GM - Capstone - Electrified Driveline System Modelling.

Note: Subject to a Non-Disclosure Agreement with General Motors and the Fung Institute at the University of California, Berkeley

Chi Zhang, Shakshi Himmatramka, Sumanyu Singh

May 4, 2020

Abstract

The automotive industry is in the midst of a revolution with increasing levels of investment in electric power units to replace internal combustion engines in current and future vehicles. While range anxiety is the leading factor preventing mass adoption by drivers worldwide, automotive manufacturers are placing a premium on drivability to ensure that the change from gas vehicles to electric vehicles is effortless. Drivability is the smoothness of a vehicle's operation, and in electric vehicles, the low-frequency vibrations from the electric driveline is the source of discomfort. This paper presents different control strategies to dampen low-frequency vibrations perceived by the human body while the vehicle is in operation. The work includes dynamic modeling of the system, linearization, system identification, classical control strategies, and advanced control strategies. The simulation results demonstrate the efficacy of each approach and assess controller performance in dampening low-frequency vibrations.

1 Introduction:

Powerful Motors have inherent performance advantages but come with unique challenges affecting the drivability of Electric Vehicles

Powerful motors are examples of high-speed and quick-reacting components that improve the performance of electric vehicles. These components profoundly affect the dynamic behavior of the vehicle, leading to high oscillations at the resonant frequency. These oscillations can lead to several problems, such as driver discomfort, wheel slip, and component fatigue. Given the importance, it is necessary to determine a solution and motivate the team to work on it.

The oscillations observed due to the presence of high-inertial components in the vehicle fall under the low-frequency domain (around 10 Hz). The limiting factor of mechanical dampers to dampen the oscillations motivated the team to design a feedback control system modulating motor torque to mitigate the problem. The team from the University of California, Berkeley, collaborated with the industry partner, General Motors, to formulate a solution to this problem.

2 Prerequisite Theory

2.1 Resonance

Resonance is the amplification of the amplitude when the frequency of a periodic force matches the natural frequency of the system on which it is applied. At the resonant frequency, the system oscillates at significantly higher amplitudes than non-resonant frequencies. Observers will notice the response of the system ballooning at resonance [1]. This phenomenon leads to heavy vibrations at resonance frequency and we aim to design a controller successful in amplitude reduction at the resonant frequencies. As seen in the open-loop bode plot behavior figure 12, at 9Hz frequency, the vehicle experiences resonance of 30dB amplitude. The goal is to reduce the overshoot (peak) and achieve a flatter closed-loop bode plot. The following equation defines the system percentage overshoot(M_p):

$$M_p = \exp(-\frac{\zeta\pi}{\sqrt{1-\zeta^2}}) * 100 \tag{1}$$

where ζ is the damping coefficient. The overshoot is only dependent on ζ and as ζ increases the overshoot M_p decreases. For the system with zero overshoot, $\zeta > 0.7$.

3 Problem Statement:

Low-frequency vibrations of range 0 Hz to 50 Hz in the front to rear direction of motion and require novel damping approaches

Mechanical dampers are effective in damping oscillations in the vertical axis (y-axis). While hardware components also provide passive damping in the longitudinal direction, the biggest advantage of using control techniques is to save on component cost, improving profit margins for each vehicle sold. Armed with this knowledge, the team focused on controlling oscillations in the longitudinal direction (x-axis). Before developing the problem statement, it was necessary to determine sources that excite these disturbances. Besides the high-inertial rotating motors, fast-changing input from the driver to the motor, gear backlash, road perturbations, and aerodynamics. Compliance characterized by mechanical stiffness and damping in the driveline components is one of the primary factors which affects the vehicle dynamics.

Based on researched information, the team, along with the technical advisor, panned out the project objectives and determined the project deliverables mentioned below:

1. **Development of Simulation and Modelling Environment**

   - Identification of linear and non-linear components of the vehicle dynamics to be considered for modelling.
   - Linearization of the non-linear components.
   - Vehicle dynamic modelling and state space representation.
   - Model system and control constraints in Simulink.

2. **Fine-tuning the simulated driveline to Real Vehicle**
    - Record and analyse vehicle data for system response.
    - Compare with the simulation environment and tune the plant model to match the vehicle
3. **Open loop analysis of the driveline system**
    - Development of bode plot to perform analysis.
    - Simulation of the open-loop model for various test conditions.
4. **Design and simulation of feedback control strategy**
    - Develop a feedback control system to dampen the oscillations at the resonant frequency.
    - Analysis of the performance.
    - Analysis of robustness and stability.
        (a) Sensor Noise
        (b) Hardware parameter variation
        (c) What happens if we could only decrease torque?
5. **Develop new and advanced control strategy to improve performance**

4 Scope and Boundaries:

Damping vibrations for the Chevrolet Bolt in the fore/aft direction

The vehicle investigated in this project is the Chevrolet Bolt EV. The variables used to construct the model and the data used to validate the model were from Bolt's specifications. A vehicle in motion has oscillations in all directions. For this project, we focused solely on the fore/aft direction of the vehicle, ignoring any dynamics in any other direction. The stiffness and damping coefficients for the components in the driveline were provided to model the fore/aft behavior and characteristics.

5 Project Plan:

Dependencies required to dampen longitudinal vibrations

The project plan is as follows:

1. **Simulation Environment:** Derivation of the state-space representation from the dynamic model after linearization.
2. **Correlate Simulated Driveline to Real Vehicle:** We are using bode plot analysis

to compare the open-loop driveline simulation to the physical vehicle. The response of the simulation model is calculated from the state-space representation of the model. The response of the physical vehicle is calculated from data collected during the constant speed frequency sweep test.

3. **Analysis of the Open-Loop response of the driveline system:** Using system identification tools and Fast Fourier Transformation, convert the time domain data to frequency domain data, and create bode plots to depict real-world data. Calculate the error by overlapping the model with the new bode plot.

4. **Basic Feedback Control to provide damping to the system:** Through the analysis of the bode plots. Identify feedback control strategies to reduce/eliminate vibrations throughout the driveline.

5. **Develop new control strategy/design to improve performance:** Using optimization and MPC, develop a new control design that has a safe set (control invariant set) capable of reducing vibrations across multiple frequencies and in partially known models.

6 Development and Evaluation

6.1 Driveline Model:

Dynamic model using mass-node analysis

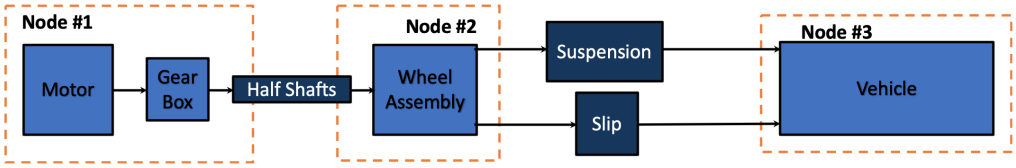


Figure 1: Vehicle Driveline Dynamic Model

The first step towards developing control strategy to provide damping in the longitudinal direction was to develop the vehicle dynamics and represent their behavior using mathematical equations. The team decided to initially develop a low-fidelity model that only considers components having major contribution to the longitudinal resonances. These vibrations are majorly generated due to rotational component of the drive train such as electric motor and wheel which affects user driving experience. Therefore, motor, wheel assembly, vehicle chassis and their intermediate connection are considered for the vehicle modeling.

To construct the vehicle driveline model, certain assumptions need to be made.

- Electric motor, wheel assembly and vehicle chassis have mass while other intermediate components such as shaft, gear, and suspension are assumed to be mass-less.

- Quarter car characteristics are used to model the vehicle dynamics

- Displacement and its derivative terms are considered in rotational coordinate axis.

For this project, the system of equations was derived based on the vehicle dynamic model shown in figure 1.

Since, only three components namely motor, wheel assembly and vehicle are assumed to have mass, we performed the nodal analysis with each mass system as a node. Therefore, the system is divided into 3 nodes and each node is analyzed individually. Before nodal analysis, the components connecting these nodes are analysed and represented using mathematical equations.

Dynamics of Connecting Components:

1. **Half Shafts**

   The connection between the electric motor and wheel assembly is established by half-shafts. It is considered as a spring-damper system. The spring and the damper torques from these half-shafts act on the Electric Motor Node and Wheel Assembly Node. Motor angular position and velocity, reduced by the gear ratio forms the input to the half-shafts.

   The spring and damper torque is given by:

$$T_{shaftspring} = k_{shaft}(\frac{\theta_m}{r_t} - \theta_w) \tag{2}$$

$$T_{shaftdamper} = b_{shaft}(\frac{\dot{\theta}_m}{r_t} - \dot{\theta}_w)) \tag{3}$$

   where r_t is gear ratio for gear set, $T_{shaftspring}$ is the torque generated by spring of the half shaft, $T_{shaftdamper}$ is the torque generated by damper of the half shaft, b_{shaft} is the damping coefficient of the shaft, k_{shaft} is the stiffness of the shaft, $\dot{\theta}_m$ is the angular velocity of motor, $\dot{\theta}_w$ is the angular velocity of the wheel.

2. **Suspension**

   The connection between the wheel assembly and vehicle chassis is established by suspension. It is considered as a spring-damper system. The spring and the damper torques from this suspension act on the wheel assembly node and vehicle chassis node. Wheel angular position and velocity forms the inputs to the suspension.

   The spring and damper torque is given by:

$$T_{suspspring} = k_{susp}(\theta_w - \theta_v) \tag{4}$$

$$T_{suspdamper} = b_{susp}(\dot{\theta}_w - \dot{\theta}_v)) \tag{5}$$

   where $T_{suspspring}$ is the torque generated by spring of the suspension, $T_{suspdamper}$ is the torque generated by damper of the suspension, b_{susp} is the damping coefficient of the suspension, k_{susp} is the stiffness of the suspension, $\dot{\theta}_w$ is the angular velocity of wheel, $\dot{\theta}_v$ is the angular velocity of the vehicle.

3. **Tire Slip**

An additional non-linear relation between the wheel assembly and vehicle chassis is given by tire slip(viscous slip). Viscous slip is a non-linear component which is defined as below [2]:

$$T_{tire} = mg\mu(\lambda)R_w \tag{6}$$

where m is the mass of the vehicle, g is the gravity constant, R_w is the radius of the wheel, and μ is the friction coefficient which is a function of the slip ratio, λ. A widely used longitudinal μ-λ relationship [3] is adopted as [2]:

$$\mu(\lambda) = c_1(1 - \mathrm{e}^{-c_2\lambda}) - c_3\lambda \tag{7}$$

where parameters of c_1, c_2, c_3 depend on the road conditions and those values are listed in table 1 [2].

Table 1: Tire model parameters for various surfaces and conditions

Surface Condition	c_1	c_2	c_3
Dry asphalt	1.2801	23.99	0.52
Dry concrete	1.1973	25.16	0.5373
Wet asphalt	0.857	33.82	0.347
Snow	0.1946	94.12	0.0646
Ice	0.05	306.3	0

Slip ratio, λ, can be expressed as:

$$\lambda = \frac{(\dot{\theta}_w - \dot{\theta}_v)}{\max(\dot{\theta}_w, \dot{\theta}_v)} \tag{8}$$

Detailed discussion on linearization is shown in the section 6.3.

Nodal Anlysis:

1. **Node 1**

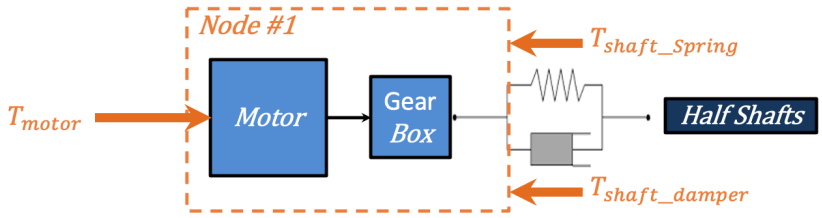


Figure 2: Node 1 Schematic Diagram

Node 1 comprises the motor and the gear set and connects to node 2 with the half shaft as shown in figure 2. The acceleration input from the driver as motor torque is the input to the electric motor (Node1). Addition to the input, the reaction torque from the spring and damper of the half-shafts act on the node. The equation describing node 1 is shown below:

$$J_m\ddot{\theta}_m = T_m - \frac{1}{r_t}(T_{shaftspring} + T_{shaftdamper}) \tag{9}$$

where J_m is motor inertia, $\ddot{\theta}_m$ is motor acceleration, T_m is motor torque, r_t is gear ratio for gear set, and $T_{shaftspring}$ is the spring torque of the half shaft, $T_{shaftdamper}$ is the spring torque of the half shaft. The equation describing half shaft is shown in (2) and (3).

2. **Node 2**

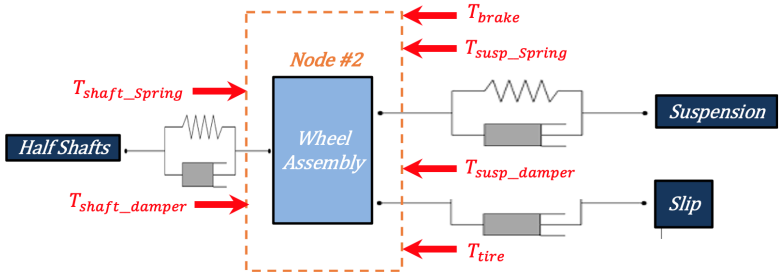


Figure 3: Node 2 Schematic Diagram

Node 2 comprises the wheel assembly which connects to Node 1 through half shaft and connects to node 3 through suspension. We assumed tire slip, extra torque applied to the wheel due to the speed difference between the wheel and vehicle chassis, to be placed parallel with suspension. The braking input from the driver as motor torque is the input to the electric motor (Node1). The schematic for node 2 is shown in figure 3. The equation describes Node 2 is shown below:

$$J_w\ddot{\theta}_w = (T_{shaftspring} + T_{shaftdamper}) - T_{suspspring} - T_{suspdamper}) - T_{tire} - T_{brake} \tag{10}$$

where J_w is wheel inertia, $\ddot{\theta}_w$ is wheel acceleration, and $T_{suspspring}$ is the spring torque of the suspension, $T_{suspdamper}$ is the torque of the suspension, T_{tire} is the tire slip torque, and T_{brake} is the brake torque applied to the wheel. The equations describing the suspension is shown in (4) and (5).

3. **Node 3**

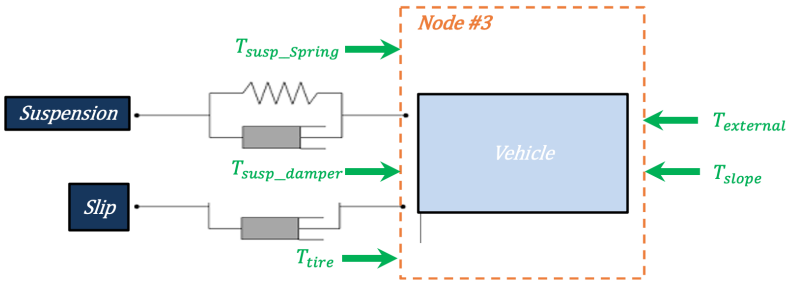


Figure 4: Node 3 Schematic Diagram

Node 3 comprises the vehicle chassis which connects to node 2 through suspension. The spring and damper torque from suspension and tire torque is transmitted to Node 3. The external torques due to wind drag and slope resist the forward motion of the vehicle. The translated motion of the vehicle is converted to the rotational coordinate system using $v = rw$. the schematic for node 3 is shown in figure 4. The equation describes Node 3 is shown below:

$$\ddot{\theta}_v m R_w^2 = T_{tire} + T_{suspspring} + T_{suspdamper} - T_{ext} - T_{slope} \tag{11}$$

where T_{ext} is the wind drag torque and T_{slope} is the torque applied to the vehicle due to road slope.

6.2 State Space Model:

State Space Representation of system dynamics

The derived vehicle dynamics without tire slip component was used to develop the equation of motion for the system dynamics and modelled using state space.

The states for the system dynamics are x:

- Motor Speed $\dot{\theta}_m$
- Wheel Speed $\dot{\theta}_w$
- Vehicle Speed $\dot{\theta}_v$
- Half-Shaft Spring torque $T_{shaftspring}$
- Suspension Spring torque $T_{suspspring}$

The inputs to the system dynamics are u:

- Motor Torque T_{motor}
- Brake Torque T_{brake}

- Road Gradient T_{slope}

The outputs of the system dynamics are y

- Motor Speed $\dot{\theta}_m$
- Wheel Speed $\dot{\theta}_w$

The state-space model is represented using equation:

$$\dot{x} = Ax + Bu \tag{12}$$

$$y = Cx \tag{13}$$

where matrix A represent the system dynamics and matrix B represent actuator dynamics. y is the system output governed by matrix C.

$$A = \begin{bmatrix} \frac{-b_{shaft}}{r^2 J_m} & \frac{b_{shaft}}{J_m r} & 0 & \frac{-1}{r * J_m} & 0 \\ \frac{b_{shaft}}{r J_w} & \frac{-(b_{shaft}+b_{susp})}{J_w} & \frac{b_{susp}}{J_w} & \frac{1}{J_w} & \frac{-1}{J_w} \\ 0 & \frac{-b_{susp}}{m r_w^2} & \frac{-(b_{susp}+b_{external})}{m r_w^2} & 0 & \frac{1}{m r_w^2} \\ \frac{k_{shaft}}{r} & -k_{shaft} & 0 & 0 & 0 \\ 0 & k_{susp} & -k_{susp} & 0 & 0 \end{bmatrix} \tag{14}$$

$$B = \begin{bmatrix} \frac{1}{J_m} & 0 & 0 \\ 0 & \frac{-1}{J_w} & 0 \\ 0 & 0 & \frac{-1}{m r_w^2} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{15}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{16}$$

6.3 Linearization of Driveline Model:

Damper analogy vs. First Order Taylor Expansion

Two types of linearization methods have been considered for this project. The first method was to model the non-linear component, viscous slip, as a damper. The second method was to apply the first-order Taylor expansion.

1. **Damper Analog**

   The plot for $\mu(\lambda)$ vs. λ for different road conditions is shown below in figure 5.

   As we can observe from the plot, the relationship between friction coefficient and slip ratio is linear when slip ratio is small. Vehicles under control normally have small slip ratio and therefore it is safe to assume that the vehicle for this project is operating within small slip ratio range. Thus, the linear relationship between friction coefficient

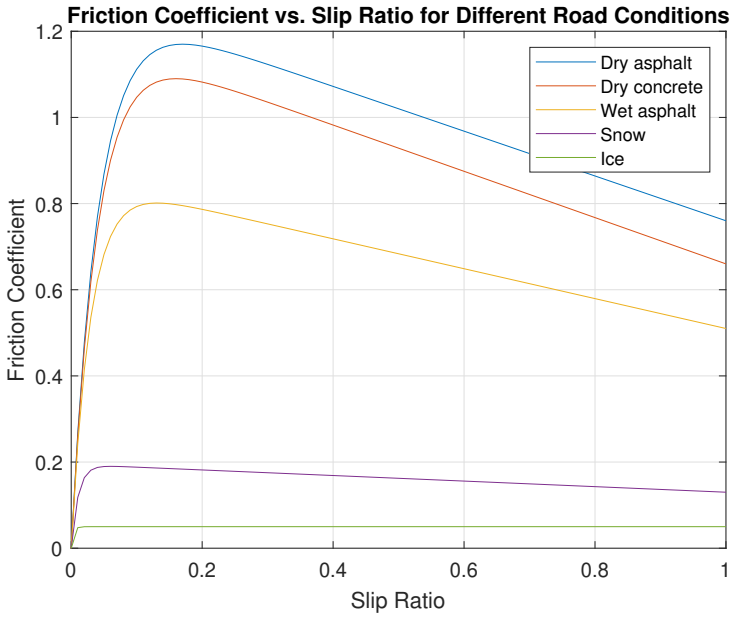


Figure 5: Friction Coefficient vs Slip Ratio for Different Road Conditions

and slip ratio can be derived from the slope, defined as a, for different road conditions using the data from figure 5.

The tire model defined in equation (6) can then be expressed as:

$$T_{tire} = mga\lambda R_w \tag{17}$$

Using λ from equation (8) into equation (17), T_{tire} is then expressed as

$$T_{tire} = mgR_w a \frac{(\dot{\theta}_w - \dot{\theta}_v)}{\max(\dot{\theta}_w, \dot{\theta}_v)} \tag{18}$$

The general equation used to calculate the force by a damper is:

$$F = b\Delta v \tag{19}$$

where b is the damping coefficient and Δv is the speed difference. Comparing equation (18) and equation (19), the damping coefficient for viscous slip can then be defined as:

$$b_{tire} = \frac{mgR_w a}{\max(\dot{\theta}_w, \dot{\theta}_v)} \tag{20}$$

By giving the operating speeds for both the vehicle and the wheel, we can then calculate b_{tire} using equation (20). The equation to calculate T_{tire} is then linearized.

2. **Taylor Expansion**

We also applied first-order Taylor expansion to linearize our system of equations around a set of equilibrium points. Our system of equations have the form:

$$\dot{x} = f_i(x_1, x_2, ..., x_n, u_1, u_2, ..., u_m) \tag{21}$$

where x_n are the states and u_m are the inputs. We found $\bar{x}_1, \bar{x}_2, ..., \bar{x}_n$ and $\bar{u}_1, \bar{u}_2, ..., \bar{u}_m$ so that

$$\dot{\bar{x}} = 0 \qquad \forall i \in \{i = 1, 2, ...n\} \tag{22}$$

We then applied first order Taylor expansion [4] around the equilibrium points found in equation (22).

$$\begin{aligned} \dot{x} &= \underbrace{f(\bar{x}, \bar{u})}_{=0} + \underbrace{\left.\frac{\partial f}{\partial x}\right|_{\substack{x=\bar{x} \\ u=\bar{u}}}}_{=A} \underbrace{x - \bar{x}}_{\Delta x} + \underbrace{\left.\frac{\partial f}{\partial u}\right|_{\substack{x=\bar{x} \\ u=\bar{u}}}}_{=B} \underbrace{u - \bar{u}}_{\Delta u} \\ \Rightarrow \dot{x} - \underbrace{\dot{\bar{x}}}_{=0} &= \Delta\dot{x} = A\Delta x + B\Delta u \end{aligned} \tag{23}$$

The state-space matrices, A and B, can then be acquired from equation (23). For the linear discrete model, the equation used to calculate the states for the next time step is shown below:

$$x_{k+1} = A_d(x_k - \bar{x}) + B_d(u_k - \bar{u}) + \bar{x} \tag{24}$$

where k is the time step, A_d and B_d is the discrete state-space matrices.

3. **Comparing Damper Analogy and Taylor Expansion**

Both methods helped to linearize the system of equations. However, the damper analogy method only holds when slip ratio, λ, is small while the Taylor expansion method works for slip ratio for the entire range. Moreover, the implementation of Taylor expansion is more straightforward than the damper method. Industry usually applies first-order Taylor expansion to eliminate non-linearity within their system. Therefore, due to the limitation of the damper method and the advantages of the Taylor expansion, we decided to use Taylor expansion to linearize our system.

4. **Further Discussion on Taylor Expansion: Single vs. Iterative Linearization**

To have the most accurate model, we can linearize the model at every time step (i.e. 1 ms) during the simulation. However, the time required to run such a simulation is too long to be considered practical. It took around 16 minutes to run one second simulation in Simulink. To reduce the simulation time, we considered two alternatives. One was to linearize the model at one constant speed (i.e. 20 mph). The second one was to still iterative linearize the model but at wider time interval with varied speeds. To compare the performances for different linearization methods, we plotted the open loop response for single linearization and iterative linearization (linearized every 5 second) and compared them with the non-linear response given the same initial conditions and inputs. The results are shown in figure 6 and figure 7. As we can observe from figure 6, both the motor and wheel speeds matched the non-linear model at the beginning of the simulation. As the simulation time increased, however, the response for the single linearization model started to deviate from the non-linear model. For the iterative

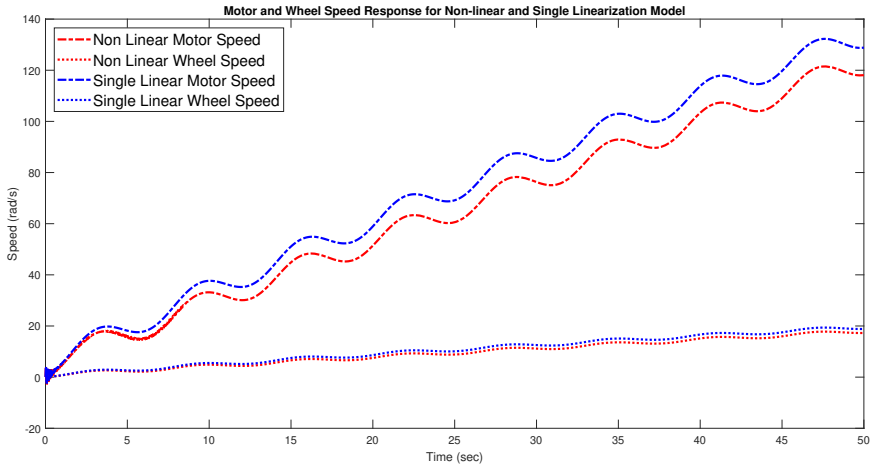


Figure 6: Motor and Wheel Speed Response for Non-linear and Single Linearization Model

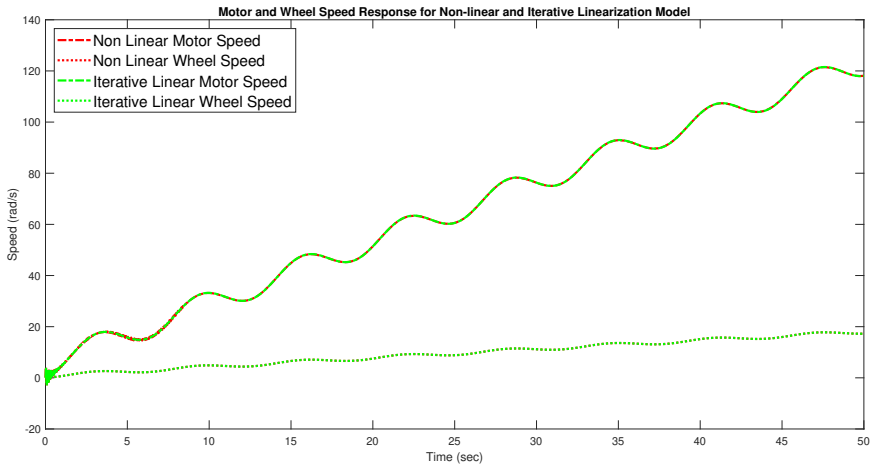


Figure 7: Motor and Wheel Speed Response for Non-linear and Iterative Linearization Model

linearization method shown in figure 7, both the motor and wheel speeds overlapped the non-linear model response for the entire simulation.

The iterative linearization took around 10 seconds to run 50 seconds of simulation and the single linearization took around 2 seconds to run 50 seconds of simulation. Even though it took sightly longer time to run the simulation with the iterative method, the accuracy of the simulation was significantly better than the single linearization method. Thus, with the advantage of being higher accuracy and less computational intensive for the iterative Taylor expansion, the result from iterative first-order Taylor expansion was used for the rest of this project.

5. **Resulting State-Space Representation of our model**

Linearizing the model with tire slip using the iterative Taylor linearization technique, similar A,B,C matrices were derived and modelled in a state space representation. These matrices are dimensional consistent with the model without slip. However, the values for the A matrix varies during the simulation due to the change in the linearization points. The A matrix for the model with slip is shown below:

$$A(ws, vs) = \begin{pmatrix} \frac{-b_{shaft}}{r^2 J_m} & \frac{b_{shaft}}{J_m r} & 0 & \frac{-1}{r * J_m} & 0 \\ \frac{b_{shaft}}{r J_w} & * & ** & \frac{1}{J_w} & \frac{-1}{J_w} \\ 0 & * * * & * * ** & 0 & \frac{1}{m r_w^2} \\ \frac{k_{shaft}}{r} & -k_{shaft} & 0 & 0 & 0 \\ 0 & k_{susp} & -k_{susp} & 0 & 0 \end{pmatrix} \tag{25}$$

where

$$* = \frac{-(b_{shaft} + b_{susp} - mgr_w(\frac{(\frac{c_3}{ws} + c_3(vs - ws))}{ws^2} - c_1 \mathrm{e}^{\frac{c_2(vs-ws)}{ws}}(\frac{c_2}{ws} + \frac{c_2(vs-ws)}{ws^2}))}{J_w}$$

$$** = \frac{b_{susp} - mgr_w(\frac{c_3}{ws} - \frac{c_1 c_2 \mathrm{e}^{\frac{c_2(vs-ws)}{ws}}}{ws})}{J_w}$$

$$* * * = \frac{b_{susp} - mgr_w(\frac{(\frac{c_3}{ws} + c_3(vs - ws))}{ws^2} - c_1 \mathrm{e}^{\frac{c_2(vs-ws)}{ws}}(\frac{c_2}{ws} + \frac{c_2(vs-ws)}{ws^2})}{m r_w^2}$$

$$* * ** = \frac{-(2b_{external} r_w^2 vs - mgr_w(\frac{c_3}{ws} - \frac{c_1 c_2 \mathrm{e}^{\frac{c_2(vs-ws)}{ws}}}{ws}) + b_{susp})}{m r_w^2}$$

where ws is the wheel speed, vs is the vehicle speed, and c_1, c_2, c_3 are parameters depend on road condition shown in Table 1 in the section 6.1. The A matrix varied in the simulation. Its value depended on the value of wheel speed and vehicle speed during the simulation.

The B and C matrices stayed the same as the model without slip and their value

remained constant during the simulation. The B and C matrices are shown below:

$$B = \begin{bmatrix} \frac{1}{J_m} & 0 & 0 \\ 0 & \frac{-1}{J_w} & 0 \\ 0 & 0 & \frac{-1}{mr_w^2} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{26}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{27}$$

6.4 System Identification and Model Correlation:

Comparision of derived Open-Loop plant with real-world data using Fast Fourier Transformation

The model described in Section 6.1, 6.3, and 6.2 is used to create our open-loop plant G(z), where z describes time in the discrete domain, as shown figure 8. The accuracy of our open-loop model is characterized through the process of system identification where the bode plot of G(z) is compared with the bode plot created using real world data through Fast Fourier Transformation. The real world data is protected by a *Non-Disclosure Agreement with General Motors* and was collected using a frequency sweep test.

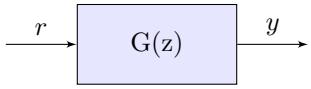


Figure 8: Open-Loop Model

The steps to perform system identification are:

1. **Create an Open-Loop Model in discrete time:**
   Sections 6.1 till 6.2 describe in detail the process for creating a plant model in continuous time. Applying zero-order hold, we can convert this model to discrete time to begin the system correlation process.

2. **Collect data using frequency sweep test:**
   At General Motors Proving Grounds at Milford, Michigan, the team collected data from a Chevrolet Bolt while running an open-loop (no feedback control) frequency sweep test. A frequency sweep test involves increasing the frequency of a sinusoidal input with fixed amplitude range to highlight and identify resonant frequencies.

3. **Filter data according to test frequency:**
   The data collected in time series data was filtered into buckets mapping multiple data points to relevant frequencies. This process also removed unobserved frequencies resulted from the inherent process of increasing frequency during the test.

4. **Perform fast fourier transformation in filtered data:**
   Fast Fourier Transformation algorithm used in this step converts signal data from

time domain to frequency domain. The filtered data calculated in the previous step is converted using the matlab command *fft*. This Fast Fourier Transformed data can now be used to identify the system.

5. **Calculate magnitude and phase of transformed data:**
   A bode plot comprises of a Magnitude and a Phase plot. Combined, These plots describe the system exhaustively. The magnitude is calculated by converting the response of the system into decibel values by taking $20\log(Magnitude)$. The phase on the other hand involves calculating phase values from the fft data and wrapping between $-\pi$ and π.

6. **Plot the magnitude and phase superimposing open-loop bode plot:**
   After calculating the magnitudes and phases, we plot it using *semilogx*. This converts the frequencies into logarithmic scale values making it easier to analyze.

7. **Calculate model correlation to identify differences between simulated and real world data:**
   The last step of the process is to plot our derived open-loop plant bode plot superimposing the semilogx plots and correcting errors in our model.

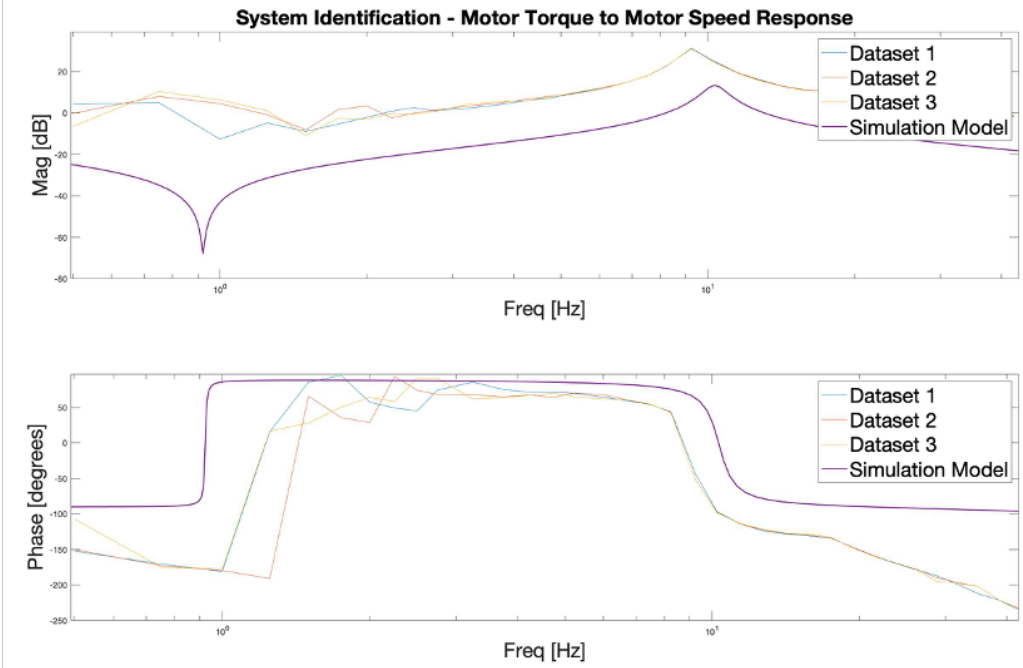


Figure 9: Bode Plot from the in-vehicle testing

Figure 9 represents the bode plot identified by in-vehicle testing. The plot shows the graph obtained from all the three data-sets (in-vehicle testing) and simulation

environment. The results from different data-sets overlap at frequency above 1Hz but differ from each other at frequency lower than 1Hz. Since it is difficult to calibrate and log data at lower frequency, we could conclude that knowledge about the real vehicle only above 1Hz frequency is accurate. Therefore, we aimed to match the simulation bode plot to the data-set bode plot at higher frequency.

The simulation bode plot characteristics is very similar to one obtained by testing which assured the correctness of the dynamics considered. But, there is a difference in magnitude for which we had to tune the system. A correction factor of 10 on our B matrix (of the state space) was required to compensate for the magnitude difference. Finally, there was slight difference in the resonance frequency and the curve shape, and to further match the plot the system characteristics such as mass, spring constant and damping coefficient was tuned. We analysed the bode-plot response by individually

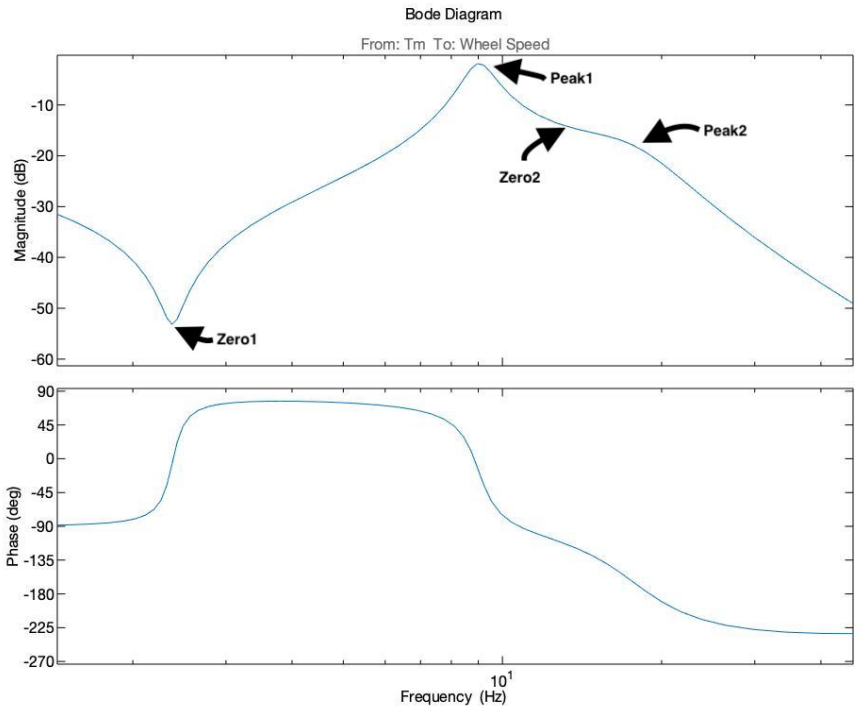


Figure 10: Wheel Speed Bode Plot Correlation

changing characteristics figure 10:

- Changes in Vehicle Mass affected the relative horizontal distance (frequency difference) between the first (highest) peak and zero.
- Changes in Wheel Mass affected the relative horizontal distance (frequency difference) between the second peak and zero.

- Spring Stiffness Coefficient mainly affected the frequency of the peak (Shaft spring constant affected the 1st peak (highest) while the suspension spring constant affected the 2nd peak)
- Damping Coefficient affected the peak magnitude (Shaft damping coefficient affected the 1st peak (highest) while the suspension damping coefficient affected the 2nd peak)

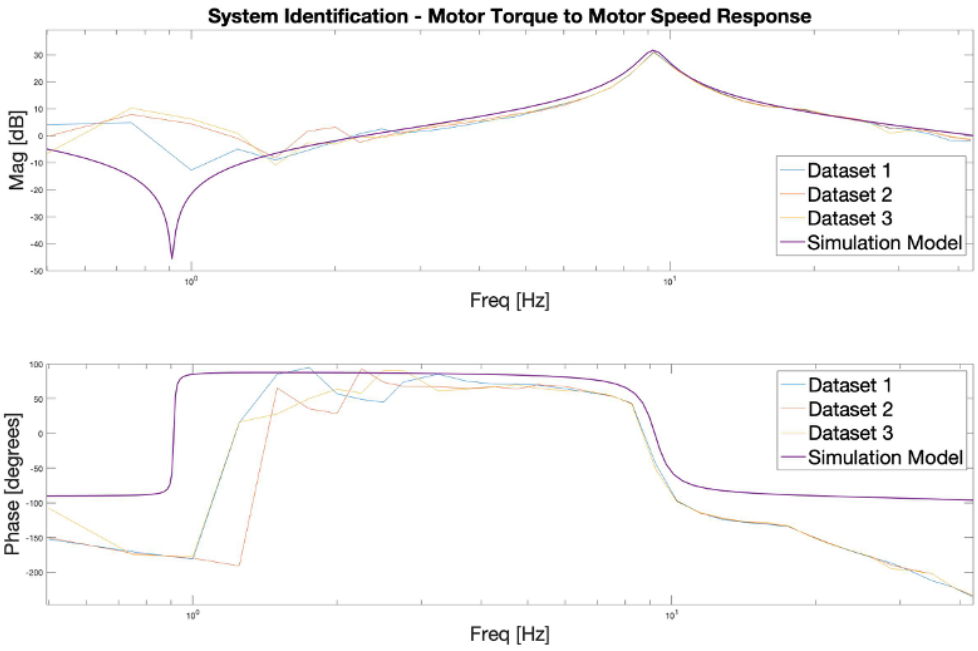


Figure 11: Bode Plot Correlation

Based on these analysis, we were able finally tune these parameters to match the simulation model behavior to the real vehicle as shown in figure 11 Following table represent the changes in the characteristics:

Characteristic	Original Value	Tuned Value
Vehicle Mass	4000	3875
Wheel Moment of Inertia	4.2	4.0
Half-Shaft Spring Coefficient	120	120
Half-Shaft Damping Coefficient	10	10
Suspension Spring Coefficient	1500	700
Suspension Damping Coefficient	200	220

7 Classical Control Strategies:

Classical Control techniques to minimize vibrations throughout the electric driveline

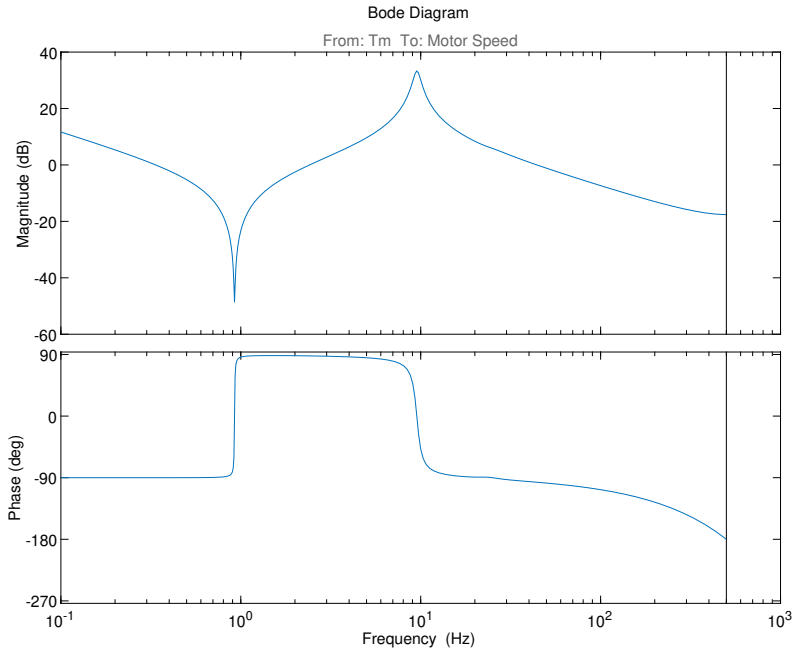


Figure 12: Open-Loop Bode Plot - Resonance from Matlab Script

The next step towards controller design is to formulate the problem statement to dampen the resonance. It was observed the resonance peak was because of the Half-Shafts characteristics which formed the connection between the motor and wheel. In an ideal vehicle configuration (without vibrations and losses), motor speed should be equal to the Wheel Speed multiplied by gear ratio $\dot{\theta_m} - r * \dot{\theta_w} = 0$. But because of the system losses mainly due to Half-Shafts, $\dot{\theta_m} - r * \dot{\theta_w} \neq 0$ and the difference account for effect of the resonance. By minimizing the difference $\min(\dot{\theta_m} - r * \dot{\theta_w})$ using a compensator, we can reduce the resonance peak.

Problem 1. *Develop a controller to dampen the resonance amplitude of the system shown in figure 12:*

$$Y = \dot{\theta}_m \tag{28}$$

$$U = T_m \tag{29}$$

$$R = \dot{\theta}_w \tag{30}$$

$$E = \dot{\theta}_m - r * \dot{\theta}_w \tag{31}$$

$$U = T_m - K * E \tag{32}$$

where, Y is the system output, $\dot{\theta}_m$ is the motor speed, U is the controllable system output,

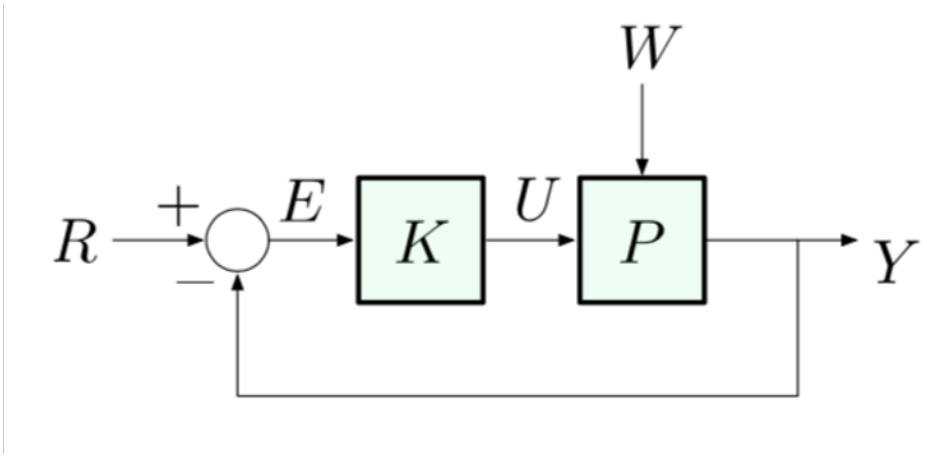


Figure 13: Standard Feedback Loop

T_m is the motor torque, R is the reference signal, $\dot{\theta}_w$ is the wheel speed, E is the measured error, and K is the gain for our controller.

Controlling a system with multiple inputs and multiple outputs can be challenging. We will be approaching the above problem by two methods.

- Classical Control: Develop a PID controller and tune the gains for the desired behavior. The linearized dynamic model will be used for this section.
- Advanced Control: Developing a Linear-Quadratic Regulator to find the gain for optimal damping of the system. Section 8 dives deeper into this area.

The vehicle is modeled in the time domain as a second order differential equations, where the system depends on previous system state, input and time. As time evolves, the state of the system and response changes. It becomes really difficult to solve this problem efficiently or observe the behavior in time domain.

To counteract this problem, classical control theory [5] uses the Laplace transform to change an Ordinary Differential Equation (ODE) in the time domain into a regular algebraic polynomial in the frequency domain. Once a given system has been converted into the frequency domain it can be manipulated with greater ease.

7.1 Simulink Model Overview - Design Considerations

To design the controller and simulate the system response, we developed a Simulink model. The model consists of following blocks:

- "Input Block" which consist of input signals from Motor Torque, Brake torque and Slope Gradient
- "Vehicle Plant" which described the vehicle behavior using state space equation

- "Signal Output Block" where the data of Motor Speed and Wheel Speed is obtained through a trigger system.
- "Error Block" where the error between the output and reference signal is calculated.
- "Controller Block" minimizes the error and the controlled output manipulates the signal from the input block.

1. **Input Block:** The Input consist of inputs from Motor Torque, Brake Torque and Road Gradient. To simplify the problem for the initial simulation, the brake torque and road gradient are constant 0 input. The input used for the frequency sweep test while logging data from the real world testing is recreated on Simulink shown in figure 14 and is used for the motor torque input. The figure 15 demonstrates the motor torque input for a 10 second simulation. The frequency of the input constantly increases as time progresses. A similar input was developed for a 100 second simulation.

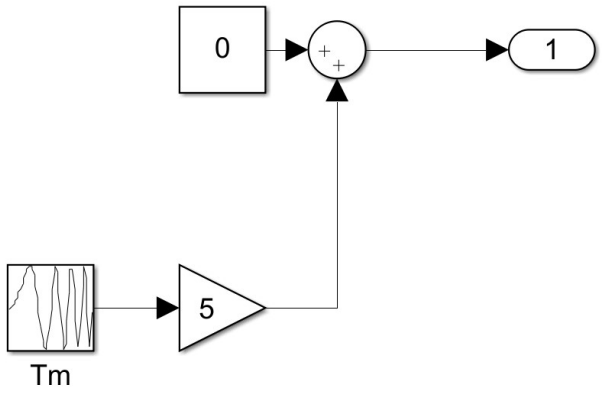


Figure 14: Motor Torque Frequency Sweep Block

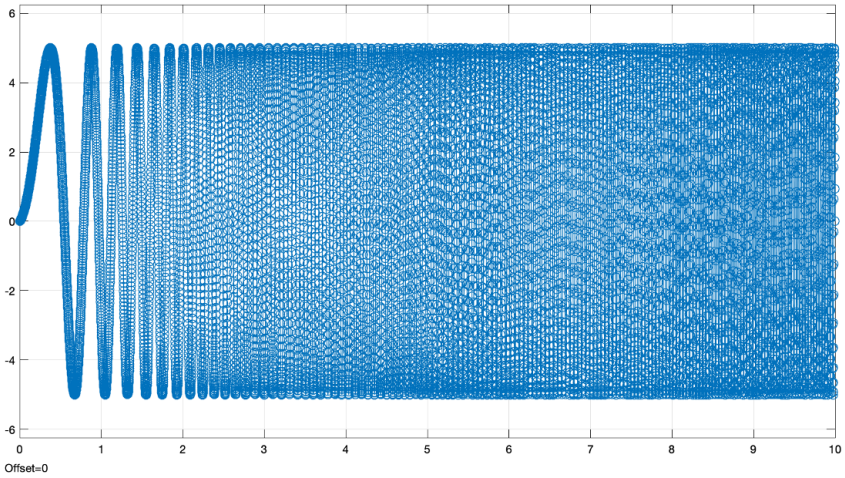


Figure 15: Motor Torque Input

2. **Vehicle Plant:** This block correlates the behavior of actual vehicle on the simulation ground. Vehicle Plant can be represented using various methods but we analysed the following two methods:

   - Transfer Functions: The plant can be represented using a transfer function for single input single output system. For a multi-input multi-output system, i.e. 3 inputs and 2 outputs for the vehicle model, 6 transfer functions can be used to describe the system. But the transfer function fails to represent the inter-related dynamics among different inputs and outputs. Since, the system dynamics plays a vital role for the problem in hand, we decided to represent the plant using state space representation.

   - State Space Representation: The plant is represented using a discrete state space block sampled at 1ms. The state space representation for the model without slip (Linear Model) and the one with slip (Non-Linear model linearized using Taylor Series) are different.
     Figure 19 shows our Model without slip which is a linear model can be represented using the standard state-space block diagram. Since the linear model is low-fidelity and faster, we decided to design the controller using the model without slip and then incorporate the results onto the Non-linear model with slip shown in section 7.3.
     Figure 27 shows our Model with slip requires re computation of A and B matrices at different time steps which complicates the problem and make it computationally slower. Along with the controller design on the model without slip, we simultaneously worked on efficiently developing a model with slip.

3. **Output Signal Block:**

   The motor speed and wheel speed output from the plant block is the input to the Output Signal Block shown in figure 16. Since the plant is sampled at 1ms, Motor

Speed and Wheel Speed data is obtained ever 1ms. But in the actual vehicle, speed sensors are used to measure the motor speed and wheel speed which are sampled every 5ms and 25ms respectively. To replicate the sensor refresh rate in the simulation environment, a trigger system is developed.

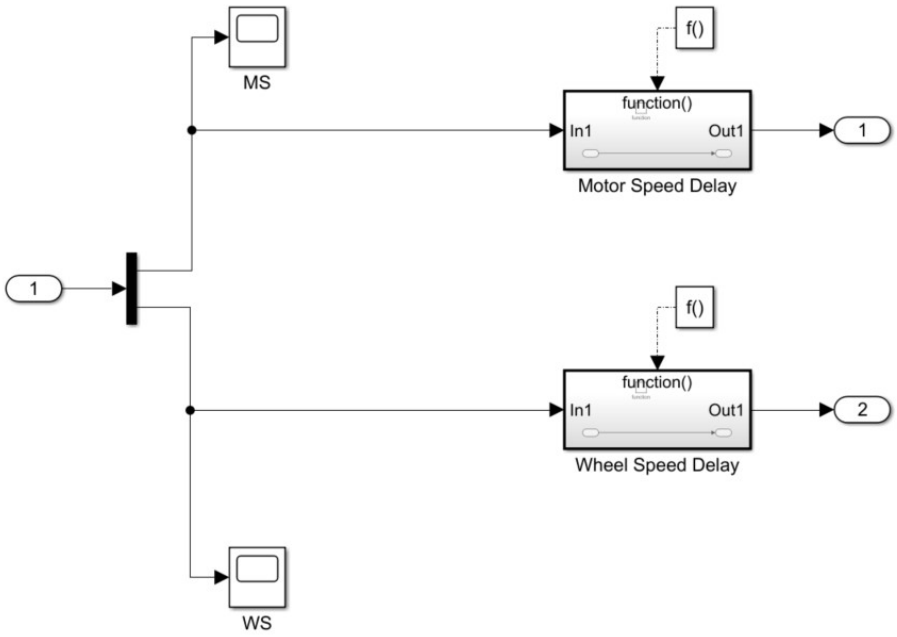


Figure 16: Output Block

4. **Error Block:** This block shown in figure 17 calculates the error between the output (Motor Speed) and reference (Wheel Speed) signal. The wheel speed data from the Output Signal Block is multiplied by the gear ratio r and this signal serves as the reference signal. The motor speed data from the Output Signal Block serves as the output signal.

5. **Controller Block** This block architecture aims to minimize the error from the error block by manipulating the input Motor torque. We developed different iterations for the controller block design which is explained in the section 7.2.

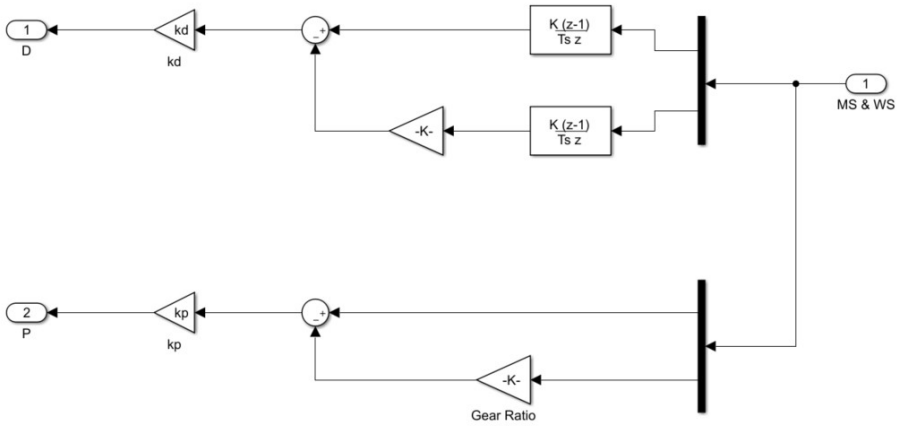


Figure 17: Error and Controller Block

The final Simulink model is shown in figure 33. The output from the "Input Block" is controlled using the output from the "Controller Block" and serves as the input to the "Plant". The "Plant" output is given to the "Output Signal Block" where it is sampled. The "Error Block" calculates the error of the sampled signal and serves as an input to the "Controller Block".

7.2 Exclusion of Non-Linear Slip

Effectively controlling the multiple input multiple output (MIMO) nature of our system involved a deep understanding of its behavior. For this reason, we chose to remove complexities and progressively add them back in our iterations using state space representation instead of transfer functions. This section describes the control strategies of our system without the inclusion of tire slip non-linearity. The methodology is described below:

Methodology: Starting with the vehicle plant model, shown in figure 19, the controller design underwent three iterations for documenting its closed-loop response.

1. **Open-Loop Response:** The open-loop response of the vehicle plant model without tire slip provides us with three insights. First, it acts as the base for our controller. Second, we can parameterize the effectiveness of our control effort. Third, the open loop response will allow us to analyse the damping characteristics of our controller. As such, we begin by setting our controller gain values to zero.

$$K = 0 \tag{33}$$

2. **Proportional Control:** Adding a proportional controller answers two questions. First, is our model controllable with a simple 'P controller'? Second, how sensitive is our model with respect to proportional gains? The error used for this method is described in the previous section. The P-Controller is defined below.

$$K = k_p * E; \tag{34}$$

3. **Proportional and Derivative Control:** To reduce the overshoot and settling time even further to meet our desired specifications, we added a derivative controller over our proportional controller to form a PD control design. The challenge with this step was that we did not have access to the derivative states and thus had to create them from our outputs. In the actual vehicle, we do not have access to these internal states and thus must use the outputs for our error calculation. The PD-Controller is defined below.

$$K = k_p * E + k_d * \dot{E}; \tag{35}$$

where, K is our controller with k_p as the proportional gain and k_d as the derivative gain illustrated in figure 17.

7.2.1 Simulink Model

For all intents and purposes, the Simulink model in this section acts as the template for further complications. The complete Simulink model is shown in figure 18.

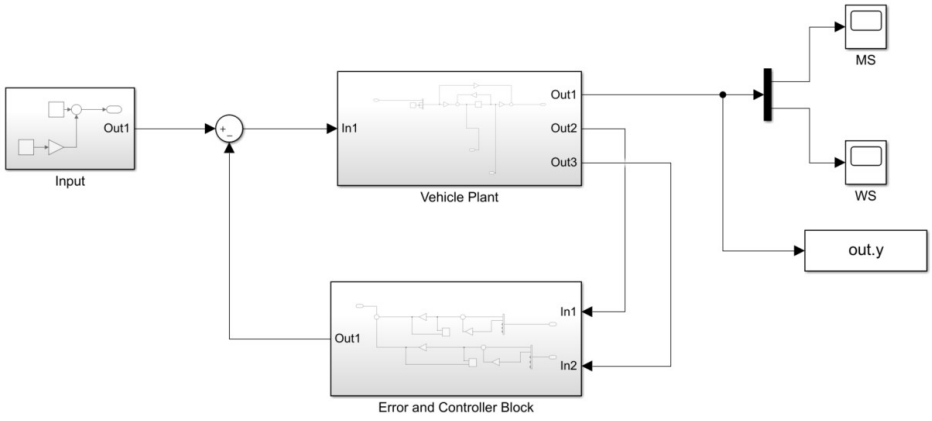


Figure 18: Simulink Model without Tire Slip

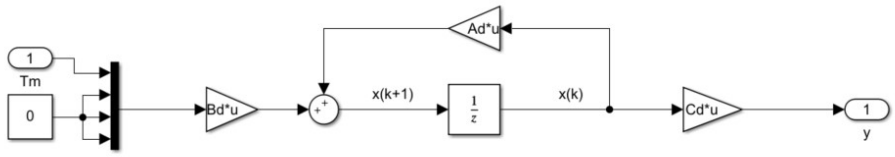


Figure 19: Vehicle Plant Model in Simulink without Tire Slip

7.2.2 Results and Discussion

1. **Open-Loop Model:** As seen in figure 20, the Motor Speed balloons around 32 seconds of simulation runtime to around ±250 rad/s. This is characterized by the

resonance witnessed in the bode plot shown in figure 21. The resonant frequency around 9.5 Hz is the target we will aim to reduce.

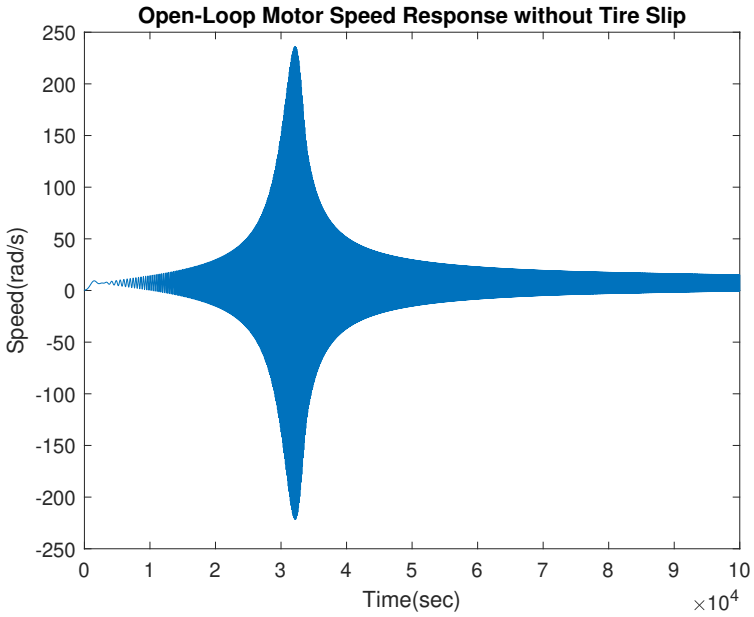


Figure 20: Open-Loop Motor Speed Response without Tire Slip

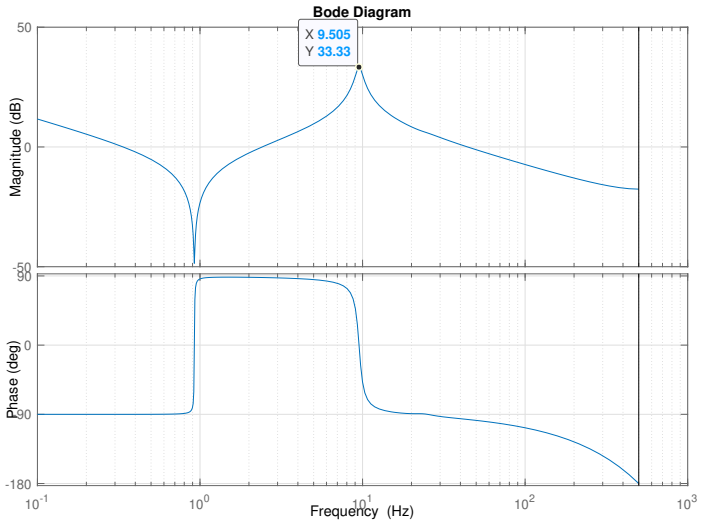


Figure 21: Open-Loop Bode Plot without Tire Slip

2. **Proportional Control:** Setting k_p to 0.3, we notice improvements to the resonant peak in figure 22 and a flatter curve around the target of 9.5 Hz. Larger gain values increase damping, significantly reducing The maximal motor speed value at the target frequency. However, by increasing damping of this system, we end up reducing drivability as it disconnects the driver from feeling the vehicle's response. Keeping these factors in mind, k_p values are limited between [0 5].

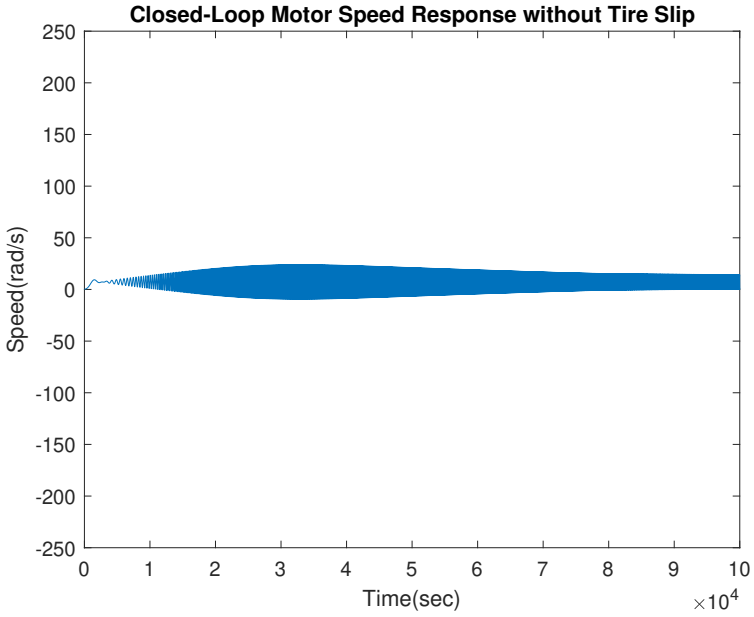


Figure 22: Closed-Loop Motor Speed Response without Tire Slip with $k_p = 0.3$

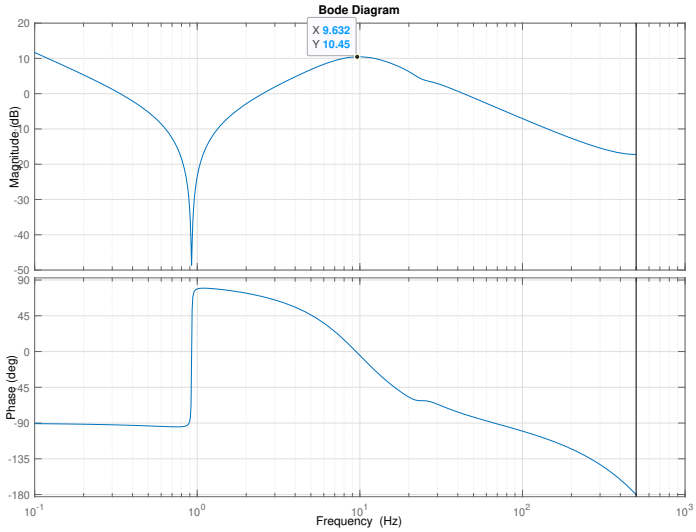


Figure 23: Closed-Loop Bode Plot without Tire Slip with $k_p = 0.3$

3. **Proportional and Derivative Control:** The improvements from solely adding a proportional gain was promising however, to decrease overshoot and the settling time, we added a derivative k_d gain as well. The figure 24 illustrates the response improvement with k_d set to 0.2. The magnitude at 9.5 Hz has also reduced by around 4 dB shown in figure 25.

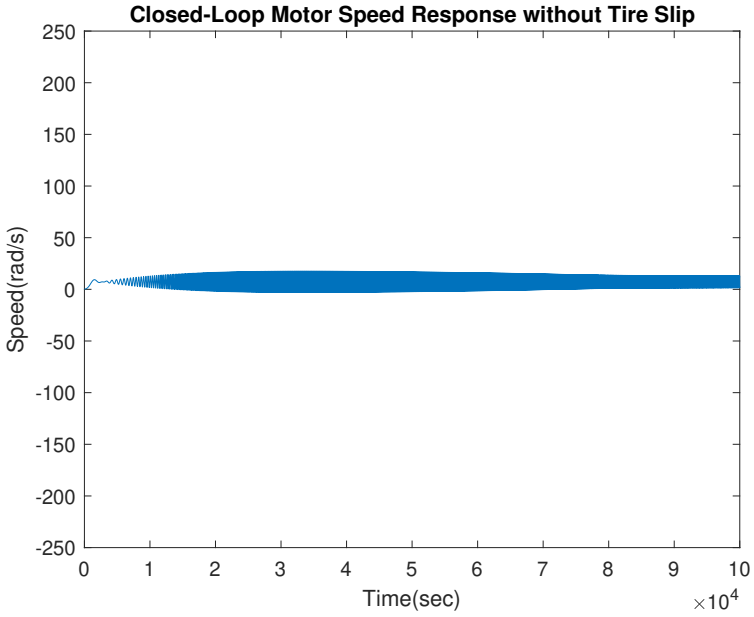


Figure 24: Closed-Loop Motor Speed Response without Tire Slip with $k_p = 0.3$ and $k_d = 0.2$

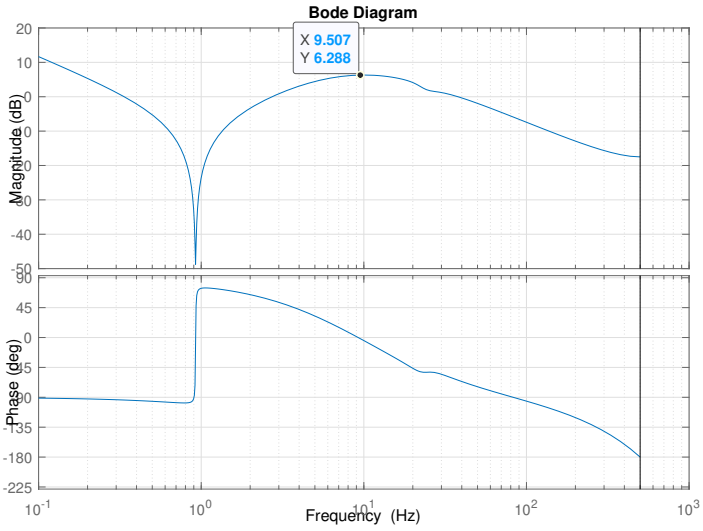


Figure 25: Closed-Loop Bode Plot without Tire Slip with $k_p = 0.3$ and $k_d = 0.2$

With desired levels of damping on the system baseline, we proceed to add the non-linear tire

slip as our first complication in our system. The tire slip alters our vehicle plant formulation and its implementation in Simulink by leveraging linearized state-space representation from section 6.2 and is described in detail in the next section.

7.3 Inclusion of Non-Linear Slip

In this section, we will discuss the construction of Simulink model with tire slip and illustrate our open-loop and close-loop results.

7.3.1 Simulink Model

The Simulink model for model with tire slip, shown in figure 26, had identical input block, output block, controller block, and error blocks as the model without tire slip.

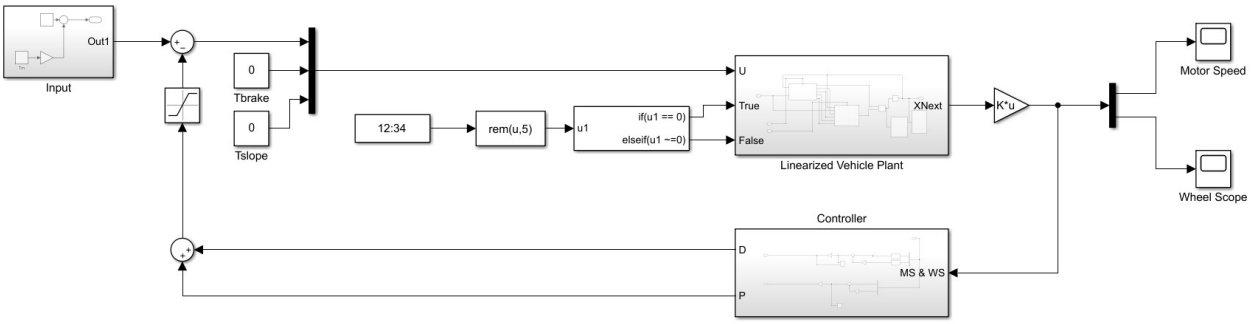


Figure 26: Simulink Model with Tire Slip

The only difference between the two models was the vehicle plant. Vehicle plant as shown in figure 27 included the nonlinear tire slip component which was then linearized every five seconds using the iterative Taylor linearization techniques discussed in Section 6.3. User-defined functions to find equilibrium points and discrete A and B matrices was called within the MATLAB script block for every five seconds.

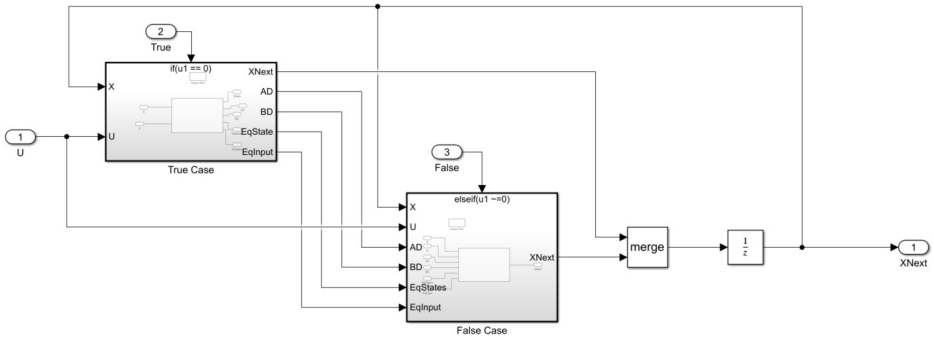


Figure 27: Vehicle Plant for Simulink Model with Tire Slip

7.3.2 Results and Discussion

1. **Open-Loop Model**

To analyze the open-loop response, we first set the controller to zero to get the open-loop motor speed response as shown in figure 28. As we can see from the plot, the motor speed reached the peak speed at around 32 seconds of the simulation. The input of the simulation was a sine wave with 0.299 Hz frequency increment per second. Therefore, the resonance frequency was at around 9.6 Hz based on the observation of the time-domain response which was verified later using the bode plot.

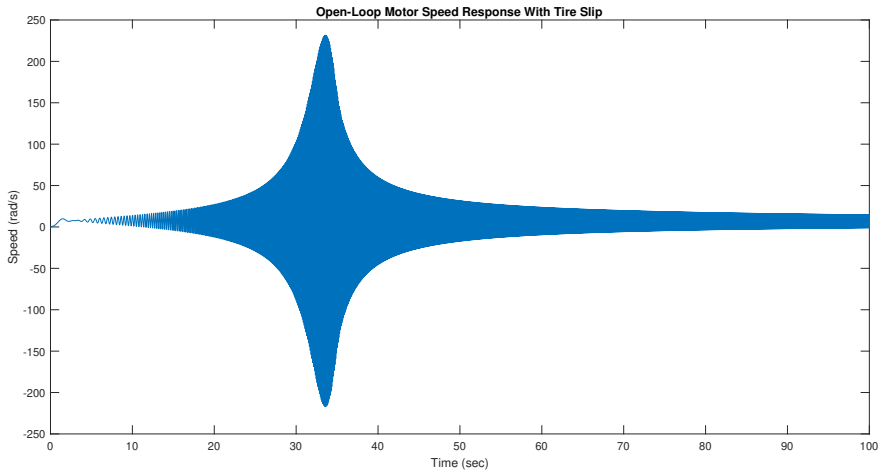


Figure 28: Open-Loop Motor Speed Response for Model with Tire Slip

The resonance frequency can be verified with the Bode plot shown in figure 29. As we can observe from this plot, the resonance was at around 9.5 Hz. Our goal was to reduce the magnitude at that resonant frequency using the controller in equation (35).

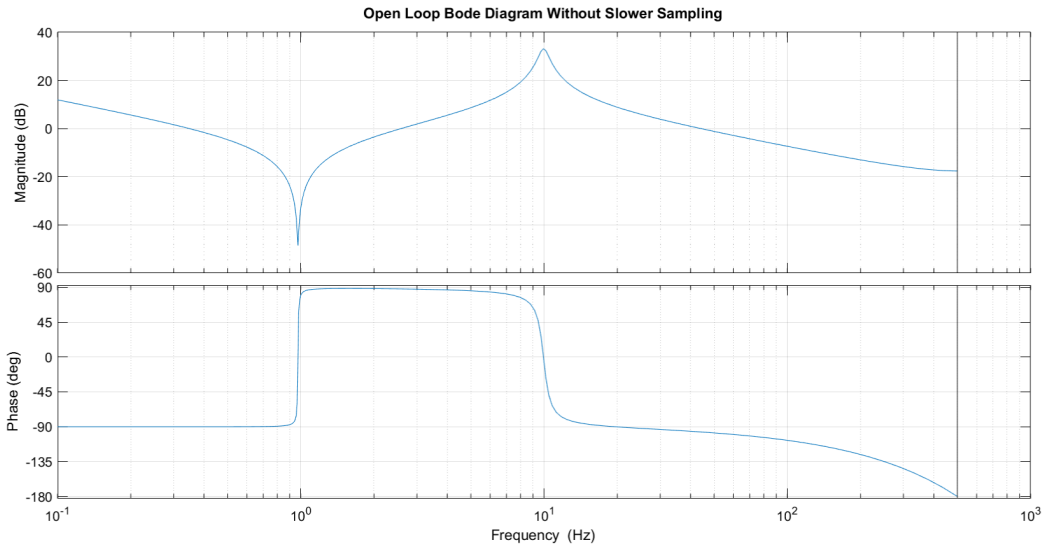


Figure 29: Open-Loop Bode Plot for Model with Tire Slip

2. **Close-Loop Model**

To reach our desired damping ratio range, 0.7 to 0.9, we varied the value of the proportional controller and calculated the damping factor directly from the Bode plot using the '3dB method'. We then checked if we satisfy the design specifications by calculating the damping ratio. The frequency at resonance, f_0, and the frequencies, f_1 and f_2, corresponding to the magnitudes that are 3dB smaller than the peak magnitude need to be extracted as shown in figure 30.

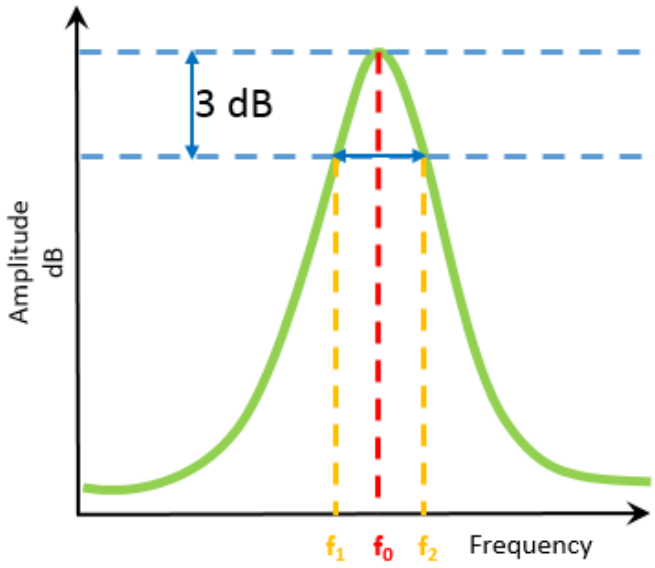


Figure 30: 3db Method to Find Damping Ratio
[6]

The damping factor, Q, and the damping ratio, ζ was then calculated using the equation shown below

$$Q = \frac{f_0}{f_2 - f_1} \tag{36}$$

$$\zeta = \frac{1}{2Q} \tag{37}$$

When k_p was in the range of [0.3,0.37], based on our simulation result and the calculation using the 3dB method, the damping ratio was in the range of [0.7,0.9]. The close-loop motor speed response for $k_p = 0.3$ is shown in figure 31

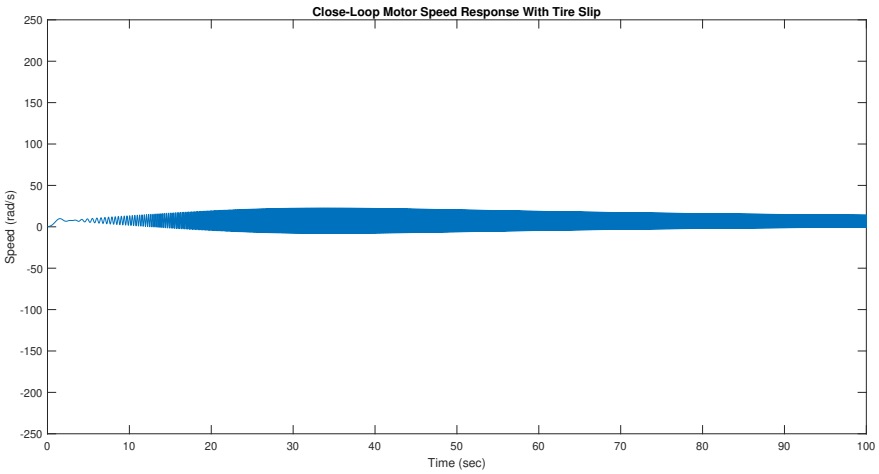


Figure 31: Close-Loop Motor Speed Response with Tire Slip when $k_p = 0.3$

The close-loop bode plot for $k_p = 0.3$ is shown in figure 32

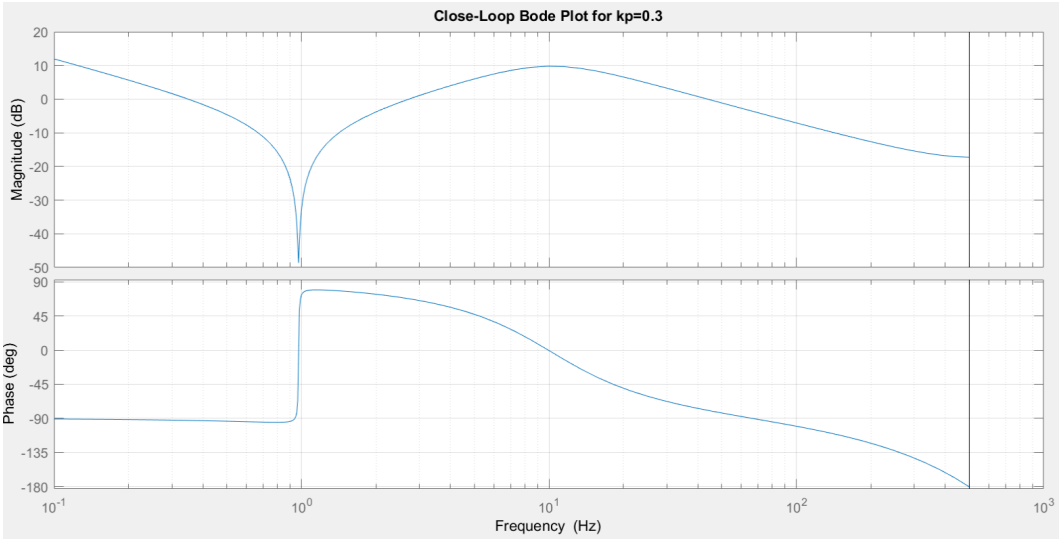


Figure 32: Close-Loop Bode Plot with $k_p = 0.3$

Comparing figure 28 with figure 31, we observed that the peak motor speed at the resonance dropped from around 230 rad/s to around 23 rad/s. The same trend can also be observed by comparing figure 29 with figure 32. The peak magnitude at the resonance decreased significantly, and the bode plot at around 10 Hz was a lot smoother after applying the controller

7.4 Adding Sensor Refresh Rates

To better simulate the real world application of our model, we also considered data refresh rate in our Simulink model. The refresh rate for motor speed and wheel speed sensor were 5 ms and 25 ms respectively. We used the Simulink model in section 7.4 and added a trigger system to simulate the different refresh rate. The overall Simulink model considering refresh rate is shown in figure 33

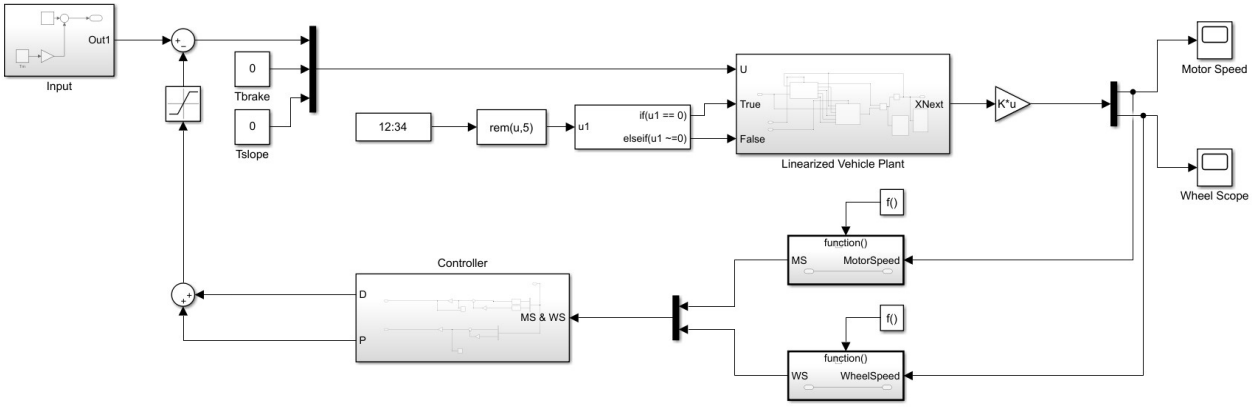


Figure 33: Simulink Model with Motor and Wheel Speed Refresh Rate

Adding the motor and wheel speed refresh rate drastically reduced the overall sample rate and led to low control on the system. The comparison between the model with and without refresh rate at same k_p is shown in figure 34. Given the same control, model without the refresh rate had less oscillation in the motor speed response than the model with the refresh rate. A slightly higher gain value was needed to reach the same result. Adding the refresh rate also decreased the control force we were able to command without the system going unstable. Any $k_p > 1.5$ made the system with refresh rate unstable while a much larger k_p value can be commanded for the system without refresh rate.

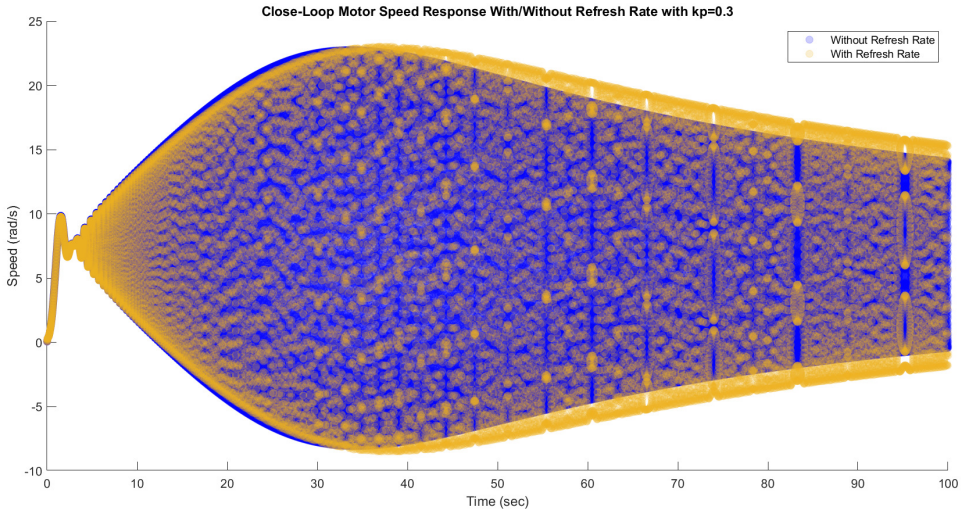


Figure 34: Close-Loop Motor Speed Response with $k_p = 0.3$ for with/without Refresh Rate

Simulink bode plot function did not work for the system with slower sampling rate. We could still generate the bode plot using the FFT method mentioned in section 6.4 and analyze the frequency-domain response. The procedure would be rather complex and we realized we were able to gain basic level of understanding for the system solely from the time-domain plots. Thus, we focused mainly on the time-domain response for the system with slower sampling rate as discussed in this section.

8 Advanced Control Strategies:

Advanced Control techniques to minimize vibrations throughout the electric driveline

There has been a lot of advancement in the field of Controls to achieve robust performance. We aim to apply optimal control, such as Linear-Quadratic Regulator and Model Predictive Control, to improve upon the performance of our classical control design. This section analyzes these strategies.

8.1 Linear-Quadratic Regulator

The linear quadratic regulator is an optimal control strategy. It consists of a quadratic cost, which when minimised gives the optimal controller policy. The optimization is subjected to system dynamics $\dot{x} = Ax + Bu$, state bounds and input bounds. LQR regulates the convergence the states with minimal control cost [7].

For a general plant the cost function is defined as:

$$J = S(x(N)) + \sum_{k=0}^{N-1} L_k(x(k), u(k)) \tag{38}$$

where $S(x(N))$ is the terminal cost and $L_k(x(k), u(k))$ represents the stage cost. Our goal is to find the optimal control input sequence which minimizes the cost J.

A finite-horizon quadratic cost function is formulated as:

$$L_k(x(k), u(k)) = \frac{1}{2}x(k)^T Q(k)x(k) + \frac{1}{2}u(k)^T R(k)u(k) \tag{39}$$

where x(k) and u(k) represent the system state and input at time k. Q represents the weighted matrix on the states while R represent input weights.

Thus when $N \to \infty$ the infinite horizon LQR problem is formulated as:

$$J = \sum_{k=0}^{\infty} \frac{1}{2}x(k)^T Q(k)x(k) + \frac{1}{2}u(k)^T R(k)u(k) \tag{40}$$

This formulation can be solved using dynamic programming and Riccati equation. Irrespective of the final cost, the solution to the Riccati equation converges to a steady state value and therefore the optimal control law can be defined as:

$$u_0(k) = -F_\infty \; x(k) \tag{41}$$

where F_∞ is the solution to Riccati Equation. On Matlab, this can be computed using the inbuilt *dlqr* function.

Substituting the optimal control to the open-loop state space yields a closed-loop state space equation given by:

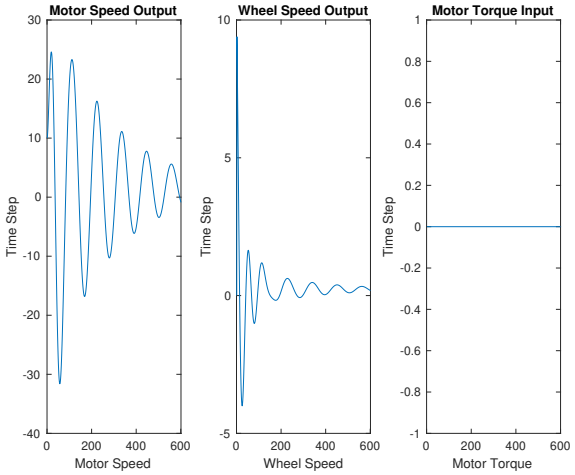
$$x(k+1) = (A - BF_\infty)x(k) \tag{42}$$

Since, we aim to minimize the output variables (Motor Speed and Wheel Speed) y, the closed-loop controller gain is given by CF_∞.

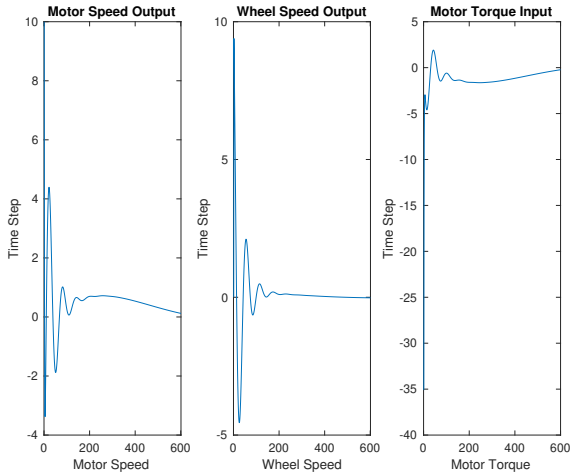
8.1.1 Without Slip

The designed controller was initiated on a simplified system model. We did not consider non-linear components and the slower-sampling of outputs in the similar fashion used for developing our classical control design.

The discrete state space model without slip was considered and the weight matrices Q and R were tuned to converge the output. For the initial condition, $\omega_{Motor} = 10$ *and* $\omega_{Wheel} = 10$. Figure 35a represents the open-loop response for Motor Speed, Wheel Speed and Motor Torque for 600 time steps.



(a) Open Loop LQR



(b) Closed Loop LQR

Figure 35: Open and Closed Loop LQR

For $Q = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$ and $R = 1$, we solved the average Riccati equation to obtain closed loop gain.

$$F_{\infty}^{x} = dlqr(A, B, Q, R) \tag{43}$$

$$F_{\infty}^{y} = C * F_{\infty}^{x} \tag{44}$$

The closed loop response is shown in figure 35b where the open-loop oscillations are reduced and outputs converge to zero with changes in the input less than 5%.

We implemented the LQR gain on the Simulink model, shown in figure 36, to analyse the system behavior in time-domain.

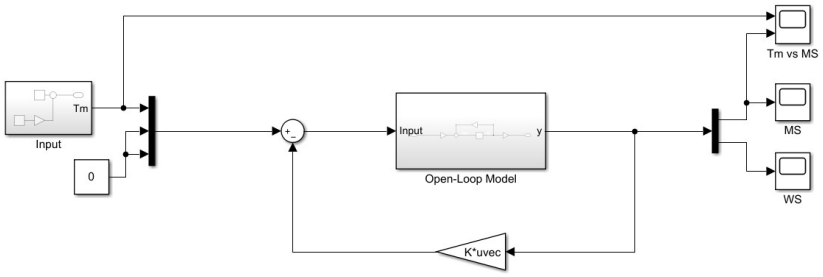


Figure 36: LQR Simulink Model

The figure 37 represents the closed loop response of Motor Speed and figure 38 represents the closed loop Bode Plot. Compared to the open-loop model figure 28, the resonance is highly damped with a damping coefficient of 0.994. Moreover, compared to the classical control, the Motor Speed mean converges to zero.

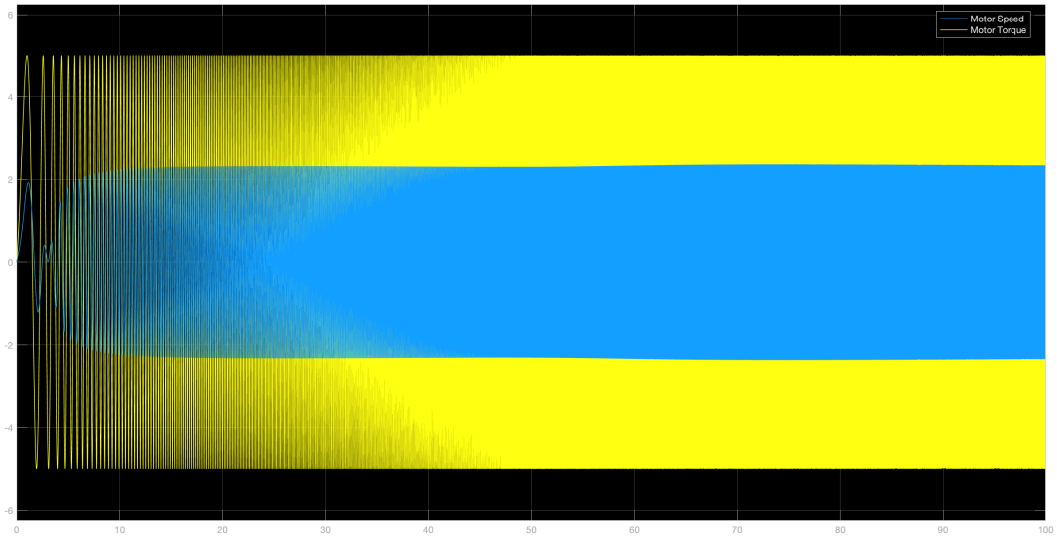


Figure 37: Motor Torque and Motor Speed Response using LQR

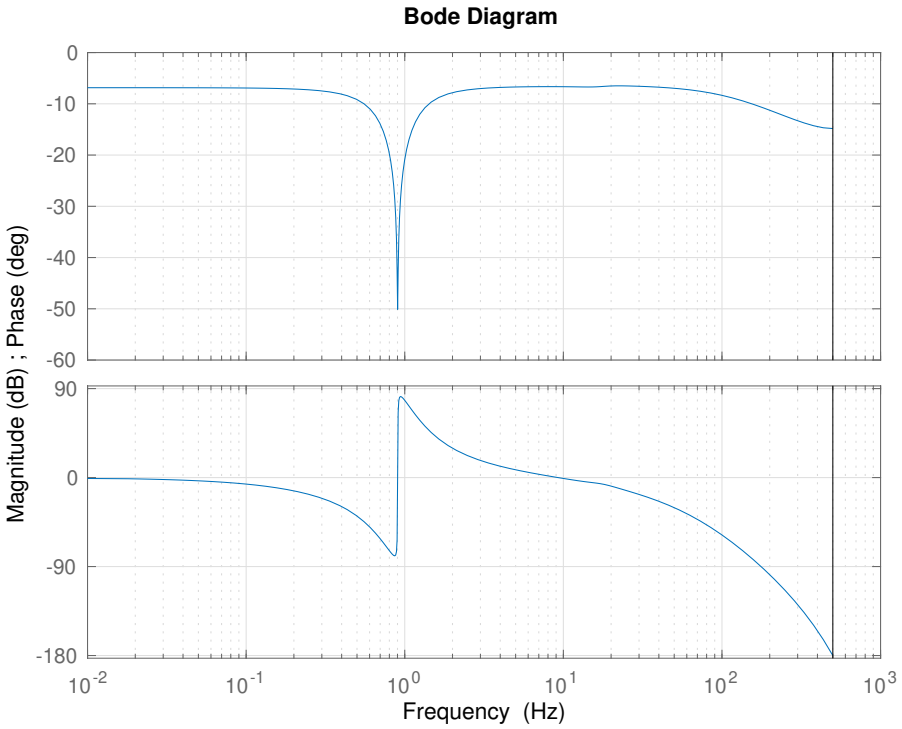


Figure 38: Bode Plot for LQR

8.2 Other Considerations

The LQR solution presented in the above section has been successful in dampening the resonance by applying multiple gain values to the system measurements. However, since the LQR gain was used on the measurements and not on the considered states, persistent feasibility and stability might not hold at all time steps. As a solution, we can analyze and prove system stability and implement an estimator to predict the states on which we can apply the LQR gain.

To enhance the robust properties of the system, we can consider process and measurement noise and extend the LQR solution to Linear-Quadratic Gaussian (LQG), which accounts for these uncertainties by minimizing the expected cost. Additionally, we can also implement Model Predictive Controller, which optimizes the control at every state based on the uncertain dynamics and noises.

9 Future Work:

Extending developed control techniques to remaining areas of electric vehicles

We can further improve our control design in the following areas.

9.1 Modelling

The team currently analyzed the system using low-fidelity system dynamics and considered wheel slip as the sole non-linear component. However, gear lash is another non-linear component which affects the longitudinal resonance in the actual vehicle. We can extend our system model to include gear lash to obtain a more accurate system dynamics representation.

9.2 System Identification

We compared bode plots for correlating our system model with the actual vehicle to tune the mass, damping coefficient, and spring coefficient parameters. Further experiments with the actual vehicle will result in better system identification results.

9.3 Classical Control

Through classical control, we successfully dampened the resonant vibrations. We made an effort to replicate the vehicle behavior in Simulink accurately. We can further improve our model by including additional sensor delays, input constraints, noise, and re-tuning our gain values to enhance system performance with a more realistic analysis of our control design.

9.4 Advanced Control

We have barely scratched the surface of the expansive domain of Advanced Control by implementing a Linear-Quadratic Regulator. Further development in this area would include considering uncertainties in our system and implementing Linear-Quadratic Regulator, Model Predictive Control, some alternate Robust Control, or Non-Linear Control to analyze the system behavior.

10 Conclusion:

Powerful Motors when controlled effectively to reduce vibrations in the driveline make the cost of switching from Internal Combustion Engines extremely feasible

This paper presented various control approaches for damping longitudinal vibrations induced by the high-inertial rotating components in electric vehicles to improve drivability. We first modeled the drivetrain dynamics to parameterize the resonant frequency and its resultant amplitude. To verify the accuracy of our modeled dynamics, we then correlated our model with Chevrolet Bolt's vehicle data collected during our visit to General Motor's Proving Ground in Michigan. We also discussed linearization techniques to model non-linear viscous tire slip. After the verification of our model accuracy, we designed classical and advanced control strategies to meet our design specifications. Our most successful control designs were P-controllers, PD-controllers, and Linear-Quadratic Regulators. Through proper damping,

we were able to reduce the longitudinal vibrations for driver comfort while retaining critical vehicle response. Although we were able to design feedback control strategies successfully, the inclusion of additional vehicle components such as gear lash, timing delays, and sensor refresh rates will add accuracy to our simulation with an increase in computational and timing complexity. More research and development in improving drivability and combining the inherent advantages of electric vehicles facilitates the replacement of internal combustion engine vehicles globally.

References

[1] "Resonance," https://en.wikipedia.org/wiki/Resonance, Wikipedia.

[2] T. D. Gillespie, "Tire-road friction," in *Fundamentals of Vehicle Dynamics*. SAE International, 1992.

[3] R. Bhandari, S. Patil, and R. K. Singh, "Surface prediction and control algorithms for anti-lock brake system," *Transportation Research Part C: Emerging Technologies*, vol. 21, no. 1, pp. 181–195, 2012.

[4] F. Borrelli, C. Jones, and M. Morari, "Model predictive control algorithm, introduction to linear and nonlinear modeling," UC Berkeley, Fall 2019.

[5] "Classical control theory," https://en.wikipedia.org/wiki/Classical_control_theory, Wikipedia.

[6] "How to calculate damping from a frf," https://community.sw.siemens.com/s/article/how-to-calculate-damping-from-a-frf, Siemens.

[7] "Linear-quadratic regulator lecture 19," https://ocw.mit.edu/courses/mechanical-engineering/2-154-maneuvering-and-control-of-surface-and-underwater-vehicles-13-49-fall-2004/lecture-notes/lec19.pdf, MIT OpenCourseWare.