Final Year Project

Submitted in fulfillment of the requirements for

ENGINEERING DEGREE FROM THE LEBANESE UNIVERSITY
Faculty of Engineering- BRANCH III

Major: Mechanical Engineering - Construction

By:

Houssein Hassan Al-Moussawi

Development of a Control System for Heavy Weight Vehicles to Prevent Roll-over and Enhance Ride Handling

Advisor:
Dr. Hassan Shraim

Defended on 8th of July 2013 at the jury:

Dr. Bassam Riachi       President
Dr. Hassan Lakkis       Member
Dr. Hassan Shraim       Member
Acknowledgements

To the light, my God, who guided me through the way and led me to accomplish this fine work, goes my greatest and faithful thanks...

I would like hereby to express my sincere gratitude to my supervisor Dr. Hassan Shraim for his firm support, encouragement and assistant throughout the duration of my study in the University. His invaluable guidance provided many ideas, which have led to the completion of this research.

I would like also to express endless gratitude to my parents, brothers, and sisters who gave me everything they could to enable me to reach the highest possible education level. I only hope that they know how their love, support and patience encouraged me to fulfill their dream.

I am also grateful to the responsible of the “Mechanical Department in the faculty of Engineering” Dr. Mohammad Hammoud, the Dean of the “Faculty of Engineering in Lebanese University” Dr. Rafic Younes, the responsible of the “Doctoral School for Sciences and Technologies EDST” Dr. Khalil El Khoury, and to every one contributed in the success of this work for their support and patience.

To every person gave me something to light my pathway, I thank him for believing in me...
And say: O my Lord! Increase my Knowledge

(Taha-114)
Abstract

Engineering is concerned with understanding and controlling the materials and forces of nature for the benefit of humankind. The twin goals of understanding and controlling are complementary because effective systems control requires that the systems be understood and modeled. Furthermore, control engineering must often consider the control of poorly understood systems such as chemical process systems.

In this medium, the project was chosen to find a control system for heavy weight vehicles which plays the role of preventing roll over in the first place, enhancing ride handling, and increasing passengers comfort in second and third places.

For this, the research is divided into several chapters in which the first one explains a brief history of vehicles suspensions systems and states the differences between passive, semi-active, and active (controlled) systems. The second chapter is dedicated for motion study of three consecutive systems: quarter, half, and full linear vehicle systems which will be modeled using Matlab Simulink software and their response to rough road excitations will be simulated.

Chapters three and four will present two types of controllers which are PID and FL controllers in which the theory of control will be explained and the way of controller’s work will be discussed. Then each controller will be implemented into each previous model and the new response due to the same road stimulation will be simulated. Afterwards, the passive, PID controlled, and FL controlled system responses will be presented for comparison. In order to fulfill this evaluation, braking and bending torques will be applied to simulate the vehicle’s velocity and cornering, and step road input will be applied too, then the systems are compared again and results are obtained and discussed in an approach to accomplish our conclusions of which the last chapter will be dedicated for.
# Table of Contents

Chapter 1. Introduction..............................................................................................................1
1.1 Vehicle Suspension Systems ...............................................................................................1
1.2 Problems and Research Objectives ....................................................................................4

Chapter 2. Mathematical Modeling........................................................................................5
2.1 Basic Considerations ..........................................................................................................5
2.2 Quarter Vehicle Model ......................................................................................................5
   2.2.1 Physical Setup ...........................................................................................................5
   2.2.2 Equations of Motion ................................................................................................6
   2.2.3 Implementation of Model using Matlab© .................................................................6
   2.2.4 Simulation and Results .............................................................................................7
2.3 Simplified Half Vehicle Model ..........................................................................................9
   2.3.1 Physical Setup .........................................................................................................9
   2.3.2 Formulation ...........................................................................................................10
   2.3.3 Implementation of Linear and Nonlinear Models into Simulink .........................11
   2.3.4 Simulations and Results ..........................................................................................12
2.4 Simplified Linear Full Vehicle Model ..............................................................................14
   2.4.1 Physical Setup and Formulation ..............................................................................14
   2.4.2 Implementation of the Process Model Using Matlab ............................................17
   2.4.3 Simulations and Results ..........................................................................................19
2.5 Summary ............................................................................................................................26

Chapter 3. PID Controller: Theory and Application ............................................................27
3.1 Control System .................................................................................................................27
3.2 Open-Loop and Closed-Loop Control Systems ...............................................................27
3.3 PID Controllers ...............................................................................................................29
   3.3.1 Proportional Controller ..........................................................................................30
   3.3.2 Integral Controller .................................................................................................31
   3.3.3 Derivative Controller .............................................................................................32
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.4 Characteristics of P, I, and D Controllers</td>
<td>33</td>
</tr>
<tr>
<td>3.3.5 PID Tuning</td>
<td>33</td>
</tr>
<tr>
<td>3.4 Implementation of the PID Controller, Simulations, and Results</td>
<td>36</td>
</tr>
<tr>
<td>3.4.1 Quarter Vehicle System</td>
<td>36</td>
</tr>
<tr>
<td>3.4.2 Half Vehicle System</td>
<td>38</td>
</tr>
<tr>
<td>3.4.3 Full Vehicle System</td>
<td>40</td>
</tr>
<tr>
<td>Chapter 4. Fuzzy Logic Controller: Theory and Application</td>
<td>45</td>
</tr>
<tr>
<td>4.1 General Definitions</td>
<td>45</td>
</tr>
<tr>
<td>4.2 Fuzzy logic system structure</td>
<td>47</td>
</tr>
<tr>
<td>4.3 Designing FLC for Suspension System</td>
<td>50</td>
</tr>
<tr>
<td>4.4 Implementation of FLC, Simulations, and Results</td>
<td>53</td>
</tr>
<tr>
<td>4.4.1 Quarter Vehicle System</td>
<td>53</td>
</tr>
<tr>
<td>4.4.2 Half Vehicle System</td>
<td>54</td>
</tr>
<tr>
<td>4.4.3 Full Vehicle System</td>
<td>56</td>
</tr>
<tr>
<td>4.5 One More Comparison Step</td>
<td>60</td>
</tr>
<tr>
<td>Chapter 5. Conclusions and Recommendations</td>
<td>63</td>
</tr>
<tr>
<td>5.1 Conclusions</td>
<td>63</td>
</tr>
<tr>
<td>5.2 Recommendations for Future Work</td>
<td>64</td>
</tr>
<tr>
<td>References</td>
<td>66</td>
</tr>
</tbody>
</table>
List of Figures

Figure 1.1 (a) Passive Suspension, (b) Semi-Active Suspension, (c) Active Suspension 2
Figure 2.1 Quarter Vehicle Suspension Model ............................................................... 6
Figure 2.2 Matlab Simulink Model of Quarter Vehicle System ......................................... 7
Figure 2.3 Time response of vertical displacement X1 ....................................................... 8
Figure 2.4 Time response of vertical displacement X2 ....................................................... 8
Figure 2.5 Time response of vertical displacement (X1-X2) compared to that of X1 & X2 ................................................................................................................................. 9
Figure 2.6 Simplified Half Vehicle suspension Model ....................................................... 9
Figure 2.7 Simulink Half Vehicle suspensions Linear Model ............................................. 11
Figure 2.8 Simulink Half Vehicle suspensions Nonlinear Model ....................................... 12
Figure 2.9 Time response of X displacement of linear and nonlinear systems ................. 13
Figure 2.10 Time response of X' vertical velocity of linear and nonlinear systems .... 13
Figure 2.11 Time response of pitch angle θ of linear and nonlinear systems ................. 13
Figure 2.12 Simplified Full vehicle Model ....................................................................... 15
Figure 2.13 Full Vehicle Linear Simulink Model ............................................................ 17
Figure 2.14 Under Mask of the Signal Outputs Dotted Rectangle of Figure 2.13 .......... 18
Figure 2.15 Under Mask of the Proposed Model Dotted Rectangle of Figure 2.13 .... 19
Figure 2.16 Time response of the vertical displacement at P1 ........................................ 20
Figure 2.17 Time response of the vertical displacement at P2 ........................................ 21
Figure 2.18 Time response of the vertical displacement at P3 ........................................ 21
Figure 2.19 Time response of the vertical displacement at P4 ........................................ 21
Figure 2.20 Time response of the vertical displacement at Pc ........................................ 22
Figure 2.21 Time response of pitch angle ........................................................................ 22
Figure 2.22 Time response of roll angle .......................................................................... 22
Figure 2.23 Braking Torque ............................................................................................. 23
Figure 2.24 Bending Torque ............................................................................................. 23
Figure 2.25 Time response of a vertical displacement at P1 ............................................ 23
Figure 2.26 Time response of a vertical displacement at P2 ............................................ 24
Figure 2.27 Time response of a vertical displacement at P3 ............................................ 24
Figure 2.28 Time response of a vertical displacement at P4 ............................................ 24
Figure 2.29 Time response of a vertical displacement at Pc ........................................... 25
Figure 2.30 Time response of pitch angle ........................................................................ 25
Figure 2.31 Time response of roll angle .......................................................................... 25
Figure 3.1 A Block Diagram of General Control System ............................................... 27
Figure 3.2 Open-Loop System ........................................................................................ 28
Figure 3.3 Closed-Loop System ..................................................................................... 28
List of Figures

Figure 3.4 The PID Controllers Block Diagram .............................................................. 29
Figure 3.5 Characterization of the unit step response by two (a) and three (b) parameters ................................................................................................................................. 35
Figure 3.6 PID Controlled Quarter Vehicle System .......................................................... 36
Figure 3.7 Time response of vertical displacement (X1-X2) ............................................ 37
Figure 3.8 Time response of vertical displacement (X1-X2) - Passive vs PID ................. 37
Figure 3.9 PID Controlled Half Vehicle System ............................................................... 38
Figure 3.10 Undermask of the Controller Block of Figure 3.9 .......................................... 38
Figure 3.11 Time response of vertical displacement X ..................................................... 39
Figure 3.12 Time response of the pitch angle ................................................................. 39
Figure 3.13 Time response of vertical displacement X - Passive vs PID ......................... 40
Figure 3.14 Time response of the pitch angle - Passive vs PID ...................................... 40
Figure 3.15 Undermask of Controller Block implemented in Full Vehicle Model ............. 41
Figure 3.16 Time response of displacement at P1 ......................................................... 41
Figure 3.17 Time response of displacement at P2 ......................................................... 41
Figure 3.18 Time response of displacement at P3 ......................................................... 42
Figure 3.19 Time response of displacement at P4 ......................................................... 42
Figure 3.20 Time response of vertical displacement at Pc ............................................. 42
Figure 3.21 Time response of the pitch angle ............................................................... 42
Figure 3.22 Time response of the roll angle .................................................................. 43
Figure 3.23 Displacement at P1 - Passive vs PID ......................................................... 43
Figure 3.24 Displacement at P2 - Passive vs PID ......................................................... 43
Figure 3.25 Displacement at P3 - Passive vs PID ......................................................... 43
Figure 3.26 Displacement at P4 - Passive vs PID ......................................................... 43
Figure 3.27 Vertical displacement at Pc - Passive vs PID ............................................. 44
Figure 3.28 Pitch angle - Passive vs PID ........................................................................ 44
Figure 3.29 Roll Angle - Passive vs PID ....................................................................... 44
Figure 4.1 Different membership functions shapes found in Matlab .............................. 47
Figure 4.2 Fuzzy Logic System ..................................................................................... 48
Figure 4.3 Membership Function for displacement and velocity input variables ............. 52
Figure 4.4 Membership Function for acceleration input variable ................................... 52
Figure 4.5 Membership Function for the desired actuator force output variable ............ 52
Figure 4.6 Input and output relations in the form of cartesian rule surfaces .................. 53
Figure 4.7 Fuzzy Logic Controlled Quarter Vehicle System .......................................... 53
Figure 4.8 Time response of vertical displacement (X1-X2) ........................................... 53
Figure 4.9 Time response of vertical displacement (X1-X2) – FLC vs Passive vs PID ... 54
Figure 4.10 FLC Block of Half Vehicle Model ............................................................... 54
Figure 4.11 Time response of vertical displacement X .................................................. 54
Figure 4.12 Time response of pitch angle ..................................................................... 55
Figure 4.13 Time response of vertical displacement X – FLC vs Passive vs PID .......... 55
Figure 4.14 Time response of pitch angle - FLC vs Passive vs PID .............................. 55
List of Figures

Figure 4.15 FLC Block in Full Vehicle Model ................................................................. 56
Figure 4.16 Time response of vertical displacement at P1.............................................. 56
Figure 4.17 Time response of vertical displacement at P2.............................................. 57
Figure 4.18 Time response of vertical displacement at P3.............................................. 57
Figure 4.19 Time response of vertical displacement at P4.............................................. 57
Figure 4.20 Time response of vertical displacement at Pc............................................. 58
Figure 4.21 Time response of pitch angle......................................................................... 58
Figure 4.22 Time response of roll angle........................................................................... 58
Figure 4.23 Time response of vertical displacement at Pc - FLC vs Passive vs PID...... 59
Figure 4.24 Time response of pitch angle - FLC vs Passive vs PID ................................ 59
Figure 4.25 Time response of roll angle - FLC vs Passive vs PID ................................... 59
Figure 4.26 Time response of vertical displacement at Pc with step input - Passive vs PID vs FLC ..................................................................................................................... 60
Figure 4.27 Time response of pitch angle with step input - Passive vs PID vs FLC...... 60
Figure 4.28 Time response of roll angle with step input - Passive vs PID vs FLC......... 61
Figure 4.29 Time response of vertical displacement at Pc with torques input - Passive vs PID vs FLC ...................................................................................................................... 61
Figure 4.30 Time response of pitch angle with torques input - Passive vs PID vs FLC 61
Figure 4.31 Time response of roll angle with torques input - Passive vs PID vs FLC ... 62
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Numerical parameters of quarter vehicle model</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>System parameters of half vehicle model</td>
<td>12</td>
</tr>
<tr>
<td>2.3</td>
<td>Numerical Values of Full Vehicle Model</td>
<td>20</td>
</tr>
<tr>
<td>3.1</td>
<td>Effects of PID Controller Terms</td>
<td>30</td>
</tr>
<tr>
<td>3.2</td>
<td>Effects of increasing PID parameters</td>
<td>33</td>
</tr>
<tr>
<td>3.3</td>
<td>Controller Parameters for Ziegler-Nichols frequency response method</td>
<td>34</td>
</tr>
<tr>
<td>3.4</td>
<td>Controller parameters for Ziegler-Nichols step response method</td>
<td>35</td>
</tr>
<tr>
<td>3.5</td>
<td>PID Parameters for the Quarter Control System</td>
<td>37</td>
</tr>
<tr>
<td>3.6</td>
<td>Parameters of PID Controllers of Half Vehicle System</td>
<td>39</td>
</tr>
<tr>
<td>3.7</td>
<td>Parameters of the PID Controllers of Full Vehicle System</td>
<td>41</td>
</tr>
<tr>
<td>4.1</td>
<td>Rule Base of the FLC Model</td>
<td>51</td>
</tr>
</tbody>
</table>
Chapter 1. Introduction

1.1 Vehicle Suspension Systems

By the early 19th century, most British horse carriages were equipped with springs; wooden springs in the case of light one-horse vehicles, and steel springs in larger vehicles. These were made of low-carbon steel and usually took the form of multiple layer leaf springs (Adams 1971).

Automobiles were initially developed as self-propelled versions of horse drawn vehicles. However, horse drawn vehicles had been designed for relatively slow speeds and their suspension was not well suited to the higher speeds permitted by the internal combustion engine (IC). In 1901, Mors of Germany first fitted an automobile with shock absorbers. In 1920, Leyland used torsion bars in a suspension system. In 1922, independent front suspension was pioneered on the Lancia Lambda car and became more common in mass market cars from 1932 (Jain and Asthana 2006). The drawbacks of excess vehicle body vibrations are that: reduced vehicle-frame life, reduced limitation of vehicle speed, negative biological effects on passengers and detrimental consequences to cargo. Modern active suspensions are tried to provide good handling characteristics and to improve riding comfort under harmful vibrations caused by road irregularities.

Very important objectives set for designing a suspension controller system deal with the riding comfort, suspension relative motion, road gripping, and body inclination. The riding comfort is directly associated with the level of vertical vibration sensed by the passengers when travelling on a rough road. The suspension relative motion describes the displacement magnitude between the sprung and unsprung masses. The road gripping is associated with the friction forces between the tire and road surface. The body inclination occurs during sudden maneuvers such as braking and cornering. The difficulties faced in the active suspension design often are: the
performance requirements are usually conflicting and the model used in the control design contains unresolved uncertainties.

As shown in Figure 1.1, there are three types of suspension systems: passive suspension systems, semi-active suspension systems and active suspension systems.

![Figure 1.1 (a) Passive Suspension, (b) Semi-Active Suspension, (c) Active Suspension](image)

The passive suspension systems consist of spring and damper or shock absorber model as illustrated in Figure 1.1(a), which are most commonly used and may be found in most vehicles. The model systems of springs and dampers are assumed to have linear characteristic, whereas most of shock absorbers exhibit nonlinear relationship between the force and velocity. In passive systems, the suspension damping and stiffness are fixed coefficients, hence, have no mechanism for feedback control.

The semi-active suspension systems have the capability of modifying the damping coefficient of the suspension systems dependent on outside demands. It can be achieved by using a mechanical device called variable damper, which is used in parallel with a conventional spring as shown in Figure 1.1(b). It requires some form of measuring devices (sensors) and controller board for tuning the damping coefficient. Many researchers proposed very attractive and efficient semi-active suspension systems featured by electro rheological (ER) fluids or magneto rheological (MR) fluids.

The active suspension systems supplies external forces to increase the passengers comfort, safety and road handling. It consists of actuators, which can be hydraulic, pneumatic, or electro mechanic, and passive suspension system (spring plus damper)
which is secured in parallel with the actuator device as shown in Figure 1.1(c). The active suspension requires sensing devices, control system and an external power source. The sensing devices, which can be accelerometers, force transducers or potentiometers, located at different points of the vehicle to measure the motions of the body. The control system is employed to supply suitable control commands to the actuators. The external power source, which can be hydraulic pump or electrical pump, draws power from the vehicle’s engine to generate appropriate external forces for the suspension systems.

In general, the road handling and safety rules request harsh suspensions, while the passengers comfort feeling require a soft damping. Driver action and vehicle speed effect on safety aspects. However, the perception of rider’s comfort depends on the environment such as road excitation. Therefore, the active suspension system should be used to solve the conflict between riding comfort and road handling requirements. Existing active suspension systems have several drawbacks, such as high energy consumption and complexity. Therefore, by using the active suspension systems in modern vehicles, these problems should be solved.

The active suspension elements demonstrate a nonlinear behavior especially in the case of large deflections. The suspension forces generated by the hydraulic actuators introduce nonlinearities to the system. The dynamic characteristics of suspension components, i.e. dampers and springs, also have nonlinear properties, which change during the vehicle life cycles. The nonlinearity behaviors of the active suspension system make the control design for the suspension system very difficult.

Several models and controllers have been developed in attempts to enhance and improve the riding and handling qualities in modern vehicles. In our study related to the active suspension control system, a quarter, half, and full vehicle models are adopted to design the control system. The quarter vehicle models only deal with vertical motions of the vehicle body and do not take into account the pitch and roll motions. While half vehicle models only deal with the vehicle body vertical motions and the pitch motions, but they do not include the roll motions of the vehicle body. On the other hand, the full vehicle models are used to design control systems that
take into account all the vehicle motions such as vertical motions, pitch motions and roll motions. Due to the complex mathematical relationships in nonlinear active suspension system, most of researches approximate the active suspension systems as linear models when designing the controllers.[1]

1.2 Problems and Research Objectives

As mentioned before, the main objective of designing the controller for a vehicle suspension system is to reduce the discomfort felt by passengers which arises from road roughness and to improve the road handling associated with the pitching and rolling movements and preventing rollover. This necessitates very fast and accurate controller to meet the key control objectives as much as possible. For that, a PID and fuzzy logic controllers for the quarter, half, and full vehicle linear active suspension systems with actuators will be designed in this project. The advantage of this research is that it allows comparing between two different types of controllers and concluding which controller best fit to such systems.

The objectives of this work are described as follow:

i. Deriving the mathematical equations for the quarter, half, and full vehicle linear active suspension systems with actuators.

ii. Deriving the mathematical equations for the quarter, half, and full vehicle nonlinear active suspension systems with actuators.

iii. Comparing the linear and nonlinear models and using the linear ones instead in order to reduce the work complexity.

iv. Studying two different types of controllers which are PID and fuzzy logic controllers.

v. Implementation of the selected controllers for suspension systems using Simulink Matlab©.

vi. Comparing the behavior of the systems after implementing each controller and analyzing the results.

These Topics will be organized and illustrated in the following chapters of this research, and all models and control systems are simulated using the Simulink part of Matlab© Program.
Chapter 2 . Mathematical Modeling

2.1 Basic Considerations

The design and analysis of control systems are based on the mathematical models of physical complex systems. A mathematical model of a dynamic system is described by a set of equations which can be obtained by using physical laws governing the particular systems. For description of the given system, several types of mathematical models can be proposed from the differential equation, the state space equation, or the transfer function and the impulse response depend on its circumstances.

For obtaining a model, a compromise between the simplicity of the model and the accuracy of results of the analysis should be made. Such that, the neglected certain inherent physical properties of the system should not affect the accuracy of the results of the experimental study of the physical system.

2.2 Quarter Vehicle Model

2.2.1 Physical Setup

Designing an automotive suspension system is an interesting and challenging control problem. When the suspension system is designed, a 1/4 model (one of the four wheels) is used to simplify the problem to a 1D multiple spring-damper system. As shown in Figure 2.1, the tire is simulated as spring of constant stiffness $k_2$ and damper with constant damping $b_2$ connected together in parallel. The road profile input $w$ excites the tire and the surrounding mechanical components that are attached to the tire. This model is for an active suspension system where an actuator is included that is able to generate the control force $U$ to control the motion of the body.
2.2.2 Equations of Motion

This system undergoes a one degree of freedom motion which is the vertical one, the dynamic equations of this oscillating motion can be governed using Newton’s 2nd Law of Motion:[3]

\[ \sum F = M \ddot{X} \] (2.1)

Hence the Oscillation motion of Sprung and Unsprung masses can be concluded:

\[ M_1 \ddot{X}_1 = -b_1 (\dot{X}_1 - \dot{X}_2) - k_1 (X_1 - X_2) + u \] (2.2)
\[ M_2 \ddot{X}_2 = b_1 (\dot{X}_1 - \dot{X}_2) + k_1 (X_1 - X_2) + b_2 (\dot{W} - \dot{X}_2) + k_2 (W - X_2) - u \] (2.3)

2.2.3 Implementation of Model using Matlab©

Matlab is a technical computing environment for high-performance numeric computation and visualization. Simulink is a part of Matlab that can be used to simulate any linear or nonlinear dynamic system. The Simulink model can be created by adding a new class of windows called the block diagram windows. In these windows, models are created and edited primarily by mouse-driven commands. To simulate any process model, a new Simulink environment should be opened by clicking on Simulink icon in the Matlab toolbar. The process model is constructed by drag-and-dropping the appropriate blocks from main Simulink Library Browser to
environment windows. Figure 2.2 shows the MATLAB SIMULINK model of the quarter vehicle active suspension system with the road profile input $W$.

![Figure 2.2 Matlab Simulink Model of Quarter Vehicle System](image)

The source block (road profile input) is a signal generator that represents a random input with 0.1 meters amplitude and 0.1 Hertz frequency.

### 2.2.4 Simulation and Results

To confirm that the design of the control system for vehicle model with actuators or control forces is necessary for meeting the control objectives or not, the time responses of the proposed model should be investigated without controller. The parameters of the system are illustrated in Table 2.1. A good automotive suspension system should have satisfactory road holding ability, while still providing comfort when riding over bumps and holes in the road. When the vehicle is experiencing any road disturbance (i.e. pot holes, cracks, and uneven pavement), the vehicle body should not have large oscillations and the oscillations should dissipate quickly. Since the distance $X_1-W$ is very difficult to measure, and the deformation of the tire ($X_2-W$) is negligible, we will use the distance $X_1-X_2$ instead of $X_1-W$ as the output in our problem. Figures 2.3-2.5 show the time outputs response of the quarter vehicle suspension system with controllers assumed to be zero signals for now.
Table 2.1 Numerical parameters of quarter vehicle model [4]

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>Body Mass</td>
<td>2500</td>
<td>Kg</td>
</tr>
<tr>
<td>M2</td>
<td>Suspension Mass</td>
<td>320</td>
<td>Kg</td>
</tr>
<tr>
<td>K1</td>
<td>Spring stiffness constant of suspension system</td>
<td>80000</td>
<td>N/m</td>
</tr>
<tr>
<td>K2</td>
<td>Spring stiffness constant of wheel and tire</td>
<td>500000</td>
<td>N/m</td>
</tr>
<tr>
<td>B1</td>
<td>Damping constant of suspension system</td>
<td>350</td>
<td>N.s/m</td>
</tr>
<tr>
<td>B2</td>
<td>Damping constant of wheel and tire</td>
<td>15020</td>
<td>N.s/m</td>
</tr>
</tbody>
</table>

Figure 2.3 Time response of vertical displacement X1

Figure 2.4 Time response of vertical displacement X2
Remark: All the time response figures are a displacement versus time sketches where the displacement is measured in meters (m) and time in seconds (s).

2.3 Simplified Half Vehicle Model

2.3.1 Physical Setup

A vehicle on a rough terrain, such as the one shown in the Figure 2.6, exhibits bounce, pitch, and roll on top of its rigid body motion. In the half vehicle analysis, we assume that the rolling motion compared to the two other types of oscillatory motions is negligible. Neglecting the rolling motion and mass of tires, and combining the stiffness and damping effects of tire and suspension system into an equivalent damping and stiffness system, a preliminary simplified half model for vehicle’s suspension system is presented in the Figure 2.6.[5]
2.3.2 Formulation

The governing system of differential equations which describe the bounce and pitch motions of the system shown in Figure 2.6 is found using Lagrange’s Equations. The generalized coordinate $x(t)$ and $\theta(t)$ are used to describe the bounce and pitch motions of the body. The kinetic energy is described in Equation 2.4 as:

$$ T = \frac{1}{2}(m\dot{x}^2 + J\dot{\theta}^2) $$

(2.4)

The potential energy is described in Equation 2.5 as:

$$ U = \frac{1}{2}[k_1(y_1 - x + l_1\theta)^2 + k_2(y_2 - x - l_2\theta)^2] $$

(2.5)

Rayleigh’s dissipation function describing viscous dissipation in the dampers is given as below:

$$ Q = \frac{1}{2}[b_1(\dot{y}_1 - \dot{x} + l_1\dot{\theta})^2 + b_2(\dot{y}_2 - \dot{x} - l_2\dot{\theta})^2] $$

(2.6)

The Lagrangian $L = T - U$ evaluated from (2.4) and (2.5), and together with (2.6) substituted in (2.7) and (2.8) allows obtaining the equations of motion.

$$ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{x}} \right) - \frac{\partial L}{\partial x} = \frac{\partial Q}{\partial x} $$

(2.7)

$$ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = \frac{\partial Q}{\partial \theta} $$

(2.8)

The application of Equations 2.7 and 2.8 yields:

$$ M\ddot{x} + (b_1 + b_2)\dot{x} + (b_2l_2 - b_1l_1)\dot{\theta} + (k_1 + k_2)x + (k_2l_2 - k_1l_1)\theta - k_1y_1 - k_2y_2 - b_1\dot{y}_1 - b_2\dot{y}_2 - u_1 - u_2 = 0 $$

(2.9)

$$ J\ddot{\theta} + (b_2l_2 - b_1l_1)\dot{x} + (b_1l_2^2 + b_2l_2^2)\dot{\theta} + (k_2l_2 - k_1l_1)x + (k_1l_1^2 + k_2l_2^2)\theta + k_1l_1y_1 - k_2l_2y_2 + b_1l_1\dot{y}_1 - b_2l_2\dot{y}_2 + u_1l_1 - u_2l_2 = 0 $$

(2.10)
2.3.3 Implementation of Linear and Nonlinear Models into Simulink

In fact the suspension model has much nonlinearity in reality and simulating a nonlinear system would give much closer view to how the real system behaves. However it is more complicated to design a controller for a nonlinear system, so in order to simplify the system as much as we can, the two systems (linear and nonlinear) were simulated and a comparison was done to the behavior of both systems while excited by the same road input. The road profile was taken as the worst condition of road inputs expressed by a signal generator which governs a random input of amplitude 0.1 meters and 0.1 Hertz frequency at $y_1$, and the same but with transport delay of 0.5 seconds at $y_2$.

Figure 2.7 Simulink Half Vehicle suspensions Linear Model

For the nonlinear model, the equations of motion will change by the way that $\theta$ will be replaced by $\sin \theta$ and $\dot{\theta}$ by $\dot{\theta} \cos \theta$, and the model will be derived from the new equations of motion. The differences between the two models can be configured by comparing figures 2.7 and 2.8:
2.3.4 Simulations and Results

The following simulations are carried out using the Simulink part of Matlab program after implementing the system parameters described in the table below:

Table 2.2 System parameters of half vehicle model

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>Body Mass</td>
<td>2000</td>
<td>Kg</td>
</tr>
<tr>
<td>J</td>
<td>Body moment of inertia around lateral axis</td>
<td>2460</td>
<td>Kg.m²</td>
</tr>
<tr>
<td>k1</td>
<td>Rear spring equivalent stiffness constant</td>
<td>80000</td>
<td>N/m</td>
</tr>
<tr>
<td>k2</td>
<td>Front spring equivalent stiffness constant</td>
<td>500000</td>
<td>N/m</td>
</tr>
<tr>
<td>b1</td>
<td>Rear damping equivalent constant</td>
<td>350</td>
<td>N.s/m</td>
</tr>
<tr>
<td>b2</td>
<td>Front damping equivalent constant</td>
<td>15020</td>
<td>N.s/m</td>
</tr>
<tr>
<td>l1</td>
<td>Longitudinal distance from the rear to the center of gravity of the body</td>
<td>1</td>
<td>m</td>
</tr>
<tr>
<td>l2</td>
<td>Longitudinal distance from the front to the center of gravity of the body</td>
<td>1.8</td>
<td>m</td>
</tr>
<tr>
<td>P1,P2,Pc</td>
<td>Respective rear, front, and center of gravity points of the body plane</td>
<td></td>
<td>/</td>
</tr>
</tbody>
</table>

Figure 2.8 Simulink Half Vehicle suspensions Nonlinear Model
Figure 2.9 Time response of X displacement of linear and nonlinear systems

Figure 2.10 Time response of X' vertical velocity of linear and nonlinear systems

Figure 2.11 Time response of pitch angle θ of linear and nonlinear systems
We can observe from the previous figures that both curves are about to be confounded which means that the two systems (linear and nonlinear) are approximately identical in their output responses with respect to time and under random and severe road profile inputs. For this, the linear system will be considered when implementing the controllers to the system in the following chapters of the research in order to reduce the complexity of the control design and make it simpler without being far from the system’s behavior in reality.

2.4 Simplified Linear Full Vehicle Model

2.4.1 Physical Setup and Formulation

To investigate the problem of balancing riding comfort and road handling, the mathematical model of four-wheel linear active suspensions system with actuators should be introduced. As shown in Figure 2.12, the tires with the suspensions system are simulated as springs with a constant stiffness $k_i$ (i = 1, 2, 3 and 4) and dampers with a constant damping $b_i$ connected in parallel with a controller $u_i$. The road profile inputs $y_i$ excite the tires and the surrounding body that is attached to these tires. The mass is considered to be the total mass of the body vehicle with its suspension components and is labeled as $M$. The control forces are applied between using actuators to minimize: the vertical motion $(X_c)$ sensed by passengers when travelling on a rough road; and vehicle body motions that are made during sharp maneuvers such as roll angle $\alpha$ and pitch angle $\vartheta$.

The plane equation will be used to derive the full model of nonlinear active suspension system which can be written as:

$$\begin{vmatrix}
 x - x_1 & y - y_1 & z - z_1 \\
 x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\
 x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\
\end{vmatrix} = 0$$

(2.11)

Any point at the body plane satisfies the above plane equation. The points P1, P2, P3, P4, and Pc are located on the body plane, so that these points satisfy the plane equation.
The coordinates of the points P1, P2, P3, P4 and Pc can be given as \((0,0,x_1)\), 
\((0,2l,x_2)\), \((L,0,x_3)\), \((L,2l,x_4)\), and \((l_1,l,x_c)\) respectively. Substituting the 
coordinates of these points in Eq. (2.11) yields:

\[
x = \frac{(x_3-x_1)}{L}y + \frac{(x_2-x_1)}{2l}z + x_1
\]  
(2.12)

The vertical displacement at the center of gravity \(x_c\) and at point P4 \(x_4\) can be 
obtained by substituting the coordinates of P3 and Pc in Eq. (2.12)

\[
x_4 = -x_1 + x_2 + x_3
\]  
(2.13)

\[
x_c = \left(0.5 - \frac{l_1}{L}\right)x_1 + 0.5x_2 + \frac{l_1}{L}x_3
\]  
(2.14)

The roll and pitch angles can be obtained from the following equations:

\[
\alpha = \frac{\partial x}{\partial z} = \frac{x_2-x_1}{2l}
\]  
(2.15)

\[
\theta = \frac{\partial x}{\partial y} = \frac{x_3-x_1}{L}
\]  
(2.16)

From equations (2.13-2.16) the vertical displacements \(x_1\), \(x_2\), \(x_3\) and \(x_4\) can be 
obtained with respect to \(x_c\), \(\alpha\), and \(\theta\) as follows:

\[
x_1 = x_c - l\alpha - l_1\theta
\]  
(2.17)
Chapter 2. Mathematical Modeling

\[ x_2 = x_c + l\alpha - l_1\theta \]  \hspace{1cm} (2.18)

\[ x_3 = x_c - l\alpha + l_2\theta \]  \hspace{1cm} (2.19)

\[ x_4 = x_c + l\alpha + l_2\theta \]  \hspace{1cm} (2.20)

The differential equations of the full vehicle nonlinear active suspension system can be obtained from the Newton’s law of motion as follows:

- **Vertical motion**

\[ M\ddot{x}_c = -\sum_{i=1}^{4} F_{ki} - \sum_{i=1}^{4} F_{bi} + \sum_{i=1}^{4} F_i \]  \hspace{1cm} (2.21)

Where \( F_{ki} \) and \( F_{bi} \) are the linear forces of springs and dampers, respectively, which can be written as:

\[ F_{ki} = k_i(x_i - y_i) \]  \hspace{1cm} (2.22)

\[ F_{bi} = b_i(\dot{x}_i - \dot{y}_i) \]  \hspace{1cm} (2.23)

Where \( y_i \) is the road profile input on the \( i \)th tire and \( F_i \) is the controller output force expressed in Figure 2.12 as \( u_i \). Hence the equation of vertical oscillation of the vehicle’s body is derived as:

\[ M\ddot{x}_c + \dot{x}_c(b_1 + b_2 + b_3 + b_4) + \dot{\theta}(l_2(b_3 + b_4) - l_1(b_1 + b_2)) + \dot{\alpha}(b_2 + b_4 - b_1 - b_3) + x_c (k_1 + k_2 + k_3 + k_4) + \theta(l_2(k_3 + k_4) - l_1(k_1 + k_2)) + \alpha(l_2 + k_4 - k_1 - k_3) - k_1y_1 - k_2y_2 - k_3y_3 - k_4y_4 - b_1\dot{y}_1 - b_2\dot{y}_2 - b_3\dot{y}_3 - b_4\dot{y}_4 - u_1 - u_2 - u_3 - u_4 = 0 \]  \hspace{1cm} (2.24)

- **Rolling motion**

\[ J_y\ddot{\theta} = (F_{k1} - F_{k2} + F_{k3} - F_{k4})l + (F_{b1} - F_{b2} + F_{b3} - F_{b4})l + (F_2 + F_4 - F_1 - F_3)l + T_c \]  \hspace{1cm} (2.25)

Where \( J_y \) is the roll moment of inertia about \( y \)-axis is, \( l \) is half the distance between the front wheels (or rear wheels) and \( T_c \) is the cornering torque. The equation of rolling motion can be rewritten as:

\[ J_y\ddot{\theta} + \dot{x}_c (b_2 + b_4 - b_1 - b_3) + \dot{\theta}(b_1l_1 + b_4l_2 - b_2l_1 - b_3l_2) + \dot{\alpha}l^2(b_1 + b_2 + b_3 + b_4) + x_c (k_2 + k_4 - k_1 - k_3) + \theta(l_2k_1 + k_4l_2 - k_2l_1 - k_3l_2) + \]
\[ \alpha l^2 (k_1 + k_2 + k_3 + k_4) + l(k_1 y_1 - k_2 y_2 + k_3 y_3 - k_4 y_4 + b_1 \dot{y}_1 - b_2 \dot{y}_2 + b_3 \dot{y}_3 - b_4 \dot{y}_4 + u_1 - u_2 + u_3 - u_4) - T_c = 0 \]  
(2.26)

- **Pitch motion**

\[ j_z \ddot{\theta} = (F_{k_3} + F_{k_4}) k_2 - (F_{k_1} + F_{k_2}) l_1 + (F_{b_3} + F_{b_4}) l_2 - (F_{b_1} + F_{b_2}) l_1 - (F_1 + F_2) l_1 + (F_3 + F_4) l_2 + T_b \]  
(2.27)

Where \( j_z \) is the pitch moment of inertia about z-axis; \( l_1 \) is the distance between the center of front wheel axle and center of gravity of the vehicle; \( l_2 \) is the distance between the center of gravity of the vehicle and the center of rear wheel axle and \( T_b \) is the braking torque. The equation of pitch motion can be rewritten as:

\[ j_z \ddot{\theta} + \dot{x}_c[l_2(b_3 + b_4) - l_1(b_1 + b_2)] + \dot{\theta}[l_2^2(b_1 + b_2) + l_2^2(b_3 + b_4)] + \dot{a}l(b_1 l_1 + b_4 l_2 - b_2 l_1 - b_3 l_2) + x_c[l_2(k_3 + k_4) - l_1(k_1 + k_2)] + \theta[l_1^2(k_1 + k_2) + l_2^2(k_3 + k_4)] + a[l(k_1 l_1 + k_4 l_2 - k_2 l_1 - k_3 l_2) + k_1 l_1 y_1 + k_2 l_1 y_2 - k_3 l_2 y_3 - k_4 l_2 y_4 + b_1 l_1 \dot{y}_1 + b_2 l_1 \dot{y}_2 - b_3 l_2 \dot{y}_3 - b_4 l_2 \dot{y}_4 + l_1(u_1 + u_2) - l_2(u_3 + u_4) - T_b = 0 \]  
(2.28)

### 2.4.2 Implementation of the Process Model Using Matlab

Figure 2.13 shows the Matlab Simulink model of the full vehicle linear active suspensions system with the road exciting profile inputs \( y_i \), bending torque input \( T_c \) and braking torque input \( T_b \).

![Figure 2.13 Full Vehicle Linear Simulink Model](image-url)
The source block is the signal generator that represents the random input with 0.1m amplitude and 0.1Hz frequency. To simulate the time delay of road exciting inputs for each tire, the time delay blocks are added. In this simulation, it is assumed that the time delay for the road exciting inputs $y_1$, $y_2$, $y_3$, and $y_4$ are 0, 0.3, 0.5 and 0.8 seconds respectively. To simulate the bending torque input, two unit step inputs have been used. The amplitude of the first unit step input is 5500 N/m and it is applied at 0 s. While, the amplitude of the second unit step input is -5500N/m and it is applied at 10 s. By adding these two signals, the amplitude of the equivalent signal is 5500 N/m and its effect is applied from 0 s to 10 s. The same technique is followed to simulate the braking torque, but the effect of the second unit step signal is applied at 5s. Therefore, the amplitude of equivalent signal, which is applied to represent the braking torque, is 5500 N/m from 0 s to 5 s. The output signals $x_1$, $x_2$, $x_3$, $x_4$, $\alpha$, and $\theta$ are sent to the workspace of Matlab program to save their data and these data will be used to plot the output responses of the proposed system. The construction of proposed model is very huge so the following figures show the under masks of the general model presented in Figure 2.13.

![Figure 2.14 Under Mask of the Signal Outputs Dotted Rectangle of Figure 2.13]
2.4.3 Simulations and Results

The Matlab-Simulink program has been used to simulate the full vehicle linear active suspension model. The open-loop responses of full vehicle suspensions system, when the random inputs are applied as road excitation, are investigated. The plots of the open-loop responses will show if the control objectives can be met without using a control system for proposed model or not. The control objectives in this case are: minimizing the vibration sensed by the passengers when travelling on the rough roads and avoidance of the rollover of the vehicle when critical maneuvers occur such as braking and cornering. Figures 2.16-2.24 illustrate the time outputs response of full vehicle suspensions system. The road profiles are assumed to be random input. Those Figures depict that the vertical displacement at each corner are unacceptable which means that the passive suspension systems are incapable of absorbing the vibrations excited by the road unevenness.
Table 2.3 Numerical Values of Full Vehicle Model

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>Body Mass</td>
<td>2000</td>
<td>Kg</td>
</tr>
<tr>
<td>J_y</td>
<td>Body moment of inertia around longitudinal axis</td>
<td>460</td>
<td>Kg.m²</td>
</tr>
<tr>
<td>J_z</td>
<td>Body moment of inertia around lateral axis</td>
<td>2460</td>
<td>Kg.m²</td>
</tr>
<tr>
<td>k1, k2</td>
<td>Left, Right rear spring equivalent stiffness constants</td>
<td>195000</td>
<td>N/m</td>
</tr>
<tr>
<td>K3, k4</td>
<td>Left, Right front spring equivalent stiffness constants</td>
<td>193000</td>
<td>N/m</td>
</tr>
<tr>
<td>b1, b2</td>
<td>Left, Right rear damping equivalent constants</td>
<td>1300</td>
<td>N.s/m</td>
</tr>
<tr>
<td>b3, b4</td>
<td>Left, Right front damping equivalent constants</td>
<td>1635</td>
<td>N.s/m</td>
</tr>
<tr>
<td>l1</td>
<td>Longitudinal distance from the rear to the center of gravity of the body</td>
<td>1</td>
<td>m</td>
</tr>
<tr>
<td>l2</td>
<td>Longitudinal distance from the front to the center of gravity of the body</td>
<td>1.8</td>
<td>m</td>
</tr>
<tr>
<td>l</td>
<td>Half the width of the track (vehicle)</td>
<td>0.75</td>
<td>m</td>
</tr>
</tbody>
</table>

By implementing these parameters in the Simulink environment and running the simulation, we get the following results:

Figure 2.16 Time response of the vertical displacement at P1
Chapter 2. Mathematical Modeling

Figure 2.17 Time response of the vertical displacement at P2

Figure 2.18 Time response of the vertical displacement at P3

Figure 2.19 Time response of the vertical displacement at P4
Figure 2.20 Time response of the vertical displacement at Pc

Figure 2.21 Time response of pitch angle

Figure 2.22 Time response of roll angle
To simulate the cornering and braking of the vehicle, a bending and braking torque have been applied as shown in Figures 2.23-2.24. The road profiles have been assumed to be random input. Figures 2.25-2.31 demonstrate the time output responses of the full vehicle suspension system. Those figures show that when the bending torque is applied, the road handling is decreased, the discomfort felt by the passengers is increased, and the contact forces between the tires and the road surface are decreased.
Figure 2.26 Time response of a vertical displacement at P2

Figure 2.27 Time response of a vertical displacement at P3

Figure 2.28 Time response of a vertical displacement at P4
Chapter 2. Mathematical Modeling

Figure 2.29 Time response of a vertical displacement at Pc

Figure 2.30 Time response of pitch angle

Figure 2.31 Time response of roll angle
2.5 Summary

Regarding the historical review, the researchers designed control systems for quarter, half or full model of suspension systems. Some researchers suggested control systems for linear suspension models, while the other suggested control systems for nonlinear suspension models. Without doubt, the nonlinearity exists in the suspension system. In this work, full vehicle linear active suspension model has been used to take into account the three movements: vertical movement, pitching movement and rolling movement. The mathematical model equations for quarter, half, and full vehicle linear active suspension systems have been derived in this chapter. The results in the previous sections show that the vibrations sensed by the passengers are sensible. When the inputs to the vehicle system are just the road random profile inputs, both the riding comfort and road handling are decreased. The amplitude of the output transient responses should be small values. Furthermore, when the bending torque is applied to simulate the cornering (or when the braking torque is applied to simulate braking) with random inputs as road profile, the output transient responses oscillate around a specific value. The desired output responses should be over damping responses with very small steady state value. Therefore, the passive suspension systems without control element are not suitable to eliminate the vibrations which arise from travelling on the rough road or sharp maneuvers such as pitching and rolling movements. In following chapters, control systems will be designed to achieve the design requirements.
Chapter 3 . PID Controller: Theory and Application

3.1 Control System

[6] A control system is an arrangement of physical components connected or related in such a manner as to command, direct, or regulate itself or another system. Control system has two important terms, which are defined as input and output. The input is the stimulus, excitation, or command applied to a control system, usually from an external energy source in order to produce a specified response from the control system. The output is the actual response obtained from a control system. It may or may not be equal to specified response implied by the input.

![Figure 3.1 A Block Diagram of General Control System](image)

3.2 Open-Loop and Closed-Loop Control Systems

Control systems are classified into two general categories, open-loop and closed-loop systems. An open-loop system is one which the control action is independent of the output. A closed-loop control system is one in which the control action is somehow dependent on the output.
Open-loop System is a control system that does not use feedback. The controller sends a measured signal to the actuator, which specifies the desired action. This type of systems is not self-correcting. If some external disturbance changes the load on machine or process being performed, some degree of physical effort of human operator is required to make necessary modifications. The system is manually controlled by human.

![Figure 3.2 Open-Loop System](image)

To avoid the problems of the open-loop controller, the feedback was added. A closed-loop controller uses feedback to control outputs of a dynamical system. Process input has an effect on the process outputs, which is measured with sensors and processed by the controller; the result (the control signal) is used as input to the process, closing the loop. Since the controller knows what the system is actually doing, it can make any necessary adjustments to keep the output where it belongs.

![Figure 3.3 Closed-Loop System](image)

Closed-loop controllers have the following advantages over open-loop controllers:

- Disturbance rejection (such as unmeasured friction in a motor)
- Guaranteed performance even with model uncertainties, when the model structure does not match perfectly the real process and the model parameters are not exact
- Unstable processes can be stabilized
- Reduced sensitivity to parameter variations
- Improved reference tracking performance
3.3 PID Controllers

A proportional-integral-derivative controller (PID controller) is a generic control loop feedback mechanism widely used in industrial control systems. A PID controller attempts to correct the error between a measured process variable and a desired set-point by calculating and then outputting a corrective action that can adjust the process accordingly.

The PID controller calculation (algorithm) involves three separate parameters: the Proportional, the Integral, and the Derivative values. The Proportional value determines the reaction to the current error, the Integral determines the reaction based on the sum of recent errors, and the Derivative determines the reaction to the rate at which the error has been changing. The weighted sum of these three actions is used to adjust the process via a control element such as the position of a control valve or the power supply of a heating element.

![Figure 3.4 The PID Controllers Block Diagram](image)

By “tuning” the three constants in the PID controller algorithm the PID can provide control action designed for specific process requirements. The response of the controller can be described in terms of the responsiveness of the controller to error, the degree to which the controller overshoots, the set-point, and the degree of system oscillation. Note that the use of the PID algorithm for control does not guarantee the optimal control of the system.
Some applications may require using only one or two modes to provide the appropriate system control. This is achieved by setting the gain of undesired control outputs to zero. A PID controller will be called a PI, PD, P, or I controller in the absence of the respective control actions. PI controllers are particularly common, since derivative action is very sensitive to measurement noise, and the absence of an integral value prevents the system from reaching its target value due to the control action.

The table below summarizes the PID terms and their effect on a control system:

Table 3.1 Effects of PID Controller Terms

<table>
<thead>
<tr>
<th>Term</th>
<th>Math Function</th>
<th>Effect on Control System</th>
</tr>
</thead>
<tbody>
<tr>
<td>P Proportional</td>
<td>$K_P \times V_{error}$</td>
<td>Typically the main drive in a control loop, $K_P$ reduces a large part of the overall error.</td>
</tr>
<tr>
<td>I Integral</td>
<td>$K_I \times \int V_{error} , dt$</td>
<td>Reduces the final error in a system. Summing even a small error over time produces a drive signal large enough to move the system toward a smaller error.</td>
</tr>
<tr>
<td>D Derivative</td>
<td>$K_D \times \frac{dV_{error}}{dt}$</td>
<td>Counteracts the $K_P$ and $K_I$ terms when the output changes quickly. This helps reduce overshoot and ringing. It has no effect on final error.</td>
</tr>
</tbody>
</table>

### 3.3.1 Proportional Controller

The proportional term makes a change to the output that is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a constant $K_P$, called the proportional gain. The proportional term is given by:

$$P_{out} = K_P \cdot e(t)$$  \hspace{1cm} (3.1)

Where:
• $P_{out}$: Proportional output
• $K_p$: Proportional Gain, a tuning parameter
• $e$: Error
• $t$: Time or instantaneous time (the present)

A high proportional gain results in a large change in the output for a given change in the error. If the proportional gain is too high, the system can become unstable. In contrast, a small gain results in a small output response to a large input error, and a less responsive (or sensitive) controller. If the proportional gain is too low, the control action may be too small when responding to system disturbances.

In the absence of disturbances pure proportional control will not settle at its target value, but will retain a steady state error that is a function of the proportional gain and the process gain. Despite the steady-state offset, both tuning theory and industrial practice indicate that it is the proportional term that should contribute the bulk of the output change.

### 3.3.2 Integral Controller

The contribution from the integral term is proportional to both the magnitude of the error and the duration of the error. Summing the instantaneous error over time (integrating the error) gives the accumulated offset that should have been corrected previously. The accumulated error is then multiplied by the integral gain and added to the controller output. The magnitude of the contribution of the integral term to the overall control action is determined by the integral gain, $K_I$. The integral term is given by:

$$I_{out} = K_I \int_0^t e(\tau)d\tau \quad (3.2)$$

Where:

• $I_{out}$: Integral output
• $K_I$: Integral Gain, a tuning parameter
• $e$: Error
• $\tau$: Time in the past contributing to the integral response
The integral term (when added to the proportional term) accelerates the movement of the process towards set-point and eliminates the residual steady-state error that occurs with a proportional only controller. However, since the integral term is responding to accumulated errors from the past, it can cause the present value to overshoot the set-point value (cross over the set-point and then create a deviation in the other direction).

### 3.3.3 Derivative Controller

The rate of change of the process error is calculated by determining the slope of the error over time (i.e. its first derivative with respect to time) and multiplying this rate of change by the derivative gain $K_D$. The magnitude of the contribution of the derivative term to the overall control action is determined by the derivative gain $K_D$. The derivative term is given by:

$$D_{out} = K_D \frac{de}{dt}$$  \hspace{1cm} (3.3)

Where:

- $D_{out}$: Derivative output
- $K_D$: Derivative Gain, a tuning parameter
- $e$: Error
- $t$: Time or instantaneous time (the present)

The derivative term slows the rate of change of the controller output and this effect is most noticeable close to the controller set-point. Hence, derivative control is used to reduce the magnitude of the overshoot produced by the integral component and improve the combined controller-process stability. However, differentiation of a signal amplifies noise in the signal and thus this term in the controller is highly sensitive to noise in the error term, and can cause a process to become unstable if the noise and the derivative gain are sufficiently large.

The output from the three terms, the proportional, the integral, and the derivative terms are summed to calculate the output of the PID controller as shown previously in Figure 3.4.
3.3.4 Characteristics of P, I, and D Controllers

The effects of each of controller parameters $K_P$, $K_I$, and $K_D$ on a closed-loop system are summarized in the table below. Note that these correlations may not be exactly accurate, because $K_P$, $K_I$, and $K_D$ are dependent on each other. In fact, changing one of these variables can change the effect of the other two. For this reason, the table should only be used as a reference for determining the values for $K_P$, $K_I$, and $K_D$.[7]

<table>
<thead>
<tr>
<th>Type</th>
<th>Rise Time</th>
<th>Overshoot</th>
<th>Settling Time</th>
<th>S-S Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_P$</td>
<td>Decrease</td>
<td>Increase</td>
<td>Small Change</td>
<td>Decrease</td>
</tr>
<tr>
<td>$K_I$</td>
<td>Decrease</td>
<td>Increase</td>
<td>Increase</td>
<td>Eliminate</td>
</tr>
<tr>
<td>$K_D$</td>
<td>Small Change</td>
<td>Decrease</td>
<td>Decrease</td>
<td>Small Change</td>
</tr>
</tbody>
</table>

3.3.5 PID Tuning

Users of control systems are frequently faced with the task of adjusting the controller parameters to obtain a desired behavior. There are many different ways to do this. One way to do this is to go through the steps of modeling and control design. Since the PID controller has so few parameters a number of special empirical methods have also been developed. A simple idea is to connect a controller, increase the gain until the system starts to oscillate, and then reduce the gains by an appropriate factor. Another is to measure some features of the open loop response and to determine controller parameters based on these features. In fact, the Ziegler-Nichols methods are the most celebrated tuning rules.

3.3.5.1 Ziegler-Nichols’ Tuning

Ziegler and Nichols developed two techniques for controller tuning in the 1940s. The idea was to tune the controller based on the following idea: Make a simple experiment, extract some features of process dynamics from the experimental data, and determine controller parameters from the features. One method is based on direct adjustment of the controller parameters. A controller is connected to the
process, integral and derivative gains are set to zero and the proportional gain is increased until the system starts to oscillate. The critical value of the proportional gain $K_c$ is observed together with the period of oscillation $T_c$. The controller parameters are then given by Table 3.3. The values in the table were obtained based on many simulations and experiments on processes that are normally encountered in process industry. There are many variations of the method which are widely used in industry.

<table>
<thead>
<tr>
<th>Controller</th>
<th>$k_p/k_c$</th>
<th>$T_i/T_c$</th>
<th>$T_d/T_c$</th>
<th>$T_p/T_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PI</td>
<td>0.4</td>
<td>0.8</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>PID</td>
<td>0.6</td>
<td>0.5</td>
<td>0.125</td>
<td>0.85</td>
</tr>
</tbody>
</table>

Remark: the parameters are given in terms of critical gain $k_c$ and critical period $T_c$. Parameter $T_p$ is an estimate of the period of damped oscillations of the closed loop system.

Another method proposed by Ziegler and Nichols is based on determination of the open loop step response of the process, as shown Figure 3.5(a). The step response is measured by applying a step input to the process and recording the response. The response is scaled to correspond to a unit step input and characterized by parameters $a$ and $T_{det}$, which are the intercepts of the steepest tangent of the step response with the coordinate axes. The parameter $T_{det}$ is an approximation of the time delay of the system and $a/T_{det}$ is the steepest slope of the step response. Notice that it is not necessary to wait until steady state to find the parameters, it suffices to wait until the response has had an inflection point. The controller parameters are given in Table 3.4. The parameters were obtained by extensive simulation of a range of representative processes.

3.3.5.2 Improved Ziegler-Nichols Rules

There are two drawbacks with the Ziegler-Nichols rules: too little process information is used and the closed loop systems that are obtained lack robustness. Substantially better tuning is obtained by fitting the model to the step response.
\[ P(s) = \frac{K}{1 + sT e^{-sT_{\text{del}}}} \]  

(3.4)

A simple way to do this is illustrated in Figure 3.5(b). The zero frequency gain of the process \( K \) is determined from the steady state value of the step response. The time delay \( T_{\text{del}} \) is determined from the intercept of the steepest tangent to the step response and the time \( T_{63} \) is the time where the output has reached 63% of its steady state value. The parameter \( T \) is then given by \( T = T_{63} - T_{\text{del}} \).

![Figure 3.5 Characterization of the unit step response by two (a) and three (b) parameters](image)

Remark: the point where the tangent is steepest is marked with a small circle.

Notice that the experiment takes longer time than the experiment in Figure 3.5(a) because it is necessary to wait until the steady state has been reached. The following tuning formulas have been obtained by tuning controllers to a large set of processes typically encountered in process control.

\[ k_p K = \min \left( 0.4 \frac{T}{L}, 0.25 \right) \]  

(3.5)

\[ T_i = \max (T, 0.5 T_{\text{del}}) \]  

(3.6)

Notice that the improved formulas typically give lower controller gain than the Ziegler-Nichols method, and that integral gain is higher, particularly for systems with dynamics that are delay dominated, i.e. \( T_{\text{del}} > 2T \).[8]

### Table 3.4 Controller parameters for Ziegler-Nichols step response method

<table>
<thead>
<tr>
<th>Controller</th>
<th>( ak_p )</th>
<th>( \frac{T_i}{T_{\text{del}}} )</th>
<th>( \frac{T_d}{T_{\text{del}}} )</th>
<th>( \frac{T_p}{T_{\text{del}}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>1</td>
<td>3</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>PI</td>
<td>0.9</td>
<td></td>
<td></td>
<td>5.7</td>
</tr>
<tr>
<td>PID</td>
<td>1.2</td>
<td>2</td>
<td>( T_{\text{del}} / 2 )</td>
<td>3.4</td>
</tr>
</tbody>
</table>
3.3.5.3 Automatic Simulink PID Tuning

You can use the Simulink Control Design PID Tuner to tune PID gains automatically in a Simulink model containing a PID Controller block. The PID Tuner allows you to achieve a good balance between performance and robustness for PID controllers. The PID Tuner:

- Automatically computes a linear model of the plant in your model. The PID Tuner considers the plant to be the combination of all blocks between the PID controller output and input. Thus, the plant includes all blocks in the control loop, other than the controller itself.
- Automatically computes an initial PID design with a good trade-off between performance and robustness. The PID Tuner bases the initial design upon the open-loop frequency response of the linearized plant.
- Provides the PID Tuner graphical user interface (GUI) to help you interactively refine the performance of the PID controller to meet your design requirements.

3.4 Implementation of the PID Controller, Simulations, and Results

3.4.1 Quarter Vehicle System

The following figure shows how the PID controller is connected in the quarter vehicle system:

![Figure 3.6 PID Controlled Quarter Vehicle System](image-url)
The Simulink made it easy on its users to tune the PID controller by the **Tune** option found in the function block of the controller which automatically finds the controller’s parameters. These parameters are presented in the following table:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>$K_P$</th>
<th>$K_I$</th>
<th>$K_D$</th>
<th>$K_N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>9420540</td>
<td>31494230</td>
<td>111358.7</td>
<td>846.1</td>
</tr>
</tbody>
</table>

Note that $K_N$ is the filter coefficient of the PID controller. After tuning the controller, the system becomes ready for experiment. Running the simulation with the same road profile input assumed in the passive system gives the following result:
3.4.2 Half Vehicle System

The implementation of the PID controller is shown below:

![PID Controlled Half Vehicle System](image)

The vertical displacements $X_1$ and $X_2$ at points P1 and P2 respectively are taken as feedbacks to the controller block subsystem, where they are derived, and the derivative is summed with a zero constant block creating by far the controller input. Figure 3.10 illustrates control method in the controller block:

![Undermask of the Controller Block of Figure 3.9](image)

In real, the half vehicle system took further time to find the optimal PID controller parameters than the quarter one. Tuning the controller many consecutive times has led to the following parameters:
Table 3.6 Parameters of PID Controllers of Half Vehicle System

<table>
<thead>
<tr>
<th></th>
<th>$K_P$</th>
<th>$K_I$</th>
<th>$K_D$</th>
<th>$K_N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID 1</td>
<td>-610694.5</td>
<td>-645681.7</td>
<td>-66258.4</td>
<td>759.6</td>
</tr>
<tr>
<td>PID 2</td>
<td>21369308.7</td>
<td>70770869.4</td>
<td>536426.8</td>
<td>884.8</td>
</tr>
</tbody>
</table>

Figures 3.11-3.14 show the simulation results and their comparison with those obtained from the passive system under the same road excitation which is assumed to be a random input of 0.1 meters amplitude and 0.1 hertz frequency.

![Figure 3.11 Time response of vertical displacement X](image1)

![Figure 3.12 Time response of the pitch angle](image2)
3.4.3 Full Vehicle System

The implementation of the PID controller in the full vehicle model is similar to the half one. However, the full model consists of four suspensions systems which means it needs four PID controllers each controlling one corner of the vehicle’s four corners. Each PID controller takes the derivative of the vertical displacement at the adjacent corner subtracted from a zero constant source as input, and delivers the necessary control to the actuator installed at that corner. To make the system we can just add a controller block to the full model presented in Figure 2.13. The components of this block are shown in the following figure:
Similarly, the PID controllers are tuned using Matlab Simulink and the parameters of the controllers are hence obtained.

<table>
<thead>
<tr>
<th>PID</th>
<th>$K_P$</th>
<th>$K_I$</th>
<th>$K_D$</th>
<th>$K_N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID 1</td>
<td>-6102978.2</td>
<td>-34282792.6</td>
<td>-68811.8</td>
<td>516.3</td>
</tr>
<tr>
<td>PID 2</td>
<td>-5292711.2</td>
<td>-68549270.9</td>
<td>-89696.2</td>
<td>449.2</td>
</tr>
<tr>
<td>PID 3</td>
<td>-7904033.9</td>
<td>-154106131.5</td>
<td>-83189.3</td>
<td>398.1</td>
</tr>
<tr>
<td>PID 4</td>
<td>-5943828.6</td>
<td>-331924279.3</td>
<td>-26442.4</td>
<td>18355.5</td>
</tr>
</tbody>
</table>

The following figures show the simulations results of the full system then compared to the passive full system under the same conditions:
Figure 3.18 Time response of displacement at P3

Figure 3.19 Time response of displacement at P4

Figure 3.20 Time response of vertical displacement at Pc

Figure 3.21 Time response of the pitch angle
**Figure 3.22** Time response of the roll angle

**Figure 3.23** Displacement at P1 - Passive vs PID

**Figure 3.24** Displacement at P2 - Passive vs PID

**Figure 3.25** Displacement at P3 - Passive vs PID

**Figure 3.26** Displacement at P4 - Passive vs PID
Figure 3.27 Vertical displacement at Pc - Passive vs PID

Figure 3.28 Pitch angle - Passive vs PID

Figure 3.29 Roll Angle - Passive vs PID
Chapter 4 . Fuzzy Logic Controller: Theory and Application

A Fuzzy logic system, which was proposed by Lotfy Zadeh in 1965, emerged as a tool to deal with uncertain, imprecise, or qualitative decision-making problems. It is a static nonlinear mapping between its inputs and outputs. The following subsections show the general concept of the fuzzy logic system.[9]

4.1 General Definitions

To deal with fuzzy logic system, the following definitions should be introduced (Passino 1998):

- **Definition 4.1 “Linguistic Variables”**
  It is a constant symbolic description that is used to specify the rules of rule based fuzzy system inputs and outputs. If the fuzzy system has two inputs (\(u_1\) and \(u_2\)) and one output (\(y\)), the following linguistic variables (for example) “\(\tilde{u}_1 = \) position error, \(\tilde{u}_2 = \) velocity error and \(\tilde{y} = \) control signal” can be used to describe \(u_1, u_2\) and \(y\) respectively.

- **Definition 4.2 “Linguistic Values”**
  Linguistic values are the symbolic domain of the linguistic variables. They are used to describe characteristics of the variables. In other words, the linguistic variable (\(\tilde{u}_i\) or \(\tilde{y}_i\)) takes on the elements from the set of linguistic values denoted by \(\tilde{H}_i = \{\tilde{H}_i^j: j = 1,2, ..., N_i\}\), where \(N_i\) is the total number of the linguistic values for each of linguistic variable. For example, if \(\tilde{u}_i\) denotes the linguistic variable “position error”, then it may assign \(\tilde{H}_i^1 = \) “negative”, \(\tilde{H}_i^2 = \) “zero” and \(\tilde{H}_i^3 = \) “positive” so that \(\tilde{u}_i\) has a value from \(\tilde{H}_i = \{\tilde{H}_i^1, \tilde{H}_i^2, \tilde{H}_i^3\}\).
• **Definition 4.3 “Linguistic Rules”**

The linguistic rules of fuzzy system can be described by using *If-Then* form:

\[
\text{If premise Then consequent}
\]

The inputs of the fuzzy set are associated with the premise (sometime called antecedents) and the outputs are associated with the consequent (sometime called action). It is considered that the MIMO form of rule can be divided into a number of MISO rules using simple rules from logic. Therefore, the general form of the linguistic \( q^{th} \) rule can be described by

\[
R_q: \text{If } \tilde{u}_1 \text{ is } \tilde{A}^{j,q}_1 \text{ and } \tilde{u}_2 \text{ is } \tilde{A}^{k,q}_2 \text{ and } \ldots \text{ and } \tilde{u}_n \text{ is } \tilde{A}^{l,q}_n \text{ Then } \tilde{y}_s \text{ is } \tilde{B}^{r,q}_s
\]

If the fuzzy system has just one output, then the total number of the possible rules which can be written for any problem depends on: the number of fuzzy system inputs and number of linguistic values for each linguistic variable. The total number of the rules in the fuzzy inference system can be calculated as:

\[
N_r = \prod_{i=1}^{n} N_{mi}
\]

where \( N_{mi} \) is the number of linguistic value of \( i^{th} \) linguistic variable and \( n \) is the number of the fuzzy system inputs. Notice that each premise can be composed of the conjunction of several terms (e.g. \( \tilde{u}_1 \) is \( \tilde{A}^{j,q}_1 \) and \( \tilde{u}_2 \) is \( \tilde{A}^{k,q}_2 \)). The “and” operator is called conjunction operator. The conjunction operator can be any other logic operators such as: “or”, “not”, etc.

• **Definition 4.4 “Membership Function”**

It is the function that maps the real numbers (crisps) of fuzzy system inputs to any numerical values between 0 and 1. The symbol \( \mu_{\tilde{A}^{j,q}_i}(u_i) \) is used to describe the membership function associated with linguistic value \( \tilde{A}^{j,q}_i \). There are many shapes can be chosen as membership function such as Sigmoid function, Gaussian function, Trapezoidal function, Triangular function, Bell-shape function...

In fact, a variety of membership functions are available in Matlab program by a way that you just need to adjust a few parameters and constants to a fixed shape function and you will get exactly the desired function shape. These different shapes of the membership functions that can be found in Matlab are given below in Figure 4.1.
• **Definition 4.5 “Fuzzy Sets”**

Given a linguistic variable $\tilde{u}_i$ with a linguistic value $\tilde{A}^i_{j,q}$ defined on the universe of discourse $U_i$ and membership function $\mu_{A^i_{j,q}}(u_i)$ that maps $U_i$ to $[0,1]$, a “fuzzy set” denoted with $A^i_{j,q}$ is defined as:

$$A^i_{j,q} = \{ (u_i, \mu_{A^i_{j,q}}(u_i)) : u_i \in U_i \}$$

![Figure 4.1 Different membership functions shapes found in Matlab](image)

4.2 **Fuzzy logic system structure**

The fuzzy system consists of three main blocks: Fuzzification, Inference Mechanism and Defuzzification as shown in Figure 4.2. One of the famous fuzzy systems is the Mamdani fuzzy system which is constructed to work with crisp inputs and it takes one or more real value inputs and transforms them into fuzzy sets in the fuzzification part. These sets are then programmed to the inference system where the actual computation is performed.
The inputs and outputs of the fuzzy system are crisps (real numbers) not fuzzy sets. The fuzzification block transfers the crisp inputs ($u_1, u_2, ..., u_n$) values to fuzzified inputs while the inference engine block uses the fuzzy rules from the rule-based block to generate fuzzy conclusions and the defuzzification block converts these fuzzy conclusions into the crisp outputs ($y_1, y_2, ..., y_m$). These crisp outputs values represent the outputs of the fuzzy system. Each part of fuzzy logic system will be explained in detail below.

- **Fuzzification**

  Fuzzification is a mapping from the observed numerical input space to the fuzzy sets that are defined in the corresponding universes of discourse. The fuzzifier maps numerical values (universe of discourse) of any input denoted by ($u_1, u_2, ..., u_n$) into fuzzy sets represented by membership functions. Therefore, the fuzzifier’s duty is to transform (encode) the crisp valued inputs into fuzzy sets. The encoded information is then used in the fuzzy inference process.

- **Inference Mechanism**

  Inference mechanism has two main tasks. In the first task, the premises of all rules are compared to the controller inputs $u_i$ to determine which rules can be applied to the current situation (this step is called matching). In the second task, the conclusions (related to control action) are determined using the active current rules.
(this step is called inference). The main item to focus on is how to quantify the logical operation “and” in the premise part of the rules. There are several ways to define the value of the premise part of $q^{th}$ fuzzy rule ($\mu_{\text{premise}(q)}$) when the “and” operator is used.

1. **Minimum:** Define $\mu_{\text{premise}(q)} = \min\{\mu_{A_1}(u_1), \mu_{A_2}(u_2), ..., \mu_{A_n}(u_n)\}$

2. **Product:** Define $\mu_{\text{premise}(q)} = \mu_{A_1}(u_1) \times \mu_{A_2}(u_2) \times ... \times \mu_{A_n}(u_n)$

The value of the conclusion part of $q^{th}$ fuzzy rule ($u(q)(y_s)$) can be described by using one of the following formulas (implication formulas):

1. **Minimum:** Define $u(q)(y_s) = \min\{\mu_{\text{premise}(q)}, \mu_{B^{q}r_s}(y_s)\}$

2. **Product:** Define $u(q)(y_s) = \mu_{\text{premise}(q)} \times \mu_{B^{q}r_s}(y_s)$

Where $\mu_{B^{q}r_s}(y_s)$ is the output membership function that corresponds to the current rule.

- **Defuzzification**

It maps the fuzzy conclusion values defined over a universe of discourse to crisp outputs (converting decisions into actions). In other words, defuzzification operates on the implied fuzzy sets produced by the inference mechanism and combines their effects to provide the fuzzy output. It is employed because in many practical applications a crisp output is required. There are many approaches to perform the defuzzification such as:

1. **The Center of Gravity (COG) Method**

The center of area generates the center of gravity of the possibility distribution of the implication fuzzy output. Let $b_q$ denotes the center of the membership function (at which the membership function reaches the maximum value) of the consequent of $q^{th}$ rule. The output of the fuzzy system can be calculated by:

$$
y_{s}^{\text{crisp}} = \sum_{q} b_q \int u(q)(y_s) \frac{\int u(q)(y_s)}{\sum_{q} \int u(q)(y_s)}
$$

(4.1)

Where $\int u(q)(y_s)$ is the area under the membership function $u(q)(y_s)$. 

2. The Center Average (CA) Method

By using this method the output of the fuzzy system can be given by:

\[ y_s^{\text{crisp}} = \frac{\sum q b_q \mu_{\text{premise}(q)}}{\sum q \mu_{\text{premise}(q)}} \]  \hspace{1cm} (4.2)

The fuzzy logic controller is usually based on the operator's knowledge. Therefore, it is difficult to determine the appropriate parameters of the fuzzy controller (constants of membership functions) which are used to generate the control action especially when there are changes in the parameters of the controlled process. In this case, the fuzzy controller will be improper controller unless its parameters are modified to overcome the changes in the plant parameters.[10]-[11]

4.3 Designing FLC for Suspension System

The FLC used in the active suspension has three inputs. Let's assume that these inputs are \( e \) (body deflection), \( \dot{e} \) (body velocity), \( \ddot{e} \) (body acceleration), and one output which is desired actuator force \( u \). The control system itself consists of three stages: fuzzification, fuzzy inference machine and defuzzification as mentioned in the previous section.

The fuzzification stage converts real-number (crisp) input values into fuzzy values, while the fuzzy inference machine processes the input data and computes the controller outputs in cope with the rule base and data base. These outputs, which are fuzzy values, are converted into real-numbers by the defuzzification stage.

The rule base used in the active suspension system can be represented by the following table with fuzzy terms derived by modeling the designer's knowledge and experience. The linguistic control rules of the FLC obtained from the Table used in such a case are as follows:

\( R_1: \text{If } (e = \text{PM}) \text{ and } (\dot{e} = \text{PM}) \text{ and } (\ddot{e} = \text{ZE}) \text{ Then } (u = \text{ZE}) \)

\( R_2: \text{If } (e = \text{PS}) \text{ and } (\dot{e} = \text{PM}) \text{ and } (\ddot{e} = \text{ZE}) \text{ Then } (u = \text{NS}) \)

\( \cdots \)

\( R_i: \text{If } (e = A_i) \text{ and } (\dot{e} = B_i) \text{ and } (\ddot{e} = C_i) \text{ Then } (u = D_i) \)

Where \( A_i, B_i, C_i, \) and \( D_i \) are the labels of fuzzy sets representing the linguistic values of \( e, \dot{e}, \ddot{e}, \) and \( u \) respectively, and characterized by their membership functions.
Table 4.1 Rule Base of the FLC Model

<table>
<thead>
<tr>
<th>e</th>
<th>(\dot{e})</th>
<th>(\ddot{e})</th>
<th>u</th>
<th>e</th>
<th>(\dot{e})</th>
<th>(\ddot{e})</th>
<th>u</th>
</tr>
</thead>
<tbody>
<tr>
<td>PM</td>
<td>PM</td>
<td>ZE</td>
<td>ZE</td>
<td>PM</td>
<td>PM</td>
<td>P or N</td>
<td>NS</td>
</tr>
<tr>
<td>PS</td>
<td>PM</td>
<td>ZE</td>
<td>NS</td>
<td>PS</td>
<td>PM</td>
<td>P or N</td>
<td>NM</td>
</tr>
<tr>
<td>ZE</td>
<td>PM</td>
<td>ZE</td>
<td>NM</td>
<td>ZE</td>
<td>PM</td>
<td>P or N</td>
<td>NB</td>
</tr>
<tr>
<td>NS</td>
<td>PM</td>
<td>ZE</td>
<td>NM</td>
<td>NS</td>
<td>PM</td>
<td>P or N</td>
<td>NB</td>
</tr>
<tr>
<td>NM</td>
<td>PM</td>
<td>ZE</td>
<td>NB</td>
<td>NM</td>
<td>PM</td>
<td>P or N</td>
<td>NV</td>
</tr>
<tr>
<td>PM</td>
<td>PS</td>
<td>ZE</td>
<td>ZE</td>
<td>PM</td>
<td>PS</td>
<td>P or N</td>
<td>NS</td>
</tr>
<tr>
<td>PS</td>
<td>PS</td>
<td>ZE</td>
<td>NS</td>
<td>PS</td>
<td>PS</td>
<td>P or N</td>
<td>NM</td>
</tr>
<tr>
<td>ZE</td>
<td>PS</td>
<td>ZE</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>P or N</td>
<td>NM</td>
</tr>
<tr>
<td>NS</td>
<td>PS</td>
<td>ZE</td>
<td>NM</td>
<td>NS</td>
<td>PS</td>
<td>P or N</td>
<td>NB</td>
</tr>
<tr>
<td>NM</td>
<td>PS</td>
<td>ZE</td>
<td>NM</td>
<td>NM</td>
<td>PS</td>
<td>P or N</td>
<td>NB</td>
</tr>
<tr>
<td>PM</td>
<td>ZE</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>ZE</td>
<td>P or N</td>
<td>PM</td>
</tr>
<tr>
<td>PS</td>
<td>ZE</td>
<td>ZE</td>
<td>ZE</td>
<td>PS</td>
<td>ZE</td>
<td>P or N</td>
<td>PS</td>
</tr>
<tr>
<td>ZE</td>
<td>ZE</td>
<td>ZE</td>
<td>ZE</td>
<td>ZE</td>
<td>ZE</td>
<td>P or N</td>
<td>ZE</td>
</tr>
<tr>
<td>NS</td>
<td>ZE</td>
<td>ZE</td>
<td>ZE</td>
<td>NS</td>
<td>ZE</td>
<td>P or N</td>
<td>NS</td>
</tr>
<tr>
<td>NM</td>
<td>ZE</td>
<td>ZE</td>
<td>NS</td>
<td>NM</td>
<td>ZE</td>
<td>P or N</td>
<td>NM</td>
</tr>
<tr>
<td>PM</td>
<td>NS</td>
<td>ZE</td>
<td>PM</td>
<td>PM</td>
<td>NS</td>
<td>P or N</td>
<td>PB</td>
</tr>
<tr>
<td>PS</td>
<td>NS</td>
<td>ZE</td>
<td>PM</td>
<td>PS</td>
<td>NS</td>
<td>P or N</td>
<td>PB</td>
</tr>
<tr>
<td>ZE</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>ZE</td>
<td>NS</td>
<td>P or N</td>
<td>PM</td>
</tr>
<tr>
<td>NS</td>
<td>NS</td>
<td>ZE</td>
<td>ZE</td>
<td>NS</td>
<td>NS</td>
<td>P or N</td>
<td>PM</td>
</tr>
<tr>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>ZE</td>
<td>NM</td>
<td>NS</td>
<td>P or N</td>
<td>PS</td>
</tr>
<tr>
<td>PM</td>
<td>NM</td>
<td>ZE</td>
<td>PB</td>
<td>PM</td>
<td>NM</td>
<td>P or N</td>
<td>PV</td>
</tr>
<tr>
<td>PS</td>
<td>NM</td>
<td>ZE</td>
<td>PM</td>
<td>PS</td>
<td>NM</td>
<td>P or N</td>
<td>PB</td>
</tr>
<tr>
<td>ZE</td>
<td>NM</td>
<td>ZE</td>
<td>PM</td>
<td>ZE</td>
<td>NM</td>
<td>P or N</td>
<td>PB</td>
</tr>
<tr>
<td>NS</td>
<td>NM</td>
<td>ZE</td>
<td>PS</td>
<td>NS</td>
<td>NM</td>
<td>P or N</td>
<td>PM</td>
</tr>
<tr>
<td>NM</td>
<td>NM</td>
<td>ZE</td>
<td>ZE</td>
<td>NM</td>
<td>NM</td>
<td>P or N</td>
<td>PS</td>
</tr>
</tbody>
</table>

The abbreviations used in Table 4.1 correspond to:

- NV: Negative Very Big
- NB: Negative Big
- NM: Negative Medium
- NS: Negative Small
- ZE: Zero
- PS: Positive Small
- PM: Positive Medium
- PB: Positive Big
- PV: Positive Very Big

The output of the fuzzy controller is a fuzzy set of control. In this study, for the process which usually requires a non-fuzzy value of control, the “center of gravity defuzzification method” is used.
A possible choice of the membership functions for the four mentioned variables of the active suspension system represented by a fuzzy set is as shown in Figures 4.3-4.6.

![Figure 4.3 Membership Function for displacement and velocity input variables](image1)

![Figure 4.4 Membership Function for acceleration input variable](image2)

![Figure 4.5 Membership Function for the desired actuator force output variable](image3)
4.4 Implementation of FLC, Simulations, and Results

4.4.1 Quarter Vehicle System

The following figure shows how the FLC is implemented in the quarter model:

![Diagram of FLC implementation in a quarter vehicle system](image)

**Figure 4.7** Fuzzy Logic Controlled Quarter Vehicle System

The simulation is made with the same road excitation leading to the following result:

![Graph showing time response of vertical displacement (X1-X2)](image)

**Figure 4.8** Time response of vertical displacement (X1-X2)
4.4.2 Half Vehicle System

In this model two actuators are installed and thus the number of controllers is doubled such that each one gives a control force to one actuator. The Simulink model stays the same except for the controller block which is shown below:

![Diagram of FLC block for Half Vehicle Model]

**Figure 4.10 FLC Block of Half Vehicle Model**

![Graph showing time response of vertical displacement X]

**Figure 4.11 Time response of vertical displacement X**
Figure 4.12 Time response of pitch angle

Figure 4.13 Time response of vertical displacement X – FLC vs Passive vs PID

Figure 4.14 Time response of pitch angle - FLC vs Passive vs PID
4.4.3 Full Vehicle System

Everything done previously is applicable on the full system except that we need four controllers for each actuator installed in one of the vehicle’s corners. The same controller used in quarter and half models is used here. However, the control block in this model just gets bigger as shown below:

![FLC Block in Full Vehicle Model](image)

The following are the simulation results and their comparison with previous systems:

![Time response of vertical displacement at P1](image)
Figure 4.17 Time response of vertical displacement at P2

Figure 4.18 Time response of vertical displacement at P3

Figure 4.19 Time response of vertical displacement at P4
Figure 4.20 Time response of vertical displacement at Pc

Figure 4.21 Time response of pitch angle

Figure 4.22 Time response of roll angle
Figure 4.23 Time response of vertical displacement at Pc - FLC vs Passive vs PID

Figure 4.24 Time response of pitch angle - FLC vs Passive vs PID

Figure 4.25 Time response of roll angle - FLC vs Passive vs PID
4.5 One More Comparison Step

In fact, it is very important to compare the two kinds of controllers that were implemented in the vehicle systems which are in this case the PID controller and the Fuzzy Logic one (FLC). However, the road profile which was input may be not sufficient for this comparison to be achieved although it is a very rough profile and hard to be found in reality. For this purpose, another road profile input is proposed to simulate a little road excitation that a vehicle can normally face. The input is considered as a step function of amplitude 0.1 meters and is applied after 1 second of the simulation beginning. It is enough to present the simulation done on the full system which is the one desired to be controlled. The transport delays are assumed to be the same as before: 0.3, 0.5, and 0.8 for the second, third, and fourth corner of the vehicle respectively. The comparisons of the bounce, pitch, and roll responses in passive, PID controlled, and FL controlled systems is presented in the following figures:

![Figure 4.26](image1)

Figure 4.26 Time response of vertical displacement at Pc with step input - Passive vs PID vs FLC

![Figure 4.27](image2)

Figure 4.27 Time response of pitch angle with step input - Passive vs PID vs FLC
One last comparison is made in an attempt to fulfill our goal in finding the best controller between the studied ones. This is done by applying the braking and cornering forces which were presented in chapter 3. The only difference is that the cornering torque is assumed in this section to be exactly the same as the braking one for more complexity in small interval of time (10 seconds). The following figures show the comparison between passive, PID controlled, and FL controlled systems when the mentioned torques are applied:

Figure 4.28 Time response of roll angle with step input - Passive vs PID vs FLC

Figure 4.29 Time response of vertical displacement at Pc with torques input - Passive vs PID vs FLC

Figure 4.30 Time response of pitch angle with torques input - Passive vs PID vs FLC
Figure 4.31 Time response of roll angle with torques input - Passive vs PID vs FLC
Chapter 5 . Conclusions and Recommendations

5.1 Conclusions

In this work, quarter, half and full vehicle linear active suspension models have been investigated, in which one or more of three motions, the vertical, pitching and linear active suspension systems with actuators have been investigated. The mathematical models for suspension systems with actuators have been derived. The results have shown that the passive suspension systems without control are not sufficient to effectively reduce the vibrations arisen from travelling on the rough road or to minimize the rolling and pitching movements when sharp maneuvers take place due to braking and cornering.

PID controllers are proposed to design as control systems, where all output variables including vertical displacement at the center of gravity, roll angle and pitch angle, depended on the vertical displacements at the corner points that are under control. The body deflection and deviation are used to evaluate the road handling and riding comfort of the passengers. By supplying the control signal the vertical displacements at each corner and the body deviation in the pitch and roll directions are largely reduced. The results illustrate that the output responses of the controlled system have been efficiently improved when the PID controllers are used. The vertical displacements at each suspended corner and at the center of gravity, the pitch angle, and the roll angle becomes nearly negligible, i.e. they are not sensed by the passengers anymore. The maximum displacement at the center of gravity is about 7 millimeters; the optimal pitch angle does not exceed 0.5 degree, and the largest roll angle is less than 1 degree, although the road input was supposed to be a very rough one with random and low frequency oscillations. On the other hand, when the bending and braking torques are applied, or when step input as road profile is applied, the PID shows a good result but not the desired one since the result obtained
includes an overshoot with one oscillation and it took about 2 seconds with the controller to attain the steady state.

Fuzzy Logic controller is designed. The results of the controller implementation indicated that the proposed controllers pressed the outputs of the controlled system responses to be smaller than the corresponding outputs of the passive system. The problem is that the random input simulating the road profile was too rough for the fuzzy controller leading to a delay in its response for it took about 1 to 2 seconds for the controller to give an efficient response, a response which is in real good as much as was that obtained from the PID controller. On the other hand, when the step input is applied, or even the braking and bending effects are, the fuzzy logic controller shows a very good result since the oscillations are eliminated and the steady state is attained rapidly which reveals a good road handling and provides passengers comfort and of course the roll over is surely prevented.

In brief, for random rough road inputs the fuzzy logic controller shows a similar response compared to the PID except for the first 2 seconds where the PID was better and more stable. However, with step road input or with the application of brake and bend effects, the response obtained from the FLC was improved than that of the PID controller.

5.2 Recommendations for Future Work

The aim of any work in control field is to improve the performance of a controller or to propose a new control scheme. Several recommendations about future works are listed below:

- In this work a triangular shape function is used as a membership function for accomplishment of the fuzzification process. Other membership functions can also be used, such as Gaussian function, bell shape function or trapezoid function. The formulas of those functions are more complex than that of the triangular one. Therefore, the update equations used to upgrade the parameters of the fuzzy controller will be much more complicated.

- Disturbances with dry asphalt road have been investigated. The performance of the proposed controllers can be demonstrated by simulation under other road
conditions, e.g. snowy road, wet asphalt or transitions between such conditions (the road switches from snowy road to wet asphalt) to assess the robustness of the controllers.

- The type of the actuators is not investigated in this research. Hydraulic, electromagnetic, or other types can be used and their efficiency and effectiveness can be studied to find the best actuator to be installed.

- The modeled systems can be tested on real road profiles with different disturbances and conditions to assess their effectiveness and to validate the robustness of the implemented controllers.

- In this work, genetic algorithm has been employed to obtain the values of the parameters of fuzzy logic systems. Other techniques can be proposed, such as neural network, to obtain the parameters of the fuzzy system. It might also yield good results.

- The half and full systems have been investigated in their simplest forms. Other models can be studied taking into consideration the unsprung masses which differentiate between the body mass and the suspensions masses, and more springs and dampers would be taken into account. This step would increase the simulations accuracy and make it closer to reality.

- Other types of controllers can be also investigated such as skyhook, backstepping, or sliding mode or any other controller which may lead to good results. In real, a combination of many controllers can be experienced such as PID with FLC where one can play the role of compensating the other and so on...
References

[1] “Neurofuzzy Controller Based Full Vehicle Nonlinear Active Suspension Systems”, Dr. Ammar A. Aldair, School of Engineering and Informatics, University of Sussex, Brighton, UK, March 2012
[8] “Active Suspension of Cars Using fuzzy Logic Controller Optimized by Genetic Algorithm”, R. K. Pekgökgöz, M. A. Gürel, M. Bilgehan, M. Kýsa, Harran University, Turkey 2010
[10] “Simulation on Active Suspension System for Peugeot 206”, Mohammad H. Bin Harun