PROJECT REPORT – VEHICLE DYNAMICS

MINI PROJECT

A Report Submitted for the Course ME424

BACHELOR OF TECHNOLOGY

MECHANICAL ENGINEERING

MEMBERS

VAMSHIK M (201ME364)
RUPANKAR DAS (201ME148)
YASHASWINI BAVANDLA(201ME213)
NIHAL R (201ME138)

DEPARTMENT OF MECHANICAL ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE-575 025

NOVEMBER, 2023

Table of contents:

1. Introduction

In the contemporary landscape of automotive engineering, this project, executed within the MATLAB environment, scrutinizes the intricate facets of ride comfort, acknowledging its conventional role as an imperative consideration within the domain of vehicle dynamics. Defined by the ease and fluidity of motion experienced by vehicle occupants, ride comfort assumes a pivotal standing at the intersection of engineering precision and user contentment. This study adopts a comprehensive approach to vehicle design, scrutinizing critical elements such as suspension systems, tire characteristics, structural design, and seating comfort—all encapsulated within the rigorous framework of MATLAB.

While not a groundbreaking innovation, the project recognizes the enduring significance of ride comfort. It stands as a determining factor influencing consumer preferences, augmenting safety measures, and contributing to market competitiveness. A vehicle that seamlessly aligns with the needs of its occupants, mitigating discomfort and fatigue, not only allures customers but substantially fortifies brand perception and customer loyalty. Addressing the commonplace challenges associated with discomfort, the project meticulously quantifies parameters such as vertical acceleration, pitching acceleration, power spectral density, natural frequencies, and the bump toe range.

Through the lens of MATLAB, the project seeks to refine, leveraging the robust capabilities of this computational tool to meticulously analyze and optimize ride comfort. The results are stated for specific cases but the model can be utilized for mass-data based optimization algorithms since it is a parametric model.

2. Literature review

2.1 Vehicle Dynamics and Ride Comfort:

Within the realm of vehicle dynamics, seminal contributions from researchers such as Pacejka and Sharp have meticulously unraveled the intricate dynamics governing tire-road interactions. Serving as the bedrock of understanding vehicular motion, their work lays the foundation for comprehending the fundamental forces steering a vehicle's trajectory. Complementing this theoretical framework, Wong and Khajepour provide invaluable tools, including quarter-car, half-car, and full-car models, empowering engineers to fine-tune suspension parameters for an optimal amalgamation of ride comfort and handling prowess.

2.2 Simulation Techniques:

In the pursuit of elevating ride comfort, simulation techniques emerge as transformative tools. Mendes underscores the versatility of MATLAB, portraying it as a robust platform for modeling vehicle behavior and dissecting the influence of variable parameters. Gillespie and Milliken deepen this exploration by delving into multibody dynamics simulations, offering a meticulous examination of the intricate interplay between suspension systems and structural elements shaping the overall ride experience.

2.3 Quantifiable Metrics:

The assessment of ride comfort necessitates objective metrics, and herein lies the significance of the work by Crolla and Jazar. Their emphasis on quantifiable parameters like Root Mean Square (RMS) Acceleration and Pitch Acceleration provides a rigorous framework for evaluating the smoothness and stability of a vehicle. Augmenting this approach, Guiggiani and Lotfi introduce additional metrics, employing Power Spectral Density (PSD) analysis to discern the frequency distribution of vibrations on the road.

2.4 Suspension Parameter Optimization:

Gillich and Li assume the role of adept orchestrators, manipulating and optimizing suspension parameters to craft the ideal recipe for a smooth ride. Delving into the nuanced interplay of stiffness and damping coefficients, they contribute to the ongoing quest for achieving an optimal balance between comfort and stability. In alignment with this pursuit, our project navigates the same terrain, striving to identify the most effective configuration for a comfortable and stable ride.

2.5 Challenges and Future Directions:

Envisioning a future where road trips unfold seamlessly, Rakheja and Su advocate for cutting-edge technologies such as real-time data and intelligent algorithms to elevate ride comfort further. This forward-looking perspective entails the integration of sophisticated systems capable of predicting and adapting to varying road conditions, ushering in an era where rides transcend the boundaries of smoothness.

In essence, this literature review encapsulates the fundamental principles of vehicle dynamics, the application of simulation techniques for optimizing ride comfort, the importance of quantifiable metrics in evaluating comfort levels, and the ongoing quest to refine suspension parameters for an optimal ride experience.

---

3. Methodology

3.1 Model Inspiration:

Our model draws inspiration from the work of André de Souza Mendes, specifically the "Half car model Version 1.0.1." While incorporating elements from Mendes's implementation, our focus diverges toward an in-depth analysis of ride comfort, with an emphasis on quantifiable metrics rather than simulation and animation aspects. This adaptation allows us to delve into the intricate dynamics of a half-car model, seeking to optimize ride comfort through systematic parameter variations.

3.2 Half-Car Model:

The Half-Car Model divides the vehicle into two main components: the front half and the rear half. Each half is treated as a separate mass-spring-damper system, connected by the vehicle's chassis. This simplification allows for a more manageable representation of the complex interactions between the vehicle and the road surface.

3.2.1 Advantages of the Half-Car Model:

1. **Simplicity and Computational Efficiency**: The Half-Car Model significantly reduces the complexity of the suspension system while retaining essential features. This simplicity enhances computational efficiency, making it a preferred choice for initial design and analysis phases.
2. **Focused Ride Comfort Analysis**: The model excels in analyzing ride comfort, particularly in terms of vertical motion and pitch behavior. It provides insights into how different parameters, such as suspension stiffness and damping, impact the vehicle's response to road irregularities.
3. **Effective for Parametric Studies**: The Half-Car Model is well-suited for parametric

studies where variations in specific parameters, such as suspension stiffness or damping, need to be systematically analyzed for their effects on ride comfort. This capability is valuable for optimization processes.

4. **Insights into Pitch Dynamics**: As pitch motion (rotation about the vehicle's longitudinal axis) significantly influences ride comfort, the Half-Car Model offers a clear representation of pitch dynamics. This insight is crucial for designing suspensions that minimize pitching acceleration and, consequently, enhance passenger comfort.
5. **Integration with Road Profiles**: The model can be integrated with different road profiles to simulate real-world conditions. This integration allows for the evaluation of how the vehicle responds to variations in road surfaces and the identification of optimal suspension settings.

3.3 Half Car Model Overview

This model captures the essential elements influencing ride comfort—mass (M), damping (C), and stiffness (K) matrices. These matrices are crucial for formulating the dynamic equation that governs the motion of the vehicle.

3.3.1 Dynamic Governing Equation:

$$Mx''(t) + Cx'(t) + Kx(t) = Fu(t)$$

Mass Matrix (M): The mass matrix represents the distribution of mass within the vehicle and is a key determinant of its inertia. For our half-car model, the mass matrix M is a 4x4 matrix, where the diagonal elements correspond to the total mass (m) of the car and the masses of the front (m1) and rear (m2) unsprung masses, along with the lateral moment of inertia (Iy).

$$M = \begin{bmatrix} m & 0 & 0 & 0 \\ 0 & I_y & 0 & 0 \\ 0 & 0 & m_1 & 0 \\ 0 & 0 & 0 & m_2 \end{bmatrix}$$

Damping Matrix (C): The damping matrix accounts for the dissipation of energy due to the internal friction within the suspension components. It includes the damping coefficients for the front (c1) and rear (c2) suspension systems.

$$C = \begin{bmatrix} c_1 + c_2 & a_2c_2 - a_1c_1 & -c_1 & -c_2 \\ a_2c_2 - a_1c_1 & a_1^2c_1 + a_2^2c_2 & a_1c_1 & -a_2c_2 \\ -c_1 & a_1c_1 & c_1 & 0 \\ -c_2 & -a_2c_2 & 0 & c_2 \end{bmatrix}$$

Stiffness Matrix (K): The stiffness matrix characterizes the resistance of the suspension components to deformation. It includes the stiffness coefficients for the front (k1) and rear (k2) suspension systems, as well as the stiffness of the front (kt1) and rear (kt2) tires.

$$K = \begin{bmatrix} k_1 + k_2 & a_2k_2 - a_1k_1 & -k_1 & -k_2 \\ a_2k_2 - a_1k_1 & a_1^2k_1 + a_2^2k_2 & a_1k_1 & -a_2k_2 \\ -k_1 & a_1k_1 & k_1 + k_{t1} & 0 \\ -k_2 & -a_2k_2 & 0 & k_2 + k_{t2} \end{bmatrix}$$

***Forcing Vector** (F):* The forcing vector represents the external forces acting on the system, primarily originating from the road profile. In our model, the forcing vector is a 4x2 matrix, accounting for the forces exerted on the front and rear tires. Here, $u_1(t)$ and $u_2(t)$ denote the displacements of the front and rear wheels, respectively. The entries k_{t1} and k_{t2} represent the tire stiffness for the front and rear wheels.

$$F = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ k_{t1}u_1(t) & 0 \\ 0 & k_{t2}u_2(t) \end{bmatrix}$$

***State Vector** (x):* The state vector x is a column vector with eight components representing dynamic variables: vertical displacement, pitch angle, vertical displacement of unsprung masses.

$$X = \begin{bmatrix} z \\ theta \\ zu1 \\ zu2 \end{bmatrix}$$

These matrices form the basis for the dynamic governing equation: Mx”(t) + Cx’(t) + Kx(t) = F(t), where x(t) represents the displacement vector, and u(t) represents the input force from the road profile. The Half-Car Model, with its detailed representation of mass, damping, and stiffness, lays the groundwork for a nuanced analysis of ride comfort parameters.

3.3 Ride Comfort Analysis:

Ride comfort analysis involves assessing the response of the vehicle to external disturbances, focusing on minimizing discomfort for occupants. The Half-Car Model, with its detailed representation of vehicle dynamics, is employed to simulate and analyze the ride comfort parameters.

3.3.1 Simulation Setup:

The governing equations of motion for the Half-Car Model are solved numerically to obtain displacement, velocity, and acceleration profiles of key components, including the body, front and rear unsprung masses, and pitch motion. The simulation incorporates a realistic road profile, characterized by bumps and irregularities, to emulate actual driving conditions.

Utilizing the developed equations, a simulation is conducted in MATLAB to analyze the response of the Half-Car Model to a given road profile. The simulation yields displacement, velocity, and acceleration profiles for various components, including the vehicle body, front and rear unsprung masses, and pitch angle.

3.3.2 Quantification of Ride Comfort Parameters:

Several ride comfort parameters are quantified to assess the performance of the vehicle suspension system. These include:

1. **Root Mean Square (RMS) Acceleration**: A measure of the average acceleration experienced by the vehicle occupants. Lower RMS acceleration values indicate a smoother ride.

$$\text{RMS Acceleration} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} a_i^2}$$

2. **RMS Pitch Acceleration**: Gauges the average angular acceleration around the pitch axis, providing insights into pitch motion and stability.

$$\text{RMS Pitch Acceleration} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \theta_i^2}$$

3. **Transmissibility (TR)**: This metric gauges the ratio of body acceleration to unsprung mass acceleration, indicating the level of isolation provided by the suspension system.

$$\text{Transmissibility} = \frac{\text{Maximum Acceleration of Body}}{\text{Maximum Acceleration of Unsprung Mass}}$$

4. **Power Spectral Density (PSD)**: Analyzes the frequency distribution of power in the vehicle's response. The PSD helps identify dominant frequencies contributing to vibrations.

$$\text{PSD}(f) = \frac{1}{T}\left|\int_{-\infty}^{\infty} x(t)e^{-j2\pi f t}dt\right|^2$$

5. **Natural Frequencies**: Represents the inherent vibrational frequencies of the vehicle structure. Optimizing these frequencies is crucial for minimizing resonance effects and enhancing ride comfort.

The quantification of these parameters allows for a comprehensive evaluation of the ride comfort performance, guiding engineers in refining the suspension design for optimal passenger comfort under varying road conditions. The simulation results provide valuable insights into the dynamic behavior of the vehicle and aid in the identification of key factors influencing ride quality.

3.4 Road Profile Generation:

The road profile is a crucial element in simulating realistic driving conditions and evaluating a vehicle's suspension system. The generation process involves synthesizing a road profile that replicates typical road irregularities, forming the basis for a comprehensive ride comfort analysis.

3.4.1 Profile Generation Process:

The road profile, denoted by $Z_r(X_r)$, is generated by combining white noise with sinusoidal bumps:

$$Z_r(X_r) = WhiteNoise + \sum_{i=1}^{n} A_i \cdot \sin(2\pi f_i X_r)$$

In this equation, X_r is the longitudinal position along the road profile. A_i is the amplitude of the ith sinusoidal bump. f_i is the frequency of the ith sinusoidal bump.

Parameters such as profile length, sampling rate, amplitude factor, number of bumps, bump amplitude, and bump frequency influence the profile's characteristics. The white noise component adds unpredictability, while sinusoidal bumps simulate specific road features like speed bumps or undulations.

3.4.2 Significance of Realistic Road Profiles:

Realistic road profiles are essential for accurate vehicle dynamic simulations. They enable a thorough evaluation of the vehicle's response to diverse driving conditions, allowing engineers to assess ride comfort, suspension effectiveness, and overall performance. The road profile's dynamic nature introduces variability, making the analysis more representative of actual road scenarios.

3.4.3 Application in Ride Comfort Analysis:

The generated road profile serves as the input to the Half-Car Model, influencing the dynamic response of the vehicle. By incorporating variations in road surfaces, the analysis becomes more nuanced, providing insights into how the vehicle interacts with different terrains and road conditions. This realistic input is crucial for a comprehensive understanding of ride comfort parameters.

---

4. Results and Analysis

4.1 Small Semicircle Bump on Flat Road

4.1.1 Vertical Displacement Analysis

The simulation of a small semicircle bump on a flat road demonstrated that the vehicle's body exhibited significantly less motion compared to the road. The graph depicting vertical displacement of unsprung and sprung masses, along with pitch, against longitudinal displacement illustrated the effective isolation of the vehicle body from road irregularities.

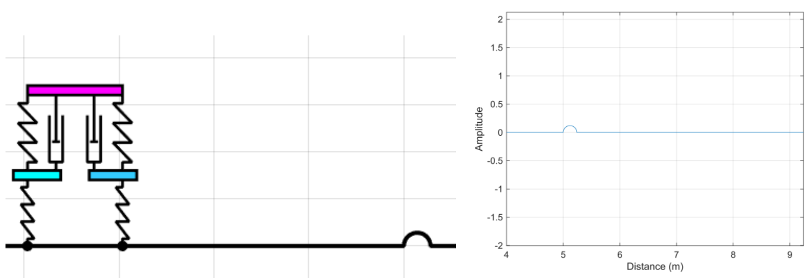


Fig: Visualization of road bump

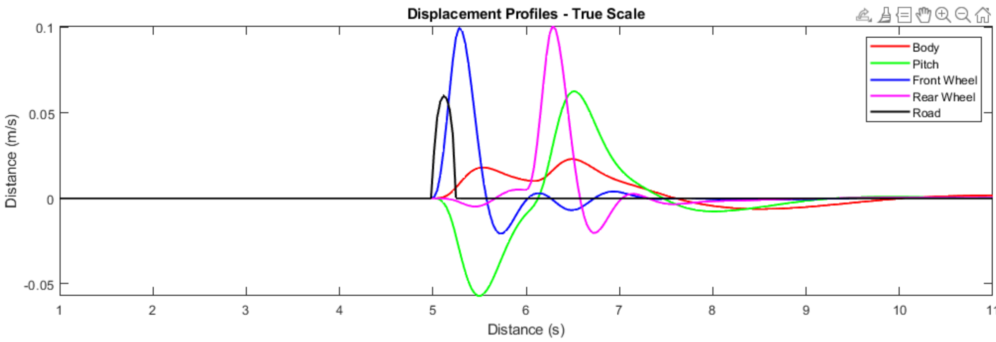


Fig: Vertical displacement v/s longitudinal position

4.1.2 Acceleration Analysis

The analysis of acceleration profiles for both sprung and unsprung masses revealed a transmissibility of 0.08. This low transmissibility value indicated that the disturbances from the road had minimal impact on the body, ensuring a smoother ride experience for passengers.

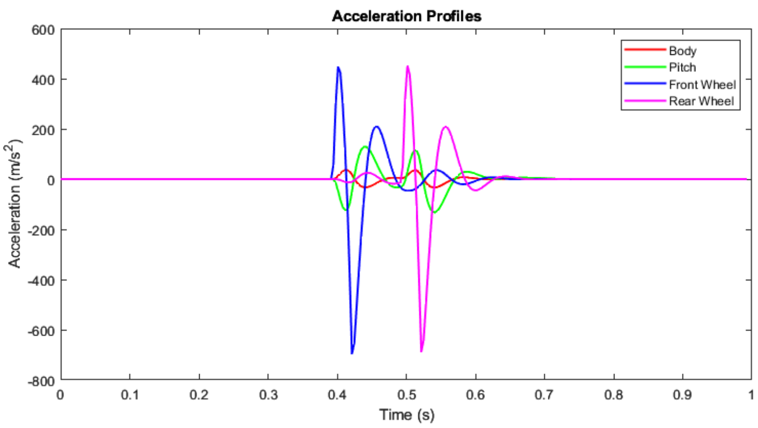


Fig: Vertical acceleration v/s time

RMS Acceleration:

1. Body: 9.0862 m/s^2
2. Pitch: 34.0564 rad/s^2

4.1.3 Power Spectral Density (PSD)

The PSD graphs provided insights into the first 5 natural frequencies of the system, showing peaks at [3.0101, 11.0369, 19.0638, 29.0973, 39.1309] in Hertz. The total spectral power of 62.49 indicated the distribution of power across different frequency components.

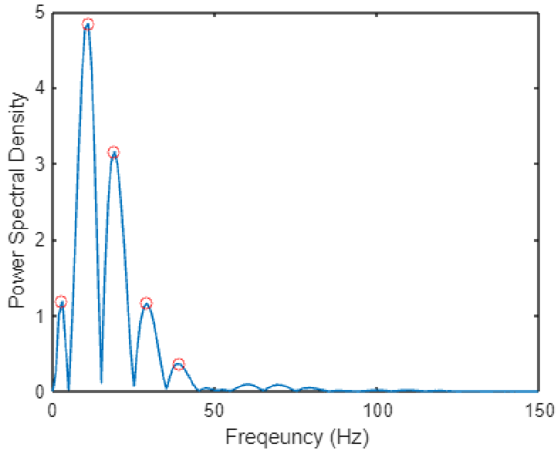


Fig: Power Spectral Density after FFT of Acceleration

4.2 White Noise Road Simulation

4.2.1 Vertical Displacement Analysis

Employing a white noise-based road input with realistic parameters, the simulation showcased reduced body motion compared to the road. The graph depicting vertical displacement of unsprung and sprung masses, along with pitch, against longitudinal displacement emphasized the effective dampening of road irregularities.

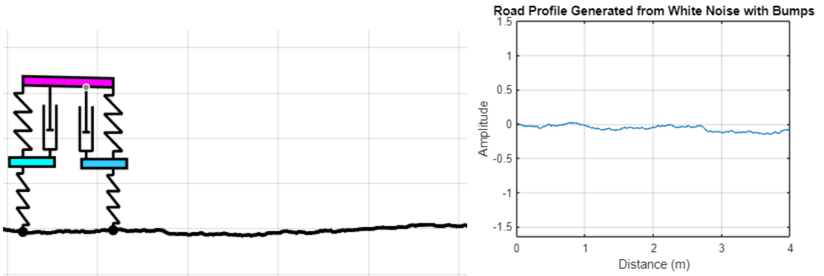


Fig: Visualization of road profile

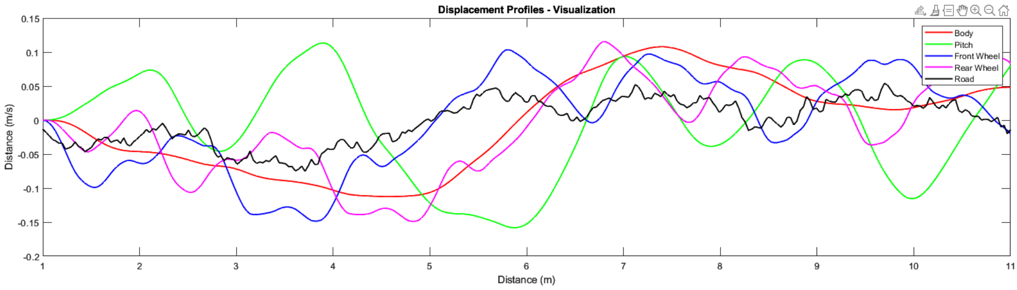


Fig: Vertical displacement v/s longitudinal position

4.2.2 Acceleration Analysis

Acceleration profiles for both sprung and unsprung masses yielded a transmissibility of 0.1147. This slightly higher transmissibility value indicated a controlled transmission of road disturbances, contributing to an acceptable compromise between comfort and road feedback.

RMS Acceleration:

1. Body: 6.8532 m/s²
2. Pitch: 29.7704 rad/s²

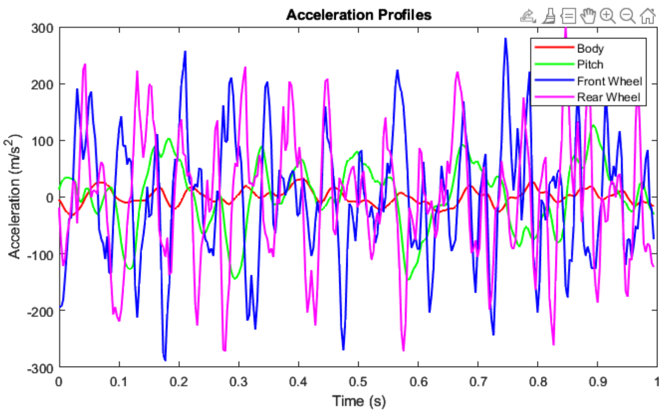


Fig: Vertical acceleration profile v/s time

4.2.3 Power Spectral Density (PSD)

PSD graphs exhibited the lowest 5 natural frequencies at [2.0067, 6.0201, 8.0268, 11.0369, 13.0436] in Hertz, with a total spectral power of 52.1049. The distribution of power across various frequencies provided insights into the system's response to road inputs.

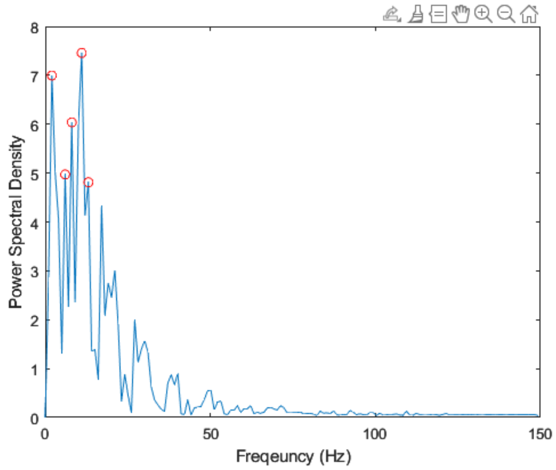


Fig: Power Spectral Density after FFT of Acceleration

4.3 Stiffness Variation Effect Analysis

4.3.1 Stiffness Variation

Systematic variation of stiffness from 200 to 30000 N/m highlighted several key observations.

1. RMS acceleration:

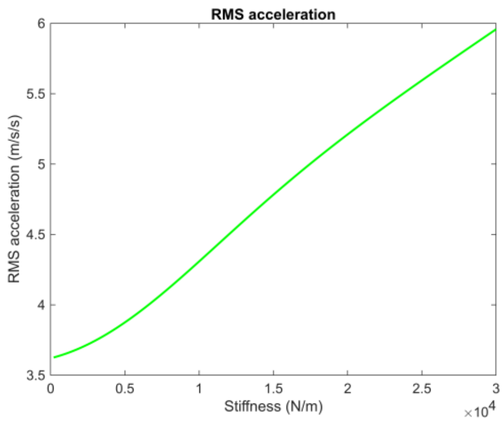


Fig: Stiffness Variation Analysis - RMS acceleration

Exhibited a monotonically increasing trend with stiffness, indicating that higher stiffness resulted in more pronounced accelerations.

2. RMS Pitch Acceleration:

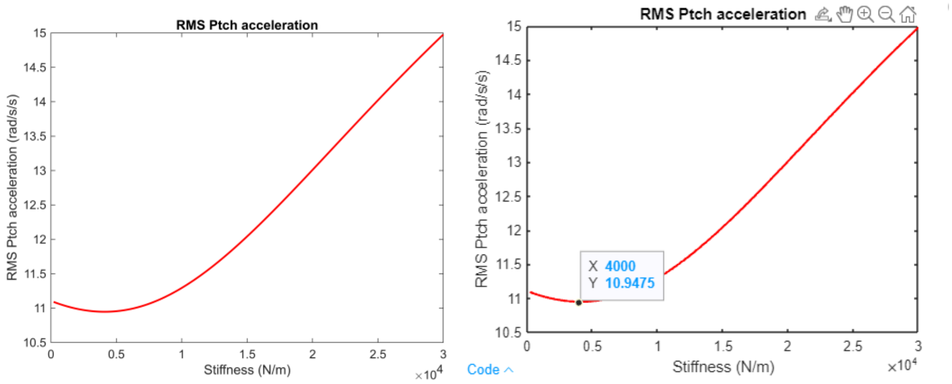


Fig: Stiffness Variation Analysis - RMS pitch acceleration

Reached a minimum at around 4000 N/m, suggesting an optimal stiffness value for minimizing pitch motion.

3. Transmissibility:

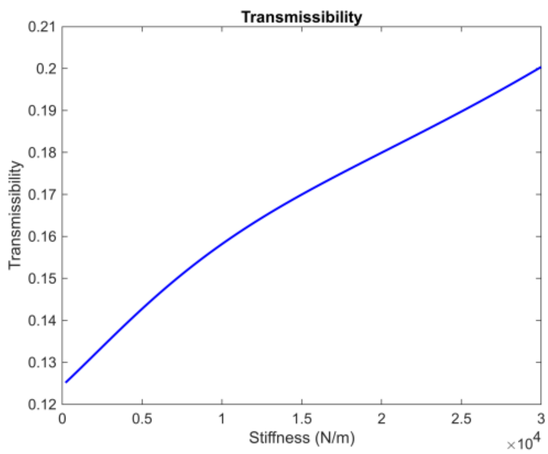


Fig: Stiffness Variation Analysis - Transmissibility

Increased with stiffness, but the slope decreased, indicating a balance between suspension stiffness and ride comfort.

4. Total Spectral Power:

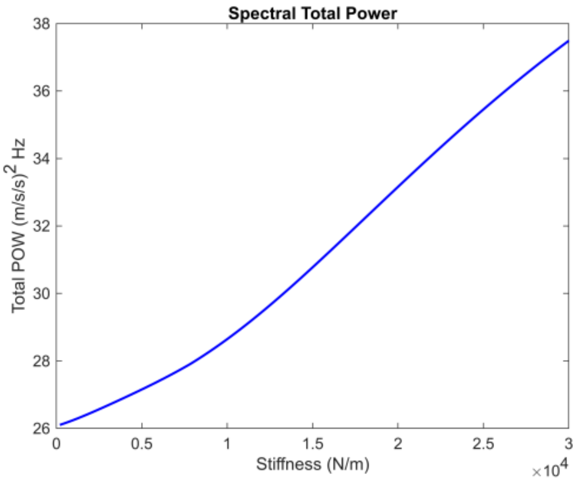


Fig: Stiffness Variation Analysis - Total Spectral Power

Monotonically increased with stiffness, indicating a higher distribution of power across various frequency components.

4.4 Damping Coefficient Variation Effect Analysis

4.4.1 Damping Coefficient Variation

Systematic variation of damping coefficients from 0 to 10000, with a constant stiffness of 15000, resulted in noteworthy observations.

1. The RMS acceleration:

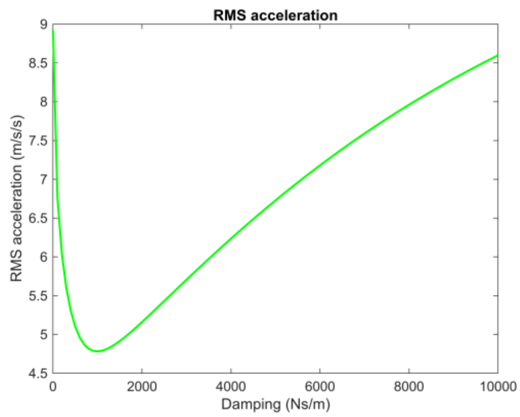


Fig: Stiffness Variation Analysis - RMS Acceleration

Demonstrated a decreasing trend to a minimum at around 900 Ns/m, suggesting an optimal damping value for minimizing accelerations.

2. RMS Pitch Acceleration:

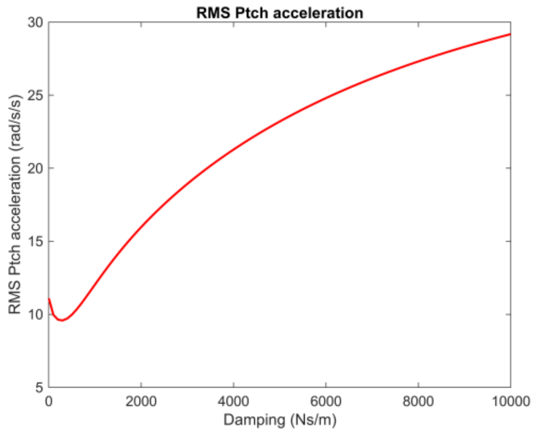


Fig: Stiffness Variation Analysis - RMS Pitch Acceleration

Exhibited a minimum at approximately 300, indicating an optimal damping coefficient for minimizing pitch motion.

3. Transmissibility:

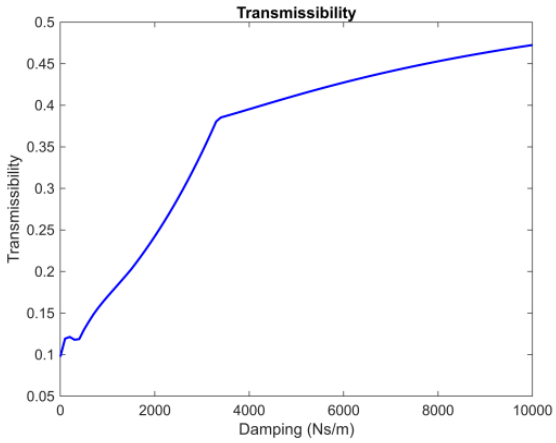


Fig: Stiffness Variation Analysis - Transmissibility

Increased with damping but showed an irregularity at low damping, likely due to superposition effects.

4. Total Spectral Power:

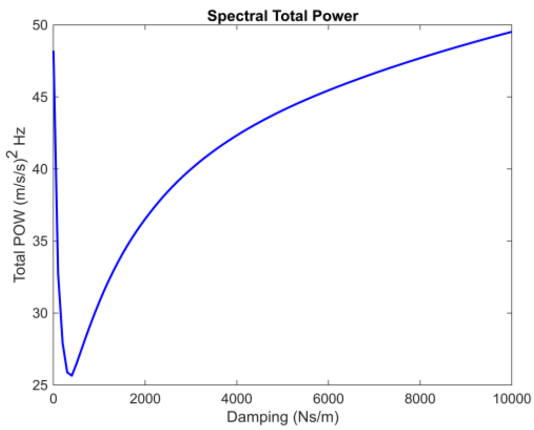


Fig: Stiffness Variation Analysis - Spectral Total Power

Decreased to a minimum at 400 Ns/m, suggesting an optimal damping value for enhanced overall ride comfort.

5. Conclusions

In conclusion, this research delves into the realm of ride comfort analysis with a focus on a parametrically modeled half-car system. Drawing inspiration from the Half Car Model Version 1.0.1 by André de Souza Mendes, this implementation departs from simulation aspects, emphasizing real-world dynamics. The half-car model, encapsulated by the dynamic equation $Mx'' + Cx' + Kx = Fu$, provides a comprehensive representation of the vehicle's response to external forces.

5.1 Half-Car Model Insights

Section 3.2 elucidates the matrices involved in the dynamic equation. The mass (M), damping (C), and stiffness (K) matrices, along with the road input (F), collectively determine the system's behavior.

5.2 Ride Comfort Analysis

The ride comfort analysis in Section 3.3 involves road profile generation, utilizing white noise for realistic simulations. The obtained results, displayed in Section 4, underscore the model's efficacy. A semicircle bump simulation elucidates the system's response, while a white noise road input introduces complexity.

5.3 Parametric Variation Analysis

Sections 4.1 and 4.2 explore the effects of stiffness and damping coefficient variations on ride comfort. Notably, damping optimization proves critical, showcasing diverse minima in RMS acceleration, RMS pitch acceleration, transmissibility, and PSD power. This variability highlights the potential for a combined optimization algorithm to enhance overall suspension system performance.

5.4 Future Directions

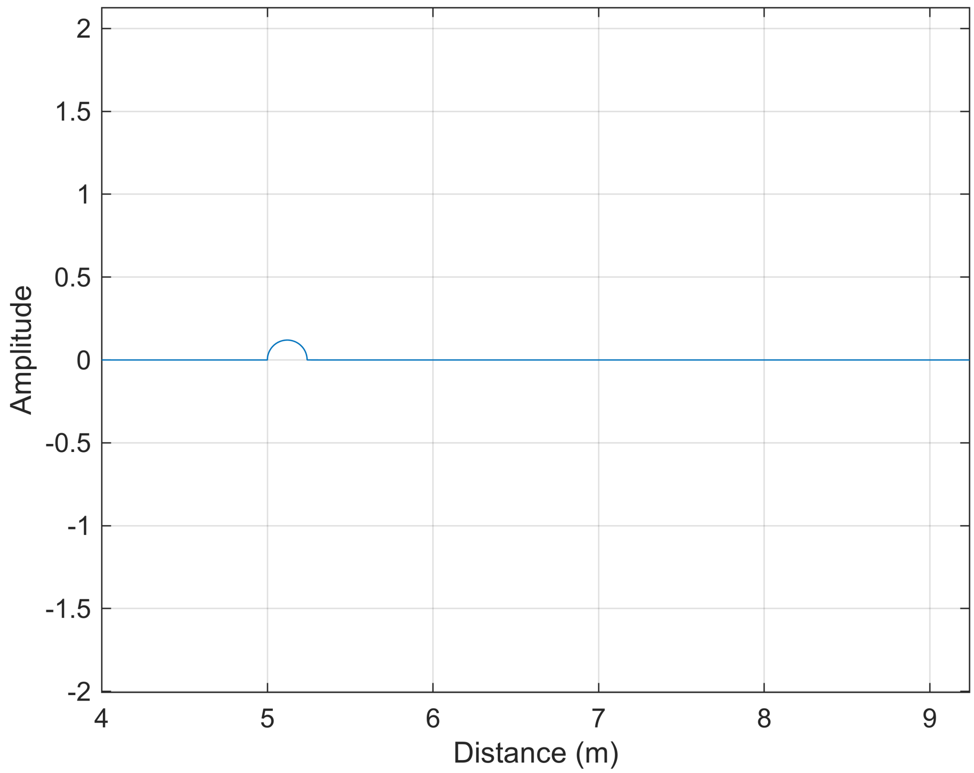
This research opens avenues for further exploration. Future work could delve into comprehensive optimization algorithms considering both stiffness and damping. Moreover, a closer integration with real-world data acquisition systems and subjective evaluations can provide a holistic understanding of ride comfort.

In summary, this study underscores the importance of a parametrically modeled half-car system in analyzing ride comfort, offering valuable insights for future advancements in vehicle suspension design and optimization.

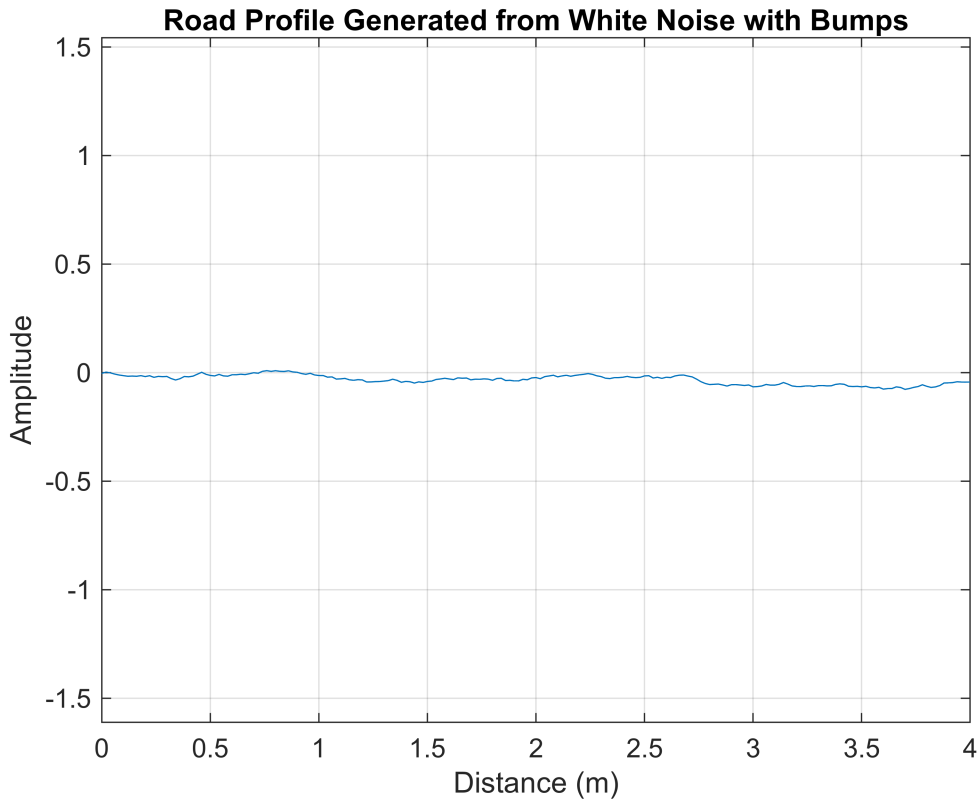
APPENDIX

Road Profile - Generation

```
[X_r, Z_r] = bump_road_input(5,0.12,15);
```




```
profile_length = 200;       % Length of the road profile (meters)
sampling_rate = 50;         % Numberof points per meter
amplitude_factor = 0.005;     % Roughness
num_bumps = 5;              % Number of bumps
bump_amplitude = 0.001;       % Amplitude of bumps
bump_frequency = 0.5;       % Frequency of bumps

[X_r, Z_r] = generateRoadProfileWithBumps(profile_length, sampling_rate, amplitude_factor, num_
```




```
road.X_r = X_r;
road.Z_r = Z_r;
```

Car Model

```
%Vehicle
scooter.mass = 150;
scooter.front_unsprung_mass = 15;
scooter.rear_unsprung_mass = 15;
scooter.Lateral_MOI = 20;
scooter.CG_2_Front = 0.5;
scooter.CG_2_Rear = 0.5;

stiffness.front_strut = 15000;
stiffness.rear_strut = 15000;
stiffness.tire_front = 100000;
stiffness.tire_rear = 100000;

damping.strut_front = 1000;
damping.strut_rear = 1000;

%Velocity
initial_vel = 10; %velocity in m/s
acc  = 0;
```

SOLVING

```
[displacement, velocity, acceleration] = Ride_Comfort_Analysis(scooter, stiffness, damping, roa
```

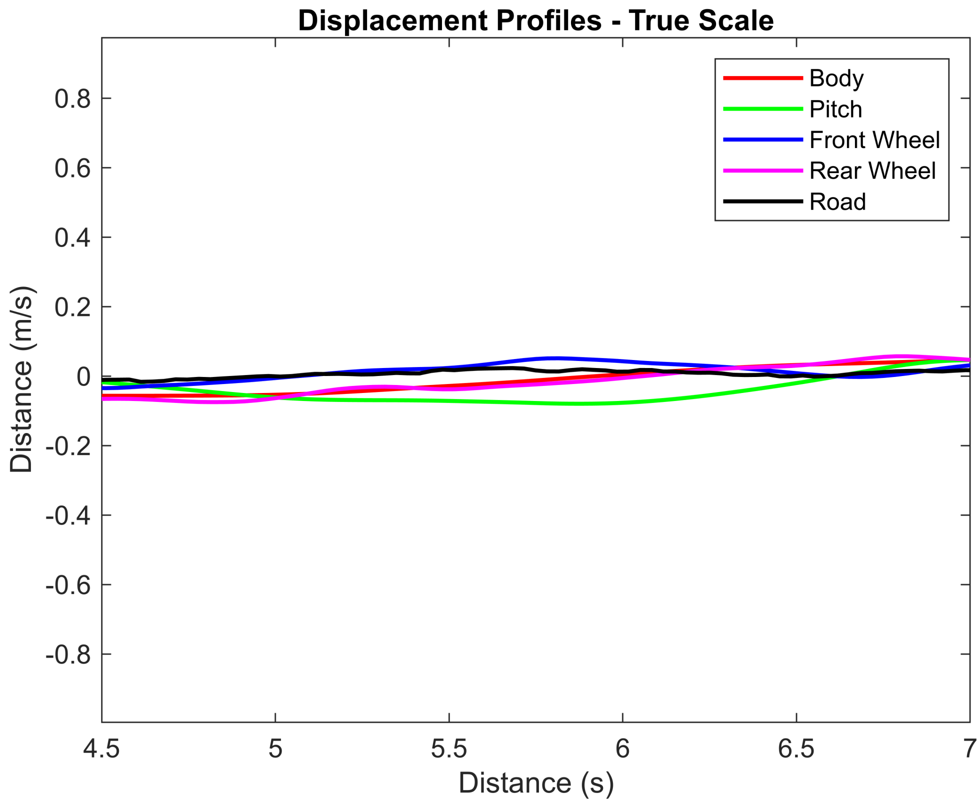
```
displacement = struct with fields:
                  z_body: [300×1 double]
        z_unsprung_front: [300×1 double]
         z_unsprung_rear: [300×1 double]
                   theta: [300×1 double]
                    time: [300×1 double]
              tire_front: [-0.0066 -0.0091 -0.0114 -0.0142 -0.0154 -0.0170 -0.0169 -0.0214 -0.0206 -0.0193 -0.016
               tire_rear: [-0.0016 5.6040e-05 -0.0041 -0.0069 -0.0080 -0.0078 -0.0089 -0.0096 -0.0088 -0.0090 -0.
    longitudinal_pos_front: [1 1.0334 1.0669 1.1003 1.1338 1.1672 1.2007 1.2341 1.2676 1.3010 1.3344 1.3679 1.4013 1
     longitudinal_pos_rear: [0 0.0334 0.0669 0.1003 0.1338 0.1672 0.2007 0.2341 0.2676 0.3010 0.3344 0.3679 0.4013 0
velocity = struct with fields:
              v_body: [299×1 double]
    v_unsprung_front: [299×1 double]
     v_unsprung_rear: [299×1 double]
            v_thetha: [299×1 double]
                time: [299×1 double]
acceleration = struct with fields:
              a_body: [298×1 double]
    a_unsprung_front: [298×1 double]
     a_unsprung_rear: [298×1 double]
             a_theta: [298×1 double]
                time: [298×1 double]
```

```
%ANd DONEEEEEEEE...
```

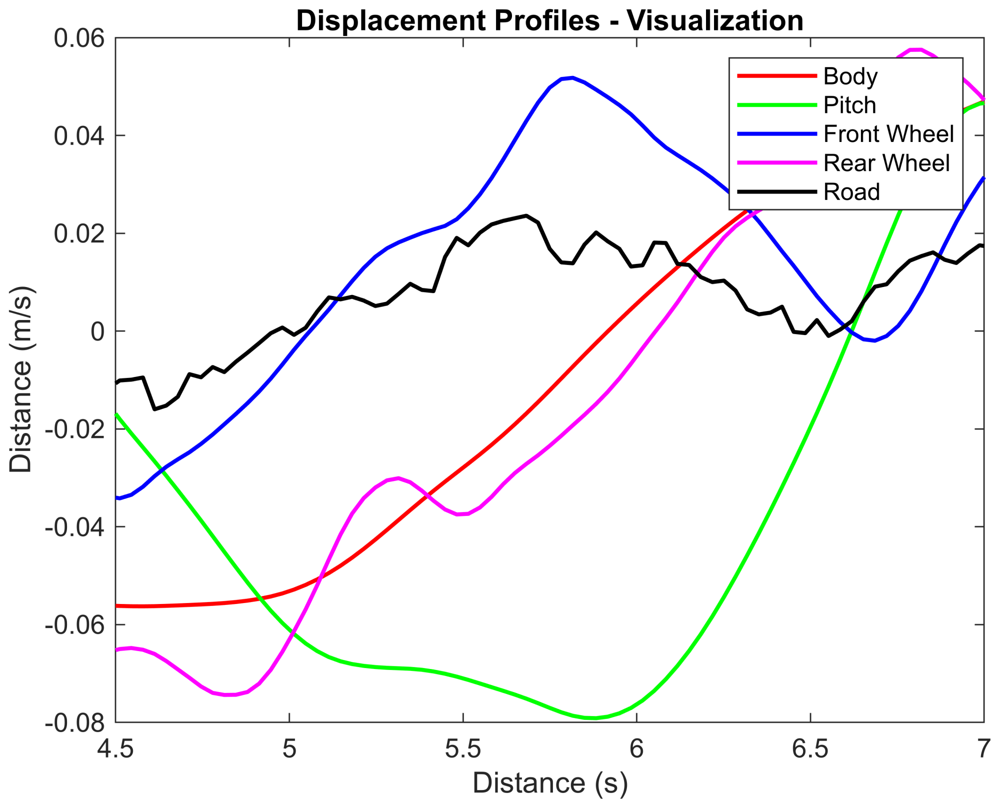
Plotting & Analyzing

Position Based Analysis

```
figure
u = displacement.longitudinal_pos_front;
plot(u , displacement.z_body, 'r', 'LineWidth', 1.5);
hold on;
plot(u, displacement.theta, 'g', 'LineWidth', 1.5);
plot(u, displacement.z_unsprung_front, 'b', 'LineWidth', 1.5);
plot(u, displacement.z_unsprung_rear, 'm', 'LineWidth', 1.5);
plot(u, displacement.tire_front, 'k', 'LineWidth', 1.5);
xlabel('Distance (s)');
ylabel('Distance (m/s)');
legend('Body', 'Pitch', 'Front Wheel', 'Rear Wheel','Road');
title('Displacement Profiles - True Scale');
axis equal
xlim([4.5 7])
hold off
```

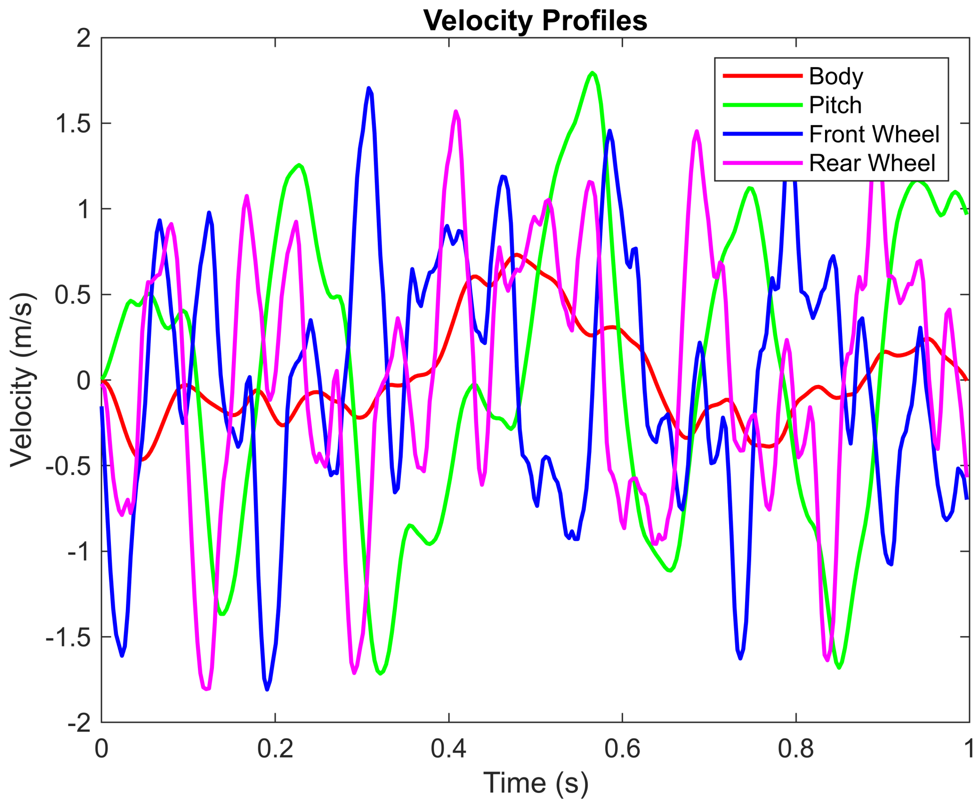



```
figure
u = displacement.longitudinal_pos_front;
plot(u , displacement.z_body, 'r', 'LineWidth', 1.5);
hold on;
plot(u, displacement.theta, 'g', 'LineWidth', 1.5);
plot(u, displacement.z_unsprung_front, 'b', 'LineWidth', 1.5);
plot(u, displacement.z_unsprung_rear, 'm', 'LineWidth', 1.5);
plot(u, displacement.tire_front, 'k', 'LineWidth', 1.5);
xlabel('Distance (s)');
ylabel('Distance (m/s)');
legend('Body', 'Pitch', 'Front Wheel', 'Rear Wheel','Road');
title('Displacement Profiles - Visualization');
xlim([4.5 7])
hold off
```

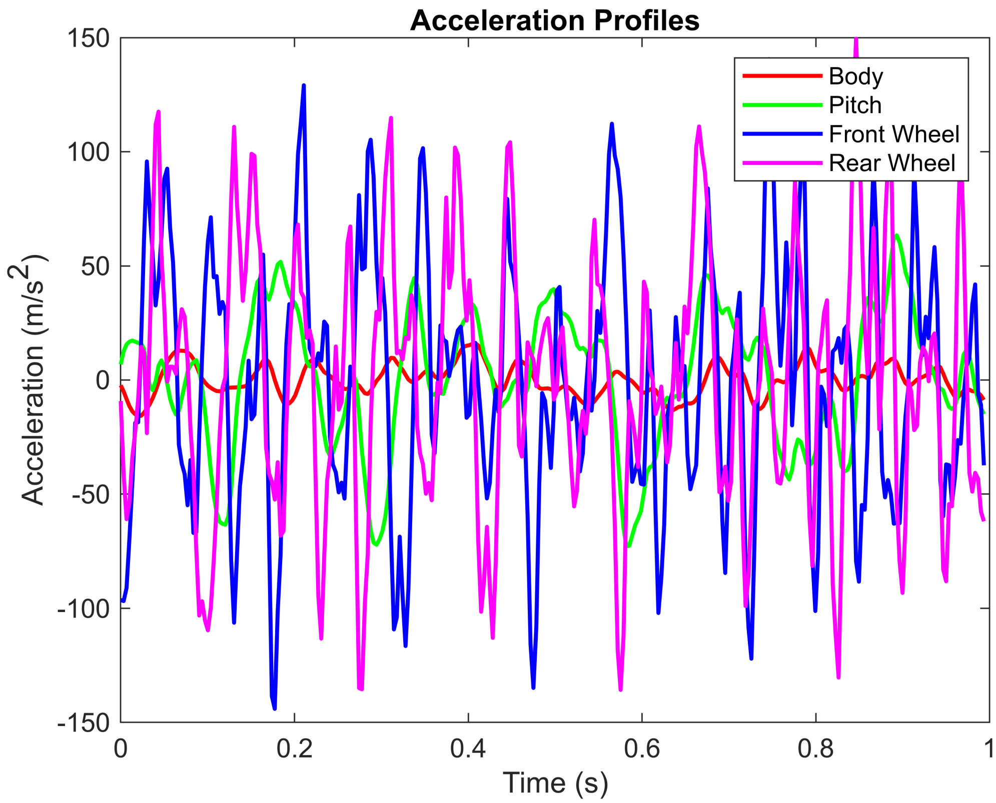


Time Based Analysis

```
figure;
plot(velocity.time, velocity.v_body, 'r', 'LineWidth', 1.5);
hold on;
plot(velocity.time, velocity.v_thetha, 'g', 'LineWidth', 1.5);
plot(velocity.time, velocity.v_unsprung_front, 'b', 'LineWidth', 1.5);
plot(velocity.time, velocity.v_unsprung_rear, 'm', 'LineWidth', 1.5);
xlabel('Time (s)');
ylabel('Velocity (m/s)');
legend('Body', 'Pitch', 'Front Wheel', 'Rear Wheel');
title('Velocity Profiles');
hold off
```




```
figure;
plot(acceleration.time, acceleration.a_body, 'r', 'LineWidth', 1.5);
hold on;
plot(acceleration.time, acceleration.a_theta, 'g', 'LineWidth', 1.5);
plot(acceleration.time, acceleration.a_unsprung_front, 'b', 'LineWidth', 1.5);
plot(acceleration.time, acceleration.a_unsprung_rear, 'm', 'LineWidth', 1.5);
xlabel('Time (s)');
ylabel('Acceleration (m/s^2)');
legend('Body', 'Pitch', 'Front Wheel', 'Rear Wheel');
title('Acceleration Profiles');
hold off
```



RIDE COMFORT QUANTIZATION

Root Mean Square Acceleration

```
%Body
RMS_Acc_Body = rms(acceleration.a_body)
```

```
RMS_Acc_Body = 6.8532
```

```
%Pitch
RMS_Acc_Pitch = rms(acceleration.a_theta)
```

```
RMS_Acc_Pitch = 29.7704
```

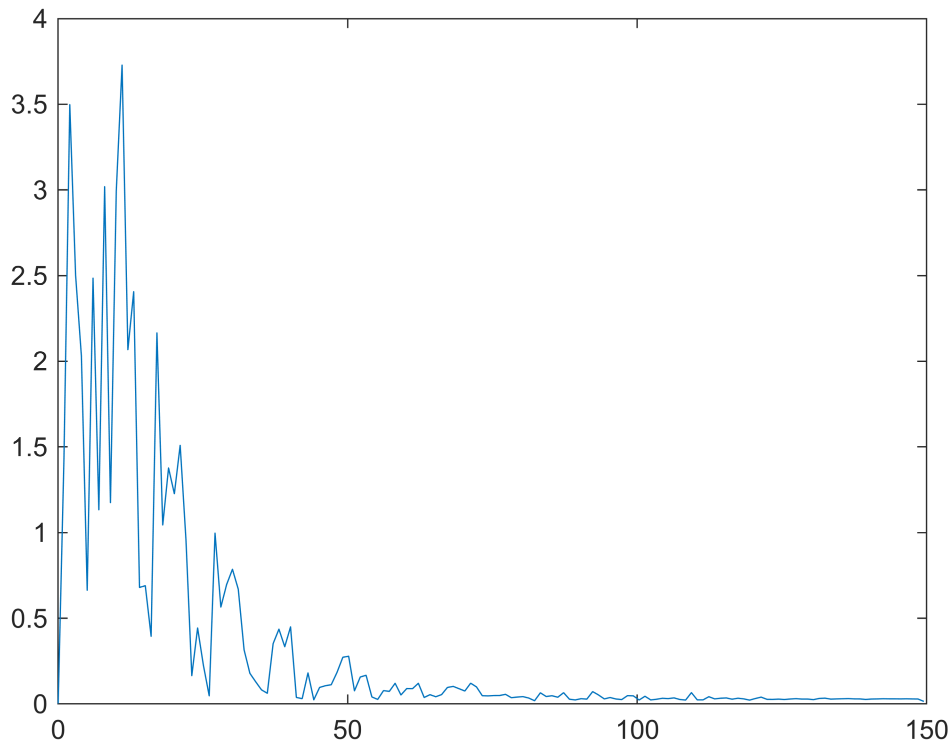
Transmissibility (acceleration based)

```
max(acceleration.a_body)/max(acceleration.a_unsprung_front)
```

```
ans = 0.1147
```

Power Spectral Density - Frequency based

```
[Weighted_PSD, frequency_arr, PSD] = Frequency_analysis(acceleration.a_body,acceleration.time);
plot(frequency_arr, PSD)
```



Total Power (PSD integral)

```
Total_Power = trapz(frequency_arr, PSD)
```

```
Total_Power = 52.1049
```

Finding Natural Frequencies

```
% Find and display the first 5 peaks in the amplitude spectrum (simple peak detection)
peaks = [];
locs = [];

for i = 2:length(frequency_arr)-1
    if PSD(i) > PSD(i-1) && PSD(i) > PSD(i+1)
        peaks = [peaks, PSD(i)];
        locs = [locs, frequency_arr(i)];
    end
end

n_modes = 5;
```

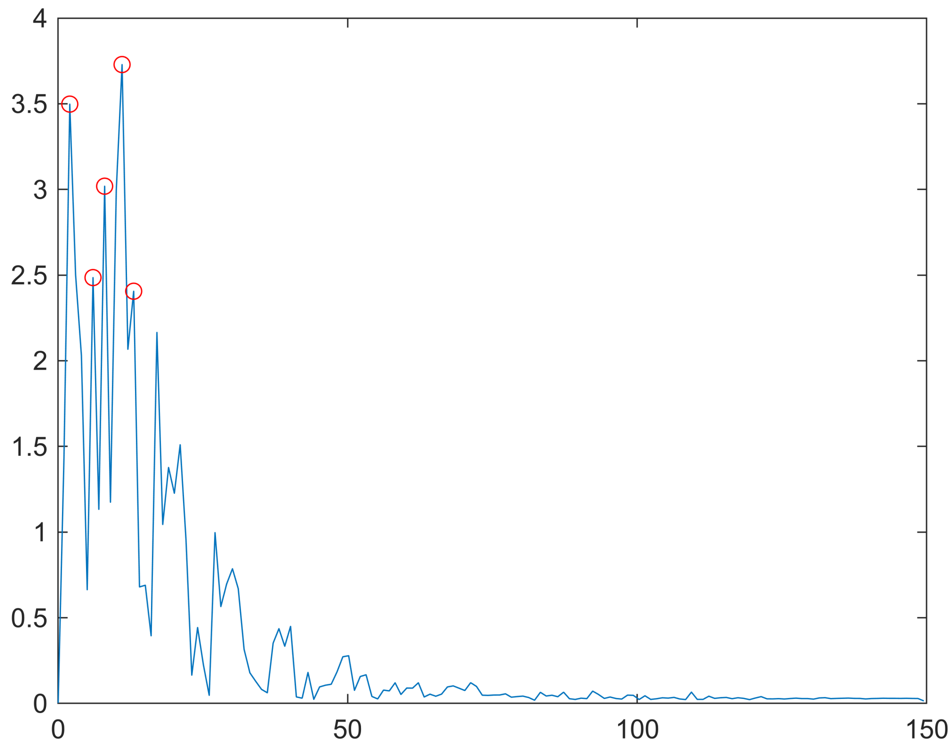
Natural Frequencies (First 5)

```
peaks = peaks(1:n_modes);
freqs = locs(1:n_modes)
```

```
freqs = 1×5
```

```
    2.0067    6.0201    8.0268   11.0369   13.0436
```

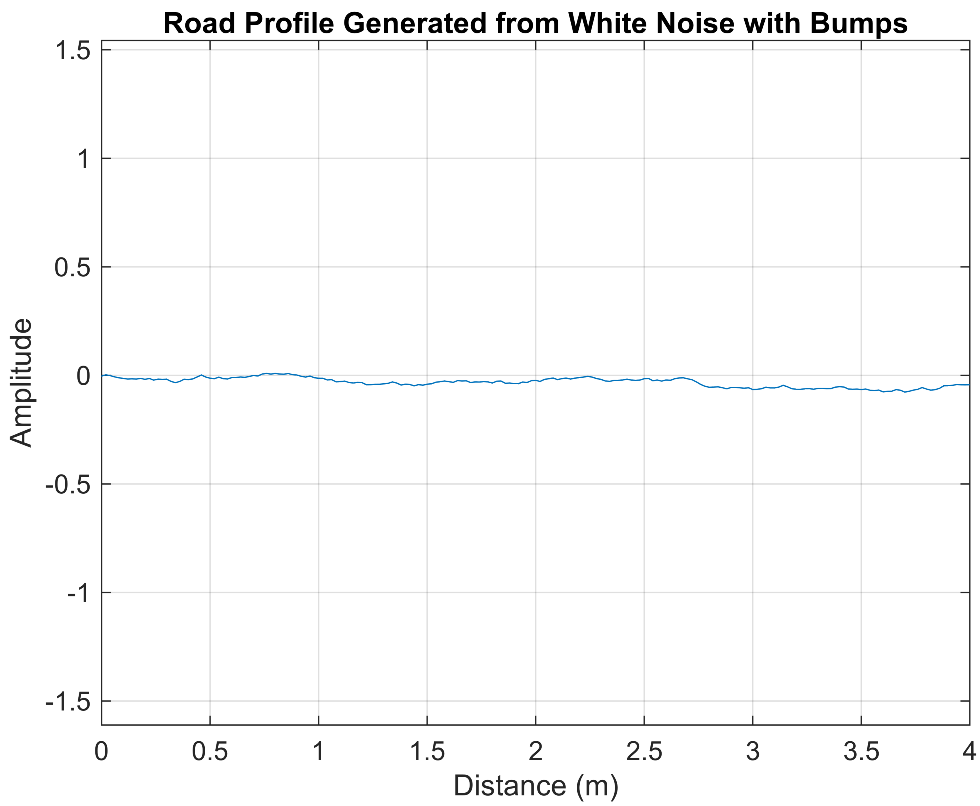
```
plot(frequency_arr, PSD)
hold on;
plot(freqs, peaks, 'ro');
hold off;
```



PARAMETER EFFECT ANALYSIS

Road Profile - Generation

```
% [X_r, Z_r] = bump_road_input(5,0.12,15);

profile_length = 200;       % Length of the road profile (meters)
sampling_rate = 50;        % Numberof points per meter
amplitude_factor = 0.005;     % Roughness
num_bumps = 5;             % Number of bumps
bump_amplitude = 0.001;       % Amplitude of bumps
bump_frequency = 0.5;       % Frequency of bumps

[X_r, Z_r] = generateRoadProfileWithBumps(profile_length, sampling_rate, amplitude_factor, num_
```




```
road.X_r = X_r;
road.Z_r = Z_r;
```

Car Model

```
%Vehicle
scooter.mass = 150;
scooter.front_unsprung_mass = 15;
scooter.rear_unsprung_mass = 15;
scooter.Lateral_MOI = 40;
scooter.CG_2_Front = 0.5;
```

```
scooter.CG_2_Rear = 0.5;

stiffness.front_strut = 15000;
stiffness.rear_strut = 15000;
stiffness.tire_front = 44000;
stiffness.tire_rear = 44000;

damping.strut_front = 1000;
damping.strut_rear = 1000;

%Velocity
initial_vel = 10; %velocity in m/s
acc  = 0;
```

Stiffness Variation

```
% Stiffness variation from 200 to 30000
% Damping kept at 1000 for analysis
k = 200:100:30000;
rms_acc = k;
natural_frequency = [k;k];
rms_pitch = k;
TRS = k;
Total_POW = k;
```

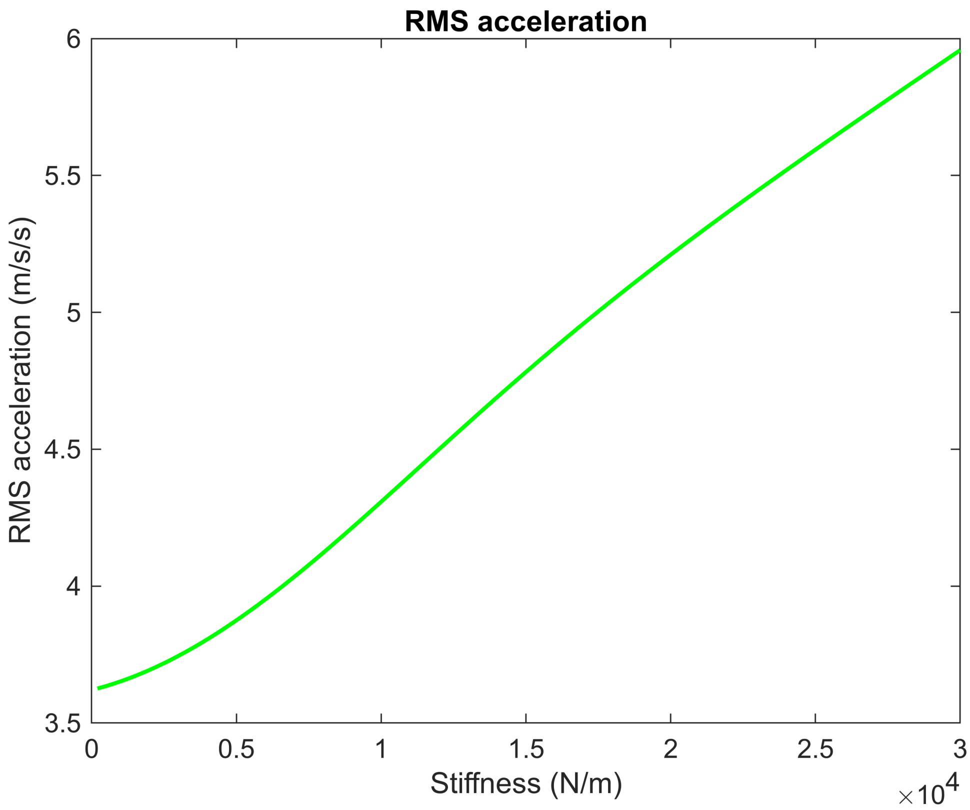
Simulating

```
for i = 1:length(k)

    stiffness.front_strut = k(i);
    stiffness.rear_strut = k(i);
    [displacement, velocity, acceleration] = Ride_Comfort_Analysis(scooter, stiffness, damping,
    rms_acc(i) = rms(acceleration.a_body);
    rms_pitch(i) = rms(acceleration.a_theta);
    TRS(i) = max(acceleration.a_body)/max(acceleration.a_unsprung_front);

    [Weighted_PSD, frequency_arr, PSD] = Frequency_analysis(acceleration.a_body,acceleration.ti
    Total_POW(i) = trapz(frequency_arr, PSD);

    peaks = [];
    locs = [];
    for j = 2:length(frequency_arr)-1
        if PSD(j) > PSD(j-1) && PSD(j) > PSD(j+1)
            peaks = [peaks, PSD(j)];
            locs = [locs, frequency_arr(j)];
        end
    end

    natural_frequency(:,i) = (locs(1:2))';
```

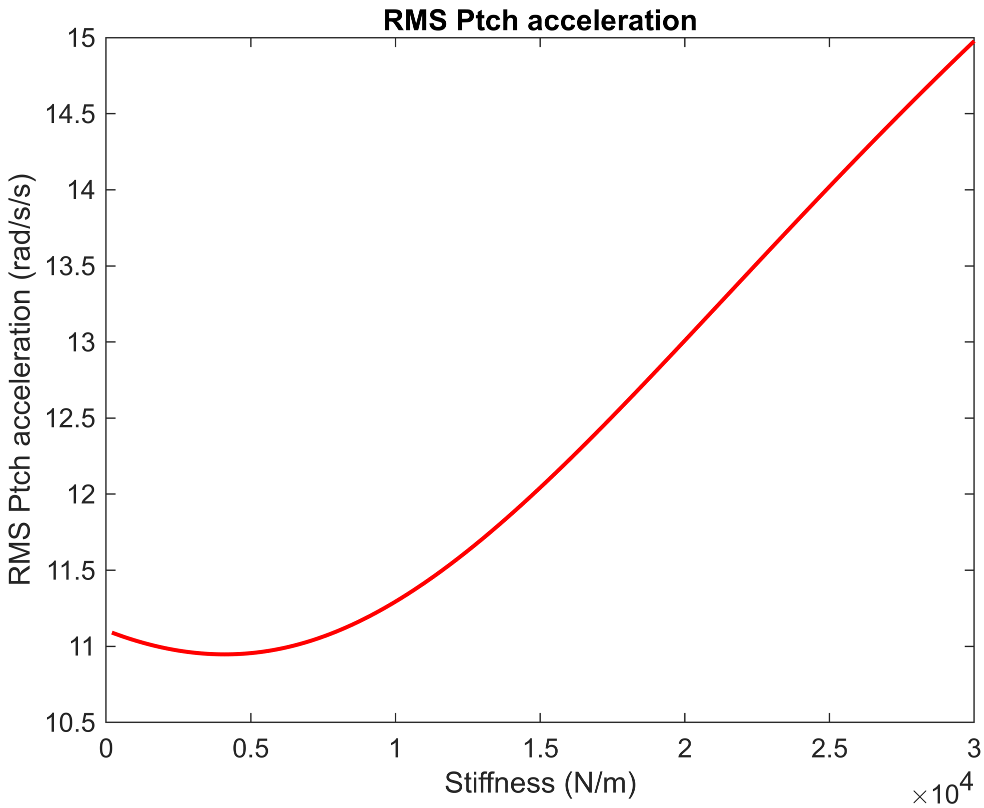
```
end
```

Plotting

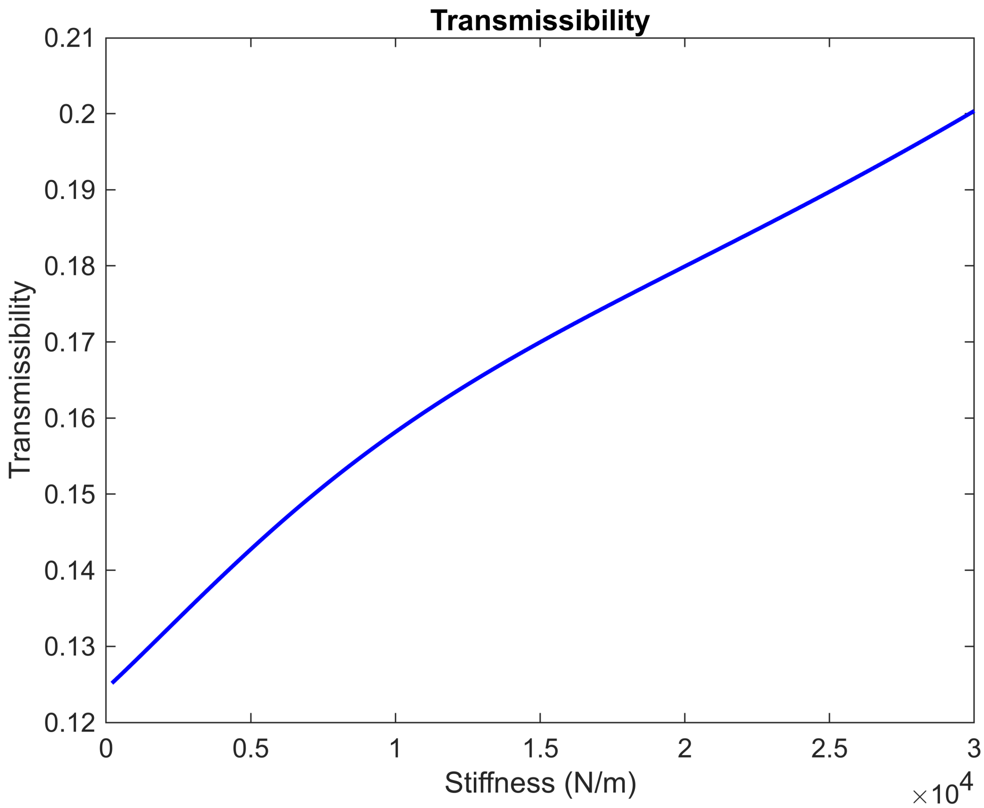
```
plot(k, rms_acc, 'g', 'LineWidth', 1.5);
xlabel('Stiffness (N/m)');
ylabel('RMS acceleration (m/s/s)');
title('RMS acceleration');
```



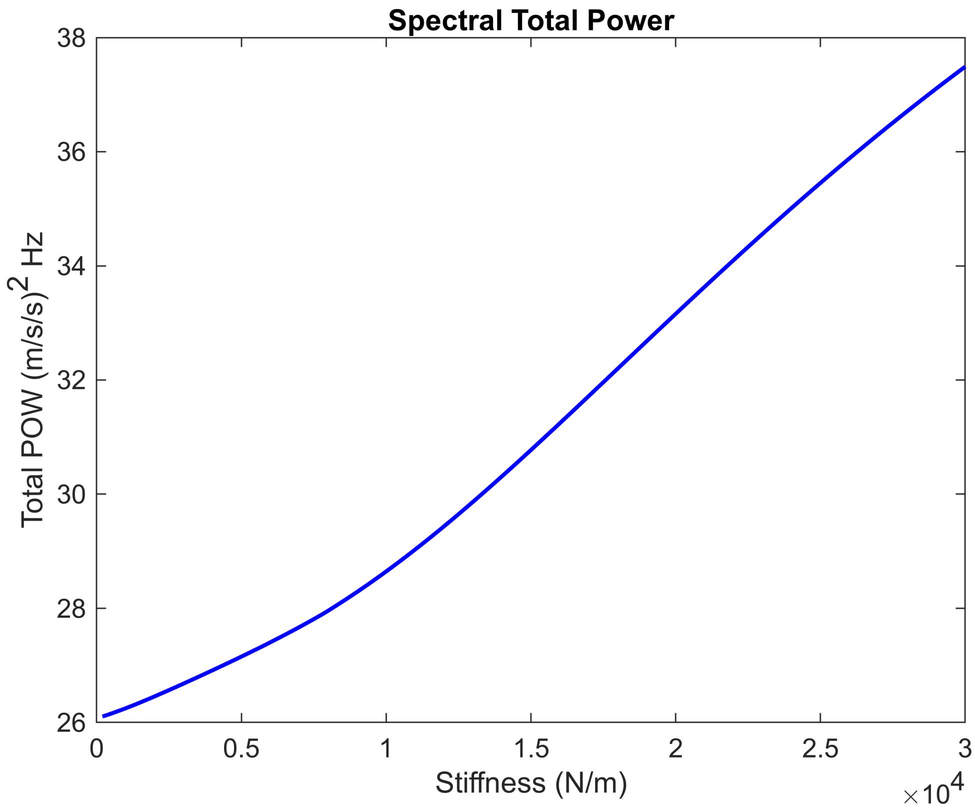

```
plot(k, rms_pitch, 'r', 'LineWidth', 1.5);
xlabel('Stiffness (N/m)');
ylabel('RMS Ptch acceleration (rad/s/s)');
title('RMS Ptch acceleration');
```



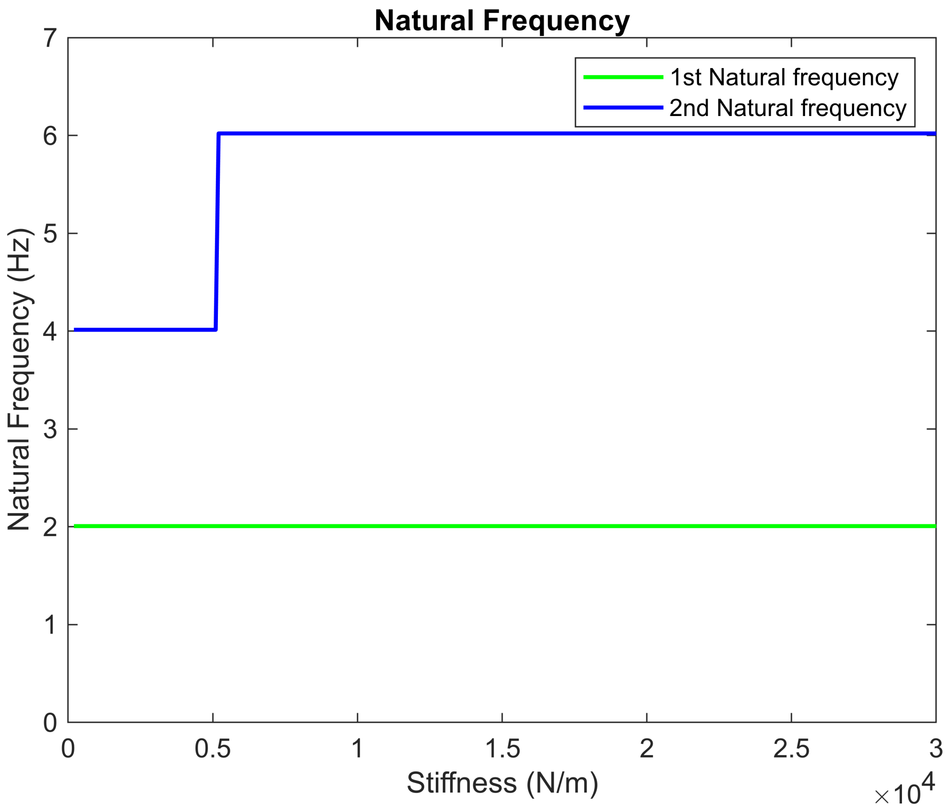

```
plot(k, TRS, 'b', 'LineWidth', 1.5);
xlabel('Stiffness (N/m)');
ylabel('Transmissibility');
title('Transmissibility');
```




```
plot(k, Total_POW, 'b', 'LineWidth', 1.5);
xlabel('Stiffness (N/m)');
ylabel('Total POW (m/s/s)^2 Hz');
title('Spectral Total Power');
```




```
plot(k, natural_frequency(1,:), 'g', 'LineWidth', 1.5);
hold on;
plot(k, natural_frequency(2,:), 'b', 'LineWidth', 1.5);
xlabel('Stiffness (N/m)');
ylabel('Natural Frequency (Hz)');
legend('1st Natural frequency', '2nd Natural frequency');
title('Natural Frequency');
ylim([0 7])
hold off
```



Damping Variation

```
% Stiffness kept at 15000 for analysis
C = 0:100:10000;
rms_acc = C;
natural_frequency = [C;C];
stiffness.front_strut = 15000;
stiffness.rear_strut = 15000;

rms_pitch = C;
TRS = C;
Total_POW = C;
```

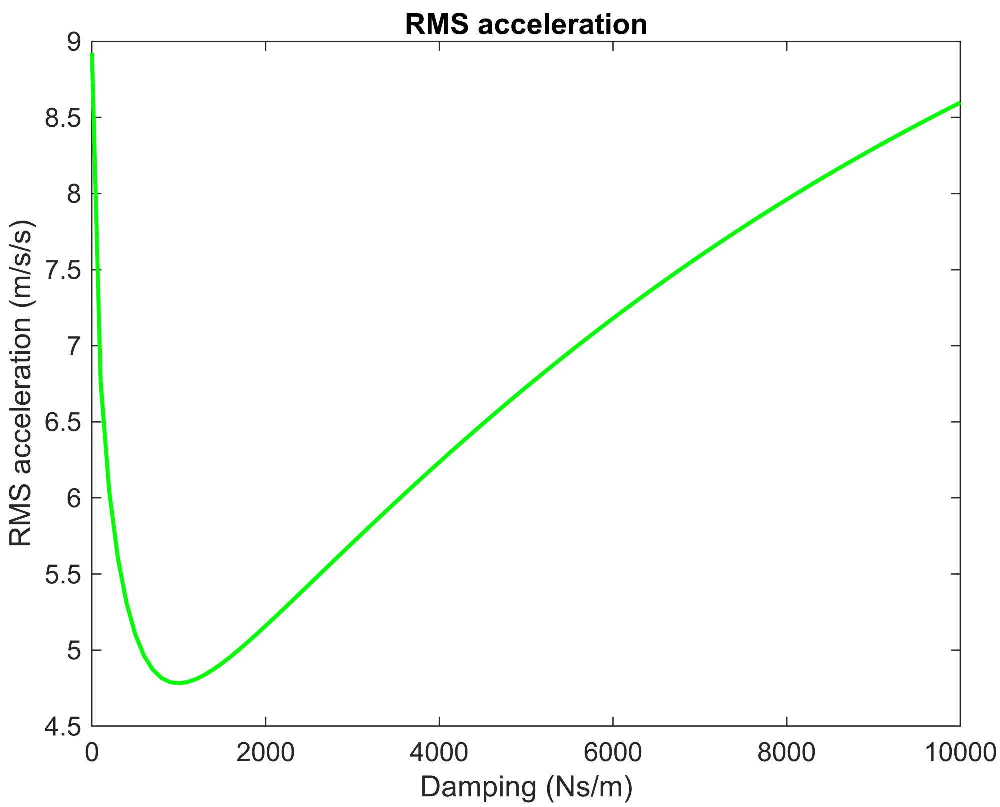
Simulating

```
for i = 1:length(C)

    damping.strut_front = C(i);
    damping.strut_rear = C(i);
    [displacement, velocity, acceleration] = Ride_Comfort_Analysis(scooter, stiffness, damping,
    rms_acc(i) = rms(acceleration.a_body);
    rms_pitch(i) = rms(acceleration.a_theta);
    TRS(i) = max(acceleration.a_body)/max(acceleration.a_unsprung_front);
```

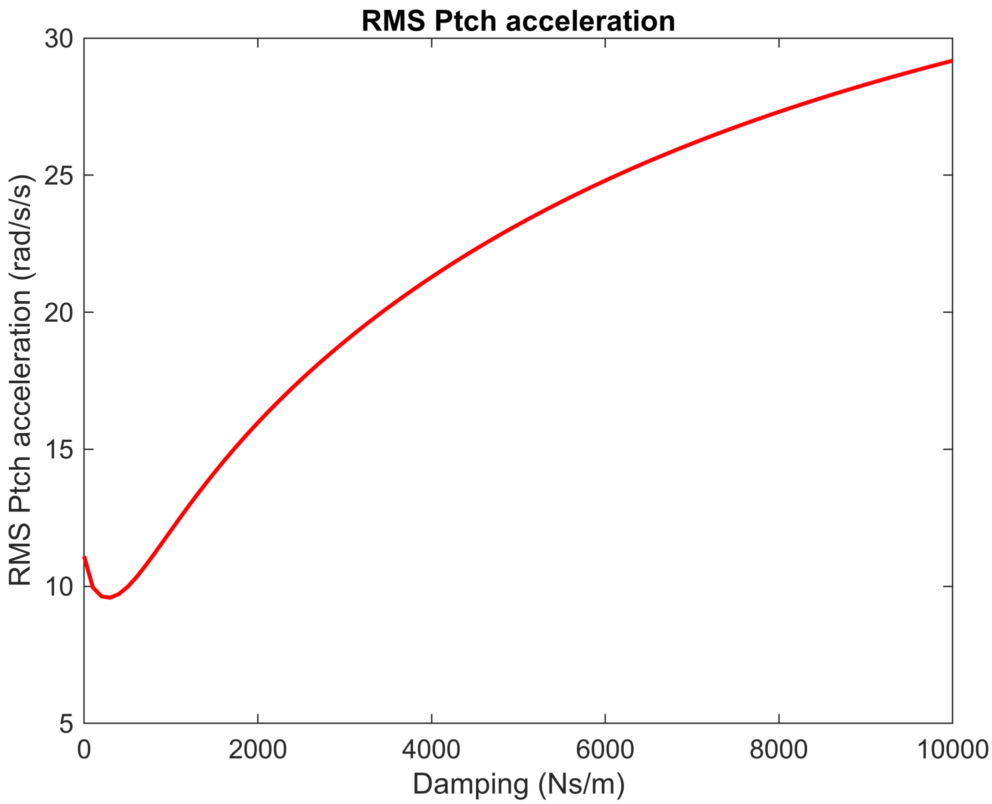
```
    [Weighted_PSD, frequency_arr, PSD] = Frequency_analysis(acceleration.a_body,acceleration.t:
    Total_POW(i) = trapz(frequency_arr, PSD);

    peaks = [];
    locs = [];
    for j = 2:length(frequency_arr)-1
        if PSD(j) > PSD(j-1) && PSD(j) > PSD(j+1)
            peaks = [peaks, PSD(j)];
            locs = [locs, frequency_arr(j)];
        end
    end
    [peaks,sortIdx] = sort(peaks,'descend');
    locs = locs(sortIdx);

    natural_frequency(:,i) = (locs(1:2))';

end
```

Plotting

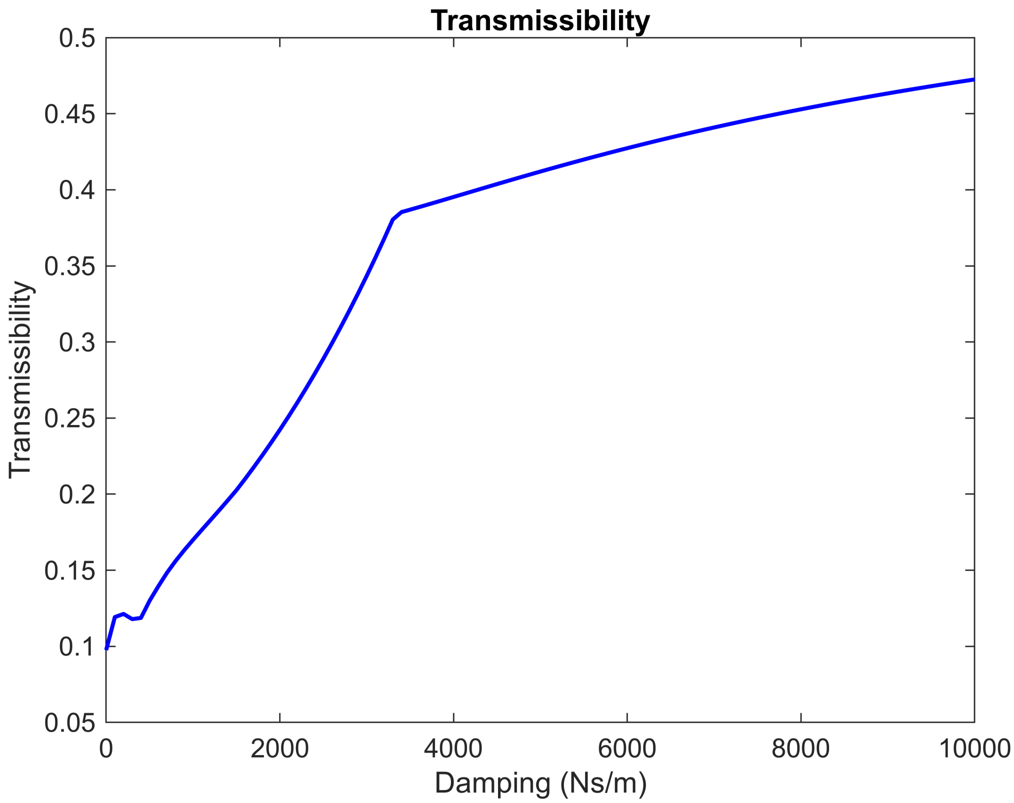
```
plot(C, rms_acc, 'g', 'LineWidth', 1.5);
xlabel('Damping (Ns/m)');
ylabel('RMS acceleration (m/s/s)');
title('RMS acceleration');
```



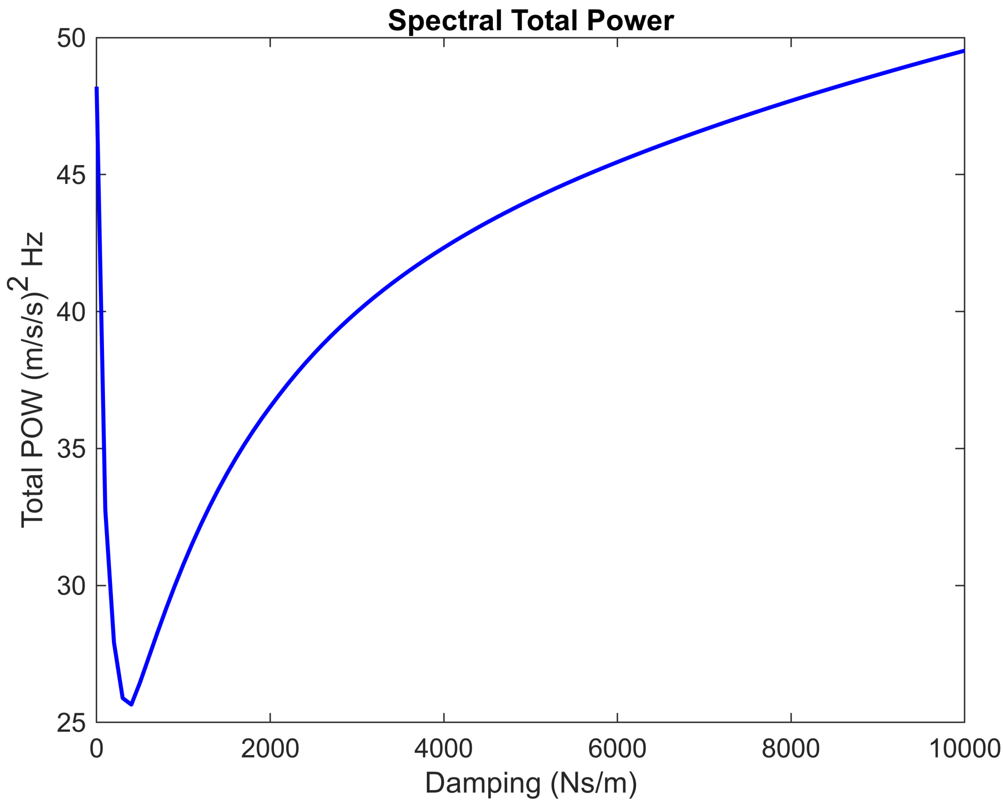
```
plot(C, rms_pitch, 'r', 'LineWidth', 1.5);
xlabel('Damping (Ns/m)');
ylabel('RMS Ptch acceleration (rad/s/s)');
title('RMS Ptch acceleration');
```



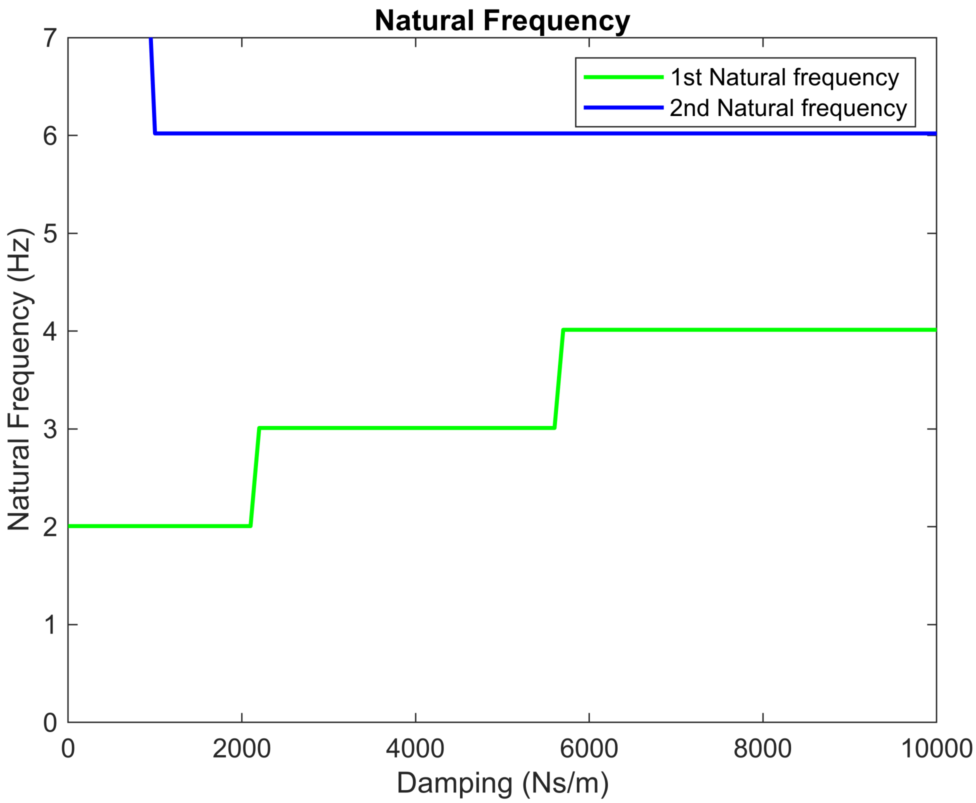

```
plot(C, TRS, 'b', 'LineWidth', 1.5);
xlabel('Damping (Ns/m)');
ylabel('Transmissibility');
title('Transmissibility');
```




```
plot(C, Total_POW, 'b', 'LineWidth', 1.5);
xlabel('Damping (Ns/m)');
ylabel('Total POW (m/s/s)^2 Hz');
title('Spectral Total Power');
```




```
plot(C, natural_frequency(1,:), 'g', 'LineWidth', 1.5);
hold on;
plot(C, natural_frequency(2,:), 'b', 'LineWidth', 1.5);
xlabel('Damping (Ns/m)');
ylabel('Natural Frequency (Hz)');
legend('1st Natural frequency', '2nd Natural frequency');
title('Natural Frequency');
ylim([0 7])
hold off
```


Natural Frequency
1st Natural frequency
2nd Natural frequency
Natural Frequency (Hz)
Damping (Ns/m)
0
1
2
3
4
5
6
7
0
2000
4000
6000
8000
10000

```
function [displacement, velocity, acceleration] = Ride_Comfort_Analysis(car, stiffness, damping

m   = car.mass;
m1  = car.front_unsprung_mass;
m2  = car.rear_unsprung_mass;
Iy  = car.Lateral_MOI;
a1  = car.CG_2_Front;
a2  = car.CG_2_Rear;
k1  = stiffness.front_strut;
k2  = stiffness.rear_strut;
kt1 = stiffness.tire_front;
kt2 = stiffness.tire_rear;
c1  = damping.strut_front;
c2  = damping.strut_rear;

playback_speed = 0.1;
tF      = 1;
fR      = 30/playback_speed;
dt      = 1/fR;
time    = linspace(0,tF,tF*fR);

X_r = road.X_r;
Z_r = road.Z_r;

M = [   m   0   0   0   ;
        0   Iy  0   0   ;
        0   0   m1  0   ;
        0   0   0   m2  ];

C = [   c1+c2           a2*c2-a1*c1         -c1         -c2     ;
        a2*c2-a1*c2     c1*a1^2+c2*a2^2     a1*c1       -a2*c2  ;
        -c1             a1*c1               c1          0       ;
        -c2             -a2*c2              0           c2      ];

K = [   k1+k2           a2*k2-a1*k1         -k1         -k2     ;
        a2*k2-a1*k1     k1*a1^2+k2*a2^2     a1*k1       -a2*k2  ;
        -k1             a1*k1               k1+kt1      0       ;
        -k2             -a2*k2              0           k2+kt2  ];

F = [   0       0   ;
        0       0   ;
        kt1     0   ;
        0       kt2 ];

% State space model
A = [   zeros(4,4)      eye(4,4)   ;
        -M\K         -M\C    ];
B = [   zeros(4,2)  ;
        M\F      ];
C = [   1 0 0 0 0 0 0 0 ;
```

```
        0 1 0 0 0 0 0 0 ;
        0 0 1 0 0 0 0 0 ;
        0 0 0 1 0 0 0 0 ;
        0 0 0 0 0 0 0 0 ;
        0 0 0 0 0 0 0 0 ;
        0 0 0 0 0 0 0 0 ;
        0 0 0 0 0 0 0 0 ];
D = zeros(8,2);

sys = ss(A,B,C,D);

% Input
lon_pos_2 = vel*time + 0.5*acc*(time.*time);           % Longitudinal position of the rear axle
lon_pos_1 = lon_pos_2 + a1+a2;   % Longitudinal position of the front axle   [m]
% OBS: Added wheelbase!
%
u1 = interp1(X_r,Z_r,lon_pos_1);
u2 = interp1(X_r,Z_r,lon_pos_2);

u_vet = [u1' u2'];

[y,time,x] = lsim(sys,u_vet,time);

z       = y(:,1); % Body vertical motion coordinate         [m]
theta   = y(:,2); % Body pitch motion coordinate            [rad]
zu1     = y(:,3); % Front wheel vertical motion coordinate  [m]
zu2     = y(:,4); % Rear wheel vertical motion coordinate   [m]

% Time step
dt = mean(diff(time));

% Velocity calculation
v_z = diff(z) / dt;  % Body vertical velocity
v_theta = diff(theta) / dt;  % Body pitch velocity
v_zu1 = diff(zu1) / dt;  % Front wheel velocity
v_zu2 = diff(zu2) / dt;  % Rear wheel velocity

% Acceleration calculation
a_z = diff(v_z) / dt;  % Body vertical acceleration
a_theta = diff(v_theta) / dt;  % Body pitch acceleration
a_zu1 = diff(v_zu1) / dt;  % Front wheel acceleration
a_zu2 = diff(v_zu2) / dt;  % Rear wheel acceleration

% Time vector for velocity and acceleration (one element less due to differentiation)

displacement.z_body =z;
displacement.z_unsprung_front = zu1;
displacement.z_unsprung_rear = zu2;
displacement.theta = theta;
displacement.time = time;
```

```
displacement.tire_front = u1/2;
displacement.tire_rear = u2/2;
displacement.longitudinal_pos_front = lon_pos_1;
displacement.longitudinal_pos_rear = lon_pos_2;

velocity.v_body = v_z;
velocity.v_unsprung_front = v_zu1;
velocity.v_unsprung_rear = v_zu2;
velocity.v_thetha = v_theta;
velocity.time = time(1:end-1);

acceleration.a_body = a_z;
acceleration.a_unsprung_front = a_zu1;
acceleration.a_unsprung_rear = a_zu2;
acceleration.a_theta = a_theta;
acceleration.time = time(1:end-2);

end
```