MXET 375
Applied Dynamic Systems

Multidisciplinary
Engineering Technology
COLLEGE OF ENGINEERING

LABORATORY # 2

Translational & Rotational Systems

Submission Date: 09/19/2024

Name: Kyle Rex
Section: 906
UIN: 932008894

Introduction

The purpose of Lab 2 is to provide general problems that are meant to expand an individual's knowledge on how to model translation and rotational systems using different libraries in Simulink and MATLAB. The objectives include gaining a general spring and damping functionality understanding, learning to create block diagram models using different Simulink libraries, and appropriately configuring values for specific applications. Task 1A focuses on creating a model translational system using Simulink and assigning the system values using MATLAB to model. Task1B focuses on creating a model translational system using Simscape. Task 2 focuses on creating a model rotational system using Simscape. At the end of the lab the individual should have a better understanding of creating, plotting, configuring, and formatting translational and rotational models, and their results, using Simulink, Simscape, and MATLAB.

Procedure & Lab Results

This lab has 2 tasks total. Task 1 has 2 parts, Part A and Part B. Task 2 has 1 part. Each task includes a detailed description of the setup, procedure, results, relevant figures, and discussion focusing on developing a better understanding and interpretation of what the results mean and how they were derived. For this lab specifically it is important to understand and recognize that the models are representative of an actual model. and are not just simple plots. This means that the y-axis and what the lines represent are defined, disregarding Task 1 Part A.

Task 1

Based on the understanding of standard control systems Task 1 focuses on creating and modeling the translational system depicted, in the form of a free body diagram, in Figure 1 using two different methods where the mass (m) is 1 kg, spring constant (k) is 1.3 N/m, and the damping ratio (b) is 1.25 N/(m/s). The first method is done using the Simulink library and toolboxes. The second method is done using the Simscape library and toolboxes. The purpose of this task is not only to show how a single system can be modeled in different ways and still produce the same results but also to expand the understanding of modeling translational systems.

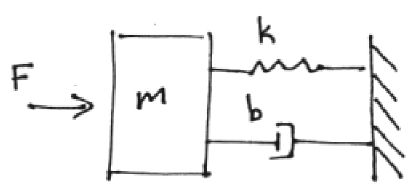

Figure 1: Task 1 System Diagram. [1]

Task 1 Part A

Task 1 Part A provided the task of creating a block diagram of a translational system using common control system elements and the Simulink Library. This system is made up of a step block, a three input (+, -, -) sum block, three gain blocks, two integrator blocks, a scope block, and a bus creator. The provided lab manual for this lab walks through the exact setup, configuration, and procedure for creating this block diagram.

The procedure for this began with importing all the required blocks and setting them up in the correct setup configuration. The incomplete setup configuration of the required block diagram to produce the correct plot, with all block names, can be seen in Figure 2. With the first part of the setup complete the next step was to configure the model with the appropriate values. In the case of this problem that is done by setting the original sum block from having two positive inputs to having one positive input and two negative inputs. Also, the final step value, which is representing the applied force (F), must be changed to 2. Then the gain blocks from top to bottom of the diagram respectively should be set to the variables 1/m, b, and k. Once this is complete the variables must be configured with their respective values as previously described. This can be done in the Simulink interface itself or with the MATLAB command window. In this case it was done using the MATLAB command window as seen in Figure 3 where m is set to 1, k is set to 1.3, and b is set to 1.25. With these values set the setup can be completed by adding a scope and a bus creator that is configured to have 5 inputs. Then connecting those inputs to their respective branches as seen in Figure 4 which shows the complete setup configuration of the required block diagram to produce the correct plot, with all block names. Once that is complete the final step is to run the system and format the result to the specified requirements described in Lab 1. This result can be seen in Figure 5.

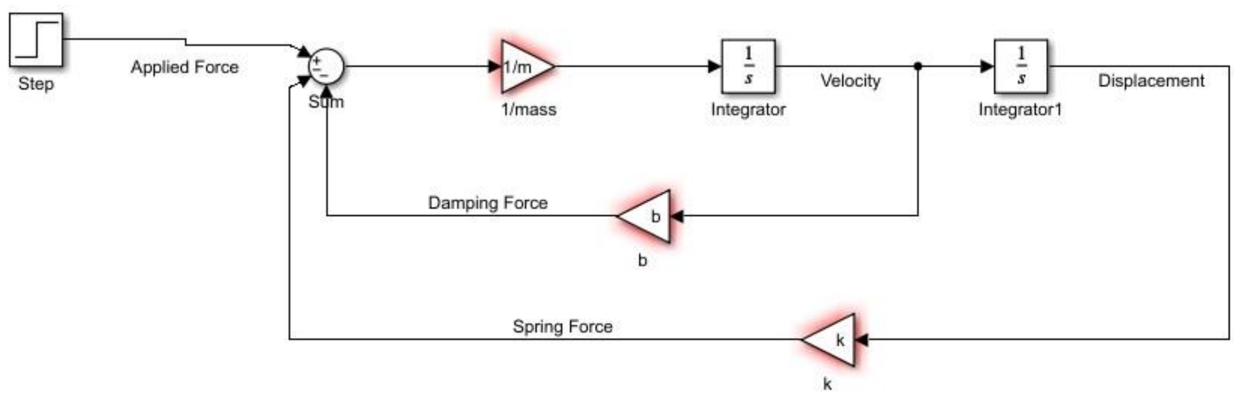

Figure 2: Task 1 Part A Incomplete Block Diagram.

Figure 3: Task 1 Part A MATLAB Variable Assignments.

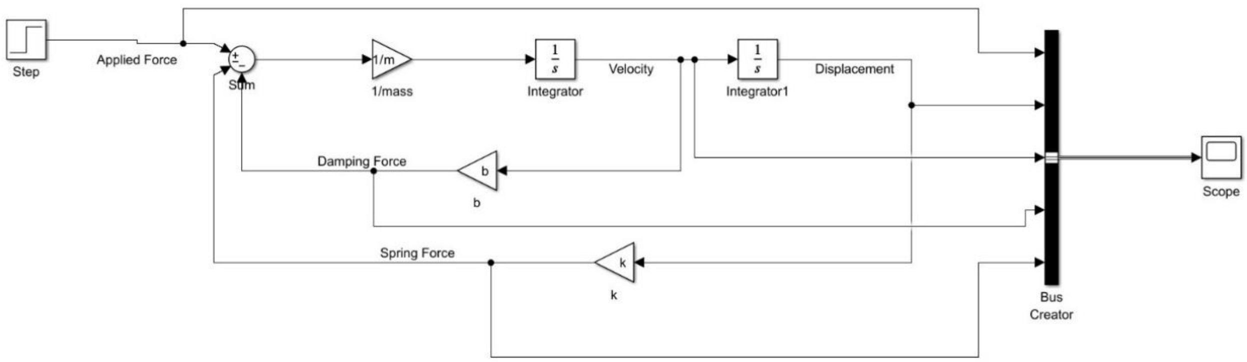

Figure 4: Task 1 Part A Complete Block Diagram.

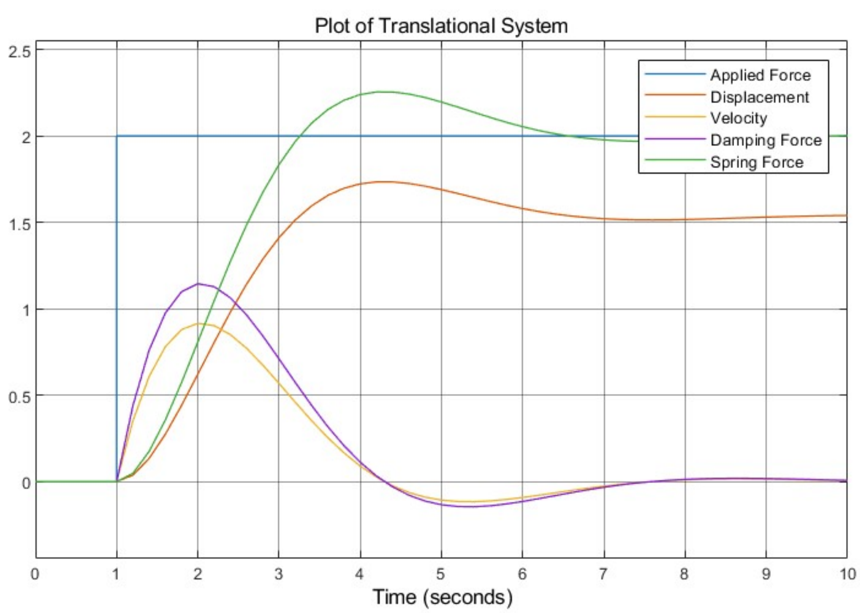

Figure 5: Task 1 Part A Translational System Plot.

Based on the resulting block diagram and plot, it can be inferred that the damping force and velocity have some type of correlation or relationship as they have a similar shape. The same can be said about the spring force and displacement. This makes sense when considering that damping force is equal to the velocity multiplied by the damping ratio of 1.25 N/(m/s) and the spring force is equal to the displacement multiplied by the spring constant of 1.3 N/m. Based on this it is clear that the damping force is directly proportional to the velocity and the spring force is directly proportional to the displacement. Apart from that it can be inferred that the object does not return to its original position given that the displacement is not 0. Analyzing the functions themselves it can be seen that due to the applied force the system's displacement and velocity initially accelerate. Then after overshooting its equilibrium position, it begins to oscillate while the damping force begins to dissipate energy before finally reaching equilibrium when the velocity is zero, spring force is equal to applied force, and the displacement has stabilized. Lastly, based on the graph, and previous conclusions, it can be inferred that this is a damped oscillatory response that enters a steady state upon reaching its final position.

Task 1 Part B

Task 1 Part B provided the task of creating a block diagram of a translational system using common control system elements and the Simscape Library. This system is made up of a step block, two mechanical translational reference blocks, a simulink-PS converter block, a mass block, an ideal force source block, two PS-simulink converter blocks, an ideal translational motion sensor block, a solver configuration block, a translational spring block, a translational damper block, and two scope blocks. The provided lab manual for this lab walks through the exact setup, configuration, and procedure for creating this block diagram.

The procedure for this began with importing all the required blocks and setting them up in the correct setup configuration. The setup configuration of the required block diagram to produce the correct plots, with all block names, can be seen in Figure 6. With the setup complete the next step was to configure each block with the appropriate values. In the case of this problem that is done by setting the final step value to 2 and assigning values of 1 kg, 1.3 N/m, and 1.25 N/(m/s) to the mass block, translational spring block, and translational damper block respectively ensuring that the units of the values are correctly entered. Lastly, in the configuration parameters window of Simulink on the solver tab the solver must be changed to "ode23t (mod.stiff/Trapesoidal)". Once that is complete the final step is to run the system and format the results to the specified requirements described in Lab 1. These results can be seen in Figure 7 and Figure 8.

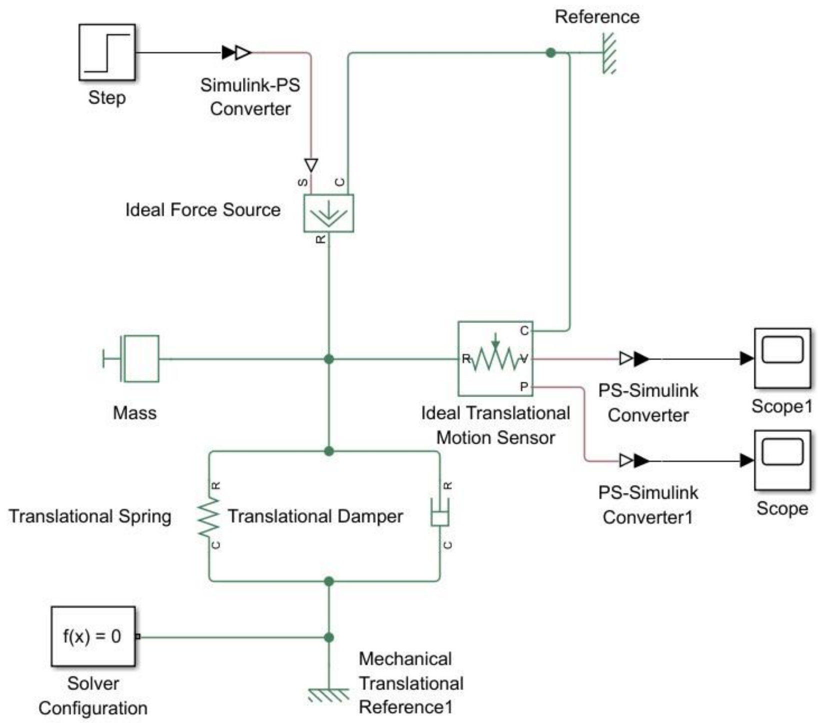

Figure 6: Task 1 Part B Complete Block Diagram.

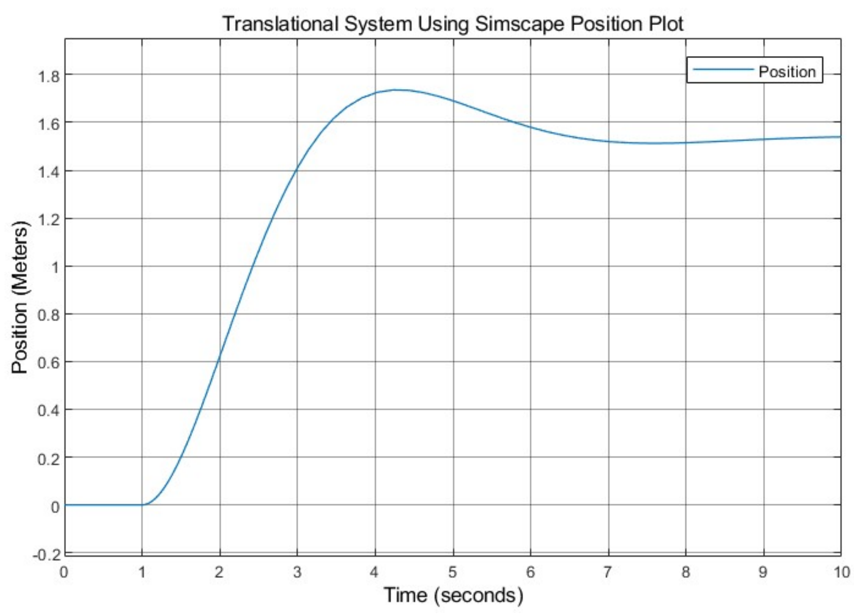

Figure 7: Task 1 Part B Translational System Position Plot.

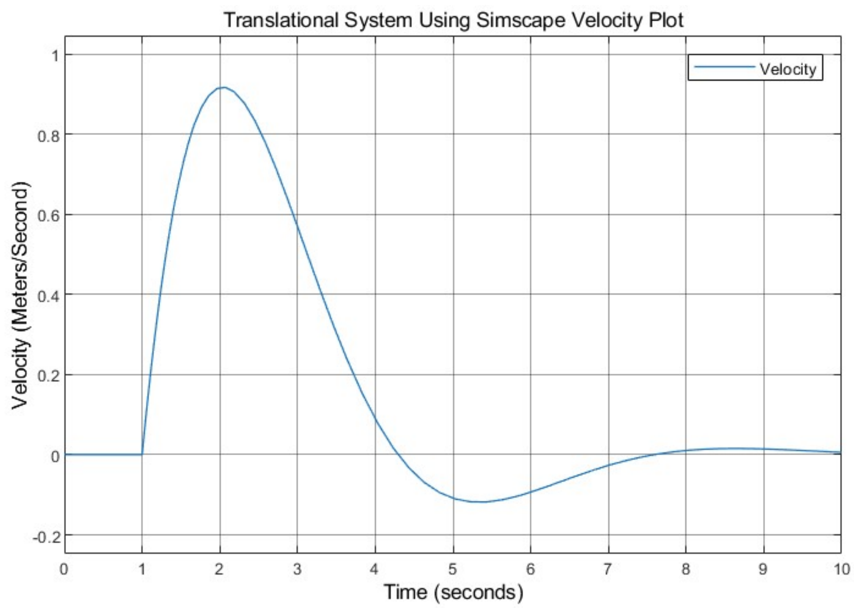

Figure 8: Task 1 Part B Translational System Velocity Plot.

Based on the resulting block diagram and plots, it can be inferred that this system is close to the exact same, if not the exact same, displacement and velocity plot as seen in Figure 5. This proves that there is more than one way to accurately and correctly model systems like these. Many of the conclusions regarding the analysis remain the same with these plots as in the first part as they are the same plots.

Task 2

With the completion of Task 1, which provided a general understanding of how to model translational systems using Simulink and Simscape, the focus of Task 2 shifts to modeling rotational systems using Simscape. Based on the understanding of standard control systems this task focuses on creating and modeling the rotational system depicted, in the form of a free body diagram, in Figure 9 where the moment of inertia (J) is 1795 g*cm^2, spring rate constant (k) is 0.01 N*m/rad, and the damping ratio coefficient (b) is 0.001 N*m/(rad/s). The purpose of this task is to expand the understanding of modeling rotational systems.

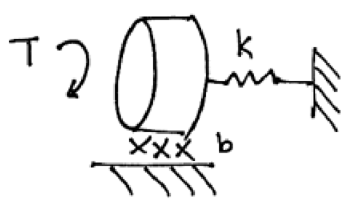

Figure 9: Task 2 System Diagram. [1]

This task provided the assignment of creating a block diagram of a rotational system using common control system elements and the Simscape Library. This system is made up of a step block, two mechanical rotational reference blocks, a simulink-PS converter block, an inertia block, an ideal torque source block, two PS-simulink converter blocks, an ideal rotational motion sensor block, a solver configuration block, a rotational spring block, a rotational damper block, and two scope blocks. The provided lab manual for this lab walks through the exact setup, configuration, and procedure for creating this block diagram.

The procedure for this began with importing all the required blocks and setting them up in the correct setup configuration. The setup configuration of the required block diagram to produce the correct plots, with all block names, can be seen in Figure 10. With the setup complete the next step was to configure each block with the appropriate values. In the case of this problem that is done by setting the final step value, the input torque, to 0.01 N*m and assigning values of 1795 g*cm^2, 0.01 N*m/rad, and 0.001 N*m/(rad/s) to the inertia block, rotational spring block, and rotational damper block respectively ensuring that the units of the values are correctly entered. Lastly, in the configuration parameters window of Simulink on the solver tab the solver must be changed to “ode23t (mod.stiff/Trapesoidal)”. Once that is complete the final step is to run the system and format the results to the specified requirements described in Lab 1. These results can be seen in Figure 11 and Figure 12.

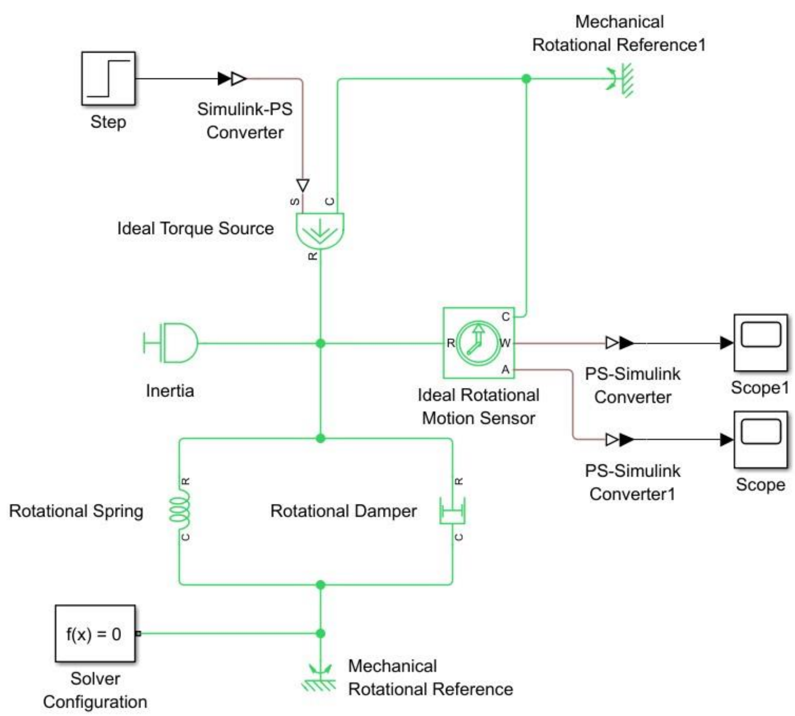

Figure 10: Task 2 Complete Block Diagram.

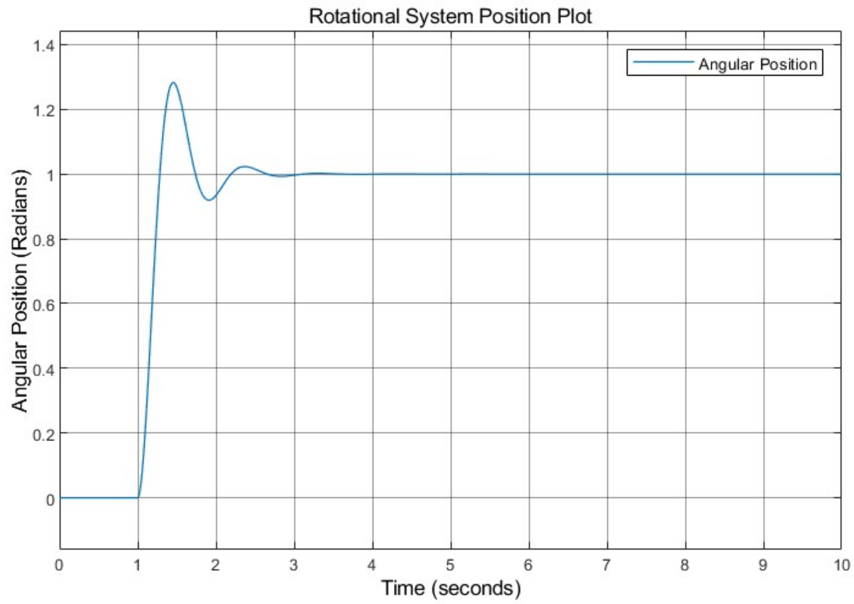

Figure 11: Task 2 Rotational System Position Plot.

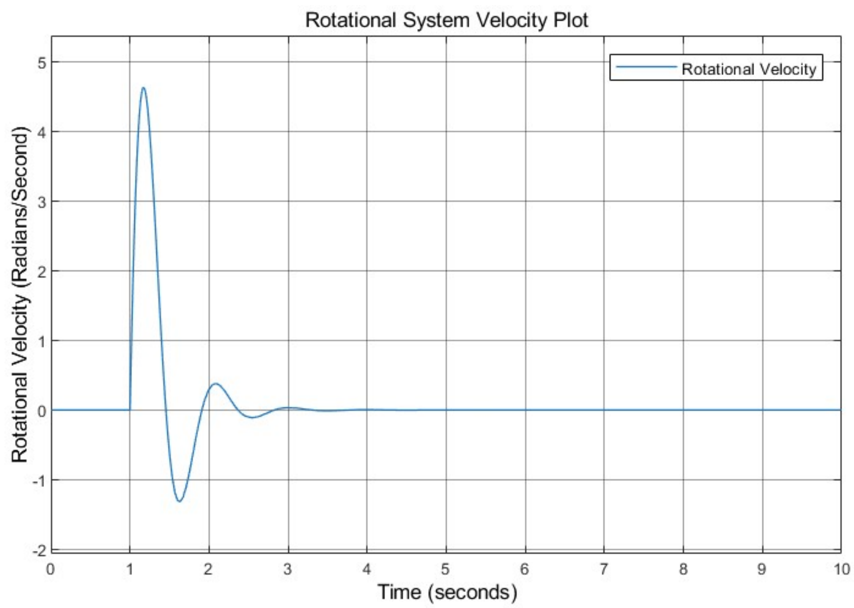

Figure 12: Task 2 Rotational System Velocity Plot.

Based on the resulting block diagram and plots, it can be inferred that the object does not return to its original position given that the displacement is not 0 but instead 1 Radians. Analyzing the functions themselves it can be seen that due to the applied force the system's displacement and velocity initially accelerate. Then after overshooting its equilibrium position, it begins to oscillate while the damping force begins to dissipate energy before finally reaching equilibrium when the velocity is zero and the displacement has stabilized. Lastly, based on the graph, and previous conclusions, it can be inferred that this is a damped oscillatory response that enters a steady state upon reaching its final position.

Post-Lab Questions

From the 5 signals plot from task 1a:
1. Why do spring force and displacement have similar curves? Why do velocity and damping force have similar curves?

Based on Hooke's Law, which states that the force of a spring is directly proportional to its displacement, where the spring force is equal to the spring constant multiplied by the displacement, it can be concluded that the spring and displacement curves are similar because they have a linear relationship. Like Hooke's Law the force of damping is directly proportional to its velocity, where the damping force is equal to the damping coefficient multiplied by the velocity, it can be concluded that the damping and velocity curves are similar because they have a linear relationship.

2. At the end of the simulation, why does the velocity return to zero but the displacement does not return to zero?

The velocity returns to zero at the end of the simulation because of the loss of all its energy from the damping force. Whereas the displacement does not return to zero at the end of the simulation because of damping preventing the system from returning to its original position.

3. How can you manipulate the damping force to increase the number of oscillations in velocity? Verify by trial.

To increase the number of oscillations in velocity the damping force must be reduced.

From the velocity plot from task 1b:
4. About how many seconds does it take to reach peak velocity? Increase the mass from 1kg 5kg. Now, about how many seconds does it take to reach peak velocity?

It takes around 2 seconds to reach peak velocity with a mass of 1kg for the system. Increasing the mass to 5kg resulted in the time changing to around 3.65 seconds for the system. So, the time it took to reach peak velocity increased.

From the position plot from task 1b (with original 1kg mass):
5. What is the final resting displacement (approximate to 1 decimal place)? What are the units?

The final resting displacement is 1.5 Meters.

6. After 10 seconds (end of the simulation) there is still a tiny movement, so your approximation in the previous question is not exact. Calculate the exact final resting displacement to 4 decimal places. Hint: when the system comes to rest, the equation to describe it is just a balance of the spring force against the input force. Show your calculation including units for each term.

Based on the provided information regarding the system it can be concluded that the spring constant (k) is equal to 1.3 N/m and the input force (F) is equal to 2.0 N. Using Hooke's law and reconfiguring it to find the displacement (x) the final resting position can be found. The equation for this will look like x = 2.0 N / 1.3 N/m = 1.5385 m. So, the final resting displacement to 4 decimal places is 1.5385 meters.

Conclusion

In conclusion, not only did this lab focus on developing a better understanding of creating, plotting, configuring, and formatting translational and rotational system models, and their resulting plots, using Simulink, Simscape, and MATLAB but it also provided the opportunity to begin to learn how to analyze these plots. This analysis helped develop a better understanding of how these systems work, specifically on how they are affected by springs, dampers, and other forces. This can be seen holistically in Task 1 and the post lab questions where the focus is on analyzing the plot depicting the applied force, displacement, velocity, damping force, and spring force of the translational system 1kg mass from 0 to 10 seconds. All the tasks in this lab focused on how to create the correct system configuration for desired designs, how to properly use & configure a diverse range of blocks to produce the correct results for those designs, and how to accurately depict relevant results of said system designs. The final conclusion from this lab is the understanding that MATLAB and Simulink are useful simulation tools for solving many practical engineering problems and furthermore developing the skills required to properly utilize them is extremely important.

References

[1] Author not listed, *Texas A&M University MXET375 - Lab 02 - Translational & Rotational Systems*. College Station, TX, USA: Date not listed.
[2] Rex K., *MXET375 - Kyle Rex - Lab Report 1*. College Station, TX, USA: 09/05/2024.