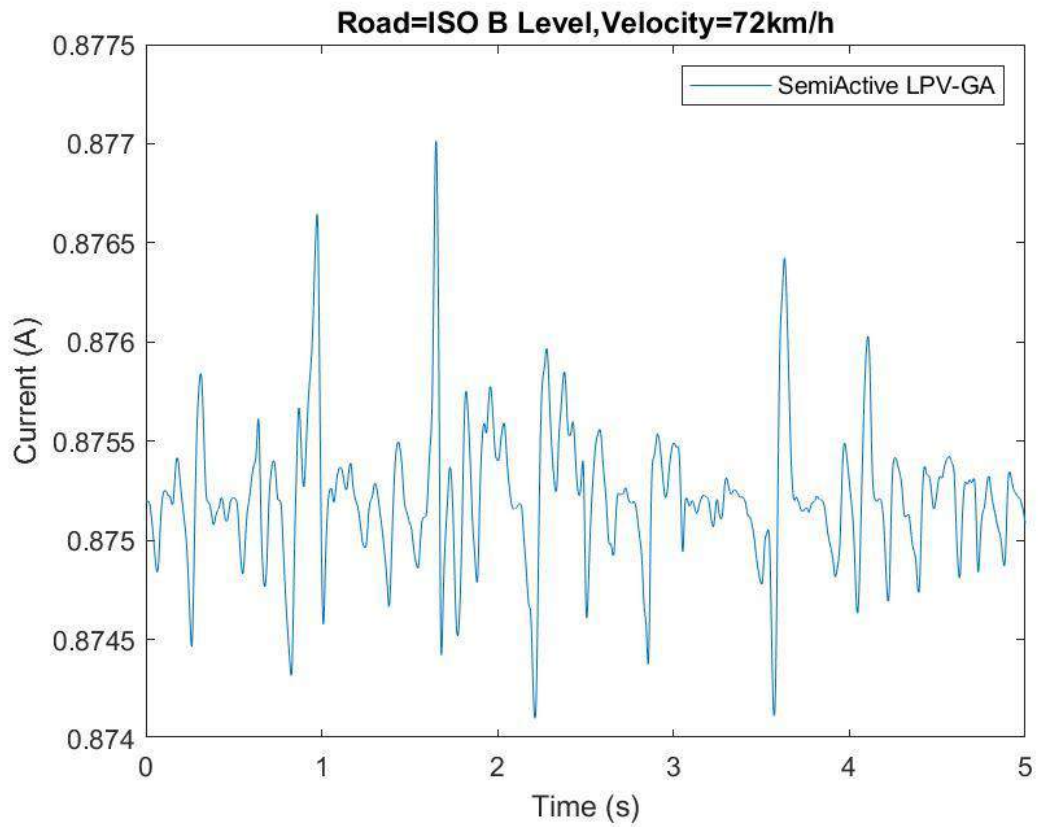


Figure 92: ISO B Road Level Current

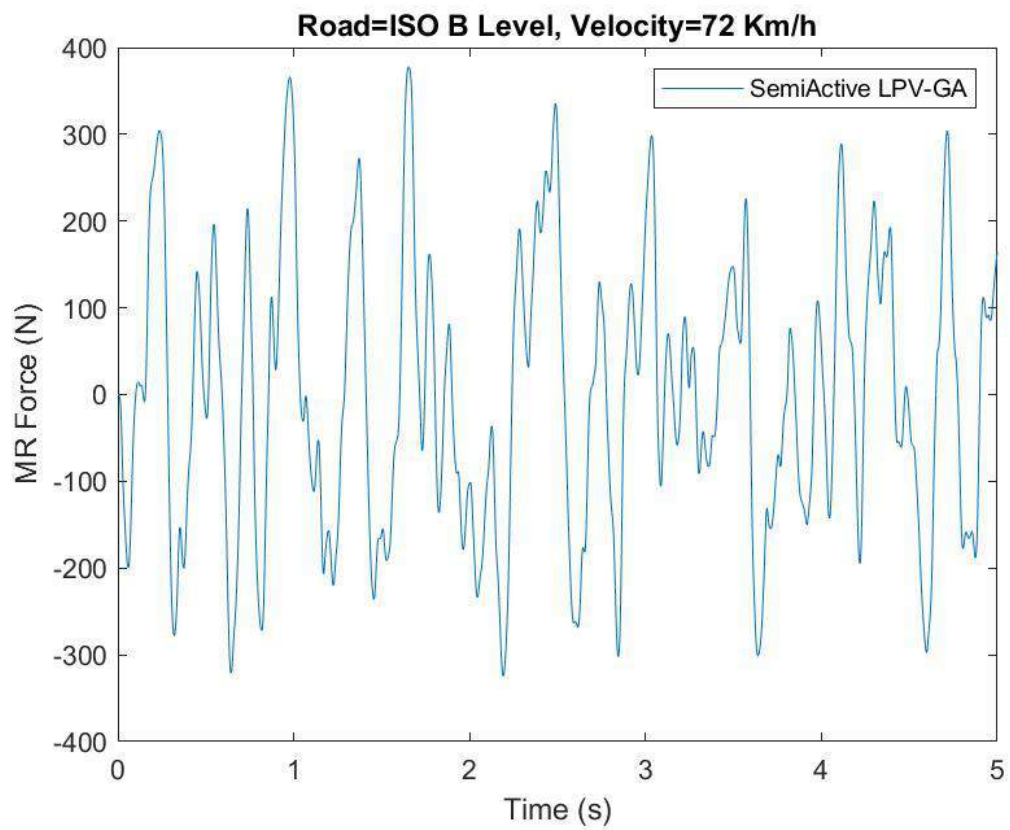


Figure 93: ISO B Road Level MR Force

Table 11: RMS Values for 5 cm Bump Input

5 Cm Bump, Velocity=18 Km/h			
System	Acceleration	Suspension Deflection	Tire Deflection
LPV/ H^∞ Semi-active	0.1716	0.001	0.2818
Passive	0.5439	0.0056	0.8446
Improvement	68.45%	82.14%	66.65%

Table 12: RMS Values for 8Hz Sinus Wave Input

8 Hz Sinus Wave			
System	Acceleration	Suspension Deflection	Tire Deflection
LPV/ H^∞ Semi-active	4.3855	0.041	0.007
Passive	6.6125	0.0676	0.0103
Improvement	33.68%	39.35%	32.04%

Table 13: RMS Values for ISO B Road Input

ISO B Road Level, 72 Km/h			
System	Acceleration	Suspension Deflection	Tire Deflection
LPV/ H^∞ Semi-active	0.4962	0.0021	0.0008
Passive	0.6514	0.0065	0.0011
Improvement	23.83%	67.69%	27.27%

Simulation results demonstrate that LMI based LPV- H^∞ semi-active suspension with MR damper performs better than passive suspension under these test conditions. Disturbance input is selected from different profiles in order to make performance analysis and observe the system under different disturbance conditions in different frequencies. The best improvements for all performance criteria is observed in bump test among all other disturbance inputs for this test. 8Hz sinus wave input is selected specially to observe vertical acceleration in this frequency because of ISO standard for human health. ISO road profile is chosen to observe the system in high frequency spectrum. Body vertical acceleration and tire deflection gain decreases in all simulations. Figures shows that the performance objective ride comfort and road holding is achieved under these simulation conditions.

10. Conclusion

In this thesis, LMI based LPV/ H^∞ gain scheduled control method for car suspension system equipped with MR damper has been implemented. The main steps of thesis may be summarized as follows:

- Static-Guo model is used to define MR damper characteristic for realistic approach
- Dissipativity and saturation constraint is determined for semi-active suspension system with MR damper
- Control and performance objectives are chosen as ride comfort and road holding
- LMI based LPV/ H^∞ gain scheduled controller synthesis is proposed in order to achieve needed requirements
- Polytopic LPV model is built to implement chosen control method
- Genetic algorithms is used to find needed weight functions by solving multi objective problem due to trade-off between ride comfort and road holding
- Controller is implemented with chosen weight functions

MR damper model proposed with control signal f_I which is the function of applied current into system. It makes the model more realistic and can be extended for other current dependent semi-active dampers. On the other hand, other MR damper models can be used to get model easier since control signal f_I must be recast and divided with control input u_I to satisfy dissipativity constraint in this model.

However, LPV control methods are easily implemented in semi-active suspension systems with displacement sensor to measure suspension deflection and suspension deflection velocity in terms of practical view. Suspension deflection velocity can be calculated as a derivative of measured suspension deflection.

Selection of weighting function is the most important and the hardest part for robust controller. Common way to find appropriate weight functions is trial by error. Genetic algorithms give the opportunity the find suitable and more accurate weight functions easier than classical methods for multi objective optimization problems. This method is applied on this study and gives achievable results.

The semi-active suspension with MR damper simulations on the quarter-car model have shown good results both in frequency and time. Especially, ride comfort and road holding gives better results than passive suspension model in low and middle frequencies 0-12 Hz. Ride comfort between 4-8 Hz is specially important which is the most sensitive frequency range for human health in line with ISO 2631. Proposed controller achieved this performance objective very well.

LMI based LPV/ H^∞ gain scheduled controller achieved the performance objectives in general but study can be improved to get better results in higher frequencies and still can be found better weight functions by changing the bounds of optimization problem.

References

- [1] Sergio M. Savaresi, Charles Poussot-Vassal, Cristiano Spelta, Olivier Sename, Luc Dugard. **Semi-Active Suspension Control Design for Vehicles**. Butterworth-Heinemann, (2010). pp. 1-32
- [2] Guo, S., Yang, S., Pan, C.: Dynamic modeling of magnetorheological damper behaviors. *Journal of Intelligent Material Systems and Structures* 17, 3–14 (2006)
- [3] O. Sename, Dugard. **An LPV control approach for semi-active suspension control with actuator constraints**. Baltimore, Maryland, USA (2010)
- [4] Do, A.L., Spelta, C., Savaresi, S., Sename, O., Dugard, L., Delvecchio, D. **An LPV control approach for comfort and suspension travel improvements of semi-active suspension systems**. , Atlanta, GA, (2010), pp. 5660–5665
- [5] Do, A.L., Lozoya-Santos, J., Sename, O., Dugard, L., Ramirez-Mendoza, R.A., MoralesMenendez. **Modélisation et commande LPV d'un amortisseur magnéto-rhéologique**. In: **Proceedings de la Conference Internationale Francophone d'Automatique**, Nancy, France (2010)
- [6] Anh-lam Do, Oliver Sename, Luc Dugard [et.al] **LPV Modeling and Control of Semi-active Dampers in Automotive Systems** . DOI: 10.1007/978-1-4614-1833-7_15, (2014) pp. 1-10
- [7] Lozoya-Santos, J.J., Aubouet, S., Morales-Menendez, R., Oliver Sename, Ramirez-Mendoza, R.A., Dugard, L. **Simulation performance of a quarter of vehicle including an MR damper model with hysteresis**. Prague, Czech Republic (2010)
- [8] Anh-lam Do, Oliver Sename, Luc Dugard [et.al] **LPV Modeling and Control of Semi-active Dampers in Automotive Systems** . DOI: 10.1007/978-1-4614-1833-7_15, (2014) pp. 13-14
- [9] Guglielmino, E., Sireteanu, T., Stammers, C., Ghita, G., Giuclea, M.: **Semi-Active Suspension Control**. Springer Verlag (2008)
- [10] Wu, Fen. **Control of Linear Parameter Varying systems**. Univ. of California, Berkeley, (1995)
- [11] Da-Wei Gu Petko H. Petkov Mihail M. Konstantinov. **Robust Control Design with MATLAB® Second Edition**. Springer, (2013)
- [12] *Mathworks-Matlab*
URL: <https://www.mathworks.com/help/control/ref/lpvsystem.html>
- [13] Da-Wei Gu Petko H. Petkov Mihail M. Konstantinov. **Robust Control Design with MATLAB® Second Edition**. Springer, (2013)

- [14] Pascal M Gahinet, Arkadi Nemorivski, A.J Laub, M.Chilali. **LMI control toolbox user's guide**. The MathWorks, Inc. (1995).
- [15] Jaffar Seyyed Esmaeili, Ahmad Akbari, Mehdi Mirzaei. **LPV H2 Control of Semi-Active Suspensions**. Shiraz, Iran (2011)
- [16] Poussot-Vassal, C., Sename, O., Dugard, L., G´asp´ar, P., Szab´o, Z., Bokor, J. **New semi-active suspension control strategy through LPV technique**. Control Engineering Practice Vol.16 No. 12, (2008), pp. 1519–1534
- [17] Grimm, G., Hatfield, J., Postlethwaite, I., Teel, A., Turner, M., Zaccarian, L. **Antiwindup for stable linear systems with input saturation: An LMI-based synthesis**. IEEE Transaction on Automatic Control Vol.48 No.9, (2003), pp. 1509–1525
- [18] Kothare, M.V., Campo, P.J., Morari, M., Nett, C.N. **A unified framework for the study of anti-windup designs**. Automatica 30, (1994), pp. 1869–1883
- [19] Mulder, E.F., Tiwari, P.Y., Kothare, M.V. **Simultaneous linear and anti-windup controller synthesis using multiobjective convex optimization**. Automatica 45, (2009), pp. 805–811
- [20] Wu, F., Grigoriadis, K.M., Packard, A. **Anti-windup controller design using linear parameter-varying control methods**. International Journal of Control Vol.73 No.12, (2000), pp. 1104–1114
- [21] Doyle, J.C., Glover, K., Khargonekar, P., and Francis, B. **State-Space Solutions to Standard H2 and H ∞ Control Problems**. IEEE Trans. Aut. Contr., AC-34 (1989), pp. 831–847.
- [22] Gahinet, P., and A.J. Laub. **Reliable Computation of γ_{opt} in Singular H ∞ Control** Lake Buena Vista, Fl., (1994), pp. 1527–1532.
- [23] Gahinet, P., and P. Apkarian. **A Linear Matrix Inequality Approach to H ∞ Control**. Int. Robust and Nonlinear Contr., 4 (1994), pp. 421–448.
- [24] Iwasaki, T., and R.E. Skelton. **All Controllers for the General H ∞ Control Problem: LMI Existence Conditions and State-Space Formulas**. Automatica, 30 (1994), pp. 1307–1317
- [25] Gahinet. **P. Explicit Controller Formulas for LMI-based H ∞ Synthesis**. Proc. Amer. Contr. Conf., (1994), pp. 2396– 2400
- [26] Boyd, S., and Q. Yang. **Structured and Simultaneous Lyapunov Functions for System Stability Problems**. Int. J. Contr., 49 (1989), pp. 2215–2240.
- [27] Barmish, B.R. **Stabilization of Uncertain Systems via Linear Control**. IEEE Trans. Aut. Contr., AC-28 (1983), pp. 848–850.

- [28] Horisberger, H.P., and P.R. Belanger. **Regulators for Linear Time-Varying Plants with Uncertain Parameters**. IEEE Trans. Aut. Contr., AC-21 (1976), pp. 705–708.
- [29] Packard, A. **Gain Scheduling via Linear Fractional Transformations**. Syst. Contr. Letters, 22, (1994), pp. 79–92.
- [30] Apkarian, P., P. Gahinet, and G. Becker. **Self-Scheduled H^∞ Control of Linear Parameter-Varying Systems**. Automatica, 1995.
- [31] Becker, G., Packard, P. **Robust Performance of Linear Parametrically Varying Systems Using Parametrically Dependent Linear Feedback**. Systems and Control Letters, 23 (1994), pp. 205–215.
- [32] Apkarian, P., and P. Gahinet. **A Convex Characterization of Gain-Scheduled H^∞ Controllers**. Trans. Aut. Contr., 1995.
- [33] Carsten Scherer, Pascal Gahinet, Mahmoud Chilali. **Multiobjective Output-Feedback Control via LMI Optimization**. IFAC Proceedings Volumes Vol.29 No. 1, (1996), pp. 1691–1696
- [34] Olivier Sename, Péter Gáspár, and József Bokor. **Robust Control and Linear Parameter Varying Approaches Application to Vehicle Dynamics**. Springer, (2013).
- [35] Beaven, R.W., Wright, M.T., Seaward, D.R. **Weighting function selection in the H^∞ design process**. Control Engineering Practice 4(5), (1996), pp. 625–633
- [36] Anh Lam Do, Boussaad Soualmi, Jorge de Jesus Lozoya-Santos, Olivier Sename, Luc Dugard, Ricardo Ramirez-Mendoza. **Optimization of weighting function selection for H^∞ control of semi-active suspensions**. Budapest, Hungary, (2010), pp. 12
- [37] Alfaro-Cid, E., McGookin, E.W., and Murray-Smith, D.J. **Optimisation of the weighting functions of an H^∞ controller using genetic algorithms and structured genetic algorithms**. International Journal of Systems Science, 39:4, (2008), pp. 335–347
- [38] Goldberg, D. **Genetic Algorithms in Searching Optimisation and Machine Learning**. (1989)
- [39] Davis, L.D. **Handbook of genetic algorithms**. Van Nostrand Reinhold. (1991)
- [40] M. Canale, M. Milanese, and C. Novara. **Semi-active suspension control using ‘fast’ model-predictive techniques**. IEEE Transactions on Control Systems Technology, vol. 14, no. 6, (2006), pp. 1034–1046
- [41] Y. Hurmuzlu and O. D. Nwokah. **The Mechanical Systems Design Handbook: Modeling, Measurement, and Control**. CRC Press, Danvers, Mass, USA, (2006)

- [42] P. Michelberger, L. Palkovics, and J. Bokor. **Robust design of active suspension system.** International Journal of Vehicle Design, vol. 14, no. 2-3, (1993), pp. 145–165
- [43] Z. C. Wu, S. Z. Chen, L. Yang, and B. Zhang. **Model of road roughness in time domain based on rational function.** Transaction of Beijing Institute of Technology, vol. 29, no. 9, (2009) , pp. 795–798
- [44] E. Guglielmino, T. Sireteanu, C. W. Stammers, G. Ghita, and M. Giuclea. **Semi-Active Suspension Control: Improved Vehicle Ride and Road Friendliness.** Springer, (2008), London, UK
- [45] Shun-Hsu Tu, Wes S. Jeng, Feng Tyan, Yu-Fen Hong. **Generation of Random Road Profiles.** Computational Dynamics and Control Lab Dept. of Aerospace Engineering Tamkang University Tamshui, Taipei County, Taiwan 25147
- [46] A. Hac. **Suspension optimization of a 2-dof vehicle model using a stochastic optimal control technique.** Journal of Sound and Vibration, vol. 100, no. 3, (1985) pp. 343–357

Appendix

