MATLAB/Simulink Simulation of Semi-Active Seat Suspension System with a Variable Flow Damper Valve

by

Scott A. Carpenter

A Report Submitted to the Faculty of the

Milwaukee School of Engineering

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Engineering

Milwaukee, Wisconsin

May 2019

Abstract

This project is the focus of a Capstone Design Project for the Milwaukee School of Engineering's (MSOE) Master of Science in Engineering (MSE) program. In this project, a simulation was developed to take known chassis input in order to evaluate the suspension, damper system design, and control system design for a semi-active seat suspension system for various applications including off-highway and utility vehicles. The software MATLAB/Simulink was employed to create a system model based on the elements of the semi-active seat suspension system, including environmental inputs, as well as the chassis vibration profile and the operator profile. The system model analyzes motion in three axes to predict the resulting movement the driver/operator experiences. Components of the system are three individual dampers employing variable flow hydraulic valves, a proportional–integral–derivative (PID) control system, and the mechanical suspension system of the seat. This software model will aid in the development of the semi-active seat suspension system by allowing several iterations to be run to evaluate different valve designs and control system strategies.

From the results of the model it is seen that more work is needed to increase the fidelity of the model. This could be done by including the piston and shim stack design in the model. Other opportunities for further model development are to improve the control system methodology and to account for the interrelationships between the three axes to determine the effect that vibration in one axis may have on vibration in the other axes.

Acknowledgments

The author has worked with the support of two coworkers at Hayes Performance Systems on this project. One is an electrical engineer, Frank Molinaro, who has helped with the design of the control system and base MATLAB model that had evolved over the year prior to the start of this project. Starting with this base model, the project expanded the model to include three axes of motion, z, y, and x, to simulate a system with three dampers. Mr. Molinaro has been instrumental in providing a base understanding of the system being modeled.

The other coworker is a suspension expert, Edward Kwaterski, who has helped with the design and testing of the damper system. The model developed in this project will be used to model several iterations of the system to allow fine tuning to meet customer needs and expectations.

Table of Contents

List of Figures 5

List of Tables 7

Nomenclature 8

Introduction 9

Description of Project 10

Background 17

Project Goal 19

Review of Literature 20

Problem Statement 24

Methods 24

Results and Discussion 26

Conclusions and Recommendations 45

References 47

Bibliography 51

Appendix A: MATLAB Source Code 52

List of Figures

Figure 1: Simplified Model of Seat Suspension System with Semi-Active Damping…..…..….11

Figure 2: Simplified MATLAB Model of Sprung Mass Damper System…….……..……..12

Figure 3: Damping Force (N) versus Piston Velocity (mm/sec) for DDA Valve…….….……..13

Figure 4: Flow versus Current for TracTive DDA Valve…………..……..……….….….…..14

Figure 5: Cut-Away View of TracTive DDA Valve……………….…….…………..………...15

Figure 6: Semi-Active Seat Suspension System……………………..……………......……..15

Figure 7: Vibration Profiles in Z-axis for Various Machines………………………………..16

Figure 8: Simscape Model of Damper System……..……………………..……........………25

Figure 9: Force (N) versus Velocity (mm/s) of the TracTive DDA Valve in a Prototype Damper………………………………………………………………………………...26

Figure 10: Force (N) versus Displacement (mm) at 1.5 Amps for Various Velocities (mm/s)....27

Figure 11: Force (N) versus Displacement (mm) at 0.5 Amps for Various Velocities (mm/s)....27

Figure 12: Force (N) versus Velocity (mm/s) for Various Valve Currents………….…...……..28

Figure 13: Force (N) versus Velocity (mm/s) for Various Valve Currents……………………..29

Figure 14: Viscosity versus Temperature (°F) for Damper Fluid……………..………………...31

Figure 15: AG3 Vibration Profile in Z-axis for Agricultural Machine…………………………31

Figure 16: MATLAB Model Output Using AG3 Vibration Profile in Z-axis for Agricultural Machine……………………………………………………………………………….32

Figure 17: EM6 Vibration Profile in Z-axis for Tracked Machine……………………………..33

Figure 18: MATLAB Model Output Using EM6 Vibration Profile in Z-axis for Tracked Machine……………………………………………………………………………….34

Figure 19: J25TR5 Vibration Profile in Z-axis for Highway Truck, Smooth…………………..35

Figure 20: MATLAB Model Output Using J25TR5 Vibration Profile in Z-axis for Highway Truck, Smooth……………………………..……………………………………………..35

Figure 21: JR5TR2 Vibration Profile in Z-axis for Highway Truck, Medium………………….36

Figure 22: MATLAB Model Output Using JR5TR2 Vibration Profile in Z-axis for Highway Truck, Medium……………………………………………………………………………..37

Figure 23: J51TR1 Vibration Profile in Z-axis for Highway Truck, Rough……………………37

Figure 24: MATLAB Model Output Using J51TR1 Vibration Profile in Z-axis for Highway Truck, Rough…………………………………………………………………..……………38

Figure 25: J28TR2 Vibration Profile in Z-axis for Backhoe Loader……………………………39

Figure 26: J28TR2 Vibration Profile in Z-axis for Backhoe Loader…………………………....39

Figure 27: MATLAB Model Output Using J28TR2 Vibration Profile in Z-axis for Backhoe Loader……………………………………………………………………………………40

Figure 28: J28TR2 Vibration Profile in X-axis for Backhoe Loader………………………..…41

Figure 29: J28TR2 Vibration Profile in X-axis for Backhoe Loader…………………………...41

Figure 30: MATLAB Model Output Using J28TR2 Vibration Profile in X-axis for Backhoe Loader……………………………………………..…………………………………42

Figure 31: J28TR2 Vibration Profile in Y-axis for Backhoe Loader…………………………..43

Figure 32: J28TR2 Vibration Profile in Y-axis for Backhoe Loader…………………………..43

Figure 33: MATLAB Model Output Using J28TR2 Vibration Profile in Y-axis for Backhoe Loader……………………………………………………………………………………44

List of Tables

Table 1: ISO 2631-1 Comfort Reactions to Vibration Environments……………………18

Nomenclature

Symbols

$a(t)$ - frequency weighted acceleration

a_{xB} - RMS acceleration at the seat base

a_{xS} - RMS acceleration at the seat surface

a_{xwB} - RMS acceleration for the base in the x-direction

a_{xwS} - RMS acceleration for the seat in the x-direction

d – damper

i(t) – current

k – spring

L – inductance

m – mass of seat and driver

R – resitance

T_{ax} – Transmissibility

v_{in} (t) – voltage

Abbreviations

DDA - dynamic damping adjustment

KVL - Kirchhoff's Voltage Law

MPC - model predictive controller

MR – Magneto-rheological

PID - proportional–integral–derivative

SEAT - Seat Effective Amplitude Transmissibility factor

VDV - vibration dose value

Introduction

It has long been known that whole body vibration can cause fatigue and health issues in people employed in the transportation industry or as equipment operators [1]. There has been significant research into the health issues and methods to lessen the effects of whole body vibration [1]. This has led to numerous designs of seat suspension systems for heavy-duty trucks and equipment. The earliest systems consisted of spring mechanisms that attempted to isolate the driver and operator from the movement of the vehicle. Increasingly complex systems have been created to help mitigate the vibration effects of the vehicles. The latest systems employ active dampers to anticipate the movement of the vehicle and counteract that movement within the seat suspension system [2]. These systems can be very costly, which has limited their adoption across many industries. Semi-active systems have begun to be developed to provide many of the benefits of the active systems but at a lower price point [3, 4].

This project features a dynamic system simulation that requires mechanical, electrical, vibration, and thermal expertise to predict the system response based on the chassis input. While semi-active seat damping currently exists in over-the-road and other markets, this project looks at additional markets that have not yet been explored. Significant research has been done on active systems but little has been done with semi-active systems [5, 6, 7, 8, 9]. This project looks to fill that gap in the research between passive and active seat suspension systems.

While active seat suspension systems are very effective at reducing whole body vibration, the costs of the systems can be prohibitive. Semi-active seat suspension systems provide much of the benefit of active systems at a lower cost, making such systems more affordable, which could in turn drive increased adoption of these systems. This will result in reduced driver and operator fatigue, longer drive times before stopping, and reduced long-term health concerns for drivers and operators.

The ability to simulate semi-active seat suspension systems will allow for improved designs and a more efficient development process, reducing total development time and speed time-to-market.

The outcome of this project is a software model to simulate a seat suspension system with semi-active damping. Inputs to the model are from the chassis vibration profile, operator profile, and control system design. The output of the simulation is the system response based on these inputs.

Description of Project

The objective of this project was to create a software model that can be used to simulate seat suspension systems based on the chassis input and that will simulate the system response of a semi-active damping system. Target markets for this system are off-highway equipment and other utility markets. The system consists of three individual dampers in the vertical, side-to-side and fore-and-aft directions controlled by a proportional–integral–derivative (PID) controller. The dampers have an electro-hydraulic valve that has variable flow control of hydraulic fluid through the valve, resulting in variable damping based on the chassis inputs to the system. The valves for the dampers are TracTive dynamic damping adjustment (DDA) valves [10, 11] and are enclosed in an engineered damper. Based on a solenoid valve, the TracTive DDA valve opens a bypass proportionally with applied current. At full current of 2 amps, the bypass is fully open. This valve allows for continuously variable damping to be able to tune the response of the system. The damper system will be adapted to an existing semi-active seat suspension structure, as shown in Figure 5. A simplified model of the system is shown in Figure 1.

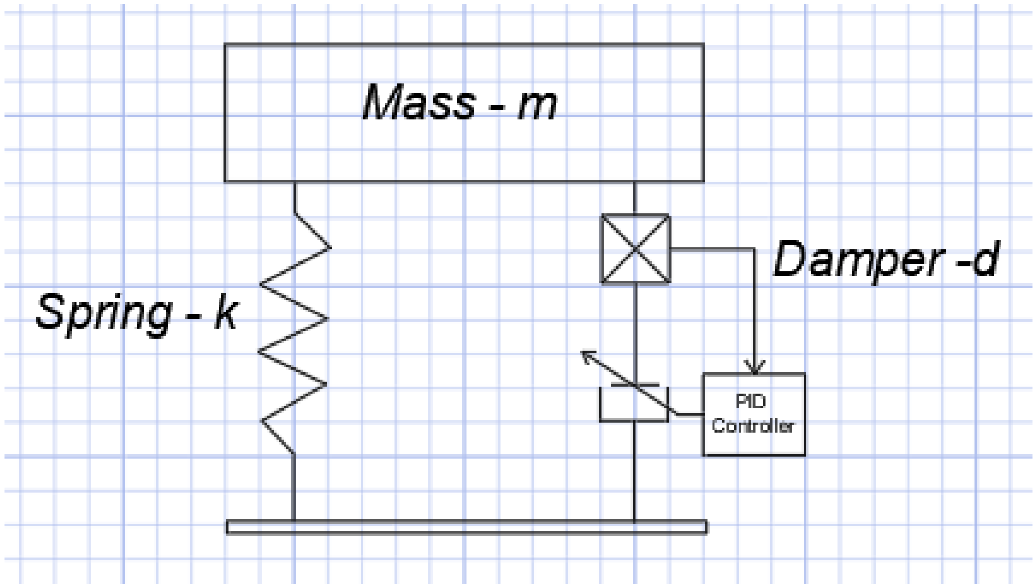

Figure 1: Simplified Model of Seat Suspension System with Semi-Active Damping.

The mathematical representation of Figure 1 is as shown:

$$mx = \sum F = -F_d - F_k, \tag{1}$$

$$m\ddot{x} = -d\dot{x} - kx, \tag{2}$$

$$m\ddot{x} + d\dot{x} + kx = 0, \tag{3}$$

where

m is the mass of the seat and operator,

d is the damping coefficient,

k is the spring constant.

Taking these equations and applying the Laplace transform theorem results in:

$$ms^2X + dsX + kX = 0, \tag{4}$$

$$X(s)(\, s^2 + s + 1) = 0, \tag{5}$$

$$X(s) = \frac{1}{(s^2+s+1)}\,. \tag{6}$$

Equation (1), according to Newton's Law, is the sum of forces in the vertical (x) direction showing that the force due to the mass, m, of the seat and operator is equal to the force due to the spring, k, and damper, d. In Equation (2), the forces are further derived as the force of the driver and seat are equal to the mass, m, times the acceleration, represented by the second derivative of x, $\ddot{x}$. This is equal to the force of the spring, calculated by the spring coefficient, -k, times the distance, x, and the force of the damper, calculated by the damping coefficient, -d, times the velocity, $\dot{x}$. Equation (3) rearranges the terms to form an ordinary differential equation, which is then converted to a Laplace transform in Equations (4) through (6).

The focus of this project is on the damping coefficient, d, using the TracTive DDA valve to provide variable damping based on system input. A simple model of this system in MATLAB appears in Figure 2.

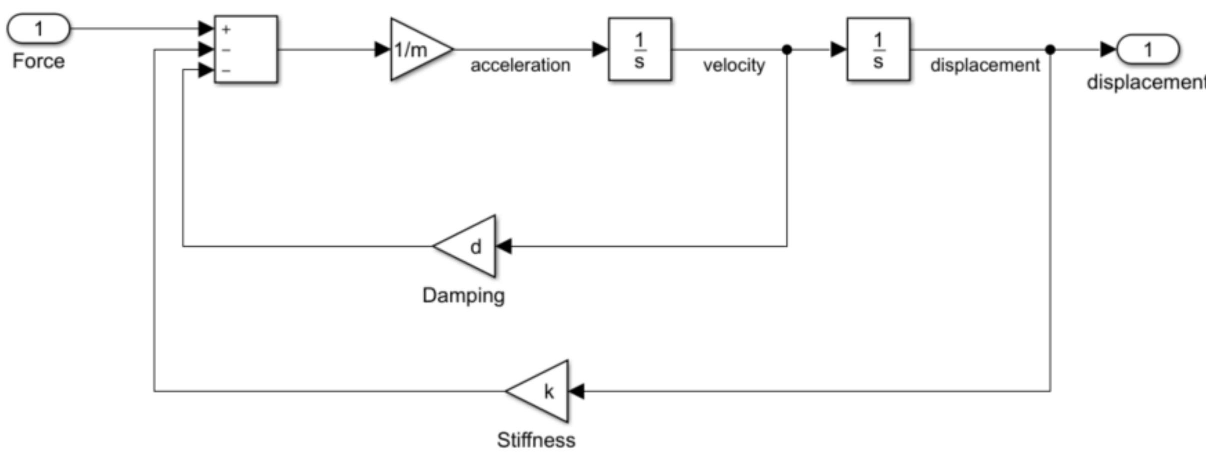

Figure 2: Simplified MATLAB Model of Sprung Mass Damper System.

The software model was created in MATLAB and includes the three individual dampers and the PID controller. Environmental variables, such as temperature of working fluid and external temperature of damper, were considered during the development of the system model.

The main component of the system is the TracTive DDA valve. To create the model for this valve, a transfer function based on experimental data provided by Tractive was employed, as seen in Figure 3 [10].

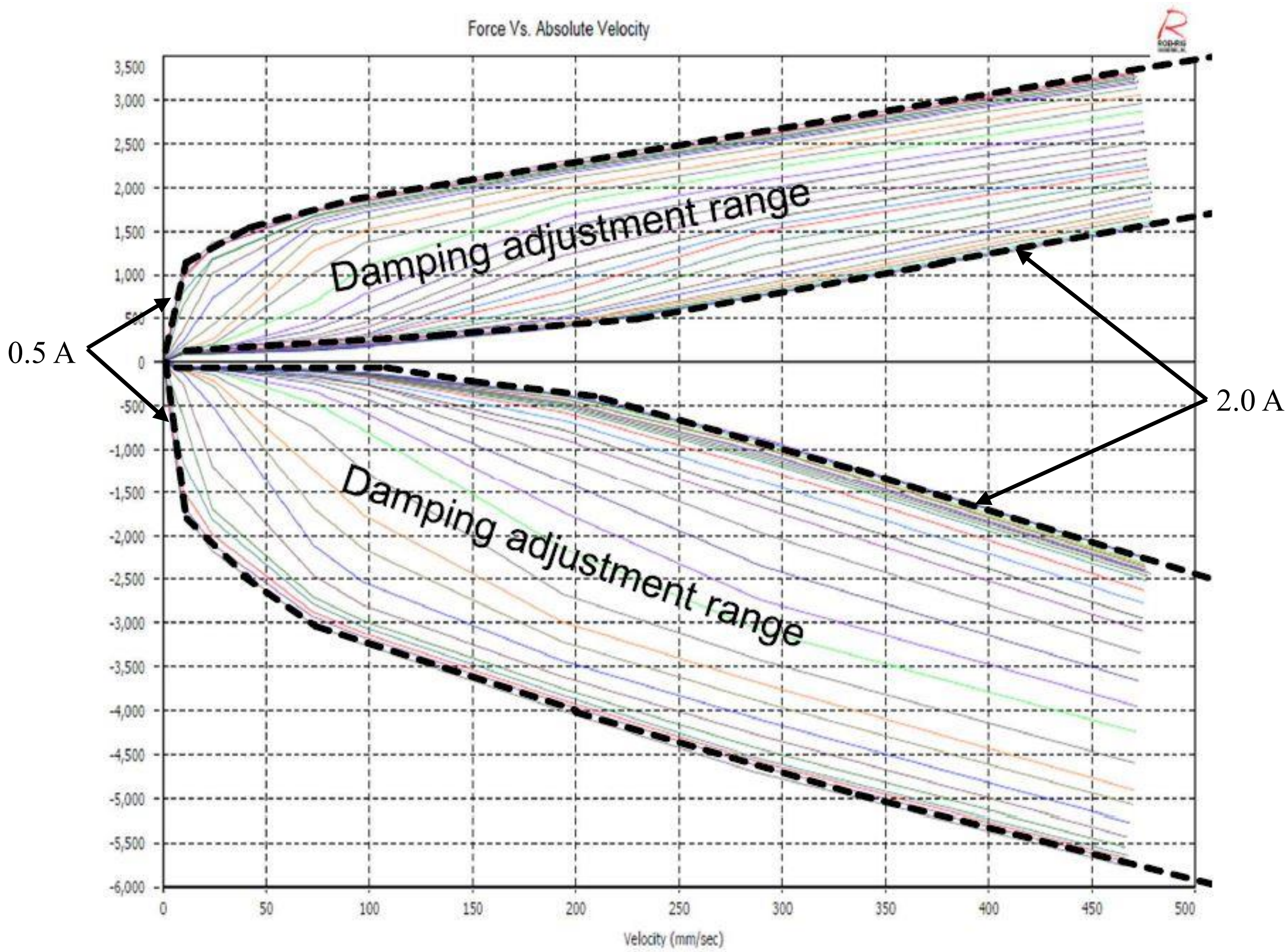

Figure 3: Damping Force (N) versus Piston Velocity (mm/sec) for DDA Valve [10].

Figure 3 shows various curves throughout the range of current applied to the valve. At the extremes is a curve for 0.5 A, reflecting a fully closed valve (maximum damping), and a curve for 2.0 A, reflecting a fully open valve (minimum damping). The upper half of the graph shows the curves for compression (positive force) and the lower half of the graph shows the curves for extension (negative force). In addition, the flow versus current graph as provided by Tractive in Figure 4 was used to develop an understanding of the area within the damping adjustment range, as seen in Figure 3.

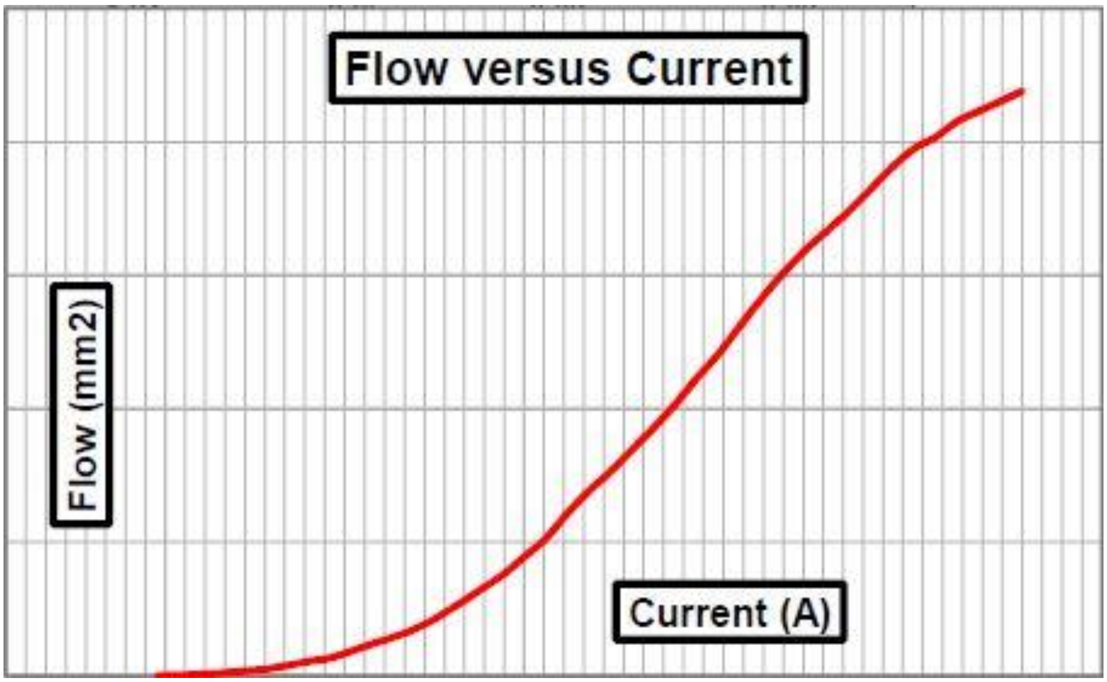

Figure 4: Flow versus Current for TracTive DDA Valve [10].

At 0.5 A, the valve is closed, flow is minimal, and damping is at the maximum. As current is increased, moving right on the graph, the flow through the valve increases up to maximum flow at 2.0 A of current, which results in minimal damping. The control system is designed to utilize the linear portion of the curve. By controlling the current applied to the valve, it is possible to produce continuously variable damping [10].

A cut-away of the TracTive DDA valve, as provided by Tractive, can be seen in Figure 5.

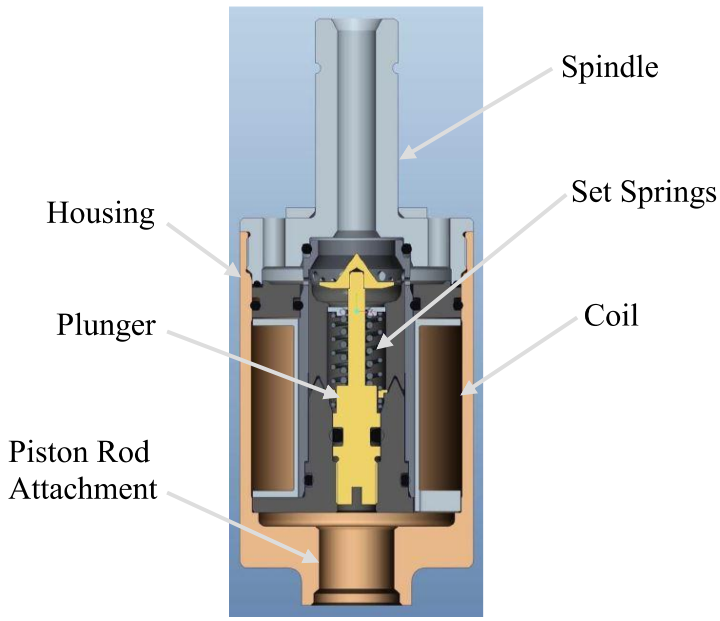

Figure 5: Cut-Away View of TracTive DDA Valve [10].

In Figure 5, the solenoid coil and other elements of the DDA valve can be seen. When the coil is energized, the plunger moves up and opens the valve, allowing hydraulic fluid to flow through the bypass [10].

The DDA valve was designed into a damper that was adapted to the existing semi-active seat suspension system. Figure 6 shows this system and calls out the main components. The new damper system will replace the semi-active damper, controller, and position sensor.

Figure 6: Semi-Active Seat Suspension System [12].

As inputs to the MATLAB model there are several options for standard vibration profiles. The standard profiles studied in this project are:

AG3 – Agriculture Machine

EM6 – Tracked Machine

J25TR5 – Highway Truck, Smooth

J25TR2 – Highway Truck, Medium

J51TR1 – Highway Truck, Rough

J28TR2 – Backhoe Loader Tractor in three axes

Of these vibration profiles, the J28TR2 profile is the only profile for which data in three axes were available and studied. All other profiles (shown in Figure 7) were examined in the z-axis only to verify the accuracy of the model when comparing to laboratory data. All laboratory data for Seat Effective Amplitude Transmissibility (SEAT) values, a performance indicator employed in the project, are only available in the z-axis.

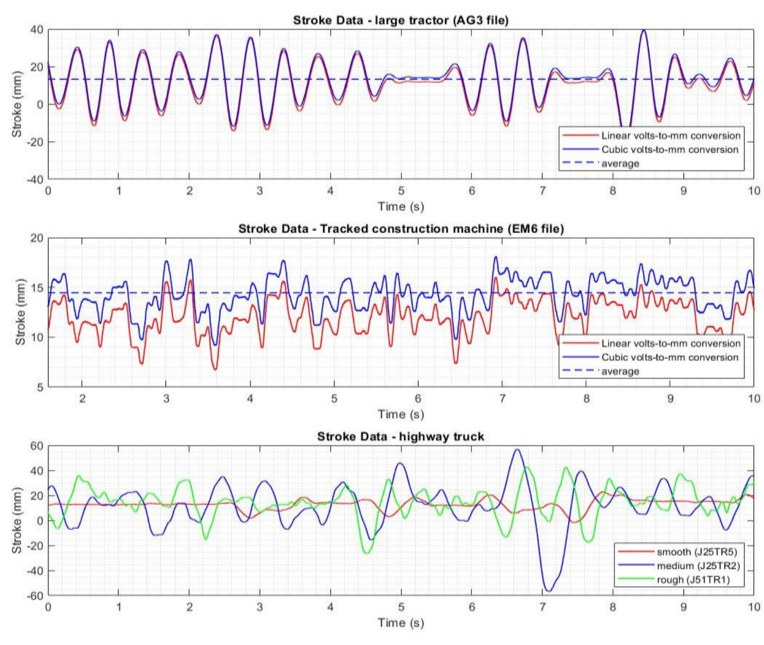

Figure 7: Vibration Profiles in Z-axis for Various Machines.

Background

In an effort to improve the comfort of drivers and operators of heavy-duty trucks and equipment, seating systems have come into focus. Typically, these vehicles exhibit low frequency vibration below 5 Hz. Low-frequency vibration in the range of 0.5 to 5 Hz results in higher risk of low back pain, which can lead to driver fatigue and injury [13]. By isolating the driver from the chassis vibration, driver fatigue and injury can be reduced. The addition of a suspension system to the driver and operator seat provides this isolation. Seat suspension systems fall into three categories: passive, semi-active, and active systems.

Passive systems are simple mechanisms that may employ springs and dampers or other isolation devices to reduce the chassis vibration that is transmitted to the driver and operator. These systems may work in limited applications and may in fact amplify low frequency vibration [5].

Active suspension systems are significantly more complex than passive systems and employ sensors, electronic controls, and various damper or motor systems to actively counteract the chassis vibration and isolate the driver and operator [5].

Semi-active systems are not as complex as active systems, but can be very effective at reducing the chassis vibration that is transmitted to the driver and operator [4]. Many of these systems are based on the use of magneto-rheological (MR) fluid in a damper to provide variable damping [4, 5, 9, 14].

In general, the seat suspension isolates the driver from the road input into the chassis of the vehicle. A rigidly mounted seat transfers all of the energy from the road surface into the driver, causing fatigue and discomfort. The frequency ranges where transmission is greatest are from 4.5 to 5.5 Hz and 9.4 to 13.1 Hz [1]. Seat suspensions were developed to reduce the strain on the driver due to whole body vibration. For the greatest effect, the resonant frequency of the seat should be less than the frequency of the vibrations of the chassis which are transmitted to the seat [1]. There are several mechanisms to achieve this state. The simplest is a system of springs mounting the seat to the cab of the vehicle. The next iteration is to add a damper to the system. This could be a passive damper, semi-active damper, or an active damper. The damper could use air, hydraulic, or magneto-rheological (MR) fluid as the working medium.

The purpose of these systems is to isolate the driver and operator from whole-body vibration. There have been numerous studies on the effects of whole body vibration on drivers and equipment operators [1]. The results are fatigue, body aches and pains, and reduced quality of life. It is these factors that have led to the development of suspension systems for seats. The studies show that by improving the ride experience of drivers and operators, the effects are greatly reduced. Drivers are more alert, can drive longer between stops, and have a reduced recovery time. The ISO 2631 part 1 "Evaluation of Human Exposure to Whole-body Vibration" standard is widely used to set allowable exposure based on "comfort reactions to vibration environments," as seen in Table 1 [15].

Table 1: ISO 2631-1 Comfort Reactions to Vibration Environments [15].

Comfort Level	Acceleration Magnitude
Not uncomfortable	Less than 0.315 m/s^2
A little uncomfortable	0.315 m/s^2 to 0.63 m/s^2
Fairly uncomfortable	0.5 m/s^2 to 1 m/s^2
Uncomfortable	0.8 m/s^2 to 1.6 m/s^2
Very uncomfortable	1.25 m/s^2 to 2.5 m/s^2
Extremely uncomfortable	Greater than 2 m/s^2

ISO 2631 part 1 uses these values to provide guidance instead of setting a limit for allowable vibration. Since the reaction to vibration depends on variables including trip duration, purpose, and activities during the trip, the allowable vibration will be different for different vehicles [15].

There are many applications for seat suspension systems in over-the-road trucks, agriculture, and off-highway vehicles. A large market for seat suspension systems is over-the-road trucks with nearly 3 million tractor-trailer combinations on the road in the United States [16].

Two competitors in this industry are the following companies:

ClearMotion, Inc. – Bose Ride™ seating system,

Lord Corporation - MR Seat Suspension systems.

The Bose Ride™ system is an active system that utilizes an electromagnetic actuator to counteract the chassis vibration [2]. This system reduces the vibration to perform better than the best air ride systems and to provide a passenger car-like ride.

Lord Corporation products are used by several system providers and employ an MR damper and related control system to provide a better ride at a much lower cost than a fully active system [17].

Project Goal

The objective of this project was to create a MATLAB/Simulink model that accurately represents a semi-active seat suspension system [18]. The suspension system was to consist of an existing Sears VRS2000 semi-active suspension system [12], three linear dampers utilizing the TracTive DDA valve, and a PID four-channel controller. These are the components of the system that were modeled in MATLAB/Simulink.

The goal of the model is to simulate the response of the system within 10% error, which represents the final tuning of the system required to accurately assess the ride profile the driver/operator experiences. This was done by characterizing the individual components, DDA valve, seat mechanism, and control system through experimentation and laboratory testing. Once testing was completed, the resulting data were compared to the final MATLAB/Simulink model results to determine if the goal of less than 10% error had been achieved.

Review of Literature

In developing seat suspension systems, the universal standard for measuring the performance of these systems is ISO 2631 [14, 15]. Another option for measuring system performance is vibration dose values (VDV) [19]. As indicated by Paddan *et al.* [19], VDV is based on the magnitude, frequency, and duration to calculate total vibration, as shown in Equation (7):

$$VDV\left(\frac{m}{s^{1.75}}\right) = \left[\int_{o}^{T} a(t)^4 dt\right]^{1/4}, \tag{7}$$

where *a(t)* is the frequency weighted acceleration.

With the latest damper technology of variable orifice hydraulic dampers and magneto-rheological (MR) fluid dampers, it is possible to have continuously variable damping [14]. When these mechanisms are modeled in MATLAB/Simulink, they should be modeled using continuously variable damping rates [14].

Roy [14] simulated a system that only included damping in the vertical direction and found that at higher frequencies, a passive damper was more effective than a semi-active damper, because the system was hitting the bump stops at higher frequencies. Roy [14] created a control system that would essentially turn off the semi-active damper at high frequencies, which resulted in better system performance.

Roy [14] states that “The contribution of the longitudinal acceleration to the total acceleration at the driver’s seat is greater than that of the vertical acceleration.” Since the system simulated was only in the vertical axis, this is significant. The system simulated in this project includes damping in the longitudinal axis as well, and should result in better overall performance.

Roy [14] evaluated three different control strategies for the semi-active seat damper. These were the Skyhook law controller, model predictive controller (MPC), and Quadrant law controller.

The Skyhook law controller is based on a concept that originated in the 1970s that uses the sky as a reference. A damper is placed between a sprung mass and the “sky” reference. This concept was initially used for the development of automotive suspensions.

An MPC was developed that used the suspension stroke limits as part of the constraint system [14]. To incorporate the suspension stroke limits, the sprung mass displacement required an added constraint to limit the relative displacement of the sprung mass to be less than the stroke limits of the suspension system [14].

The Quadrant law controller was developed as an alternative to the Skyhook law controller by Rakheja and Sankar [20]. A closed form controller was created based on this concept with the goal of minimizing the sprung mass acceleration [14].

Roy [14] concluded that the MPC was the most effective controller. However, it would also be the most complex and costly to implement, reducing the probability that this type of controller is feasible in a production environment.

In reality, because drivers and operators experience vibration in more than one direction, it is necessary to model the fore-and-aft vibration experienced by drivers and operators. Stein *et al.* [21] researched fore-and-aft vibration in a seat suspension system. They assumed that the vertical and fore-and-aft vibration are independent from each other, so only the fore-and-aft (x-direction) vibration was modeled. In this study, they also found that the bump stops influenced the vibration seen by the driver and operator, and needed to be included in the simulation. Two different seat configurations were tested, including high friction without a hydraulic damper, and low friction with a hydraulic damper. The high friction system had a friction force of 80-85 N and the low friction system had a friction force of 5-10 N [21]. Two performance indicators were used in the study. The first one was the Seat Effective Amplitude Transmissibility factor (SEAT) adapted to the x-axis direction, as shown in Equation (8):

$$SEAT_x = \frac{a_{xwS}}{a_{xwB}} , \quad (8)$$

where

a_{xwS} is the RMS acceleration for the seat in the x-direction,

a_{xwB} is the RMS acceleration for the base in the x-direction.

The second factor is transmissibility, defined by Equation (9):

$$T_{ax} = \frac{a_{xS}}{a_{xB}} , \quad (9)$$

where

a_{xS} is the RMS acceleration at the seat surface,

a_{xB} is the RMS acceleration at the seat base.

The result of the simulation shows that the correlation of the model to the lab results is high for lower frequencies but does not correlate well for higher frequencies (10-15 Hz). In addition, neither seat suspension system made an improvement in vibration in the x-direction. However, Stein *et al.* [21] do state that the model could be used for further suspension development.

With regard to modeling a damper, a paper entitled “Non-linear Characteristic Simulation of Hydraulic Shock Absorbers Considering the Contact of Valves” was reviewed [22]. The paper focuses on the internal structure of the damper and comes to the conclusion that a properly designed MATLAB simulation model can accurately predict the performance of a hydraulic shock absorber [22].

Instead of relying on mathematical formulas to represent internal valve deformation, finite element analysis (FEA) was used to model the internal structure of the dampers to model the valve deflection. This procedure provides a more accurate portrayal of the dynamic performance of the valves within the damper [22].

Liang *et al.* [22] also look at the pressure difference between the compression chamber and the rebound chamber. Damping force is derived by calculating the oil flow between the chambers by using Equation (10):

$$Q = C_d A\sqrt{\frac{2\Delta p}{\rho}}\,, \tag{10}$$

and then using the pressure in each chamber to calculate the damping force using Equation (11):

$$F = A_p P_2 - \left(A_p - A_r\right)P_1 - F_f sin(v)\ . \tag{11}$$

Problem Statement

This project expands on previous work investigating semi-active seat suspension systems by simulating damping in three axes. Does additional damping in the fore-aft and side-to-side directions improve driver comfort? This is the question that this project seeks to address.

Methods

This project involved many disciplines in the modeling of a semi-active suspension for seating systems. The project required mechanical, hydraulic, and electronic control systems knowledge to execute a simulation of the system. From the review of existing literature for this project, it was found that most studies have focused on vibration in one axis only -- typically, the vertical (z-axis) direction. For the model in this project, a system with dampers in three axes is simulated.

In addition, most of the research has been on active damping systems. This project focused on semi-active hydraulic dampers and their control system, thereby filling a knowledge gap that exists for seat suspension system modeling. There is limited research available on semi-active systems [23, 24, 25, 26, 27, 28].

This project started with a base simulation model of the seat suspension and control system with a single damper mounted in the z-axis and has been expanded to include three dampers mounted in the z, x, and y axes. Within this system model is a model for each of the damper systems using a Simscape block to represent the TracTive DDA valve, as shown in Figure 8. In addition to the transfer function representing the TracTive DDA valve, other key elements are the variable translational damper and translational spring. Other elements used are force and motion sensors for monitoring and control of the system model.

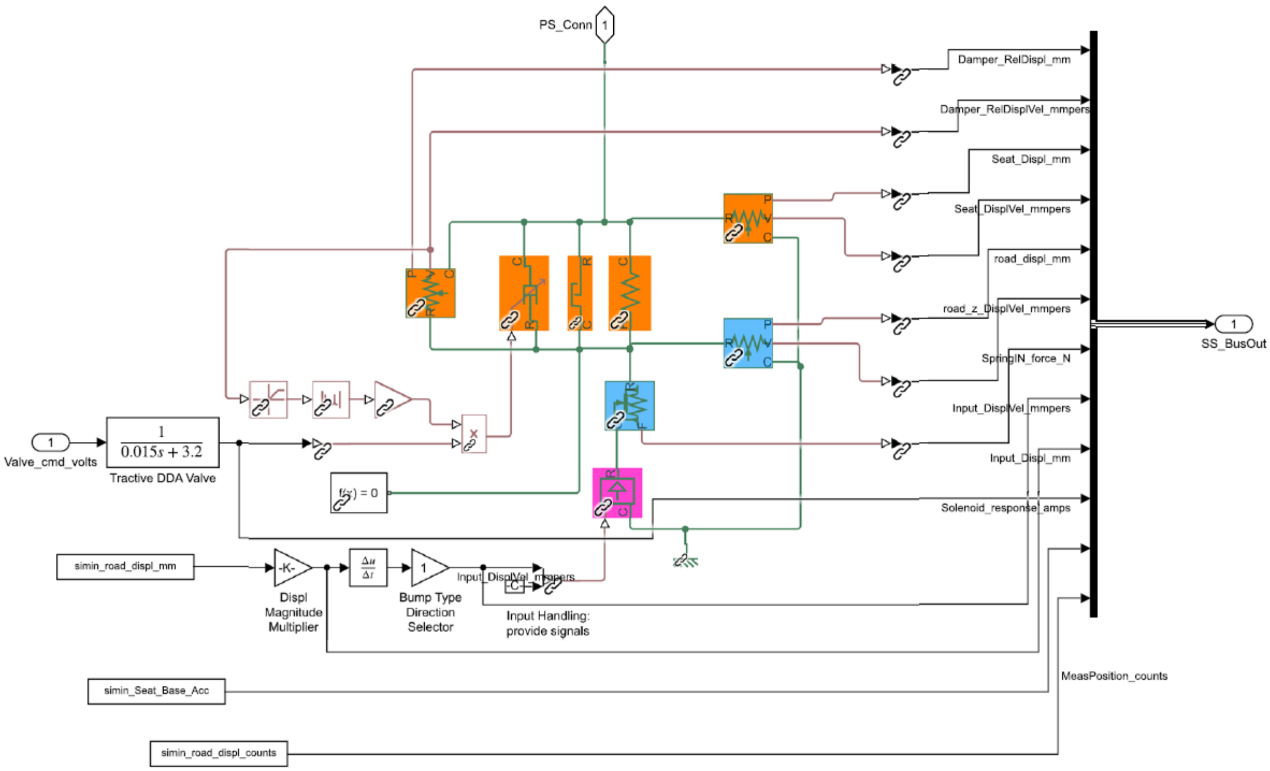

Figure 8: Simscape Model of Damper System.

This model is a sub-system in the larger simulation model which includes three damper sub-systems, inputs, the controller, and outputs.

Results and Discussion

To characterize the damper performance, a prototype damper was made using the TracTive DDA valve and tested on a shock dynamometer at Hayes Performance Systems [29]. Figure 9 shows the damping force versus velocity for applied currents from 0.5 amps to 2.0 amps.

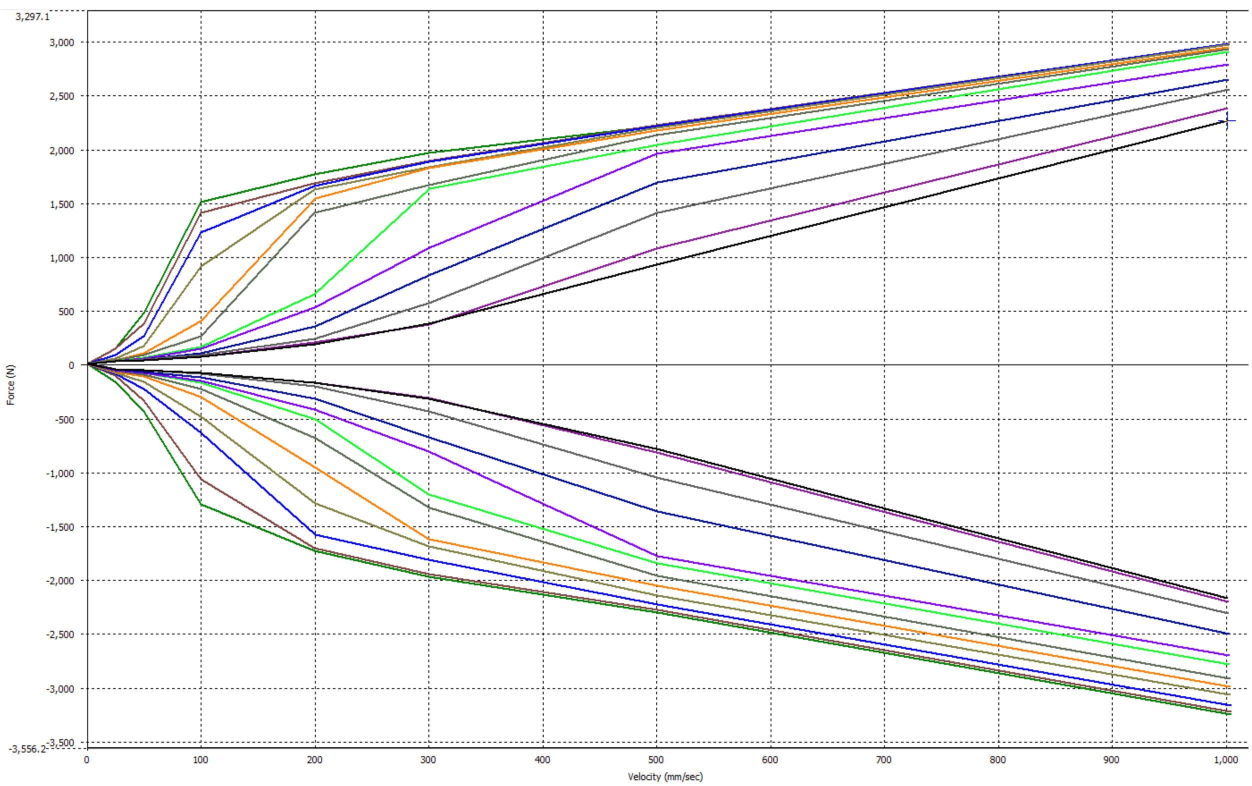

Figure 9: Force (N) versus Velocity (mm/s) of the TracTive DDA Valve in a Prototype Damper.

As seen in Figure 9, the practical range of operating current is 0.6 amps to 1.7 amps. When comparing these data to the data provided by TracTive, it is seen that the two sets of data are very similar. For the next stage of testing, input currents of 0.5 amps and 1.5 amps were selected. Holding the current constant, the shock absorber was tested at various velocities to evaluate the damping forces for a given piston velocity. Figures 10 and 11 show the force versus displacement at various velocities for currents of 1.5 and 0.5 amps.

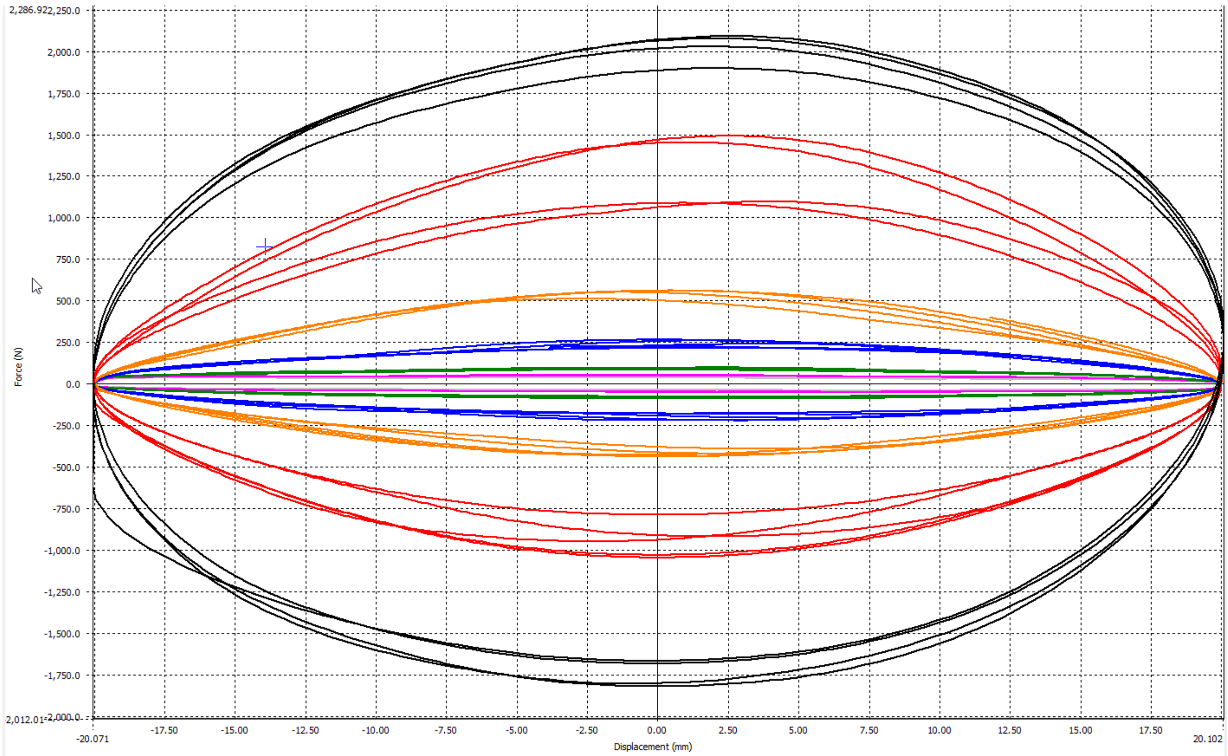

Figure 10: Force (N) versus Displacement (mm) at 1.5 Amps for Various Velocities (mm/s).

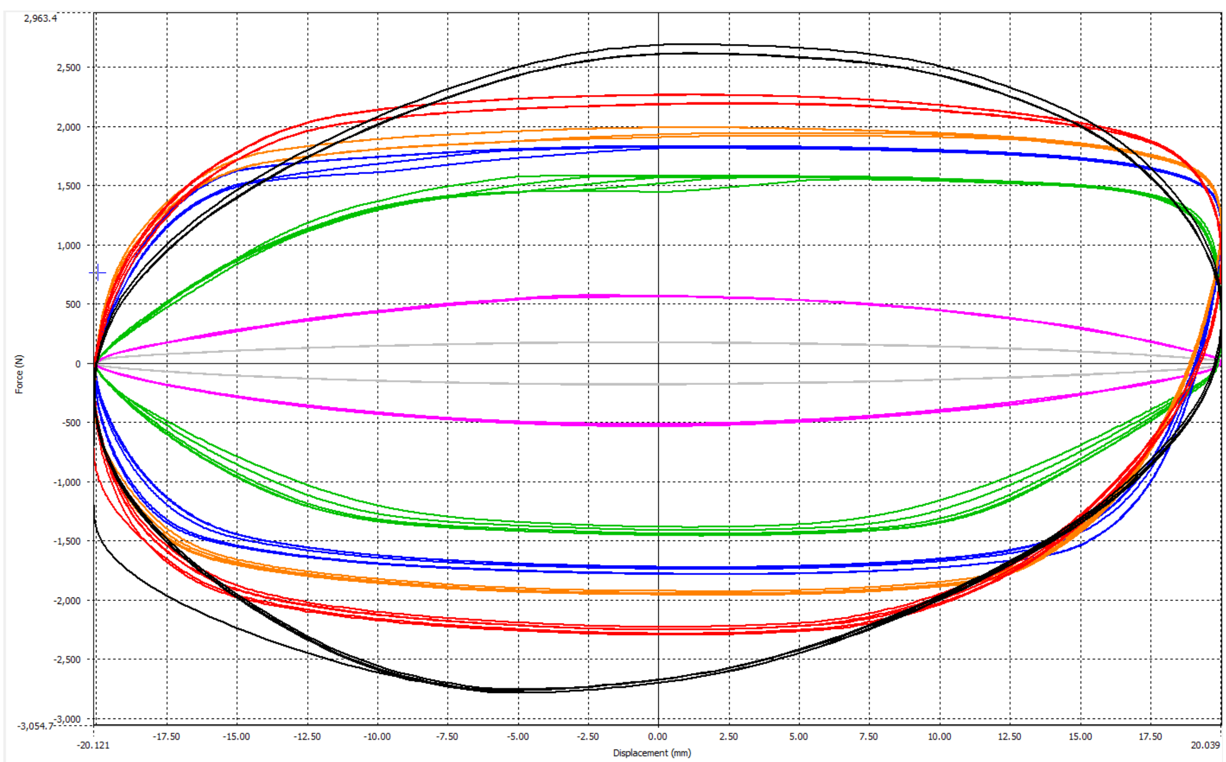

Figure 11: Force (N) versus Displacement (mm) at 0.5 Amps for Various Velocities (mm/s).

Looking again at the damping force versus the piston velocity for a given current, Figure 12 shows the performance at input currents from 0.5 amps to 1.8 amps in 0.1 amp increments for velocities from 0 to 1 m/s. Again, at 0.5 amps or less, the valve is fully closed and results in maximum damping. At 1.8 amps, the valve is nearly fully open and results in minimal damping force in addition to the damping force due to the piston and shim stack.

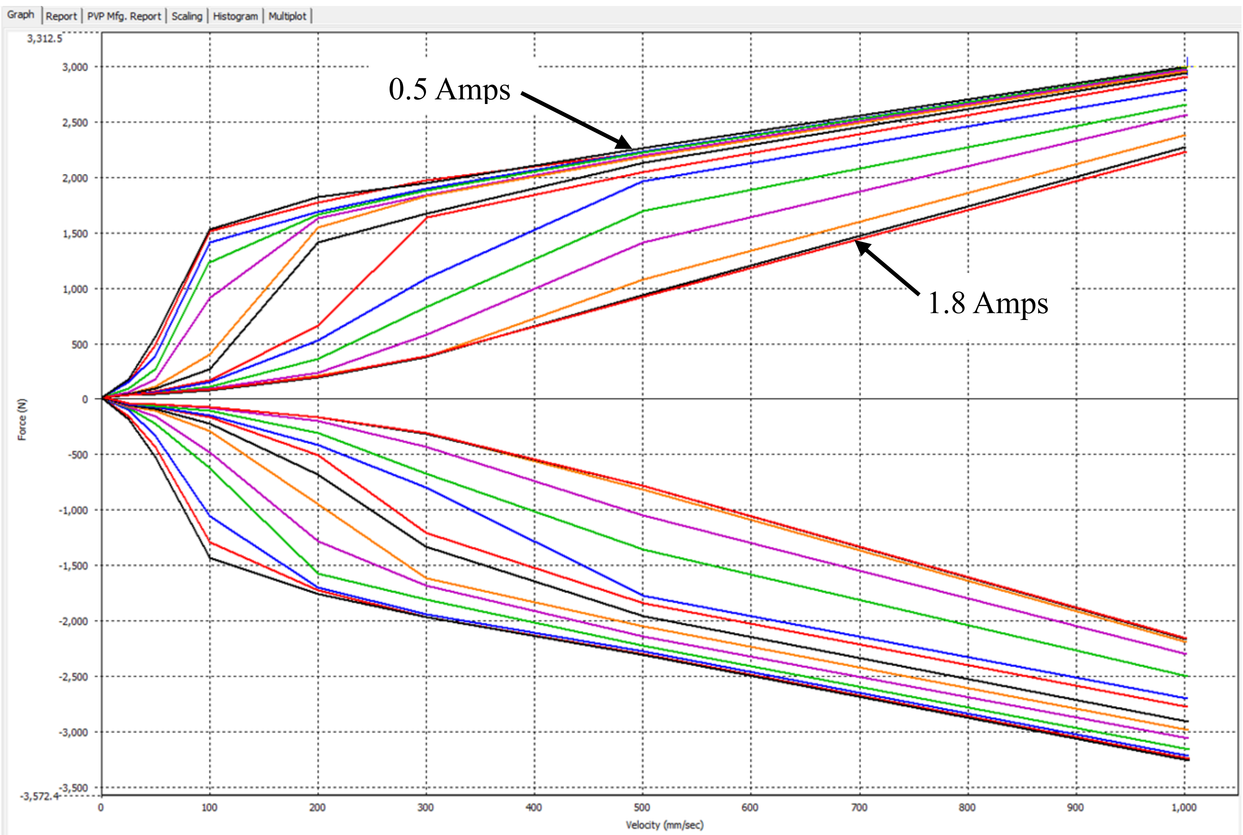

Figure 12: Force (N) versus Velocity (mm/s) for Various Valve Currents.

To look at this closer, Figure 13 zooms in to values for velocities for 0 to 300 mm/s.

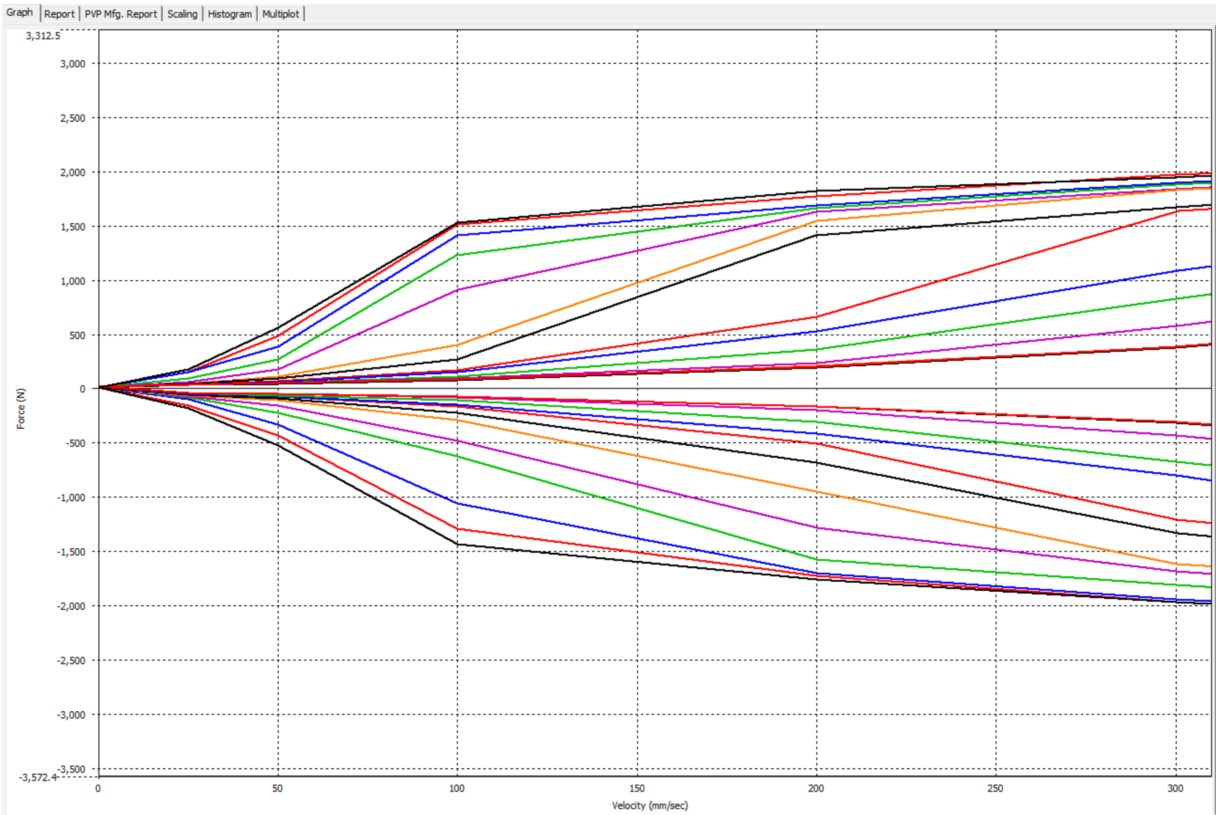

Figure 13: Force (N) versus Velocity (mm/s) for Various Valve Currents.

Since the DDA valve is essentially a solenoid, it is treated as an RL series circuit for the MATLAB model. In the model, the DDA valve is represented as a transfer function. Using Kirchhoff's Voltage Law (KVL) to define the equation representing the DDA valve, Equation (12) is derived:

$$v_{in}(t) = L\frac{di(t)}{dt} + Ri(t) \qquad (12)$$

where

v_{in} (t) is the voltage applied to the DDA valve,

L is the inductance of the DDA valve,

R is the resitance of the DDA valve,

i(t) is the current flowing through the DDA valve coil.

Applying Laplace transform theorem to each term yields Equation (13):

$$V(s) = LsI(s) + RI(s)\ . \tag{13}$$

Further symplifying the equation results in Equation (14):

$$V(s) = I(s)(Ls + R)\ . \tag{14}$$

Then the transfer function is realized in Equation (15):

$$H(s) = \frac{I(s)}{V(s)} = \frac{1}{Ls+R}\ . \tag{15}$$

Empirical data were used to further derive the transfer function that is used in the MATLAB model. The inductance of the Tractive DDA valve is 0.015 H and the resistance is 3.2 Ohms. This results in the transfer fuction shown in Equation (16):

$$H(s) = \frac{1}{0.015s+3.2}\ . \tag{16}$$

This transfer function has been used to represent the DDA valve in the model of the damper.

When considering environmental influences on the system, the most significant would be the change in viscosity of the damper fluid. The environment the system would be operating in is protected from weather inside a cab or operator compartment. Therefore, the exposure to cold temperatures would be limited. When looking at the viscosity chart in Figure 14, it is seen that for temperatures above 50° F, the curve is fairly linear. Since the steady state system temperature is typically expected to be higher than 50° F, temperature was ruled out as an input to the simulation model. Since the damper fluid is used to cool the solenoid in the DDA valve, the temperature of the fluid will always be higher than 50°F, and will remain in the linear range of the viscosity curve.

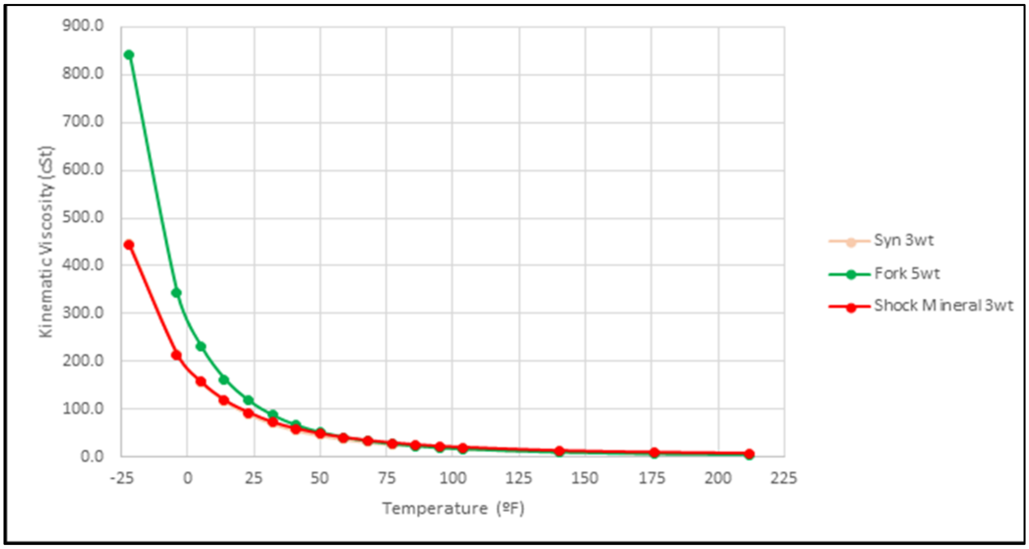

Figure 14: Viscosity versus Temperature (°F) for Damper Fluid.

To evaluate the accuracy of the model, empirical data from Z-axis testing were used. The input data used were from the AG3 vibration profile, as shown in Figure 15, with a 75 kg water bag used to simulate the operator.

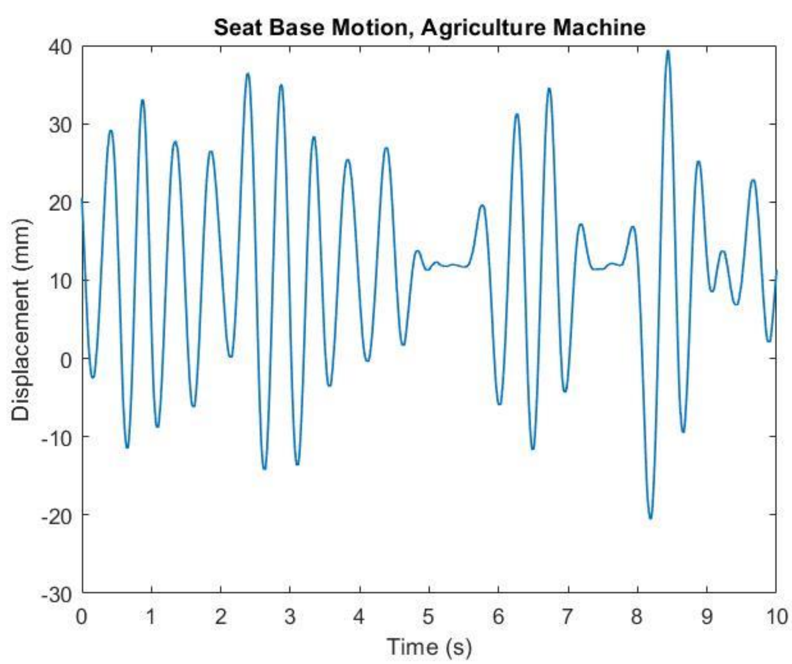

Figure 15: AG3 Vibration Profile in Z-axis for Agricultural Machine.

Results from the MATLAB model are shown in Figure 16. The graph in Figure 16 is the z-axis model output for the AG3 vibration profile input with a 75 kg operator. The dashed black line is the input AG3 vibration profile as shown in Figure 15. The solid gray line is the seat's response to the input vibration and the green line is the result that the operator experiences.

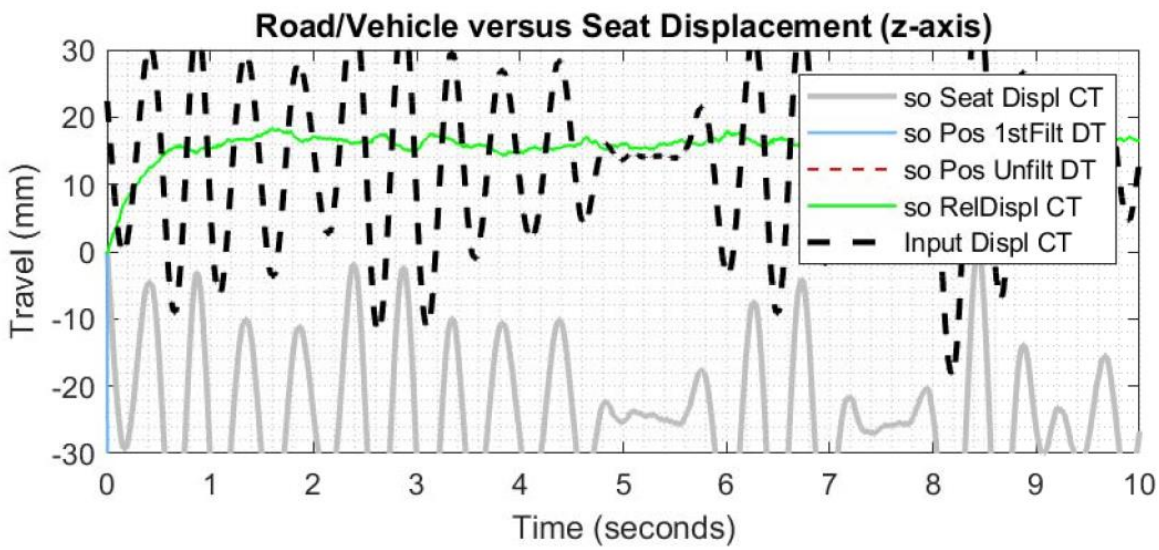

Figure 16: MATLAB Model Output Using AG3 Vibration Profile in Z-axis for Agricultural Machine.

The calculated SEAT measurement from the MATLAB model was 0.5155. Results of laboratory testing on a vibration table using the AG3 z-axis profile for the input included a SEAT value of 0.682. This indicates that the simulation is anticipating more damping than seen in laboratory testing.

Additional profiles were examined in the z-axis only based on available input data. Some vibration profiles had available laboratory data for comparison while others did not. The next profile examined was EM6 for a tracked machine, as shown in Figure 17.

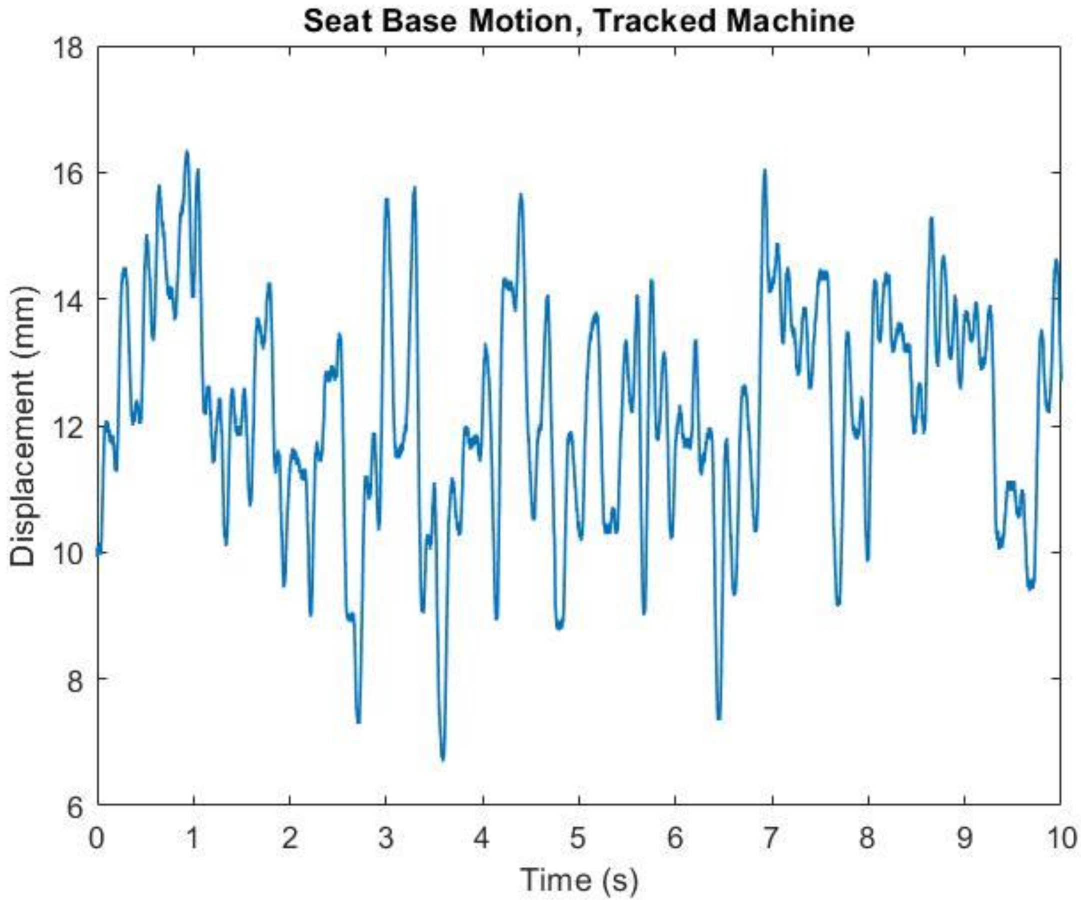

Figure 17: EM6 Vibration Profile in Z-axis for Tracked Machine.

Figure 18 shows the output profile from the MATLAB model for the z-axis based on an input of the EM6 tracked machine vibration profile, as shown in Figure 17. Again, the black dashed line is the EM6 vibration profile as shown in Figure 17, the solid gray line is the seat's reaction to the input vibration and the green line is the resultant vibration experienced by the operator. The SEAT value experienced in laboratory testing for this system using the EM6 vibration profile for an input was 0.250, indicating that this control system can effectively damp the EM6 vibration profile.

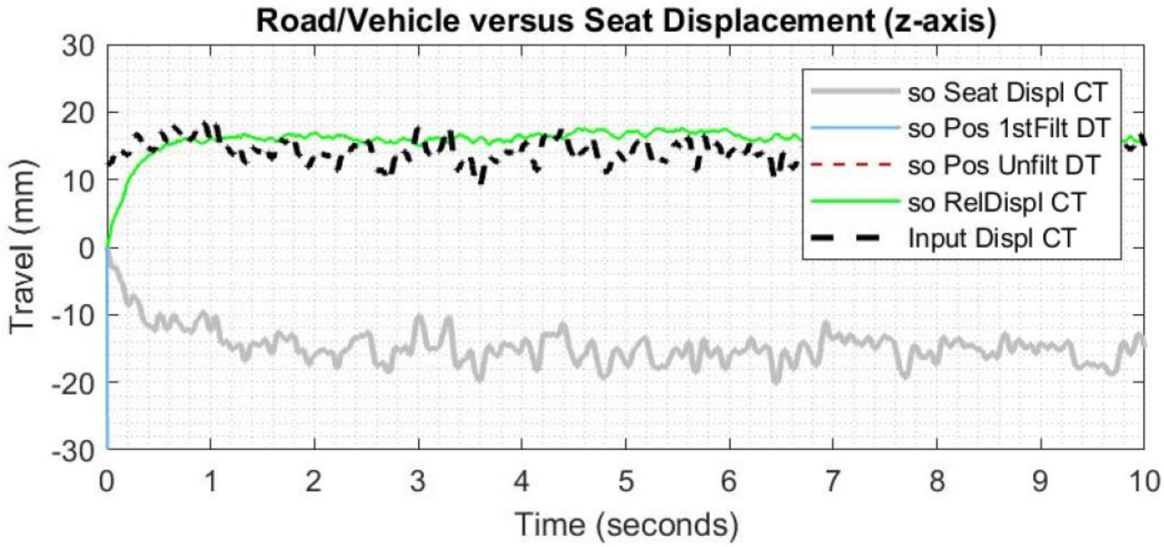

Figure 18: MATLAB Model Output Using EM6 Vibration Profile in Z-axis for Tracked Machine.

Figure 19 shows the J25TR5 z-axis vibration profile for a highway truck, smooth. When used as input to the MATLAB model, Figure 20 is the resulting output. The green line would indicate that the control system can effectively damp the vibration input from the J25TR5 profile to isolate the operator from vibration. No laboratory testing was performed for the J25TR5 vibration profile.

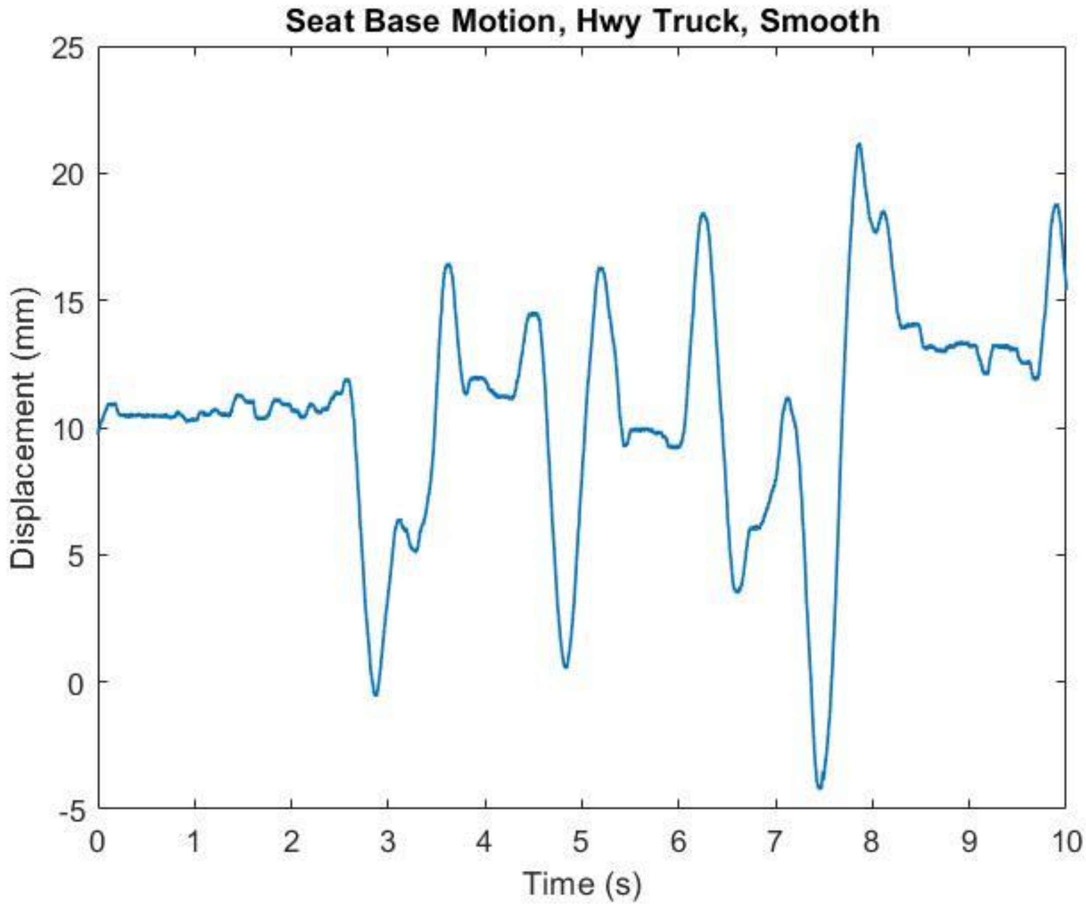

Figure 19: J25TR5 Vibration Profile in Z-axis for Highway Truck, Smooth.

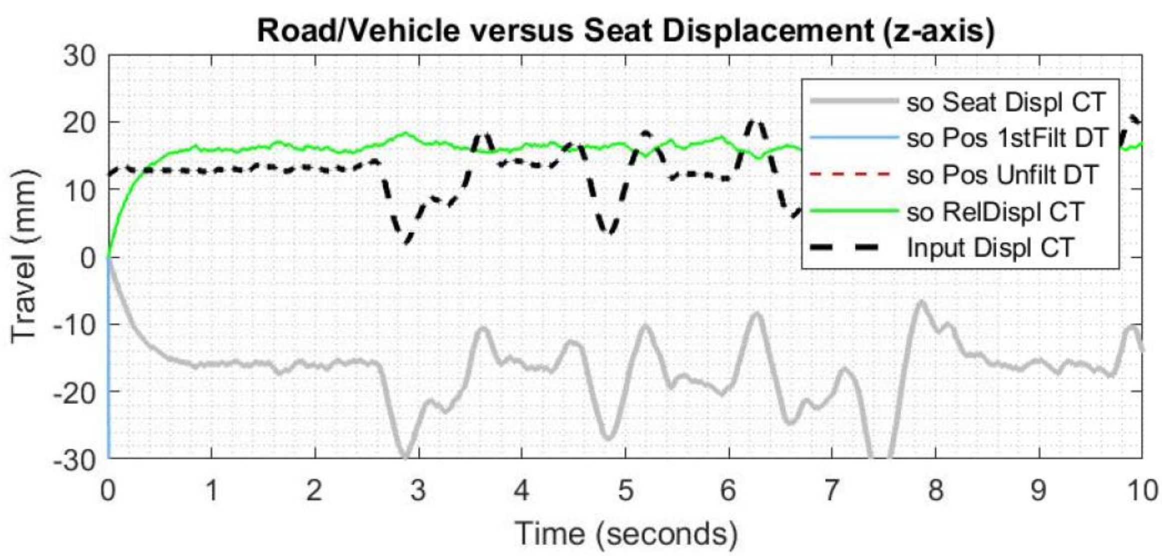

Figure 20: MATLAB Model Output Using J25TR5 Vibration Profile in Z-axis for Highway Truck, Smooth.

Figure 21 shows the JR5TR2 z-axis vibration profile for a highway truck, medium. When used as input to the MATLAB model, the resulting output is shown in Figure 22. The results indicate that the control system effectively damps the vibration from the JR5TR2 profile. Vibration experienced by the operator as represented by the green line in Figure 22 is of lesser magnitude than the JR5TR2 input profile represented by the dashed black line. Again, for this profile, no laboratory testing was performed.

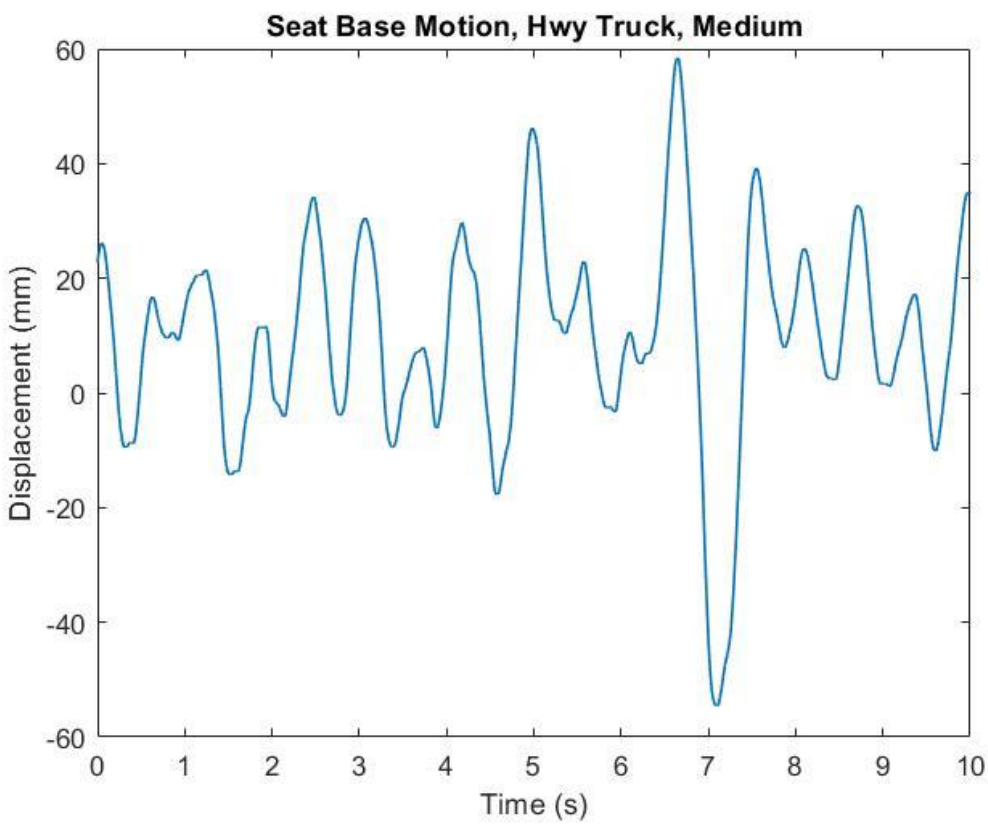

Figure 21: JR5TR2 Vibration Profile in Z-axis for Highway Truck, Medium.

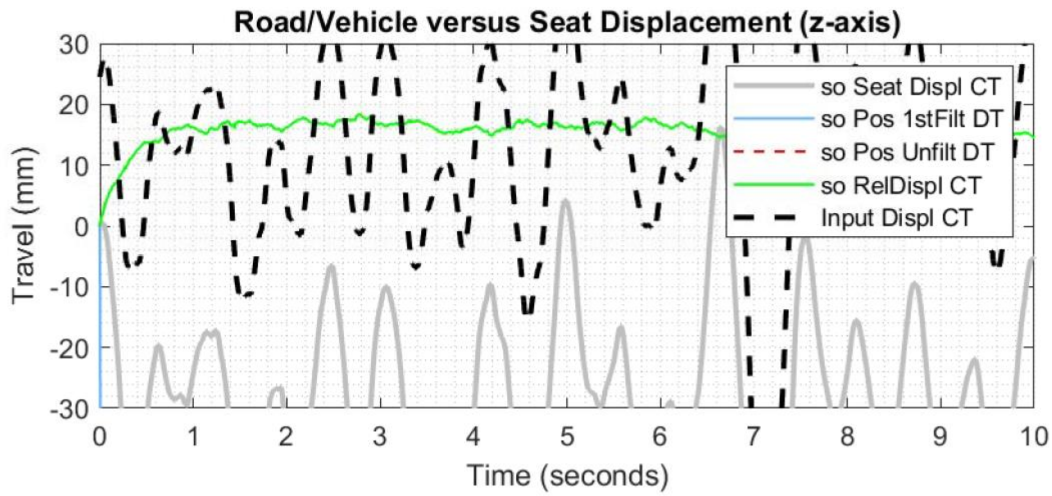

Figure 22: MATLAB Model Output Using JR5TR2 Vibration Profile in Z-axis for Highway Truck, Medium.

Figure 23 shows the J51TR1 z-axis vibration profile for a highway truck, rough. When using this profile as input to the MATLAB model, the results are as shown in Figure 24. The graph in Figure 24 indicates that the control system is able to effectively damp the vibration from the J51TR1 profile resulting in the vibration experienced by the operator as respresented by the green line. No laboratory testing was performed using this input profile.

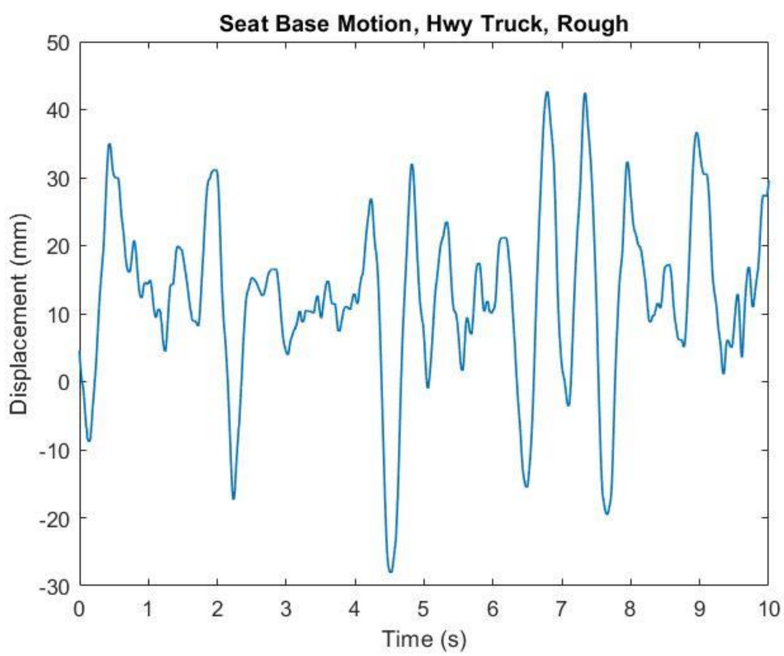

Figure 23: J51TR1 Vibration Profile in Z-axis for Highway Truck, Rough.

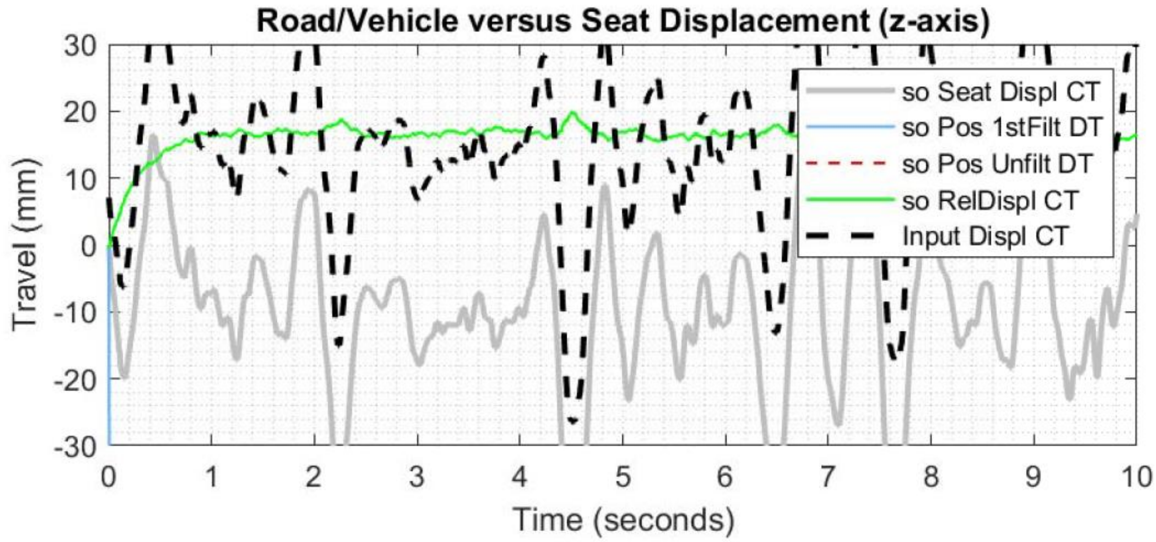

Figure 24: MATLAB Model Output Using J51TR1 Vibration Profile in Z-axis for Highway Truck, Rough.

The only profile for which 3-axis data were available is the J28TR2 vibration profile for a backhoe loader. Figure 25 shows the complete dataset for the J28TR2 z-axis vibration. For this project and simulation model, the time period is limited to the first 10 seconds of data, which are shown in Figure 26.

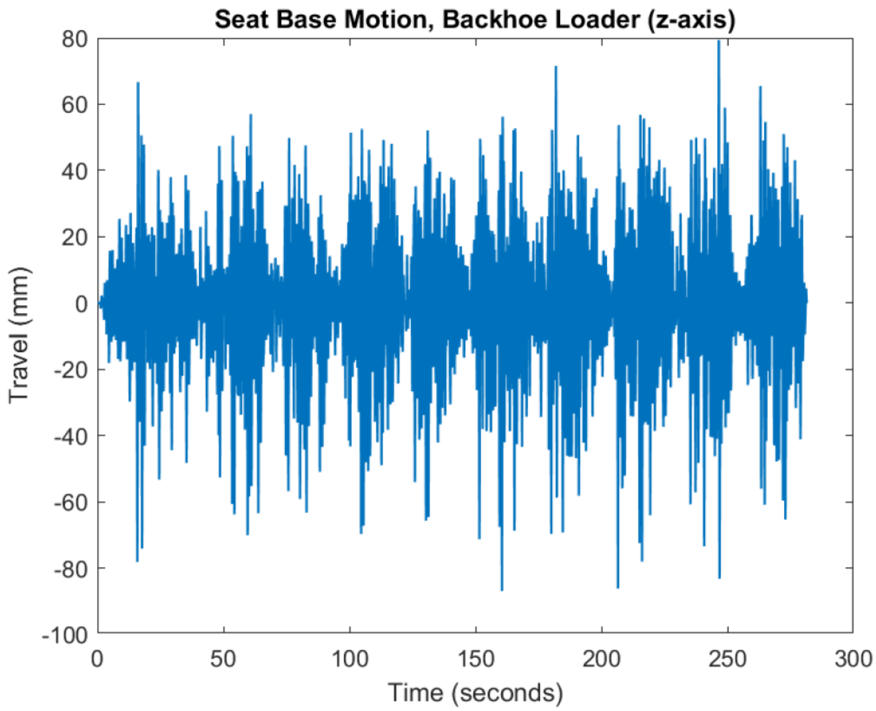

Figure 25: J28TR2 Vibration Profile in Z-axis for Backhoe Loader.

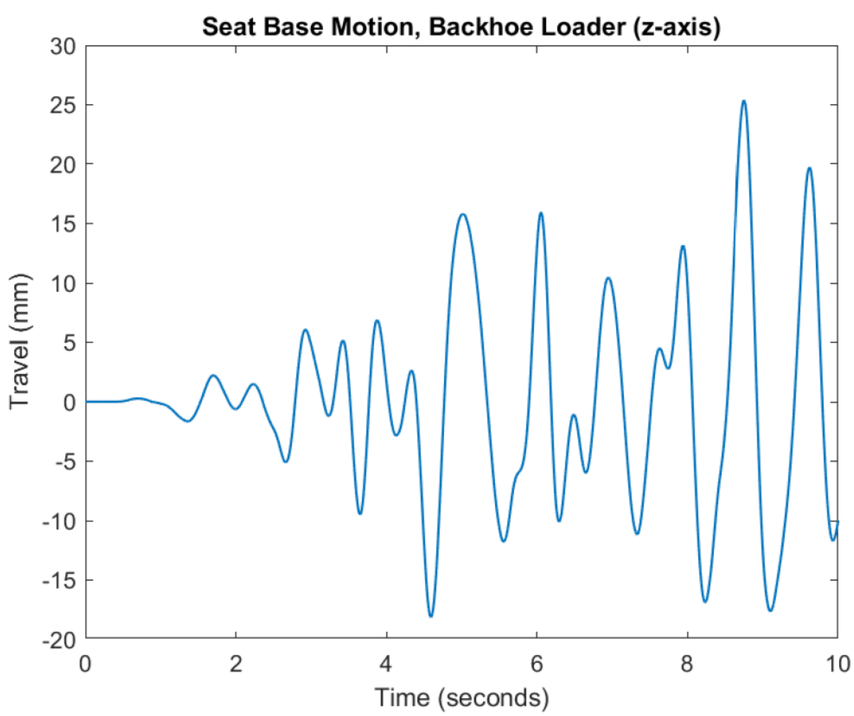

Figure 26: J28TR2 Vibration Profile in Z-axis for Backhoe Loader.

Using the data shown in Figure 26 as input to the MATLAB simulation model, the output is shown in Figure 27.

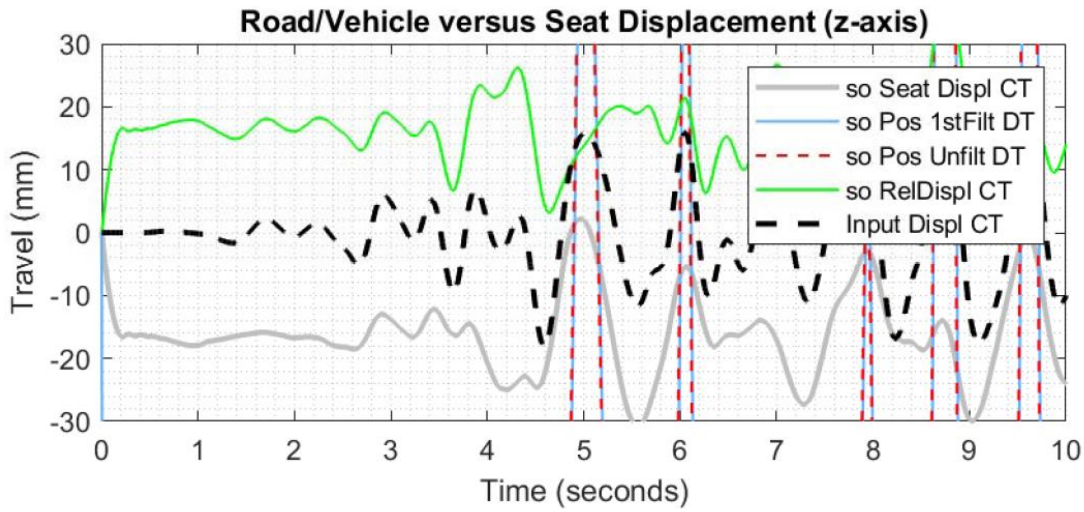

Figure 27: MATLAB Model Output Using J28TR2 Vibration Profile in Z-axis for Backhoe Loader.

The MATLAB model results shows that for the J28TR2 vibration profile, the system is not as effective at damping the vibration. This is seen in the resulting vibration the operator experiences as represented by the green line in Figure 27. The laboratory result for this profile was a SEAT value of 0.868.

Figure 28 shows the complete dataset for the J28TR2 x-axis vibration and Figure 29 shows the first 10 seconds used in the MATLAB simulation model.

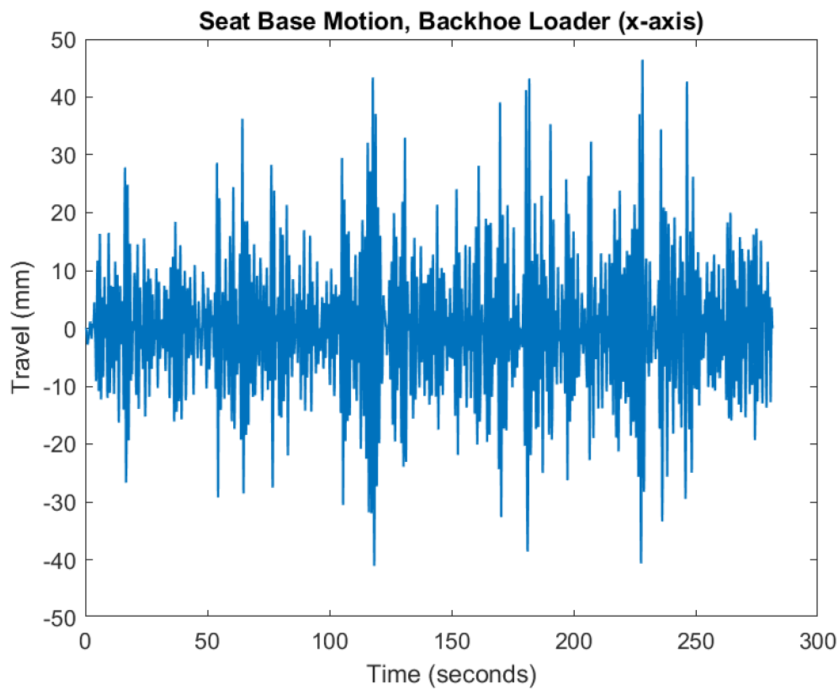

Figure 28: J28TR2 Vibration Profile in X-axis for Backhoe Loader.

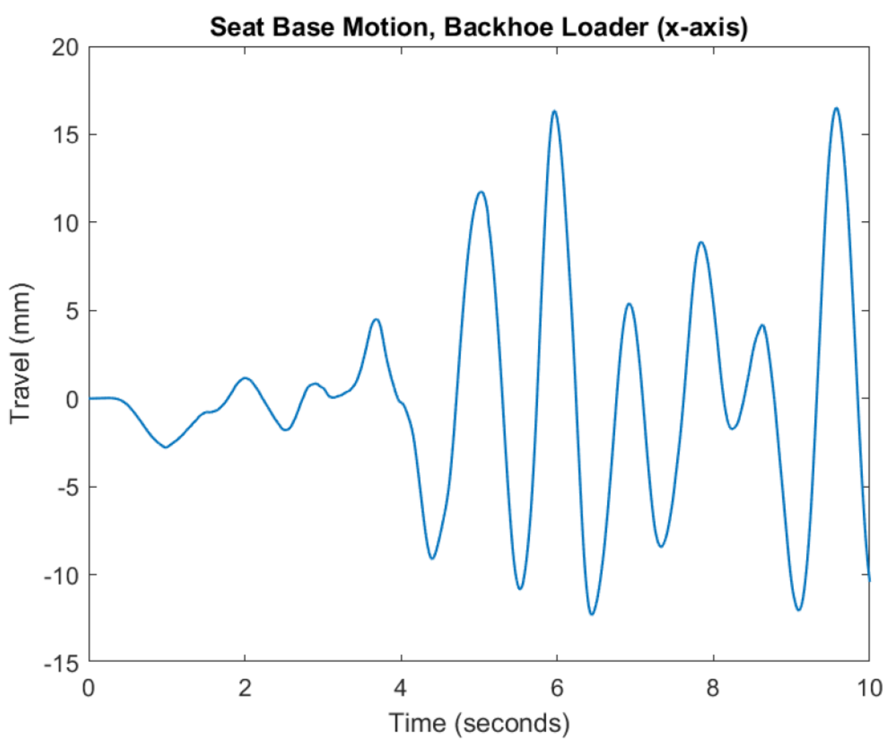

Figure 29: J28TR2 Vibration Profile in X-axis for Backhoe Loader.

The output from the MATLAB simulation model using the J28TR2 x-axis vibration profile is seen in Figure 30. Again the model results would indicate that the control system is not as effective at damping the vibration from the J28TR2 in the x-axis. No laboratory data are availble for the x-axis of this profile.

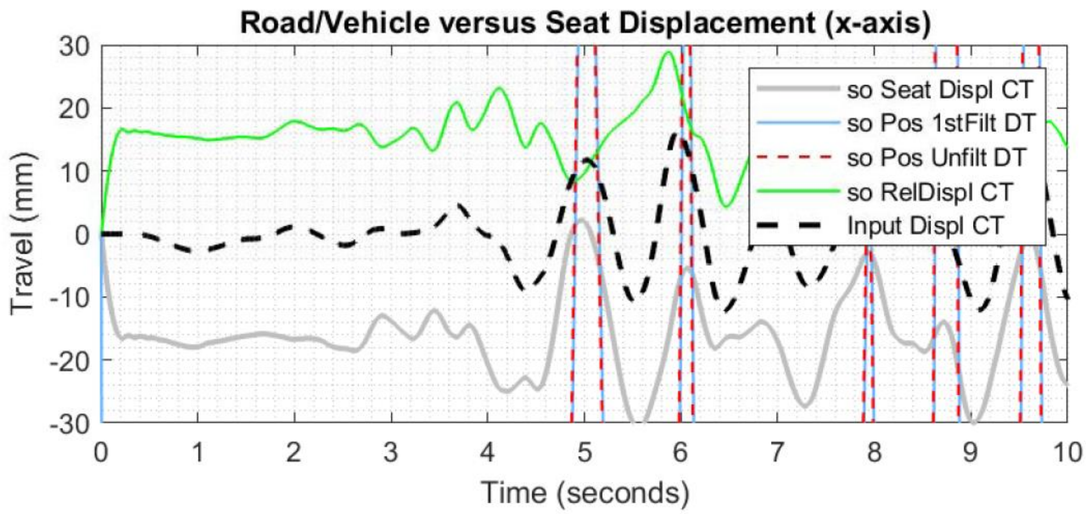

Figure 30: MATLAB Model Output Using J28TR2 Vibration Profile in X-axis for Backhoe Loader.

Figure 31 shows the complete dataset for J28TR2 Y-axis vibration and Figure 32 shows the first 10 seconds as used in the MATLAB simulation model.

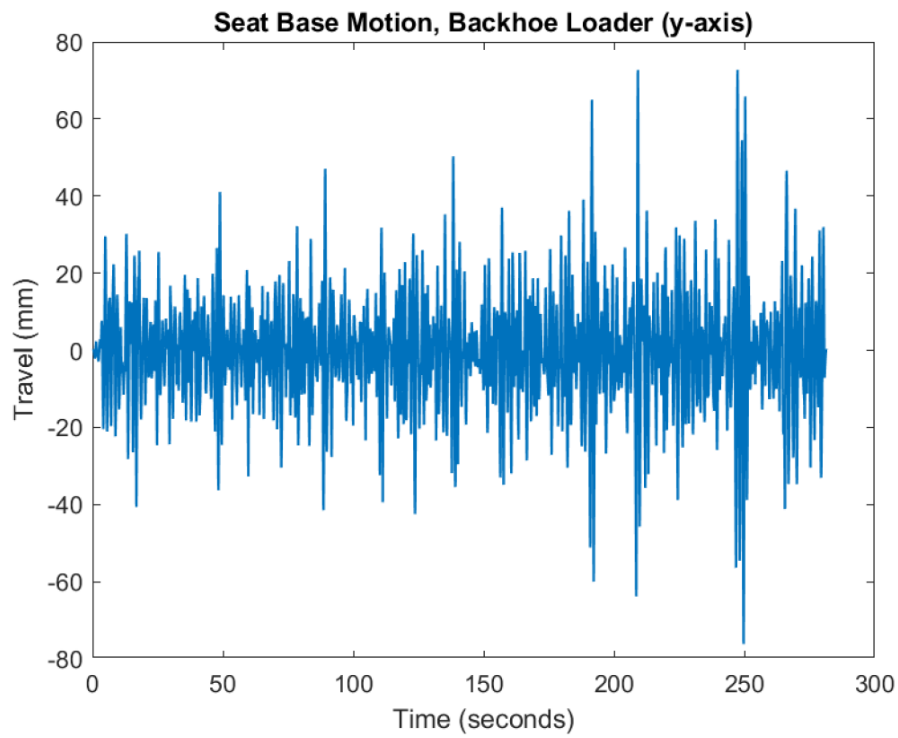

Figure 31: J28TR2 Vibration Profile in Y-axis for Backhoe Loader.

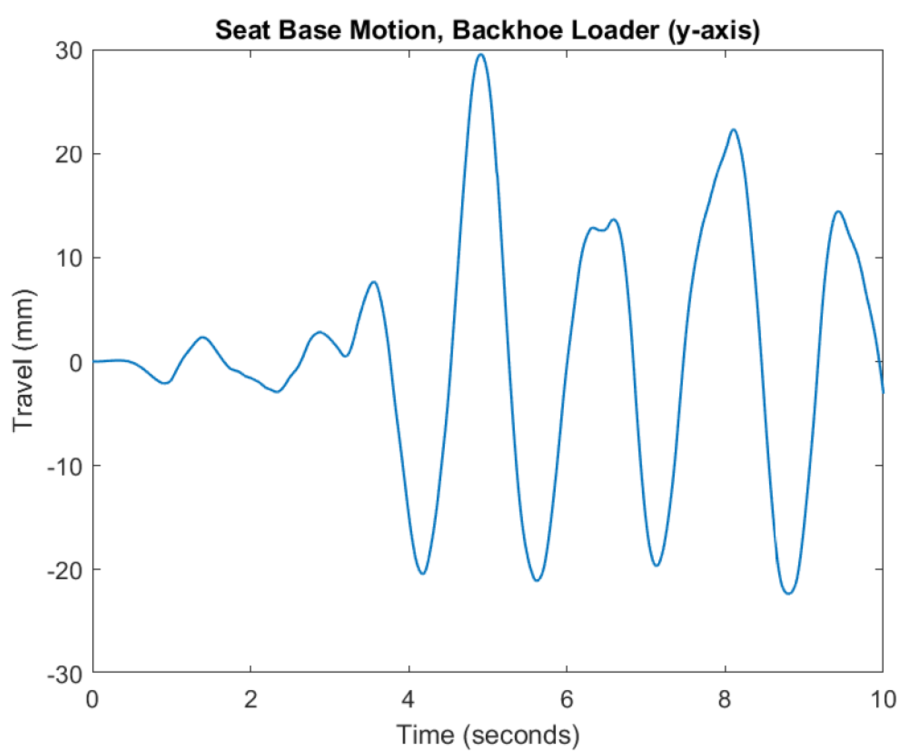

Figure 32: J28TR2 Vibration Profile in Y-axis for Backhoe Loader.

The output from the MATLAB simulation model using the J28TR2 y-axis data is shown in Figure 33. Reviewing these results, it is also seen that the control system is not as effective at damping the vibration from the J28TR2 profile in the y-axis. No laboratory data are available for the J28TR2 y-axis profile.

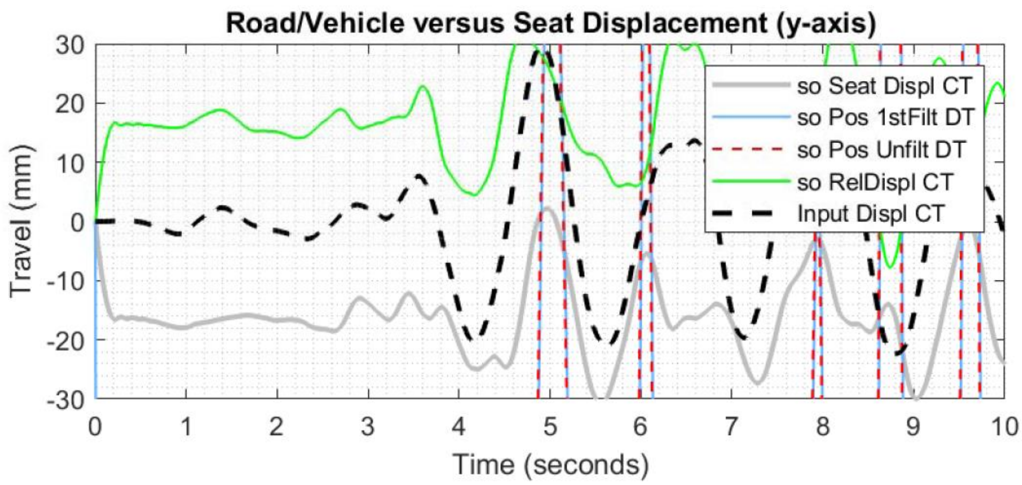

Figure 33: MATLAB Model Output Using J28TR2 Vibration Profile in Y-axis for Backhoe Loader.

Conclusions and Recommendations

From the initial results of the damper testing, it is seen that the actual performance of the TracTive DDA valve is close to the expected performance based on data provided by TracTive for similar applications. However, when simulated in MATLAB, the control system effectiveness represented by the model decreases as the severity of the vibration increases.

For the least severe vibration profile, AG3 for an agricultural machine, it is seen that the model results indicate that the control system can effectively damp the vibration. This is confirmed by the laboratory result of a SEAT value of 0.682.

When reviewing the MATLAB model results for additional z-axis vibration profiles, it is seen that the control system can effectively damp the vibration input to the system. There are limited laboratory data available to confirm these results.

For the only vibration profile for which all three axes of data were available, the J28TR2 profile, the MATLAB model indicates that the control system is not as effective in damping the vibration. The laboratory result of a SEAT value of 0.868 in the z-axis would indicate that the control system is not as effective at damping vibration from this profile compared to the laboratory results for the AG3 vibration profile.

With the limited laboratory data available, it is difficult to determine the accuracy of the model. When looking at the relative results, the MATLAB model correctly identifies that the control system is not as effective at damping vibration from the J28TR2 when compared to the other profiles. The laboratory results confirm this trend when comparing SEAT values. Additional laboratory results should be reviewed to determine the effectiveness of the model. A review of the resulting relative vibration profile of the seat for each of the input profiles should be done to compare to the model results. This may be able to better quantify the effectiveness of the MATLAB model at predicting laboratory results and actual performance of the seat suspension system.

When considering the potential error between the simulation model and laboratory testing, what is not taken into account is the shim stack and piston design inside the damper. These components can have a significant influence on the performance of the damper and were not simulated in this model design. Further work to simulate the internal design of the damper could increase the fidelity of this model [22].

Another factor is the design of the control system. For this project, a simple PID controller is used. The MATLAB model indicates that this control system is not effective for all input vibration profiles. To improve the accuracy of the model and performance of the system across all vibration inputs, a more complex control system could be modeled [14, 20, 30].

In addition, this project looked at the three axes independently. Further work should consider the interrelationships between the three axes and developing a control system to account for the interdependencies. This could potentially improve the predictive ability of the model with regard to what the machine operator actually experiences.

References

[1] Teschke, K., Nicol, A.-M., Davies, H., & Ju, S. (1999, April 14). Whole body vibrations and back disorders among motor vehicle drivers and heavy equipment operators: a review of the scientific evidence [R]. https://dx.doi.org/10.14288/1.0048193

[2] Parison, J. (2010, January 27). The Bose Ride™ system [PDF document]. Retrieved from http://cdn2.hubspot.net/hub/460706/file-2335930475-pdf/files/Bose-Ride-system-Technical-Overview-White-Paper.pdf?t=1423677098810

[3] Ryba, D. (1992, Nov. 10). Semi-active suspension for a seat with a controlled electromagnetic damper. *Periodica Polytechnica.Transportation Engineering, 21*(3), 247-264.

[4] Maciejewski, I. & Krzyzynski, T. (2011). Control design of semi-active seat suspension systems. *Journal of Theoretical and Applied Mechanics 49*(4), 1151-1168.

[5] Ning, D. (2016-12-01). An active seat suspension design for vibration control of heavy-duty vehicles. *Journal of low frequency noise and vibration*, *35*(4), 264-278. https://doi.org/10.1177/0263092316676389

[6] Pan, G. Y., & Hao, X. L. (2011). Research on active control of driver's seat suspension system. *Applied Mechanics and Materials*, *701*, 105-107 https://doi.org/10.4028/www.scientific.net/AMM.105-107.701

[7] Pan, G. Y., & Hao, X. L. (2011). Research on modeling and simulation of active seat suspension system based on LQG control, *2011 International Conference on Remote Sensing, Environment and Transportation Engineering*, 656-659. https://doi.org/10.1109/RSETE.2011.5964362

[8] Pan, G. Y., Yang, H., & Yong, T. L. (2013). Study of commercial vehicle active seat suspension. *Applied Mechanics and Materials*, *441*, 641-644. http://doi.org/10.4028/www.scientific.net/AMM.441.641

[9] Valero, B. (2008). *Testing, control and modeling of a pneumatic active seat suspension and ride comfort evaluation* (Doctoral dissertation). Retrieved from ProQuest Dissertations & Theses Global. (Accession No. 304327462).

[10] Tractive Suspension. (2016, March 23). The art of suspension: DDA valve [PDF document]. Retrieved from https://www.gl-suspension.de/app/download/5256407366/Tractive+DDA+valve+23-3-2016.pdf?t=1506455536

[11] Lindeman, E. [Inventor]. (2015, October 27). Continuously variable damper device. US Patent 9,168,808. Washington, DC: U.S. Patent and Trademark Office.

[12] Sears Seating. (2018). Semi-active suspension system. Retrieved from https://www.searsseating.com/technology/innovations/

[13] Yan, Z., Zhu, B., Li, X., & Wang, G. (2015). Modeling and analysis of static and dynamic characteristics of nonlinear seat suspension for off-road vehicles. *Shock and Vibration*, *2015*, 1-13. https://doi.org/10.1155/2015/938205

[14] Roy, J. (2015). *Design of cab suspensions and semi-active seat damping control strategies for tractor semi-trailers* (Doctoral dissertation). Retrieved from ProQuest Dissertations & Theses Global. (Accession No. 1719458997).

[15] International Organization for Standardization. (1997). *Mechanical vibration and shock – evaluation of human exposure to whole-body vibration – Part 1: General requirements.* (ISO Standard No. 2631). Retrieved from http://www.iso.org/standard/7612.html

[16] Bureau of Transportation Statistics. (2017). *Freight Facts and Figures 2017*. Retrieved from https://www.bts.gov/product/freight-facts-and-figures

[17] Lord Corporation. (2018). *Seat Suspensions*. Retrieved from https://www.lord.com/products-and-solutions/active-vibration-control/industrial-suspension-systems/seat-suspensions

[18] The MathWorks Inc. *Simscape*. Retrieved from https://www.mathworks.com/products/simscape.html

[19] Paddan, G.S., & Griffin, M.J. (2002). Evaluation of whole-body vibration in vehicles, *Journal of Sound and Vibration, 253*(1), 195-213. https://doi.org/10.1006/jsvi.2001.4256.

[20] Rakheja S., & Sankar S. (1985). Vibration and shock isolation performance of a semi-active "On-Off" damper. *Journal of Vibration, Acoustics Stress and Reliability in Design 107*(4), 398-403. https://doi.org/10.1115/1.3269279.

[21] Stein, G.J., Zahoranský, R., Gunston, T.P., Burström, L. & Meyer, L., (2008). Modelling and simulation of a fore-and-aft driver's seat suspension system with road excitation, *International Journal of Industrial Ergonomics*, *38*(5–6), 396-409. https://doi.org/10.1016/j.ergon.2007.10.016.

[22] Liang, L, Xiaohong, Z, Manlong, P & Gang, Q. (2012) Non-linear Characteristic Simulation of Hydraulic Shock Absorbers Considering the Contact of Valves. *The 2nd International Conference on Computer Application and System Modeling (2012)*. 1132-1134.

[23] Wang, W.L., & Xu, G.L. (2009). Fluid formulae for damping changeability conceptual design of railway semi-active hydraulic dampers. *International Journal of Non-Linear Mechanics, 44*(7), 809-819. https://doi.org/10.1016/j.ijnonlinmec.2009.05.003

[24] Els, P.S., Theron, N.J., Uys, P.E., & Thoresson, M.J. (2007). The ride comfort vs. handling compromise for off-road vehicles. *Journal of Terramechanics, 44*(4), 303-317. https://doi.org/10.1016/j.jterra.2007.05.001

[25] Jahanpour, J., Porghoveh, M., & Ilbeigi, S. (2016). Forced vibration analysis of a system equipped with a Nonlinear Displacement-Dependent (NDD) damper. *Scientia Iranica. Transaction B, Mechanical Engineering, 23*(2), 633-647. https://doi.org/10.24200/sci.2016.3850

[26] Giliomee, C.L., & Els, P.S. (1998). Semi-active hydropneumatics spring and damper system. *Journal of Terramechanics, 35*(2), 109-117. https://doi.org/10.1016/S0022-4898(98)00016-0

[27] M. Zapateiro, M., Luo, N., Karimi, H.R., & Vehi, J. (2009). Vibration control of a class of semiactive suspension system using neural network and backstepping techniques. *Mechanical Systems and Signal Processing, 23*(6), 1946-1953. https://doi.org/10.1016/j.ymssp.2008.10.003

[28] R.N. Yerrawar, R.N., & Arakerimath, R.R. Performance assessment and control policies for semiactive suspension using SIMSCAPE. In *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT).* https://doi.org/10.1109/icacdot.2016.7877769

[29] Hayes Performance Systems. Retrieved from www.hayesbrake.com.

[30] Rajendiran, S., & Lakshmi, P. (2016). Simulation of PID and fuzzy logic controller for integrated seat suspension of a quarter car with driver model for different road profiles. *Journal of Mechanical Science and Technology, 30*(10), 4565-4570. https://doi.org/10.1007/s12206-016-0927-6.

Bibliography

Jain, P. (2007). Design and analysis of a tractor-trailer cabin suspension. *SAE Technical Paper, 2007-26-047*. https://doi.org/10.4271/2007-26-047.

LaPlante, J. [Inventor], & Larkins, W. [Inventor] (2004, May 4). Semi-active shock absorber control system. US Patent #6,732,033. Washington, DC: U.S. Patent and Trademark Office.

Appendix A: MATLAB Source Code

Main program code:

```
%% This is HAYES' standard matlab SimMaster script for simulations.
% >>>>>> HAYES CONFIDENTIAL & PROPRIETARY <<<<<<
%% PURPOSE
% Standardizes, centralizes, coordinates and controls all components needed
% to enable rapid, efficient and iterative analysis of simulation for semi-active seat suspension
control
%% REV BLOCK
% original template: HAYES_Standard_SimMaster_Script_rev20160407.m (path:
\\Mequon01\Engineering\20-Mechatronics\01 - Models\04 - MIL Simulation Models\) with
% script content drawn from SimMaster_825i_Vehicle.m script
%
% DDMMMYY - author initials - brief description (original source)
% 25MAY18 - FM - simulation of semi-active control of simple spring-mass-damper
% 22OCT18 - SC - Created Rev 2
% 11Apr19 - SC - Created rev 5 for 3 axis

%% SCRIPT (or FUNCTION) INPUTS & OUTPUTS
% INPUT: workspaces, init scripts, plant models, control models
% OUTPUT: workspace parameters, plots, objects, etc.
%% INPUT DATA FILE PREP
% raw input signal data must be trimmed down to frame the event of interest to minimize wasted
simulation execution time
% model parameters & constants
%% SCRIPT OUTLINE & ORGANIZATION
% Section 1 - Simulation Setup: Configs, Load Workspace & Run Parallel Scripts
% Section 2 - Declares, Initialize, Pre-allocate, scale
% Section 3 - Launch & Control Simulation
% Section 4 - Plot Results
%% ======================================
close all
clear
clc

%% SECTION 1 - SIMULATION SETUP

    % -->>> Load Plant Model Parmeter Value Initialization to workspace
            %run('Init_solenoid_model.m'); %
            %run('Init_valve_model.m'); %
            %run('Init_PWM_model.m');  %
            run('Init_SpringMassDamper_rev05_3_axis.m');

    % -->>> Choose & Load Test Surface to workspace
```

```
        %  0 = no road input...static
        %  1 = step up/down input
        %  2 = simple sinusoidal road input(set freq in Hz and amplitude in mm with
Init_test_surface_0.m script)
        %  3 = simple sinusoidal road input derived from ECU sampled LVDT or stroke sensor in
counts)
        %  4 = agriculture machine
        %  5 = tracked machine
        %  6 = hwy truck smooth
        %  7 = hwy truck medium
        %  8 = hyw truck rough
        %  9 = J28TR2 3 axes

          test_surface = 9;

          run('Init_TestSurfaces_rev04.m');   % plots the selected test surface for simulation
          input_road_displ_mm = travel_cubic_mm;
          t_data = time_s;
          num_samples = length(t_data);
          T_sampling_for_sim = t_data(2) - t_data(1);

    % -->>> Signal Processing Parameters
        % Filtering Coef

    % -->>> Controller Functions & Parmeters
        % Control Features (future)
          SAS_control = 1;  % 0 = turn OFF / 1 = turn ON
          SAS_CL_control = 1;  % 0 = turn OFF / 1 = turn ON
          Dithering = 0; % units: 0 = OFF/1 = ON

          run('Init_Controller_rev01.m');

%% SECTION 2 - DECLARES, INITIALIZE, PRE-ALLOC, SCALING
      % flow control

      % select the correct number of samples from the appropriate trimmed raw data set.
%          if Test_Condition == 1
%              num_samples = length(t_data);
%          elseif Test_Condition == 2
%              num_samples = length(t_data);
%          elseif Test_Condition == 3
%              num_samples = length(t_data);
%          elseif Test_Condition == 4
%              num_samples = length(t_data);
%          end
       duration = 10 %num_samples*T_sampling_for_sim;
```

```
        t_sim = t_data;  % the vector of simulation time stamps per data sample
%          sim_dec = double(1);         % used in ZOH blocks
%
%      % simulation time
        sim_StartTime = t_data(1);                 %programmatically sets the simulation's start time
(typically zero, but can enter any start time)
        calc_stoptime = t_data(num_samples);
        sim_StopTime = 10 %single(calc_stoptime);       %programmatically sets the simulation's
stop time (typically length*sample_rate)

%% SECTION 3 - LAUNCH & CONTROL SIMULATION

    % -->>> Specify the model's name in the string below (do not include the extension ".slx")
       model = 'SimEnviron_SpringMassDamper_rev05_3_axis'; %'Damper';

       set(0,'DefaultlineLinewidth',1);

    % -->>> Launch Simulink and open the model to be tested
       open_system(model);

    % -->>> Programmatically configure this model's simulation parameters
       % see full list of potential parameter names at
"http://www.mathworks.com/help/simulink/slref/model-parameters.html"
       % all the values must be strings (e.g. 'string')

set_param(model,'StartTime',num2str(sim_StartTime),'StopTime',num2str(sim_StopTime));%
sets the duration of the simulation to run for sim_time seconds
       set_param(model,'SolverType','Variable-step','Solver','ode23t'); % 'SolverType','Fixed-
step','FixedStep',num2str(T_sampling_for_sim));
       set_param(model,'MaxStep','1e-3','RelTol','1e-3','AbsTol','1e-3');
       set_param(model,'ZeroCrossAlgorithm','Adaptive');

    % -->>> Execute and Save simulation results to a separate object
       SimResults = sim(model);

    % find seat and base acceleration values
       sz = simout_Seat_z_Acc.Data(:,1).*0.0001019;
       sz_sq = sz.^2;
       sz_mean = mean(sz_sq);

       sb = simin_Seat_Base_Acc(:,1);
       sb_sq = sb.^2;
       sb_mean = mean(sb_sq);
       seat_acc_Z = sqrt(sz_mean/sb_mean);
```

```
   % find SEAT value

        sd = simout_SEAT_Displ_mm.Data;
        sd_sq = sd.^2;
        sd_mean = mean(sd_sq);

        rd = simin_road_displ_mm(:,1);
        rd_sq = rd.^2;
        rd_mean = mean(rd_sq);

        SEAT = sqrt(sd_mean)/sqrt(rd_mean);

%% SECTION 4 - PLOT RESULTS

% --->>> MEASURED DATA vs MODEL PREDICTION
        % trim plotted signals

        t1 = 0;
        t2 = 10;

        h_fig_Susp = figure('Position',[014, 50, 600, 840],'Name','Damping
Effect'); %,num2str(Test_Condition),'/Cal Version = ', num2str(cal_version)]);

        ax(1) = subplot(3,1,1);
        hold on
        plot(simout_SEAT_Displ_mm,'Color',[.75 .75 .75],'DisplayName','so Seat Displ
CT','LineWidth',2);
        plot(simout_Position_1stFilt_Ofs_mm,'Color',[.4 .698 1],'DisplayName','so Pos 1stFilt
DT','LineWidth',1);
        plot(simout_Position_Unfilt_Ofs_mm,'--r','DisplayName','so Pos Unfilt
DT','LineWidth',1);   % use parameter: simout_SEAT_Displ_mm for sim rev03 and earlier
        plot(simout_DAMPER_Displ_mm,'-g','DisplayName','so RelDispl CT','LineWidth',1);
        plot(Input_Displ_mm,'--k','DisplayName','Input Displ CT','LineWidth',2);
        ylim([-30 30]);
%       xlim([t1 t2]);
        title({'Road/Vehicle versus Seat Displacement (z-axis)'});
        xlabel('Time (seconds)')
        ylabel('Travel (mm)')
        grid minor
        legend show
        box on

        ax(2) = subplot(3,1,2);
        hold on
        plot(simout_SPRINGIn_Force_N,'-b','DisplayName','Sprung Mass');
        ylim([-5e3 -4e3]);
```

```
%       xlim([t1 t2]);
        title({'Force Input to Seat Base (z-axis)'});
        xlabel('Time (seconds)')
        ylabel('Force (N)')
        grid minor
        legend show
        box on

        ax(3) = subplot(3,1,3);
        hold on
        h(2) = plot(simout_SEAT_DisplVel_mmpers,'Color',[.75 .75 .75],'DisplayName','Seat
DisplVel','LineWidth',2);
        plot(simout_DAMPER_DisplVel_mmpers,'-r','DisplayName','Damper
DisplVel','LineWidth',1);
        h(1) = plot(simout_ROAD_DisplVel_mmpers,'--k','DisplayName','Road
DisplVel','LineWidth',2);
        uistack(h(1),'top');  % puts filtered signal on top in the plot
        ylim([-400 400]);
%       ylim([0 10]);
%       xlim([t1 t2]);
        title('Road/Vehicle versus Seat Velocity (z-axis)');
        xlabel('Time (seconds)')
        ylabel('Velocity (mm/s)')
        grid minor
        legend show
        box on

        linkaxes(ax,'x'); %[ax1,ax2,ax3]

% ------ 2D Lookup table Functionality
        h_fig_CalTable = figure('Position',[600, 100, 600, 840],'Name','2D Table
LookUp'); %,num2str(Test_Condition),'/Cal Version = ', num2str(cal_version)]);

        ax(1) = subplot(3,1,1);
        hold on
        plot(simout_Position_Unfilt_Ofs_mm,'-k','DisplayName','Unfilt Disp','LineWidth',1);
        plot(simout_Position_1stFilt_Ofs_mm,'-g','DisplayName','Filt Disp','LineWidth',1);
%      ylim([0 30]);
%       xlim([t1 t2]);
        title({'2D Table LookUp horizontal'});
        xlabel('Time (seconds)')
        ylabel('Travel (mm)')
        grid minor
        legend show
        box on
```

```
        ax(2) = subplot(3,1,2);
        hold on
        plot(simout_Velocity_Unfilt_mmpers,'-k','DisplayName','Unfilt Velocity')
        plot(simout_Velocity_1stFilt_mmpers,'-r','DisplayName','Filt Velocity');
        ylim([-220 200]);
%       xlim([t1 t2]);
        title({'Damper Velocity'});
        xlabel('Time (seconds)')
        ylabel('Velocity (mm/s)')
        grid minor
        legend show
        box on

        ax(3) = subplot(3,1,3);
        h(1) = plot(simout_ROAD_DisplVel_mmpers,'--k','DisplayName','Road
DisplVel','LineWidth',2);
        hold on
        h(2) = plot(simout_SEAT_DisplVel_mmpers,'Color',[.75 .75 .75],'DisplayName','Seat
Velocity','LineWidth',2);
        plot(simout_DAMPER_DisplVel_mmpers,'-g','DisplayName','Damper
Velocity','LineWidth',1);
        uistack(h(1),'top');  % puts filtered signal on top in the plot
        ylim([-400 400]);
%       ylim([0 10]);
%       xlim([t1 t2]);
        title('Road/Vehicle versus Seat Velocity (z-axis)');
        xlabel('Time (seconds)')
        ylabel('Velocity (mm/s)')
        grid minor
        legend show
        box on

        linkaxes(ax,'x'); %[ax1,ax2,ax3]
% ------
        h_fig_Cntrlr = figure('Position',[1200, 150, 600, 840],'Name','Controller
Behavior'); %,num2str(Test_Condition),'/Cal Version = ', num2str(cal_version)]);

        bx(1) = subplot(3,1,1);
        hold on
        h(1) = plot(simout_set_current,'-k','DisplayName','Iset [A]','LineWidth',2);
        %plot(simout_cmd_current,'-b','DisplayName','Icmd [A]');
        %plot(simout_MeasValveCurrent_amps,'-g','DisplayName','Imeas [A]','LineWidth',2);
        %uistack(h(2),'top');  % puts filtered signal on top in the plot
        ylim([-0.1 3]);
        %ylim([-4000 4000]);
        title('Controller Currents set, cmd & meas');
```

```
    xlabel('Time (seconds)')
    ylabel('Amps')
    grid minor
    legend show
    box on

    bx(2) = subplot(3,1,2);
    h(1) = plot(simout_cmd_voltage,'-b','DisplayName','CMD Voltage');
    hold on
    plot(simout_sys_voltage,'-k','DisplayName','System Voltage');
    %uistack(h(2),'top');  % puts filtered signal on top in the plot
    %ylim([0 2000]);
    ylim([-1 18]);
    title('Commanded Voltage to Tractive Valve');
    xlabel('Time (seconds)')
    ylabel('Volts')
    grid minor
    legend show
    box on
%       legend('Location','southeast')

    bx(3) = subplot(3,1,3);
    hold on
    plot(simout_cmd_current,'-r','DisplayName','Icmd [A]');
    plot(simout_MeasValveCurrent_amps,'--g','DisplayName','Imeas [A]','LineWidth',2);
    uistack(h(2),'top');  % puts filtered signal on top in the plot
    %ylim([-0.1 3]);
    %ylim([-4000 4000]);
    title('Controller Currents cmd & meas');
    xlabel('Time (seconds)')
    ylabel('Amps')
    grid minor
    legend show
    box on

    linkaxes(bx,'x'); %[bx1,bx2,bx3,bx4]

    %**********************Plot x axis data***********************
  % --->>> MEASURED DATA vs MODEL PREDICTION
    % trim plotted signals

    t1 = 0;
    t2 = 10;

    h_fig_Susp = figure('Position',[014, 50, 600, 840],'Name','Damping
Effect'); %,num2str(Test_Condition),'/Cal Version = ', num2str(cal_version)]);
```

```
        ax(1) = subplot(3,1,1);
        hold on
        plot(simout_SEAT_x_Displ_mm,'Color',[.75 .75 .75],'DisplayName','so Seat Displ
CT','LineWidth',2);
        plot(simout_Position_1stFilt_Ofs_mm2,'Color',[.4 .698 1],'DisplayName','so Pos 1stFilt
DT','LineWidth',1);
        plot(simout_Position_Unfilt_Ofs_mm2,'--r','DisplayName','so Pos Unfilt
DT','LineWidth',1);   % use parameter: simout_SEAT_Displ_mm for sim rev03 and earlier
        plot(simout_DAMPER_2_Displ_mm,'-g','DisplayName','so RelDispl CT','LineWidth',1);
        plot(Input_x_Displ_mm,'--k','DisplayName','Input Displ CT','LineWidth',2);
        ylim([-30 30]);
%       xlim([t1 t2]);
        title({'Road/Vehicle versus Seat Displacement (x-axis)'});
        xlabel('Time (seconds)')
        ylabel('Travel (mm)')
        grid minor
        legend show
        box on

        ax(2) = subplot(3,1,2);
        hold on
        plot(simout_SPRINGIn_x_Force_N,'-b','DisplayName','Sprung Mass');
        ylim([-5e3 -4e3]);
%       xlim([t1 t2]);
        title({'Force Input to Seat Base (x-axis)'});
        xlabel('Time (seconds)')
        ylabel('Force (N)')
        grid minor
        legend show
        box on

        ax(3) = subplot(3,1,3);
        hold on
        h(2) = plot(simout_SEAT_x_DisplVel_mmpers,'Color',[.75 .75 .75],'DisplayName','Seat
DisplVel','LineWidth',2);
        plot(simout_DAMPER_2_DisplVel_mmpers,'-r','DisplayName','Damper
DisplVel','LineWidth',1);
        h(1) = plot(simout_ROAD_x_DisplVel_mmpers,'--k','DisplayName','Road
DisplVel','LineWidth',2);
        uistack(h(1),'top');  % puts filtered signal on top in the plot
        ylim([-400 400]);
%       ylim([0 10]);
%       xlim([t1 t2]);
        title('Road/Vehicle versus Seat Velocity (x-axis)');
        xlabel('Time (seconds)')
```

```
	ylabel('Velocity (mm/s)')
	grid minor
	legend show
	box on

	linkaxes(ax,'x'); %[ax1,ax2,ax3]

	h_fig_Cntrlr = figure('Position',[1200, 150, 600, 840],'Name','Controller
Behavior'); %,num2str(Test_Condition),'/Cal Version = ', num2str(cal_version)]);

	bx(1) = subplot(3,1,1);
	hold on
	h(1) = plot(simout_set_current2,'-k','DisplayName','Iset [A]','LineWidth',2);
	%plot(simout_cmd_current,'-b','DisplayName','Icmd [A]');
	%plot(simout_MeasValveCurrent_amps,'-g','DisplayName','Imeas [A]','LineWidth',2);
	%uistack(h(2),'top');  % puts filtered signal on top in the plot
	ylim([-0.1 3]);
	%ylim([-4000 4000]);
	title('Controller Currents set, cmd & meas');
	xlabel('Time (seconds)')
	ylabel('Amps')
	grid minor
	legend show
	box on

	bx(2) = subplot(3,1,2);
	h(1) = plot(simout_cmd_voltage2,'-b','DisplayName','CMD Voltage');
	hold on
	plot(simout_sys_voltage2,'-k','DisplayName','System Voltage');
	%uistack(h(2),'top');  % puts filtered signal on top in the plot
	%ylim([0 2000]);
	ylim([-1 18]);
	title('Commanded Voltage to Tractive Valve');
	xlabel('Time (seconds)')
	ylabel('Volts')
	grid minor
	legend show
	box on
%	legend('Location','southeast')

	bx(3) = subplot(3,1,3);
	hold on
	plot(simout_cmd_current2,'-r','DisplayName','Icmd [A]');
	plot(simout_x_MeasValveCurrent_amps,'--g','DisplayName','Imeas [A]','LineWidth',2);
	uistack(h(2),'top');  % puts filtered signal on top in the plot
	%ylim([-0.1 3]);
```

```
    %ylim([-4000 4000]);
    title('Controller Currents cmd & meas');
    xlabel('Time (seconds)')
    ylabel('Amps')
    grid minor
    legend show
    box on

    linkaxes(bx,'x'); %[bx1,bx2,bx3,bx4]

        %**********************Plot y axis data************************
  % --->>> MEASURED DATA vs MODEL PREDICTION
    % trim plotted signals

    t1 = 0;
    t2 = 10;

    h_fig_Susp = figure('Position',[014, 50, 600, 840],'Name','Damping
Effect'); %,num2str(Test_Condition),'/Cal Version = ', num2str(cal_version)]);

    ax(1) = subplot(3,1,1);
    hold on
    plot(simout_SEAT_y_Displ_mm,'Color',[.75 .75 .75],'DisplayName','so Seat Displ
CT','LineWidth',2);
    plot(simout_Position_1stFilt_Ofs_mm3,'Color',[.4 .698 1],'DisplayName','so Pos 1stFilt
DT','LineWidth',1);
    plot(simout_Position_Unfilt_Ofs_mm3,'--r','DisplayName','so Pos Unfilt
DT','LineWidth',1);   % use parameter: simout_SEAT_Displ_mm for sim rev03 and earlier
    plot(simout_DAMPER_3_Displ_mm,'-g','DisplayName','so RelDispl CT','LineWidth',1);
    plot(Input_y_Displ_mm,'--k','DisplayName','Input Displ CT','LineWidth',2);
    ylim([-30 30]);
%       xlim([t1 t2]);
    title({'Road/Vehicle versus Seat Displacement (y-axis)'});
    xlabel('Time (seconds)')
    ylabel('Travel (mm)')
    grid minor
    legend show
    box on

    ax(2) = subplot(3,1,2);
    hold on
    plot(simout_SPRINGIn_y_Force_N,'-b','DisplayName','Sprung Mass');
    ylim([-5e3 -4e3]);
%       xlim([t1 t2]);
    title({'Force Input to Seat Base (y-axis)'});
    xlabel('Time (seconds)')
```

```
	ylabel('Force (N)')
	grid minor
	legend show
	box on

	ax(3) = subplot(3,1,3);
	hold on
	h(2) = plot(simout_SEAT_y_DisplVel_mmpers,'Color',[.75 .75 .75],'DisplayName','Seat
DisplVel','LineWidth',2);
	plot(simout_DAMPER_3_DisplVel_mmpers,'-r','DisplayName','Damper
DisplVel','LineWidth',1);
	h(1) = plot(simout_ROAD_y_DisplVel_mmpers,'--k','DisplayName','Road
DisplVel','LineWidth',2);
	uistack(h(1),'top');  % puts filtered signal on top in the plot
	ylim([-400 400]);
%	ylim([0 10]);
%	xlim([t1 t2]);
	title('Road/Vehicle versus Seat Velocity (y-axis)');
	xlabel('Time (seconds)')
	ylabel('Velocity (mm/s)')
	grid minor
	legend show
	box on

	linkaxes(ax,'x'); %[ax1,ax2,ax3]

	h_fig_Cntrlr = figure('Position',[1200, 150, 600, 840],'Name','Controller
Behavior'); %,num2str(Test_Condition),'/Cal Version = ', num2str(cal_version)]);

	bx(1) = subplot(3,1,1);
	hold on
	h(1) = plot(simout_set_current3,'-k','DisplayName','Iset [A]','LineWidth',2);
	%plot(simout_cmd_current,'-b','DisplayName','Icmd [A]');
	%plot(simout_MeasValveCurrent_amps,'-g','DisplayName','Imeas [A]','LineWidth',2);
	%uistack(h(2),'top');  % puts filtered signal on top in the plot
	ylim([-0.1 3]);
	%ylim([-4000 4000]);
	title('Controller Currents set, cmd & meas');
	xlabel('Time (seconds)')
	ylabel('Amps')
	grid minor
	legend show
	box on

	bx(2) = subplot(3,1,2);
	h(1) = plot(simout_cmd_voltage3,'-b','DisplayName','CMD Voltage');
```

```
    hold on
    plot(simout_sys_voltage3,'-k','DisplayName','System Voltage');
    %uistack(h(2),'top');  % puts filtered signal on top in the plot
    %ylim([0 2000]);
    ylim([-1 18]);
    title('Commanded Voltage to Tractive Valve');
    xlabel('Time (seconds)')
    ylabel('Volts')
    grid minor
    legend show
    box on
%       legend('Location','southeast')

    bx(3) = subplot(3,1,3);
    hold on
    plot(simout_cmd_current3,'-r','DisplayName','Icmd [A]');
    plot(simout_Damper3_MeasValveCurrent_amps,'--g','DisplayName','Imeas
[A]','LineWidth',2);
    uistack(h(2),'top');  % puts filtered signal on top in the plot
    %ylim([-0.1 3]);
    %ylim([-4000 4000]);
    title('Controller Currents cmd & meas');
    xlabel('Time (seconds)')
    ylabel('Amps')
    grid minor
    legend show
    box on

    linkaxes(bx,'x'); %[bx1,bx2,bx3,bx4]


% END of SCRIPT
```

Initialization file “Init_SpringMassDamper_rev05_3_axis.m” code:

```
%% This is HAYES' standard matlab SimMaster script for tuning the parameters of a plant
model.
% >>>>>> HAYES CONFIDENTIAL & PROPRIETARY <<<<<<
%% PURPOSE
% Initializes the HAYES Seat Suspension System plant model parameters into the workspace
prior to simulation.
% Must be launched by a SimMaster script.

%% REV BLOCK
```

```
% original template: HAYES_Standard_SimMaster_Script_rev20160407.m (path:
\\Mequon01\Engineering\20-Mechatronics\01 - Models\04 - MIL Simulation Models\) with
% script content drawn from HB_825i_5phaseABS_final_testing.m script (same path as for this
script.)
%
% DDMMMYY - author initials - brief description (original source)
% 02MAY18 - FM - Controller parameters
% 22OCT18 - SC - created Rev02
% 11Apr19 - SC - created Rev05 3 axis

%% SCRIPT (or FUNCTION) INPUTS & OUTPUTS
% INPUT: SimMaster script
% OUTPUT: workspace parameters, plots, objects, etc.
%% INPUT DATA FILE PREP
% none required
%
%% SCRIPT OUTLINE & ORGANIZATION
% Section 0 - Comments describes associated plant model
% Section 1 - Declares, Initialize, Pre-allocate, scale
% Section 2 - (optional) Plot Results
%% ========================================
% clear
% close all
% clc

%% SECTION 0 - ASSOCIATED PLANT MODEL
% plant_Solenoid_rev01.slx (create this at some point to capture

%% SECTION 1 - PARAMETER INITIALIZATION

    % Unit Conversion formulas
    m_to_mm = 0.001;  % convert meters to millimeters
    m_to_ft = 3.28083; % convert meters to feet
    mm_to_m = 1000;
    N_to_lbf = 0.2248089;   % convert N to pounds-force
    N_to_kgf = 0.1019716; % convert N to kilogram-force
    m_to_inches = 39.36996;  % convert meters to inches

    % Gravity variables
     gravity_mpers2 = -9.80665;  % units = m/s^2
     gravity_mmpers2 = gravity_mpers2 * m_to_mm;
     gravity_ftpers2 = gravity_mpers2 * m_to_ft;

    % Solver Configuration
```

```
   % Input Excitation Function (simple Sine block was used as road input in early stages of plant
model contruction (later replaced by siminblock
      Exciter_freq_Hz = 2;
      Exciter_freq_radpers = Exciter_freq_Hz * 2 * pi;   % radians/sec = 2?f
      Exciter_amplitude_unitless = 1;

   % Chassis mass (not yet used in plant model)
      Chassis_mass_kg = 4535.924;  % units = kilograms
      Chassis_mass_lbs = Chassis_mass_kg * 2.20462;

   % Seat mass
      Seat_mass_kg = 50;  % units = kilograms
      Seat_mass_lbs = Seat_mass_kg * 2.20462;
      Seat_force_N = Seat_mass_kg * -9.80;  % units = Newtons (or kg-m/s^2)

   % Operator mass
      Operator_mass_kg = 75;  % typ 125 kg changed to 75 kg to match empirical data
      Operator_mass_lbs = Operator_mass_kg * 2.20462;
      Operator_force_N = Operator_mass_kg * -9.80;  % units = Newtons (or kg-m/s^2)

   % Sprung mass (seat + operator)
      Sprung_mass_kg = Seat_mass_kg + Operator_mass_kg;
      Sprung_mass_lbs = Seat_mass_lbs + Operator_mass_lbs;
      Sprung_mass_force_N = Sprung_mass_kg * gravity_mpers2;  % units = Newtons (or kg-
m/s^2)

   % Spring rate
      SpringConstant_Npermm = 100; % set to 1 N/mm for zero spring effect / 1e5 N/mm for
stiff spring effect
      SpringConstant_kgfpermm = SpringConstant_Npermm * N_to_kgf;
      SpringConstant_Nperm = SpringConstant_Npermm * mm_to_m;
      SpringConstant_lbfperin = SpringConstant_Nperm * N_to_lbf * m_to_inches;

   % Shock Cylinder Stroke Parameters
      Stroke_length_mm = 67;  %Sears reqmt is 67 mm
      Stroke_length_x_mm = 30;
      Stroke_length_y_mm = 30;
      Stroke_length_upper_bound_mm = Stroke_length_mm/2;
      Stroke_length_x_upper_bound_mm = Stroke_length_x_mm/2;
      Stroke_length_y_upper_bound_mm = Stroke_length_y_mm/2;
      Stroke_length_lower_bound_mm = Stroke_length_mm/2;
      Stroke_length_x_lower_bound_mm = Stroke_length_x_mm/2;
      Stroke_length_y_lower_bound_mm = Stroke_length_y_mm/2;
      %Preload_
      Preload_SprungMass_static_force_N = Sprung_mass_kg*gravity_mpers2;    % static force
from sprung mass load
```

```
        Preload_SprungMass_static_bias_m =
Preload_SprungMass_static_force_N/SpringConstant_Nperm;  % units are mm above full stroke
center
        Preload_SprungMass_static_bias_mm = Preload_SprungMass_static_bias_m / m_to_mm;
        Preload_added_bias_mm = 0;
        Preload_SprungMass_total_bias_mm = Preload_SprungMass_static_bias_mm +
Preload_added_bias_mm;
        Preload_total_bias_mm = Preload_SprungMass_static_bias_mm +
Preload_added_bias_mm;

    % Damping_coef
        varD_Coef_slope_Npermmpers = 0; % last setting = 5.5 N/mm/s
        Damping_Coef_Npermmpers = 0;  % set to 0 or 0.001 to eliminate damping effects;
otherwise make as small as possible to allow variable damping the widest range of authoriy
        Damping_Coef_Npermpers = Damping_Coef_Npermmpers * mm_to_m;
        Damping_Coef_lbfperin = Damping_Coef_Npermmpers * N_to_lbf * m_to_inches;

    % Spring-Mass-Damper System Parameters
        temp1 = SpringConstant_Nperm/Sprung_mass_kg;
        temp2 = sqrt(temp1);
        SMD_natural_freq_Hz = temp2/2*pi; % units = Hz
        % booger factor
        SMD_natural_freq_Hz = SMD_natural_freq_Hz/10;
```

Initialization file "Init_TestSurfaces_rev04.m" code:

```
%% PURPOSE: loads and visualizes selected terrain file as measured by stroke profile:
    %  0 = no road input...static
    %  1 = step up/down input
    %  2 = simple sinusoidal road input(set freq in Hz and amplitude in mm with
Init_test_surface_0.m script)
    %  3 = simple sinusoidal relative damper stroke input derived from ECU sampled LVDT or
stroke sensor in counts)
    %  4 = agriculture machine
    %  5 = tracked machine
    %  6 = hwy truck smooth
    %  7 = hwy truck medium
    %  8 = hyw truck rough
    %  9 - 3 axis J28TR

%% File copied from Choose Surface section of SimMaster_825i_Vehicle_2016_12_14.m
% balanced behavior of 825i rear wheel
%%
% clear all
% close all
% clc
```

```
%%
if test_surface == 0  % static or zero road input
   time_s = linspace(0,10,1000000);  % 10 µs time step
   time_s = transpose(time_s);
   static = zeros(1000000,1);
   travel_cubic_mm = static;
   simin_road_displ_mm = [time_s,static];
   road_displ_mm = ones(1000000,1);
   Road_Displ_amplitude_mm = 2048*road_displ_mm;
   simin_road_displ_counts = [time_s,Road_Displ_amplitude_mm];
   no_skip = 0; % use to skip simin creation
elseif test_surface == 1 % step up (bump) or down (pothole) road input
   time_s = linspace(0,10,1000000); % 500 µs time step
   time_s = transpose(time_s);
   Road_Displ_direction = 1; % +1 = bump / -1 = pothole
   step = zeros(1000000,1);
   event_num = 1e5;   % road surface excursion starts at this event
   ramp_time = 300;  % number of 500µs samples (5000 = 2.5 s / 1000 = 0.5 s)
   increment = 0;
   increment_size = 1/ramp_time;
   if Road_Displ_direction == 1
      for i = 1:100000
         step(event_num+i+1) = step(event_num+i) + increment_size;
      end
   elseif Road_Displ_direction == -1
      for i = 1:100000
         step(event_num+i+1) = step(event_num+i) - increment_size;
      end
   end
   step(event_num+ramp_time:1e6) = 1*Road_Displ_direction;
   Road_Displ_amplitude_mm = 1*Road_Displ_direction;
   simin_road_displ_mm = [time_s,step];
   travel_cubic_mm = step;
   plot(step); grid minor
   no_skip = 0; % use to skip simin creation
elseif test_surface == 2 % sinusoidal road input
   time_s = linspace(0,10,1000000);
   time_s = transpose(time_s);
   f =0.5;  % units = Hz
   omega = 2*pi*f;
   Road_Displ_amplitude_mm = 5; % units = mm
   sinusoid = Road_Displ_amplitude_mm * sin(omega*time_s);
   travel_cubic_mm = sinusoid;
   simin_road_displ_mm = [time_s,sinusoid];
   % create signal in units of a/d counts (skip voltages of sensor)
```

```
    scaling = 37.5; %units = counts/mm; adjust this to reduce or increase to match measured
displacement data
    offset = 500; % units = counts; ajdust this to center the a/d counts results to be between 0 <>
4095
    road_displ_counts = sinusoid*scaling + offset;
    simin_road_displ_counts = [time_s,road_displ_counts];
    no_skip = 0; % use to skip simin parameter creation
elseif test_surface == 3
    load('damper_pos_test_08');
    time_s = time_6 - time_6(1);
    travel_cubic_mm = CALC_Damper_Unfiltered_position_inches_6*25.4;  % convert inches to
mm
    simin_road_displ_mm = [time_s,travel_cubic_mm];
    simin_road_displ_counts = [time_s,M_Damper_position_counts_6];
    Road_Displ_amplitude_mm = 1; %added 3-19-19
    no_skip = 0;   % use to skip simin parameter creation
elseif test_surface == 4
    load('AG3.mat');
    no_skip = 1;
elseif test_surface == 5
    load('EM6.mat');
    no_skip = 1;
elseif test_surface == 6
    load('J25TR5_zonly.mat');
    no_skip = 1;
elseif test_surface == 7
    load('J25TR2_zonly.mat');
    no_skip = 1;
elseif test_surface == 8
    load('J51TR1_zonly.mat');
    no_skip = 1;
elseif test_surface == 9
    load('J28TR2.mat');
    no_skip = 2;
end

%% convert raw signal voltage data to stroke units
    if no_skip == 1
        travel_linear_mm = -54.438*Ch1_volts + 184.26;   % linear conversion volts to mm
        travel_cubic_mm = -1.0388787115799*Ch1_volts.^3 + 6.3487654942079*Ch1_volts.^2 -
61.198405226913*Ch1_volts + 177.253498919021;  % cubic conversion
        travel_mean_cubic = mean(travel_cubic_mm);
        Seat_Base_Acc = (Ch1_volts.*1.007);
        simin_Seat_Base_Acc = [time_s,Seat_Base_Acc];
        simin_road_displ_mm = [time_s,travel_cubic_mm];
        Road_Displ_amplitude_mm = 1;
```

```
    simin_road_displ_counts = [time_s,travel_cubic_mm];
    simin_x_road_displ_mm = [time_s,travel_cubic_mm];
    simin_y_road_displ_mm = [time_s,travel_cubic_mm];
elseif no_skip == 2
    Z = J28TR2(:,4);
    X = J28TR2(:,2);
    Y = J28TR2(:,3);
    travel_linear_mm = Z{:,1}*1000;
    travel_x_linear_mm = X{:,1}*1000;
    travel_y_linear_mm = Y{:,1}*1000;
    time_s = J28TR2{:,1}*1;
    travel_cubic_mm = travel_linear_mm.^3;
    travel_mean_cubic = mean(travel_cubic_mm);
    %Seat_Base_Acc = travel_linear_mm;
    simin_Seat_Base_Acc = [time_s,travel_linear_mm];
    simin_road_displ_mm = [time_s,travel_linear_mm];
    simin_x_road_displ_mm = [time_s,travel_x_linear_mm];
    simin_y_road_displ_mm = [time_s,travel_y_linear_mm];
    Road_Displ_amplitude_mm = 1;
    simin_road_displ_counts = [time_s,travel_cubic_mm];

   figure(1);
    plot(time_s,travel_linear_mm);
    xlim([0 10]);
    hold on;
    title({'Seat Base Motion, Backhoe Loader (z-axis)'});
    xlabel('Time (seconds)')
    ylabel('Travel (mm)')
    hold off;
    figure(2);
    plot(time_s,travel_x_linear_mm);
    xlim([0 10]);
    hold on;
    title({'Seat Base Motion, Backhoe Loader (x-axis)'});
    xlabel('Time (seconds)')
    ylabel('Travel (mm)')
    hold off;
    figure(3)
    plot(time_s,travel_y_linear_mm);
    xlim([0 10]);
    hold on;
    title({'Seat Base Motion, Backhoe Loader (y-axis)'});
    xlabel('Time (seconds)')
    ylabel('Travel (mm)')
    hold off;
end
```

```
%% for plotting purposes, plot all terrain profiles and highlight the test surface that has been
selected for simulation

%     h_terrain = figure('Position',[100 100 1000 800],'Name','Terrain Profiles');
%
%      % Visualize Stroke data of large Ag Tractors
%          subplot(3,1,1)
%          plot(time_s,travel_linear_mm,'r','DisplayName','Linear volts-to-mm conversion');
%          hold on
%          plot(time_s,travel_cubic_mm,'-b','DisplayName','Cubic volts-to-mm conversion');
%          plot([0,10],[travel_mean_cubic, travel_mean_cubic],'--b','DisplayName','average');
%          grid minor
%          ylim([-40 40]);
%          xlabel('Time (s)');
%          ylabel('Stroke (mm)');
%             if test_surface == 1
%                title('Stroke Data - large tractor (AG3 file)');
%             elseif test_surface == 2
%                title('Stroke Data - Tracked construction machine (EM6 file)');
%             elseif test_surface == 3
%                title('Stroke Data - highway truck SMOOTH');
%             elseif test_surface == 4
%                title('Stroke Data - highway truck MEDIUM');
%             elseif test_surface == 5
%                title('Stroke Data - highway truck ROUGH');
%             end
%          legend show
%          box on
%          legend('Location','southeast')
%          clear
%
%       % Visualize Stroke data of Tracked Construction machines
%          subplot(3,1,2)
%          load('EM6.mat');
%          travel_linear_mm = -54.438*Ch1_volts + 184.26;   % linear conversion volts to mm
%          travel_cubic_mm = -1.0388787115799*Ch1_volts.^3 +
6.3487654942079*Ch1_volts.^2 - 61.198405226913*Ch1_volts + 177.253498919021;  % cubic
conversion
%          travel_mean_cubic = mean(travel_cubic_mm);
%          plot(time_s,travel_linear_mm,'r','DisplayName','Linear volts-to-mm conversion');
%          hold on
%          plot(time_s,travel_cubic_mm,'-b','DisplayName','Cubic volts-to-mm conversion');
%          plot([0,10],[travel_mean_cubic, travel_mean_cubic],'--b','DisplayName','average');
%          grid minor
%          %ylim([3 3.3]);
```

```
%          xlabel('Time (s)');
%          ylabel('Stroke (mm)');
%
%          legend show
%          box on
%          legend('Location','southeast')
%          clear
%
%      % Visualize Stroke data of over-the-road highway trucks
%          subplot(3,1,3)
%          load('J25TR5_zonly.mat')
%          travel_cubic_mm = -1.0388787115799*Ch1_volts.^3 +
6.3487654942079*Ch1_volts.^2 - 61.198405226913*Ch1_volts + 177.253498919021;  % cubic
conversion
%          plot(time_s,travel_cubic_mm,'-r','DisplayName','smooth (J25TR5)'); % 'LineWidth',2,);
%          hold on
%          clear
%          load('J25TR2_zonly.mat'); % 'LineWidth',2,);
%          travel_cubic_mm = -1.0388787115799*Ch1_volts.^3 +
6.3487654942079*Ch1_volts.^2 - 61.198405226913*Ch1_volts + 177.253498919021;  % cubic
conversion
%          plot(time_s,travel_cubic_mm,'-b','DisplayName','medium (J25TR2)');
%          clear
%          load('J51TR1_zonly.mat');
%          travel_cubic_mm = -1.0388787115799*Ch1_volts.^3 +
6.3487654942079*Ch1_volts.^2 - 61.198405226913*Ch1_volts + 177.253498919021;  % cubic
conversion
%          plot(time_s,travel_cubic_mm,'-g','DisplayName','rough (J51TR1)'); %
'LineWidth',2,););
%          grid minor
%          ylim([-60 60]);
%          legend show
%          box on
%          legend('Location','southeast')
%          xlabel('Time (s)');
%          ylabel('Stroke (mm)');
%          title('Stroke Data - highway truck');
```

Initialization file “Init_Controller_rev01.m” code:

```
%% This is HAYES' standard matlab SimMaster script for tuning the parameters of a plant
model.
% >>>>>> HAYES CONFIDENTIAL & PROPRIETARY <<<<<<
%% PURPOSE
```

```
% Initializes the HAYES Seat Suspension System Signal Processing model block parameters
into the workspace prior to simulation.
% Must be launched by a SimMaster script.

%% REV BLOCK
% original template: HAYES_Standard_SimMaster_Script_rev20160407.m (path:
\\Mequon01\Engineering\20-Mechatronics\01 - Models\04 - MIL Simulation Models\) with
% script content drawn from HB_825i_5phaseABS_final_testing.m script (same path as for this
script.)
%
% DDMMMYY - author initials - brief description (original source)
% 16JUL18 - FM - signal processing model parameters

%% SCRIPT (or FUNCTION) INPUTS & OUTPUTS
% INPUT: SimMaster script
% OUTPUT: workspace parameters, plots, objects, etc.
%% INPUT DATA FILE PREP
% none required
%
%% SCRIPT OUTLINE & ORGANIZATION
% Section 0 - Comments describes associated plant model
% Section 1 - Declares, Initialize, Pre-allocate, scale
% Section 2 - (optional) Plot Results
%% ========================================
% clear
% close all
% clc

%% SECTION 0 - ASSOCIATED PLANT MODEL
% Solenoid_plant_rev.slx (not yet created)
% plant_SpringMassDamper_rev01.slx

%% SECTION 1 - PARAMETER INITIALIZATION

  % Solver Configuration
    Ts = 0.001;

  % System Power
    V_battery = 14; % units = volts

  % Controller Functions Enable/Disable
    if SAS_control == 1
      SAS_ON = 1;  % create in SimEnviron model
    else
      SAS_ON = 0;
    end
```

```
    if SAS_CL_control == 1
       CL_ON = 1;  % create in SimEnviron model
    else
       CL_ON = 0;
    end

  % Rate Transistions Physical System to Controller back to Physical System
    control_loop_rate = 0.001;  % units = seconds (used in all rate transition blocks in
SimEnviron)
    sim_dec = 1;  % demoninator in zero-order hold blocks

  % PWM Configuration
    PWM_freq_Hz = 2500; % enter integer 1000 < PWM_freq < 5000 units = H
    Dither_amplitude_percent = 0; % units = percent of PWM waveform's duty cycle / enter
value 0% < ampl < 5%
    Dither_freq_Hz = 0; % units = Hertz / enter value between 0 Hz < freq < 200 HzPWM_

  % Current PID controller Gains
    % gain values currrently writtn directly in the PID block
```

Model file "SimEnviron_SpringMassDamper_rev05_3_axis":

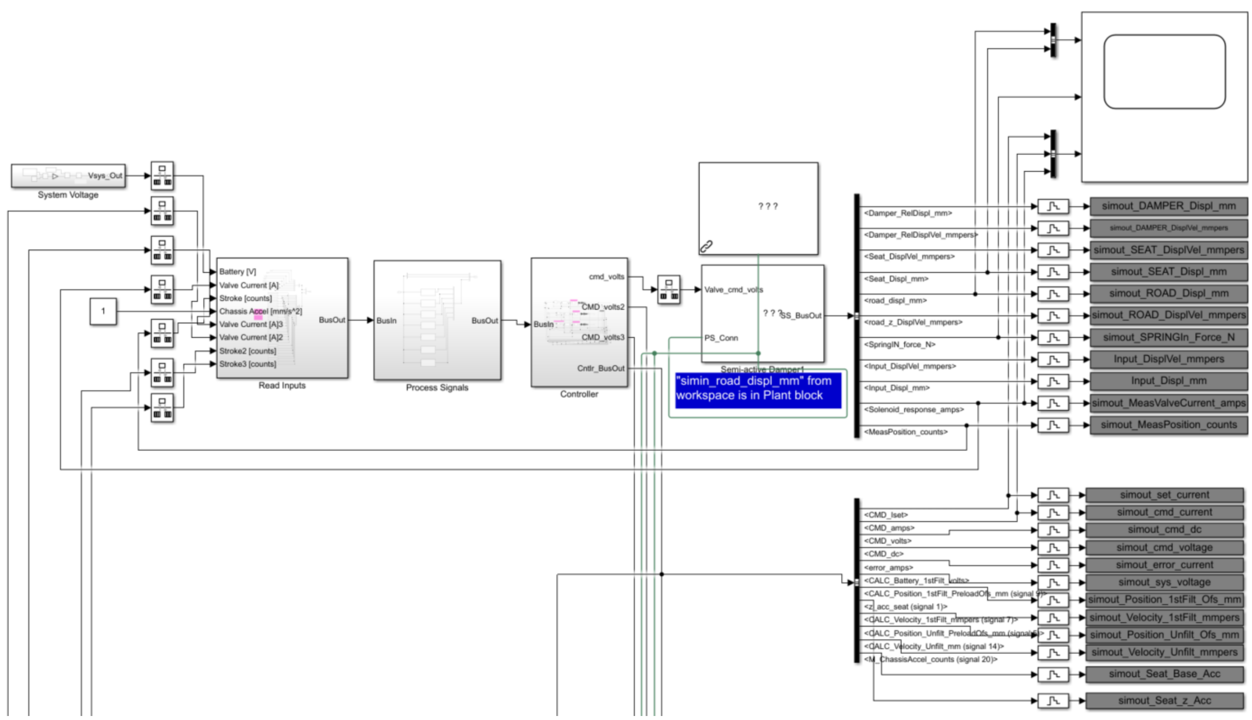

Figure A-1: Z-Axis Damper and Control System.

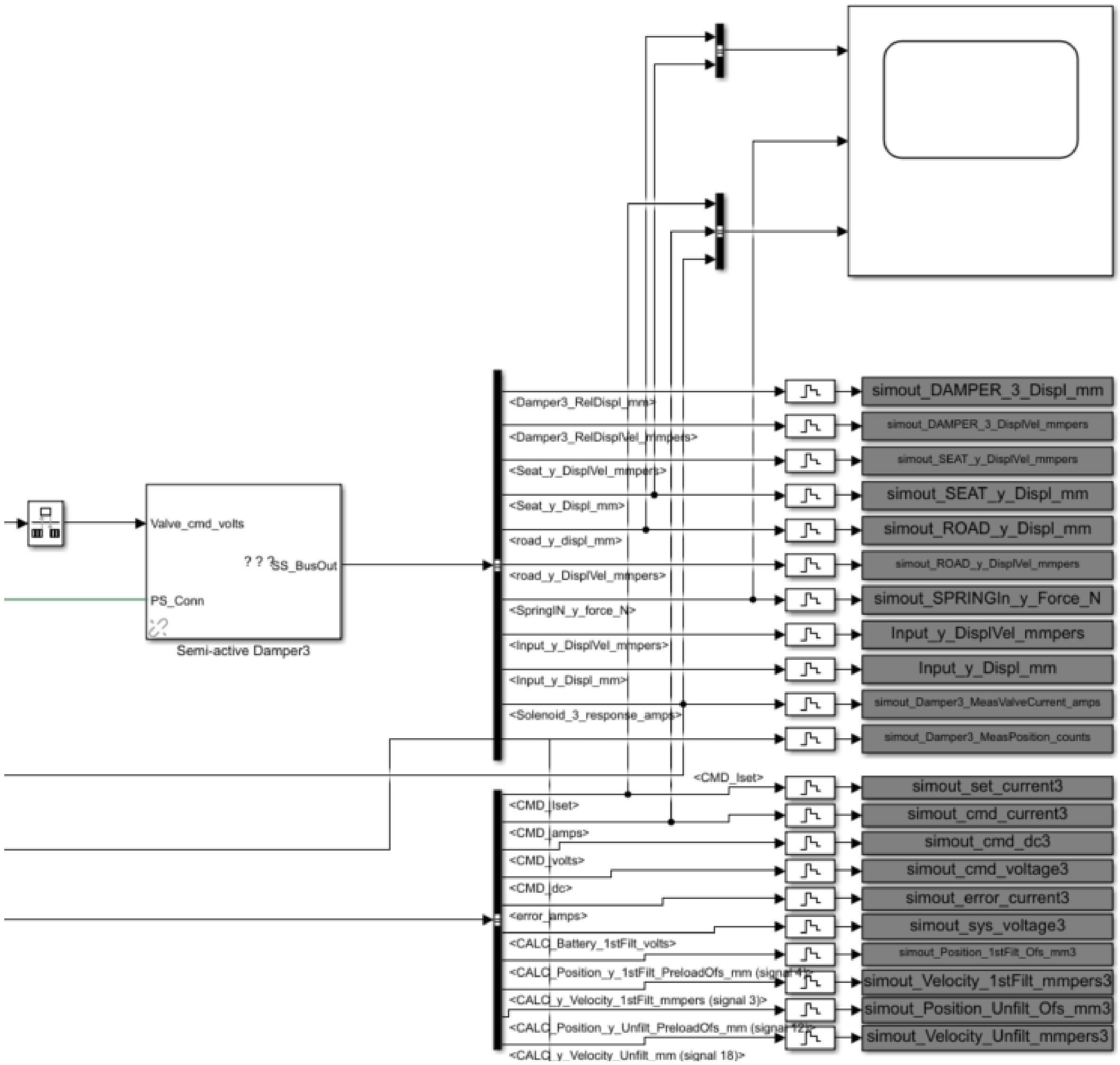

Figure A-2: Y-Axis Damper.

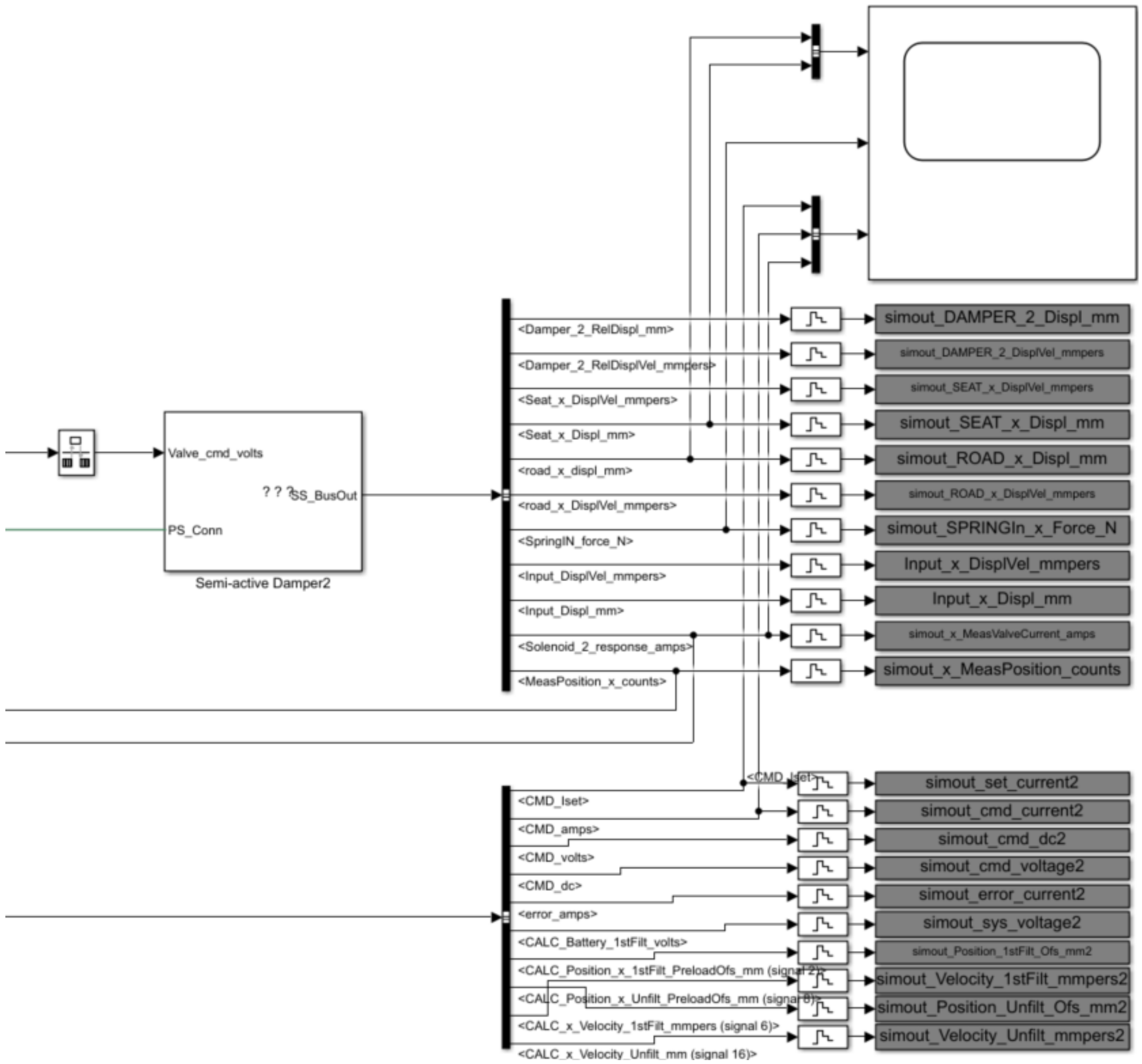

Figure A-3: X-Axis Damper.

Engineering

Capstone Report Approval Form

Master of Science in Engineering -- MSE

Milwaukee School of Engineering

This capstone report, entitled "MATLAB/Simulink Simulation of Semi-Active Seat Suspension System with a Variable Flow Damper Valve," submitted by the student Scott A. Carpenter, has been approved by the following committee:

Faculty Advisor: ______________________ Date: _______________

Dr. Subha Kumpaty, Ph. D.

Faculty Member: ______________________ Date: ________________

Frank Molinaro, B.S.E.E.

Faculty Member: ______________________ Date: ________________

Gary Shimek, M.L.I.S.