Project: Food serving differential drive mobile robot

Introduction

This project details the creation and analysis of a differential drive mobile robot for food serving purposes. The report outlines the construction of a navigation system, coupling it with a DC motor system model using governing equations derived to describe the robot's movement and torque distribution. It discusses the development of a Simulink model representing the navigation and motor systems, integrating theoretical elements like a theoretical damper and mechanical impedance. Additionally, the report covers the design of a suspension system, considerations for floor path and door threshold profiles, and the intricate tuning process for motor parameters and PID controllers to ensure stability and accuracy in controlling the robot's motion. The project concludes with insightful observations about the system's components, including jerk, acceleration, and damping coefficients.

Derivations

When constructing the coupled navigation and DC motor system model, the governing equations describing the robot’s navigation were determined. From that a model of the navigation system was derived. See Figure 1 below, where L is the distance of a wheel to the centre of mass of the robot, R is the radius of the wheels, J is the first moment of inertia, and M is the mass of the robot.

$$M a_u = F_{uL} + F_{uR}$$

$$M \dot{V}_u = F_{uL} + F_{uR}$$

$$\dot{V} = \frac{F_{uL} + F_{uR}}{M}$$

$$\dot{V} = \frac{\tau_L + \tau_R}{RM} \quad (1)$$

$$J \ddot{\theta} = (F_{uR} - F_{uL}) L$$

$$\ddot{\theta} = \dot{\omega} = (F_{uR} - F_{uL}) \frac{L}{J}$$

$$\dot{\omega} = (\tau_R - \tau_L) \frac{L}{JR} \quad (2)$$

Figure 1: Navigation Governing Equations

Through Newton's second law of motion, the rate of change of the planar velocity of the robot was put in terms of the torque experienced by the left wheel and the torque experienced by the right wheel. Furthermore, the rate of change of the angular velocity of the robot was also expressed in terms of the torque experienced by both wheels of the motor. This angular velocity is with respect to a vertical axis located at the robot's centre of mass. It was assumed that the centre of mass is located at the centre point between the two wheels. See Figure 2 below for the derivation.

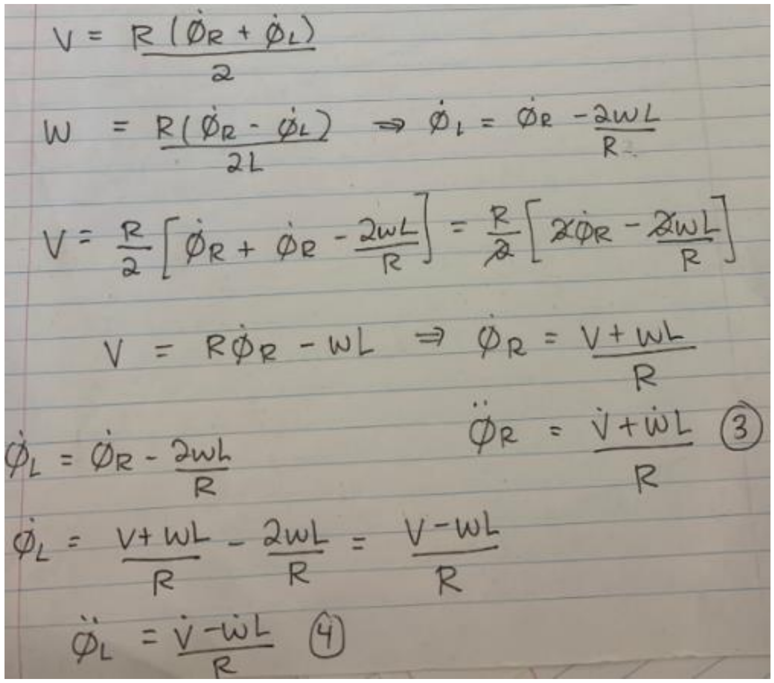

Figure 2: Navigation Equations in Terms of Torque

The angular accelerations of the left and right wheels of the robot can be expressed in terms of the planar and angular velocities of the robot. These four governing equations were used to construct the block diagram in Simulink seen in Figure 3 below.

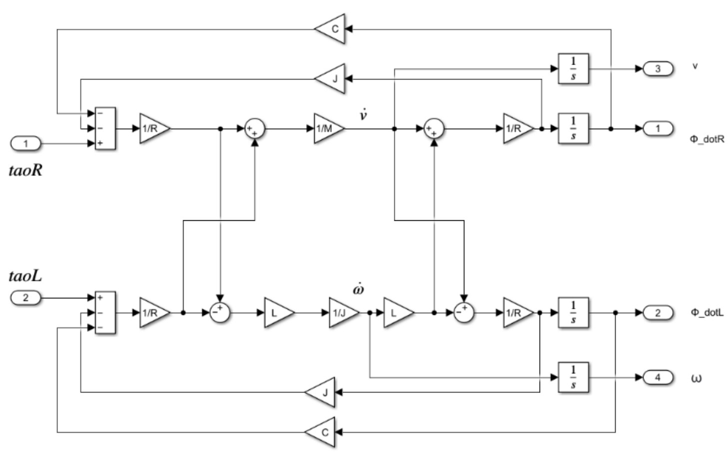

Figure 3: Simulink Model of Navigation System

Here C describes a theoretical damper that opposes the direction of torque the motor generates. Furthermore, since the wheel has a mass, it has a counteracting mass moment of inertia represented here by "J". This mass moment of inertia resists the torque being applied to it by the motor, and a counter torque is present that is equal to $J\ddot{\Phi}$. These two are subtracted from the input torque and represent the mechanical impedance of the load on the motor. The DC motors of the robot is what will be providing the torque to the wheels which allows them to turn. Therefore, the DC motor system will be able to be coupled to the navigation system by having the torque the motor produces be an input to the block diagram created for the robot's navigation model.

The equations in Figure 4 describe the behaviour of a DC motor. They were rearranged as shown to put the torque of the motor in terms of input voltage to the motor and the angular velocity of the wheel the motor is spinning. Since there are two wheels, there will be two input voltage signals and two resulting torques. This information was used to construct the navigation block diagram in Simulink. The subsystem in Figure 5 holds the block diagram shown earlier for the navigation model.

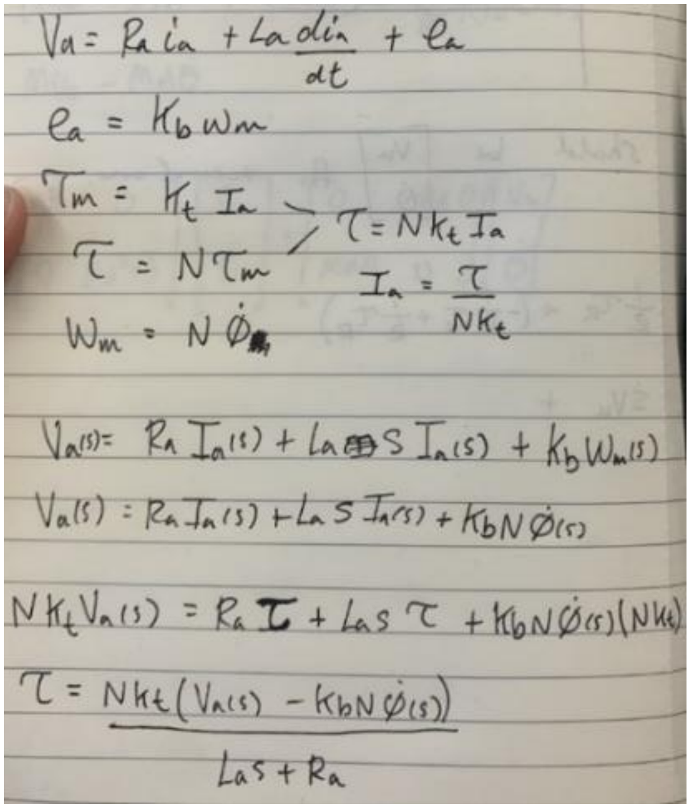

Figure 4: DC Motor Equations

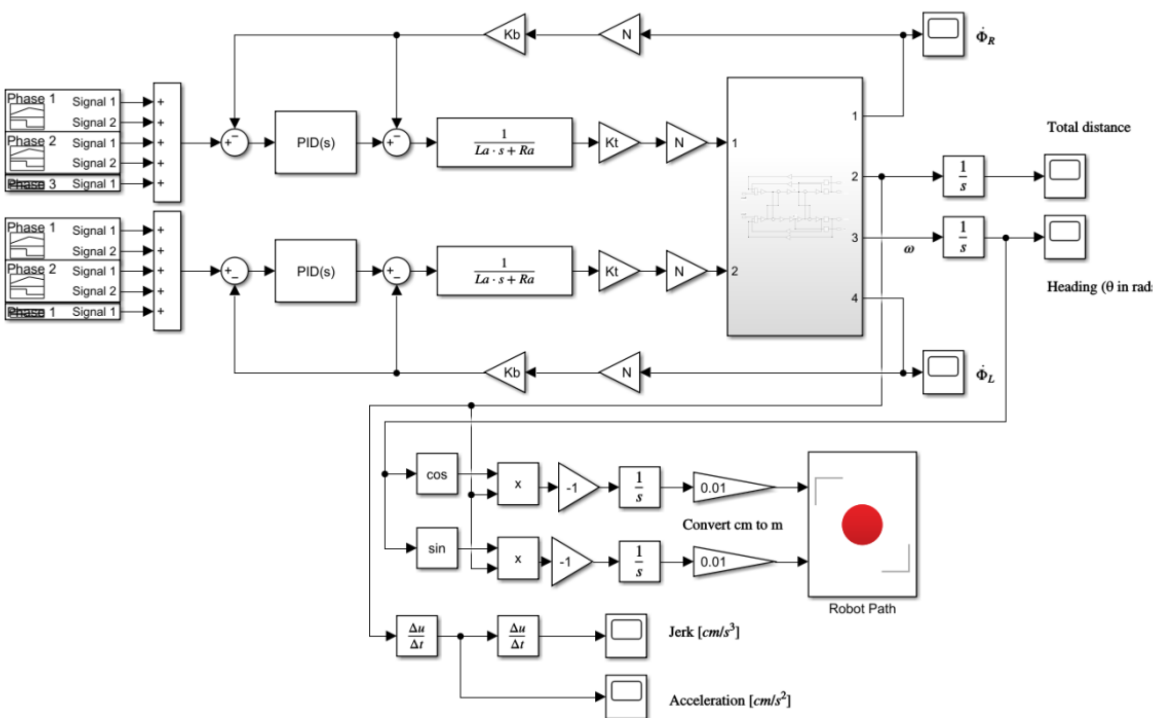

Figure 5: Full Navigation System

The model for the robot's suspension system was created by using Simscape elements for mechanical systems. The model is shown in Figure 6.

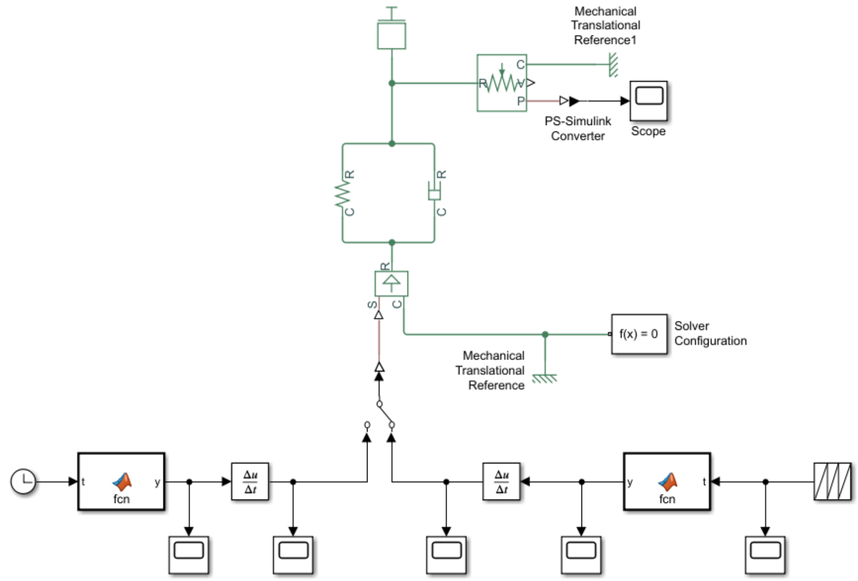

Figure 6: Simscape Suspension System

Suspension System Characterization

The robot will move on a tiled floor, and the grouting between the tiles will be considered for the path of the robot using a haversine profile. The grouting is 0.5 cm deep, and as such the amplitude was set to -0.5. Furthermore, the titles are 1ft by 1ft with 1cm grouting between the tiles. Depending on the velocity, the amount of time it takes to cover this distance will change, essentially increasing or decreasing the period. The period of the floor signal can be described with 31.48/v, which is derived from the tile dimensions (1ft = 30.48 cm, plus 1 cm grouting). Following this, the pulse width was set to (1/31.48)*100% to consider the percentage of the total distance which the robot dips (1cm of the total 31.48 cm).

For instance, with a velocity of 3 m/s and 5m of planar motion, the robot should cross 15.9 tiles in a span of 1.67 seconds. Additionally, since it is assumed that the robot begins to move right before the first tile, it crosses 16 grouts in total. See Figure 7 for the floor signal where the vertical axis represents the vertical distance of the floor signal and the horizontal axis represents time. Each dip in the signal represents grouting which is -0.5 cm deep, while the length of the flat edge at 0 on the vertical axis represents the time it took for the robot to cross one tile.

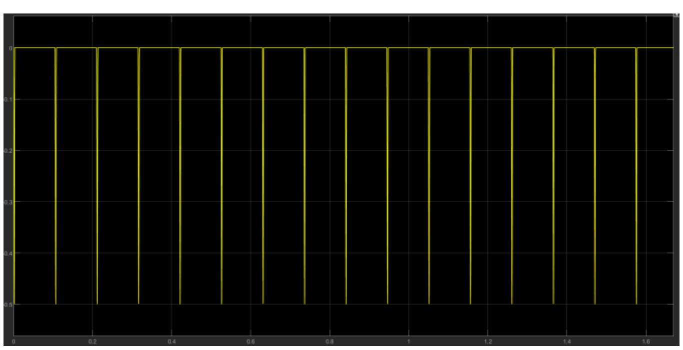

Figure 7: Floor Signal

Furthermore, the door threshold has a length of 5 cm and height of 3 cm that the robot must pass. A haversine cross section was assumed for the door threshold, and a signal corresponding to it was made. Using section 4.8.4 from the System Dynamics Fourth Edition (BY William J. Palm III), an equation can be derived for this signal:

$$y(t) = H\left(\sin\left(\frac{\mathrm{vt}}{\mathrm{L}}\right)\right)\left(\sin\left(\frac{\mathrm{vt}}{\mathrm{L}}\right)\right)$$

This equation reasonably represents a surface variation encountered when driving over the door bump, where H and L represent the height and length respectively. The haversine equation describing the door bump, and planar distance of 5m is described as (where distances are in cm):

$$y(t) = 3\left(\sin\left(\frac{\mathrm{vt}}{5}\right)\right)\left(\sin\left(\frac{\mathrm{vt}}{5}\right)\right)(heaviside(vt) - heaviside(vt-5))$$

The heaviside function in the expression was added to isolate a single bump. Finally, the signal after the robot moves a planar distance of 5m (1.67s) can be seen in Figure 8 below. The same signal with a smaller stop time of 1 second can be seen in Figure 21 in Appendix A.

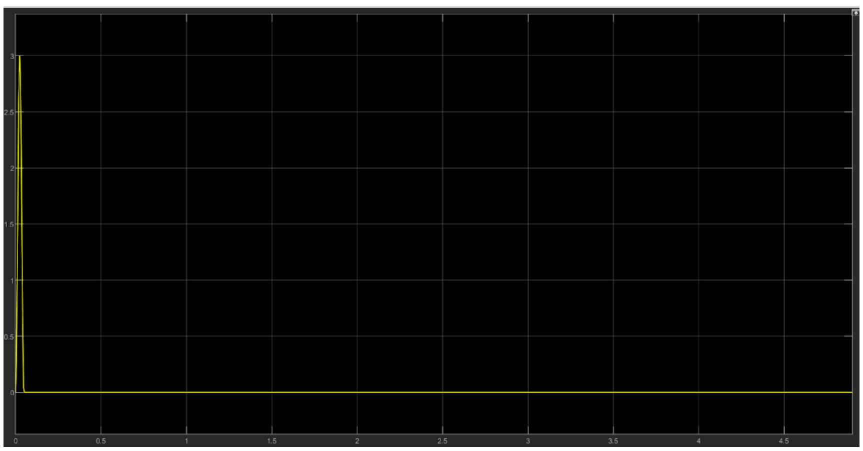

Figure 8: Door Threshold Signal

Assuming motion was in the slew phase (steady state constant planar robot velocity), the stiffness and the damping parameters of the model had to be iterated to describe the correct wheel motion for an assumed planar velocity. The velocity was initially assumed as 3 m/s, however, several combinations through trial and error of stiffness and damping parameters, gave results that exceeded the robot's allowed displacement of 0.5 cm or other errors. Most of these combinations gave erroneous damping patterns with an amplitude that exceeded 0.5 cm as shown in Figure 9:

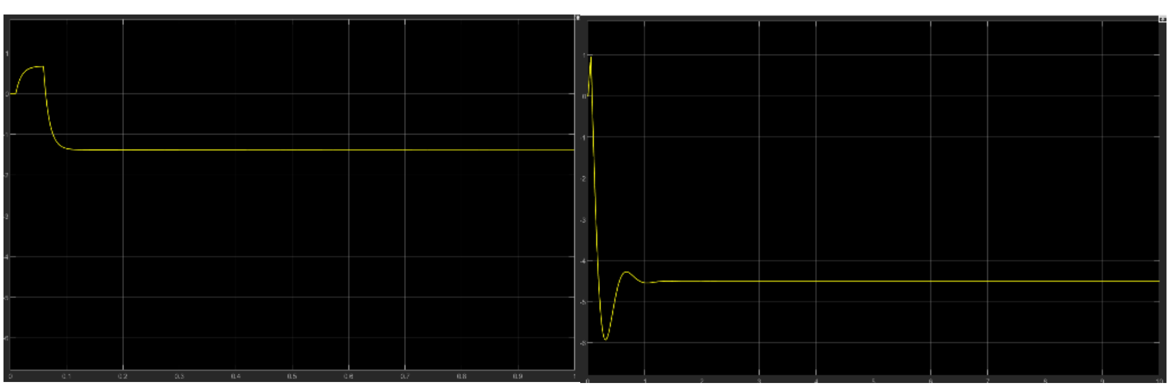

Figure 9: Erroneous Dampening Signals

A more methodical approach was chosen, which was to design the system to be critically damped as critically damped systems return a system to the equilibrium state as quickly as possible, and minimize overshooting unlike underdamped systems which oscillate before returning to equilibrium. To find the ideal parameters, the equation below was used with an arbitrary value of k = 6.9 N/m to find c.

$$c^2 = 4mk$$

Unexpectedly, these critically damped parameters, k = 6.9 N/m and c = 23.5 Ns/m still resulted in some overshooting in some of the cases. Therefore, the dampening coefficient was iterated on through trial and error until a final coefficient of 40 Ns/m was found to have a lot less overshooting. The final constants were chosen to be k = 6.9 N/m, and c = 40 Ns/m. The maximum vertical displacement was found when the robot is at its lightest, with a total mass of 10kg per wheel, resulting in a maximum vertical displacement of 0.0971 cm when travelling over the door bump, as shown in Figure 10 for 5m of planar motion. A magnified version of the graph can be seen in Figure 24 in Appendix A.

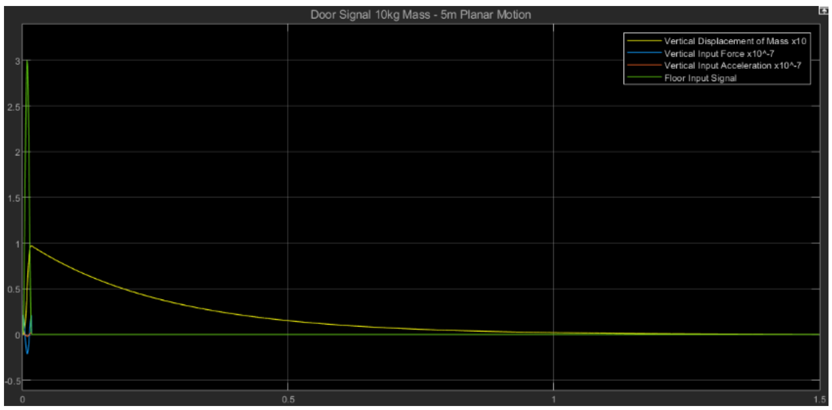

Figure 10: Vertical Displacement of 10kg Mass while travelling over the door bump

For these same parameters a vertical displacement of 1.603*10^-3 cm was experienced with the tiled floor input signal. This is shown in Figure 11 for 5m of planar motion. A magnified version can be found in Figure 22 in Appendix A.

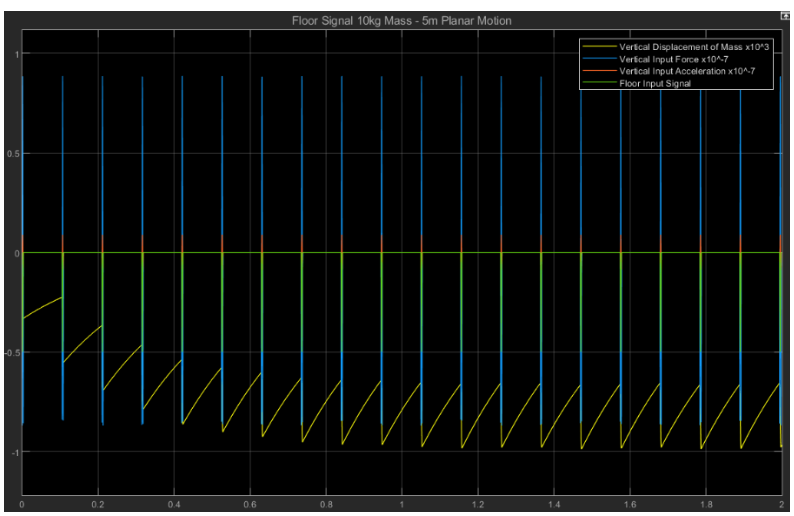

Figure 11: Vertical Displacement of a 10kg Mass while travelling over the tiled floor

A fully loaded robot with a total mass of 20kg per wheel traveling at a velocity of 3 m/s with the same k and c values as before had a maximum vertical displacement of 0.049238 cm. As shown in Figure 12 for 5m of planar motion. A magnified version of the graph can be found in Figure 25 in Appendix A.

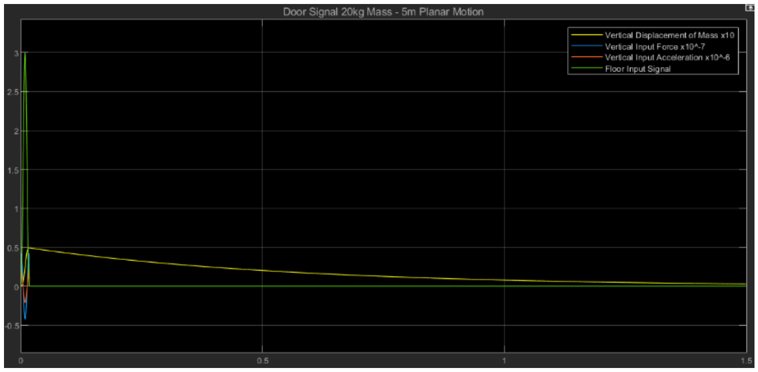

Figure 12: Vertical Displacement of 20kg Mass while travelling over the door bump

For these same parameters, a maximum vertical displacement of 0.00899 cm is experienced by the robot when passing over the floor input. This is shown in Figure 13. A magnified version can be seen in Figure 23 in Appendix A.

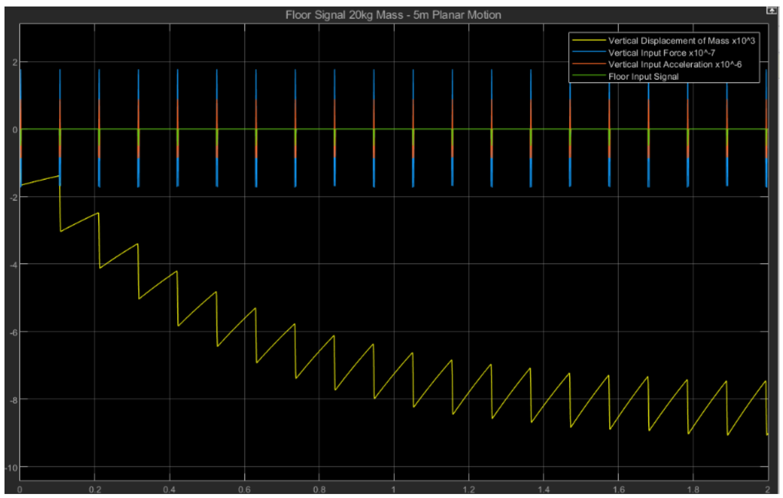

Figure 13: Vertical Displacement of a 20kg Mass while travelling over the floor

The robot's overall vertical displacement is also affected by the maximum height allowed for door thresholds as well as the required smoothness of the threshold profile. A higher threshold will increase the robot's vertical displacement. Furthermore, the vertical displacement will also depend on the threshold profile. A rectangular cross sections would have a much larger vertical displacement compared to a smoother profile such as the provided haversine function. The plots of vertical force and vertical acceleration vs time for the floor input signal for both masses can be seen in Figure 11 and 13. The plots of vertical force and vertical acceleration for door input signal for both masses can be seen in Figure 10 and Figure 12. For more zoomed in versions of the mentioned plots see Figure 22, Figure 23, Figure 24, and Figure 25 in Appendix A. Under all of the conditions discussed above, the vertical displacement of the robot does not exceed 0.5 cm. Therefore, the suspension design is valid.

Motor Parameters & Control

For the motor, hypothetical parameters were selected by initially setting all motor parameters at 1 then changed to improve overall stability or repeatability. When all values were set to 1, the motor drew upwards of 15kA which was not only unrealistic, but difficult to control; motor parameters were adjusted to reduce the current required per unit of torque to reduce current which increased stability with peak currents only reaching ~10 A, making it easier to control. The iterations resulted in the following parameters: Kb = 0.2 V·s/rad, Kt = 330 N·m/A, La = 1 H, N (gear ratio) = 1:5, and Ra = 1 ohm.

The design of the geometrical properties of the robot follow the provided outline, with the robot being modeled as a cylinder with a height of one meter, having a wheel diameter of 10 cm, and having a total diameter of 50cm. Other geometrical properties with reference to Figure 14 include the distance of the projected center of mass to point A being 0 cm and the distance to center of the robot to either wheel being 25cm (half of the robot's diameter).

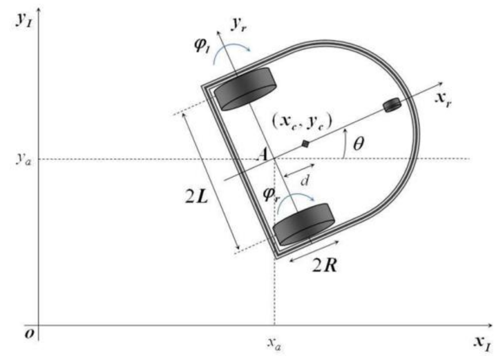

Figure 14: Diagram of Differential Drive Mobile Robot

To control the DC motor, a trapezoidal input voltage curve is ideal to avoid large accelerations, or a large influx of current when compared to applying a voltage directly to the motor without ramping up. Through trial and error of different input voltages, it was found that an input voltage of 60V resulted in the required top speed of 300 cm/s. Next, for the ramp up and down of the voltage curve, various slopes of the trapezoidal voltage curve were iterated on to minimize overshooting the required velocity. An initial value of 300 cm/s^2 was arbitrarily selected as a start, which was then iterated on, and gradually reduced until overshooting was minimized. Eventually a final acceleration value of 120m/s^2 was settled on.

To calculate the distance travelled by the robot, a relationship between distance traveled and voltage was required. Through trial and error, the area under the trapezoidal voltage curve was found to be a quarter of the distance that the robot travels. This relationship was used to design the voltage signals for the planar path.

To better control the motor, a PID controller was implemented in Simulink to improve the repeatability and stability of the speed and acceleration of the motor. Adding the controller converts the system from an open loop system into a closed loop system through the addition of a feedback loop from the left and right speeds of either motor speed to the input voltage. A big part of implementing the PID loop was tuning the P, I and D parameters along with the derivative filter coefficient; with trial and error, the P and I parameters were changed to reduce overshoot and oscillation as much as possible, then the rest of the filtering was done with done through the D term and derivative filter constant. After implementing and tuning the PID controller the results were minimal velocity overshoots and dampened oscillations in current and speed, providing a very accurate platform to control the motor.

The trapezoidal voltage curve would accelerate the robot at 1.2m/s^2 to a top speed of 3m/s, travel at 3 m/s, and then decelerate at the same rate to cross the first 15m distance. Then turn in place, where the right wheel moved backwards, with the left wheel moving forward. Since the turns were small and did not required the robot to travel a great distance, the acceleration was reduced by one third to 40 cm/s^2. It is represented by a change in slope that goes in the opposite direction of the input voltage signal for the right wheel and a change in slope that goes in the same direction for the left wheel as seen in Figure 16 and Figure 17.

Navigation

The wheels rotate until the robot has turned 90 degrees clockwise. When rotating, the area under the input voltage curves were integrated so that the angular displacement was 90 degrees. Figure 15 shows the angular displacement with respect to the robot's initial heading. After the first 15m, the robot turns until the angular displacement reaches -pi/2, or 90 degrees to the right, then again at 25m (10m after the first path). The signal follows the same design process for each segment that the robot travels (15m, 10m, and 8m). The complete input voltage signals for the left and right wheel can be seen in Figure 16 and Figure 17.

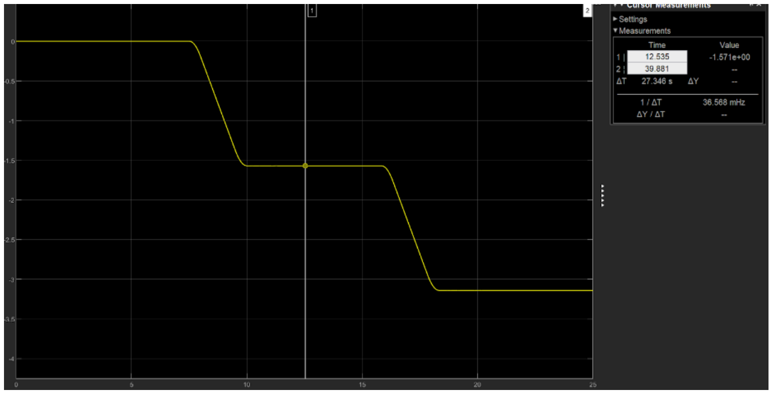

Figure 15: Plot of Angular Displacement

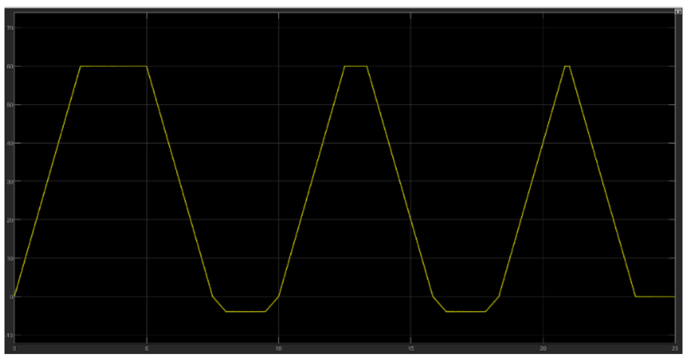

Figure 16: Voltage vs Time for Right Wheel

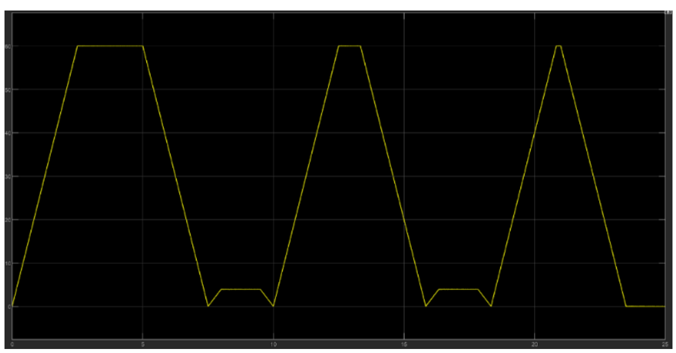

Figure 17: Voltage vs Time for Left Wheel

Using the plot of velocity of the robot, plots of the robot's acceleration and jerk were found by differentiating the velocity to find acceleration and differentiating again to find jerk. To plot the path of the robot, the heading of the robot was put through a sine and a cosine block then both outputs of the blocks were multiplied by distance travelled which resulted in the displacement in the X and Y axis of the robot over time, or overall path travelled. The plots of instantaneous acceleration (cm/s^2) vs time(s), jerk of the robot (cm/s^3) vs time(s), and robot path (X and Y) can be seen in Figure 18, 19, and 20, respectively.

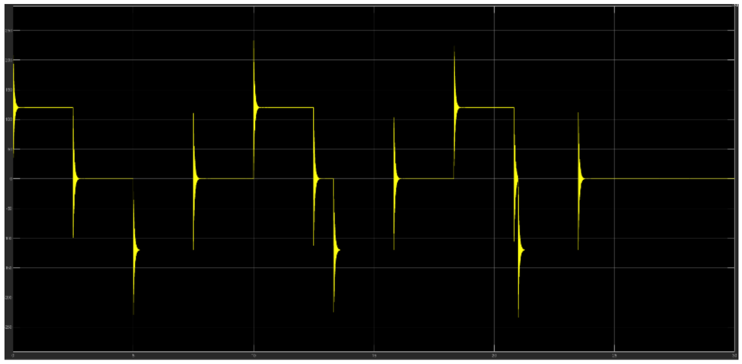

Figure 18: Plot of Instantaneous Acceleration vs Time

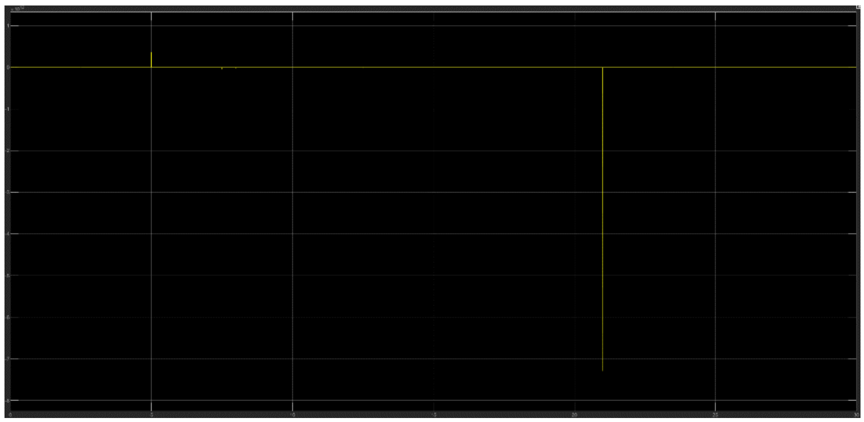

Figure 19: Plot of Jerk vs Time

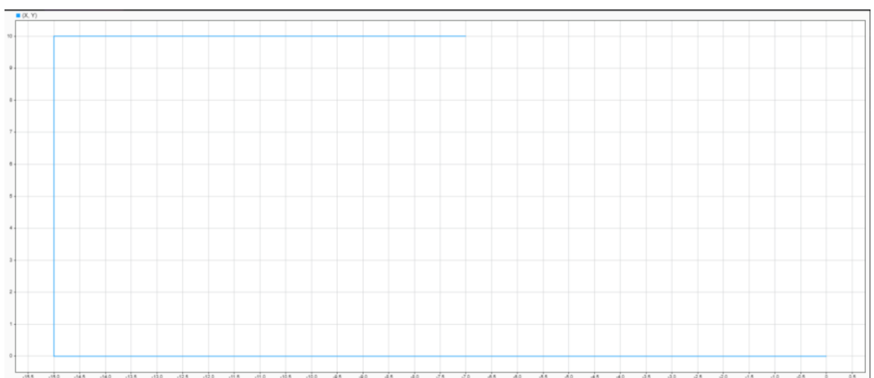

Figure 20: Path of the Robot

Conclusion

To analyze and model the path of the food serving robot, several steps were taken. The robot was modelled using the given parameters in the provided outlines (robot body, mass, etc.). First, governing equations were found that coupled the robot's navigation and DC motor system. These equations were used to model a navigation system in Simulink. The DC motor system was coupled to the navigation system by having the motor torque be an input to the navigation model.

Next, equations were found that modelled the behaviour of the DC motors. These equations were used to complete the DC motor system model in Simulink. As well, the suspension system was designed using elements from SimScape.

To make the floor signal of the grouting and tiles, several parameters were chosen such as amplitude, period, and pulse width which were set to -0.5, 31.48/v, and (1/31.48)*100%, respectively.

The next step involved identifying the mechanical parameters in the suspension system (spring constant and coefficient of dampening) that allowed the robot to travel at 3m/s with minimal displacement. The process took several iterations of trial and error, however, after realizing the tediousness and inefficiency of this process, a new design process was used. The problem was attacked with a more methodical approach which was to use knowledge of dampening system equations and ideal dampeners to systematically find a spring constant of 6.9 N/m and coefficient of dampening of 40. With these parameters, the robot at full load and minimal load experienced a max displacement less than 0.5 cm with both floor signals.

To find the voltage required, hypothetical motor parameters and geometrical properties were determined. To overcome the challenge of relating the voltage to the displacement of the robot, trial and error was performed where in each trial enough data was collected that patterns were found between the input voltage, slope, area under the curves, and distance. It was found that an input voltage of 60V, a forward acceleration of 1.2 m/s^2, and a 40 cm/s^2, results in the required top speed of 3 m/s. Thus, using these parameters, trapezoidal signals were designed so that the area under the curve was made to be a quarter of the distance that the robot travels. A PID controller was implemented for each motor to convert the system into a closed loop system and improve stability of the motor. P, I, and D parameters and the derivative filter coefficient were tuned to filter the signal and reduce overshoot.

The plot of the robot's acceleration and jerk were found by differentiating and 2^{nd} order differentiating the robot's velocity, while the plot of the robot's path was found using trigonometric equations.

Some notable observations about the components and parameters of the system include a high jerk, which caused the acceleration to oscillate and drop from a high acceleration to a low acceleration very quickly. As well, the coefficient of dampening which provides critical damping was unexpectedly overshooting. Increasing the coefficient caused an overdamping system, yet reduced overshoot. Reducing the coefficient brought the system closer to critical damping, however, substantially increased overshoot. It was also found that increasing the spring stiffness coefficient resulted in an underdamped system with large overshoot, while decreasing the spring stiffness resulted in an overdamped system with a much smaller overshoot.

Appendix A:

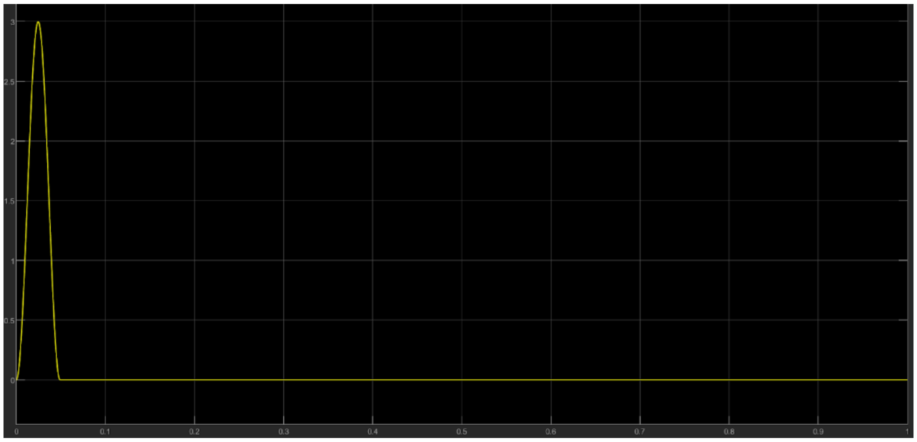

Figure 21: Door Threshold Signal Over 1 Second

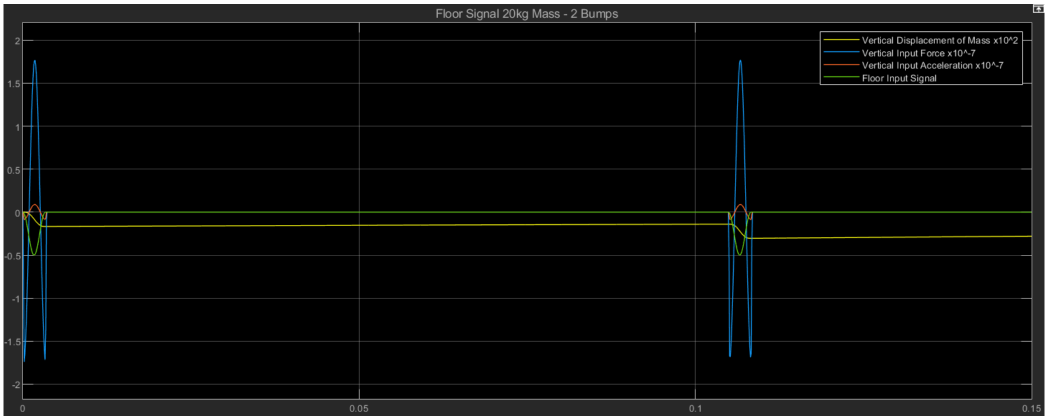

Figure 22: Plot of Vertical Acceleration and Force vs Time for 2 Bumps - 20 kg

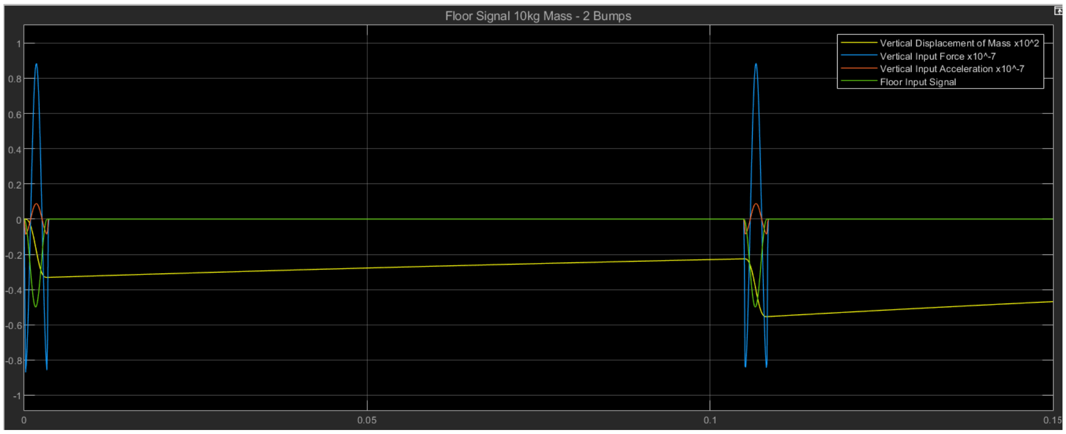

Figure 23: Plot of Vertical Acceleration and Force vs Time for 2 Bumps - 10 kg

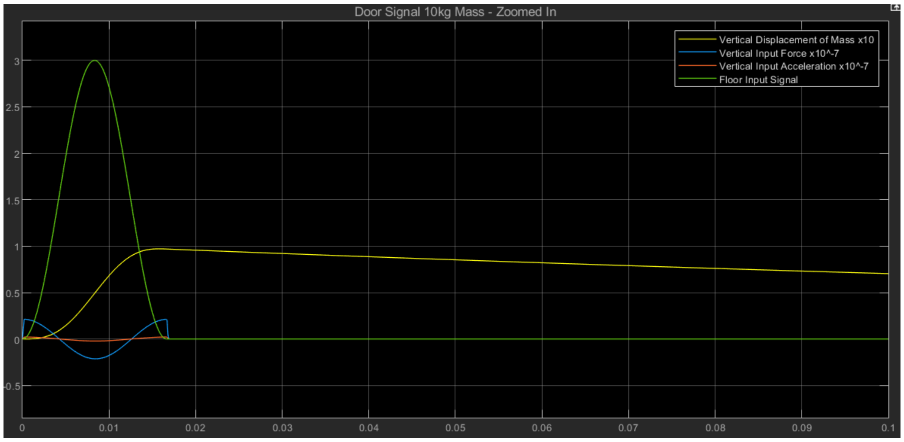

Figure 24: 10kg Cumulative Plot with Door Signal

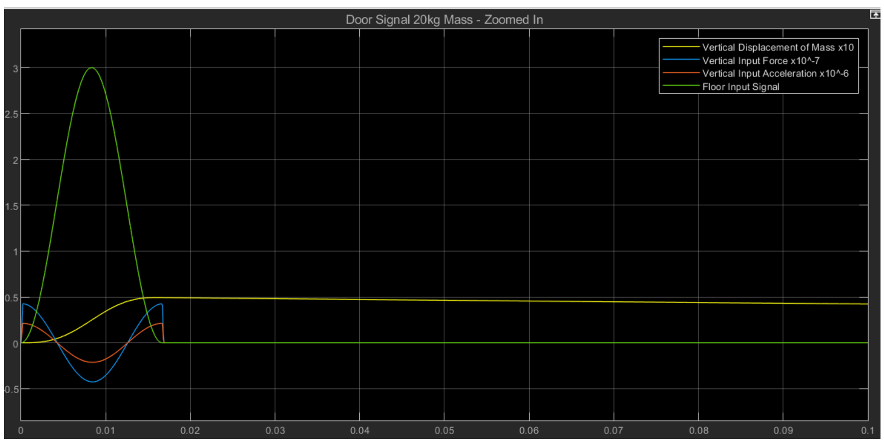

Figure 25: 20kg Cumulative Plot with Door Signal