Modeling and Simulation of a Vibrating Box

Alton Coolidge, Camilo Cuartas-Lopera, Kenta Terasaki, Mihir Vemuri

November 2024

Time Approximation

Alton Coolidge: 8 hours
Camilo Cuartas-Lopera: 10 hours
Kenta Terasaki: 10 hours
Mihir Vemuri: 11 hours

1 Introduction

In this project, we focused on modeling and simulating the dynamics of a vibrating box attached to multiple springs, incorporating both linear and nonlinear characteristics. Our approach involved linearization, modal analysis, and a comparison between linear and nonlinear models. The following sections document our methodology, results, and reflections.

2 Methodology

2.1 Spring Behavior

We start by looking at the motion of a single spring through the function *compute_spring_force*, which involves the two parameters of a spring: its stiffness, k, and its natural length, l_0.

The main three components that determine the output of our *compute_spring_force* function are:

- The current length of the spring (l).
- A unit vector pointing from Point A to Point B (e_s).
- The force exerted by the spring at Point B (F).

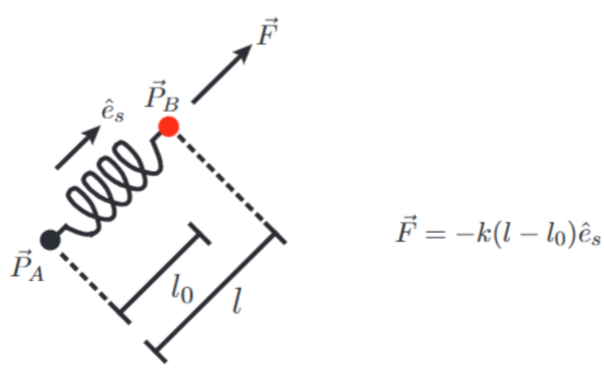

Figure 1: *This image was provided to us to model spring behavior, highlighting key variables.*

2.2 Box Kinematics

Next, we compute the forces exerted by each of the springs in our system. These forces depend on the orientation of the box, (x, y, θ), and the positions of the spring endpoints, $\vec{P}_A$ and $\vec{P}_B$. Later in this assignment, we use a total of four springs positioned around the box. Each spring is attached to:

- A static point in the environment (a known position).
- A corner of the box, whose position depends on the box's orientation.

To compute these forces, we use a rigid body transformation (via our *compute_rbt* function), which relates the box-frame and world-frame coordinates. This transformation allows us to account for both translations and rotations of the box, enabling us to map all relevant points of the box and spring endpoints effectively.

2.3 Linear/Angular Acceleration of Box

With the information regarding the positions and forces acting on the box and springs, we can determine the linear and angular acceleration of the box for *any* position and orientation. This will be useful later when we need to plot the motion of our box.

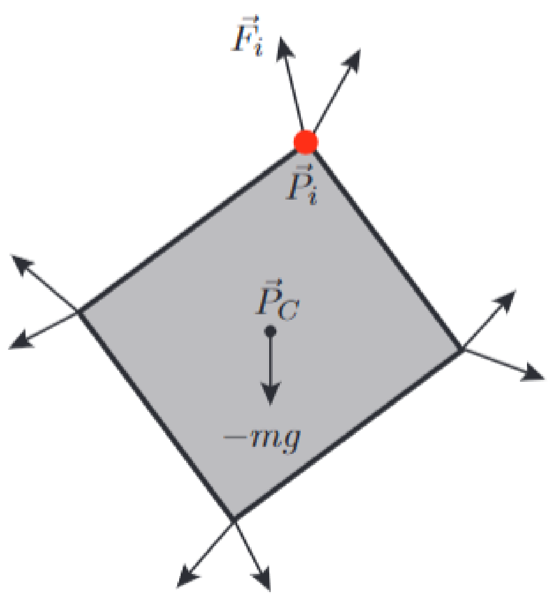

Figure 2: Free-body diagram of the system in the world frame. The key components include the center of mass, the force of gravity, and the outward forces of the springs (spring forces) at the mounting points.

By manipulating the force equations from the free-body diagram and applying Newton's Second Law, we can compute the box's accelerations.

Using the *compute_accel* function, we input the box's position (x, y), orientation (θ), and some unique properties of the box (e.g., structure, mass, moment of inertia). The function outputs the linear and angular accelerations of the box for any given position and orientation.

2.4 Numerical Simulation of Nonlinear System

The primary goal of this assignment and the visual animation is to bring our system to an equilibrium state. To achieve this, we need to identify when the system's acceleration becomes zero. We accomplish this using numerical integrators from previous assignments, after converting the necessary values into a format compatible with these integrators.

We begin by creating a vector, V, which represents the x-position, y-position, θ, and their respective derivatives (velocities), organized into a 6×1 matrix:

$$V = \begin{bmatrix} x \\ y \\ \theta \\ \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}$$

Taking the time derivative of this initial vector yields a new vector, , which is the same size as V but contains the derivatives of each term. This means represents the system's accelerations:

$$\dot{V} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{bmatrix}$$

With this setup, our numerical integrators can take time (t) and V as inputs and compute . This process is encapsulated in the function *box_rate_func*, which allows us to simulate the dynamics of our box system effectively.

3 Results

3.1 Simulation Comparisons

Small Initial Displacement Plots:

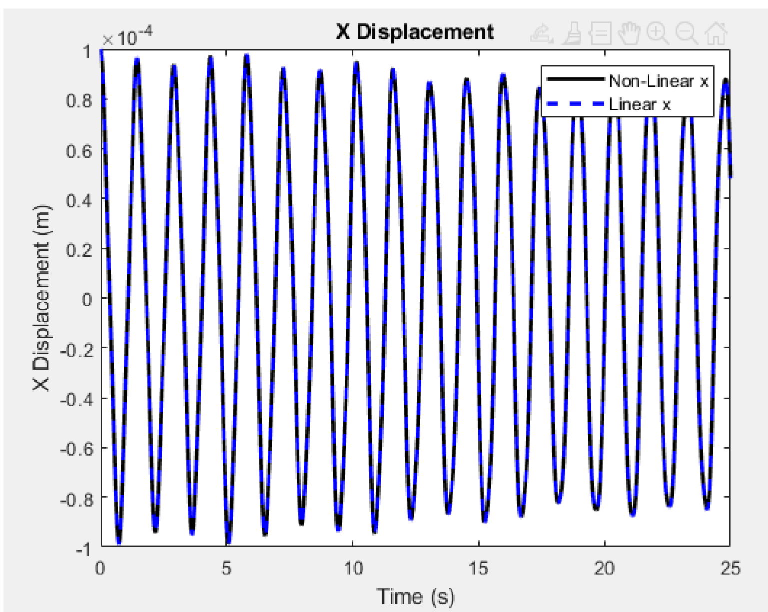

Figure 3: X displacement.

X displacement shows how far the box has moved from it's equilibirum position. All of the displacement plots operate close to equilibrium. There is no divergence in X-displacement graph, which shows the initial displacement is in a range where our linearized model is valid.

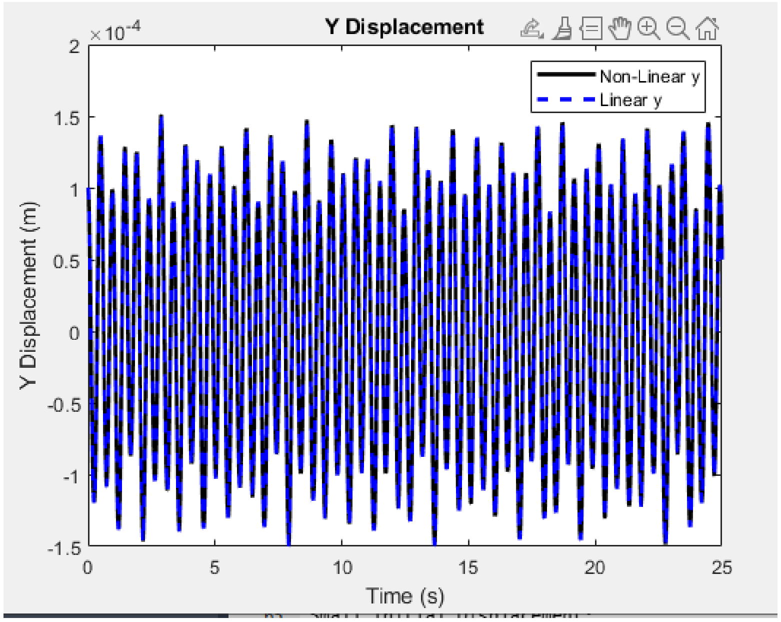

Figure 4: Y displacement.

Y displacement shows the interaction between the weight of the box and the forces that restore the spring back to natural form. Unlike the x-displacement plot, y displacement amplitude fluctuates over time. This could mean there's a coupling effect between oscillations in the different degrees of freedom, or the effects of the nonlinear spring forces.

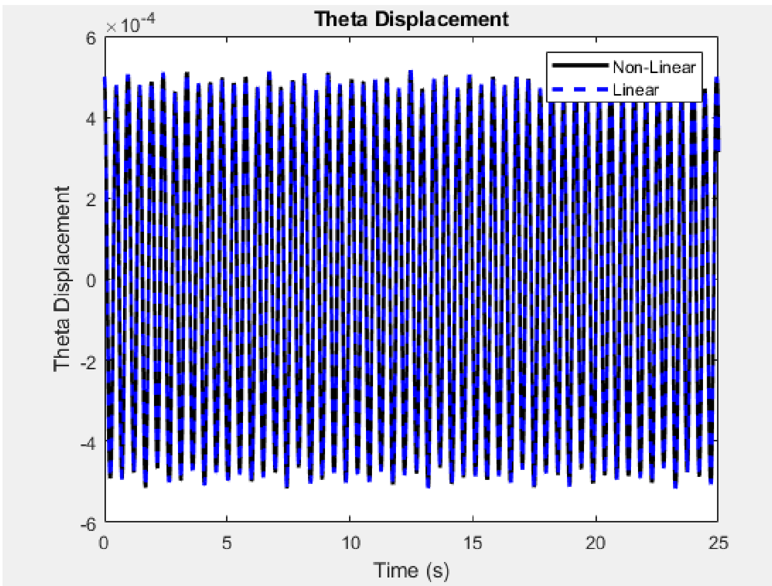

Figure 5: Theta displacement.

Theta displacement represents how much the box has rotated from its equilibrium orientation. There is a slight variation in the amplitudes on this graph, which might be because of the couples behavior of the x and y translational motions.

Large Initial Displacement Plots:

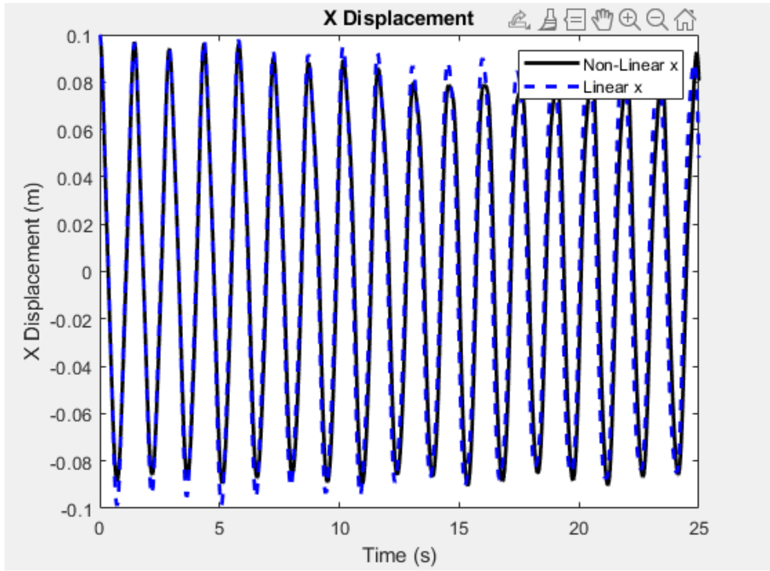

Figure 6: X displacement.

This graph shows X after a large inital perturbation. The oscillation amplitude of the nonlinear system reduces over time, which could be due to a more dampling. The linear model doesn't account for this loss, which is why the plots don't match up.

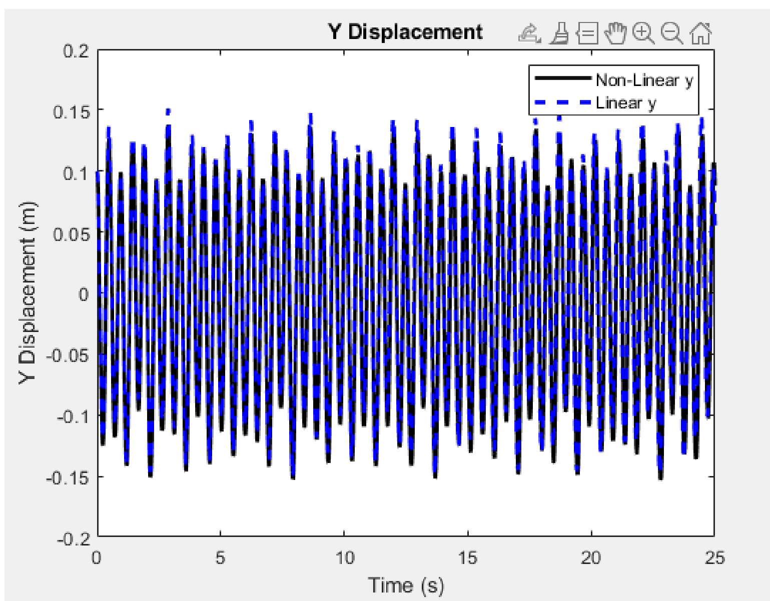

Figure 7: Y displacement.

Just like the x displacement graph, the y displacement graph diverges over time.

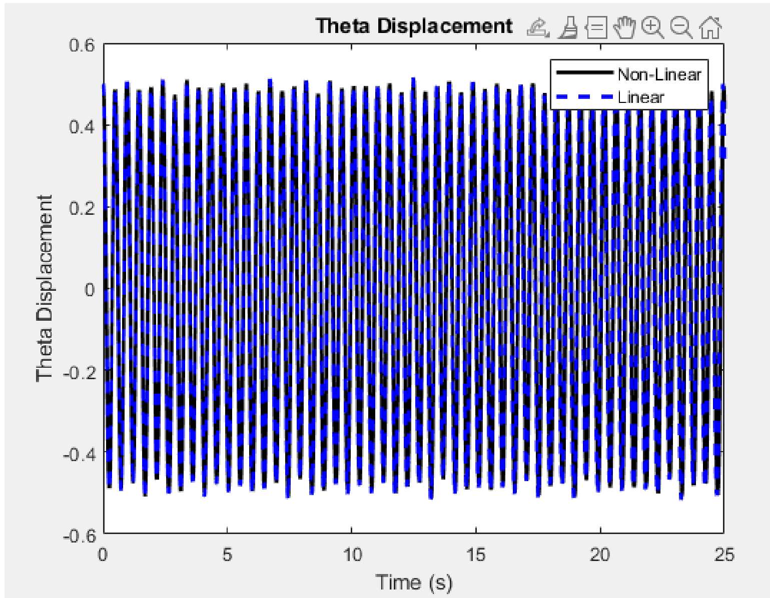

Figure 8: Theta displacement.

It looks like the increased inital perturbation mainly affects amplitude after looking at all three plots, and not oscillatory frequency. The nonlinear reduction with the theta displace ment is way more gradual, which means that any rotational damping the system might have isn't as strong as the translational damping.

3.2 Mode Shape Analysis

- Generate plots for each resonant frequency, comparing the nonlinear system to the linear solution of $U_0 \cos(\omega_n t)$. Small Initial Displacement:

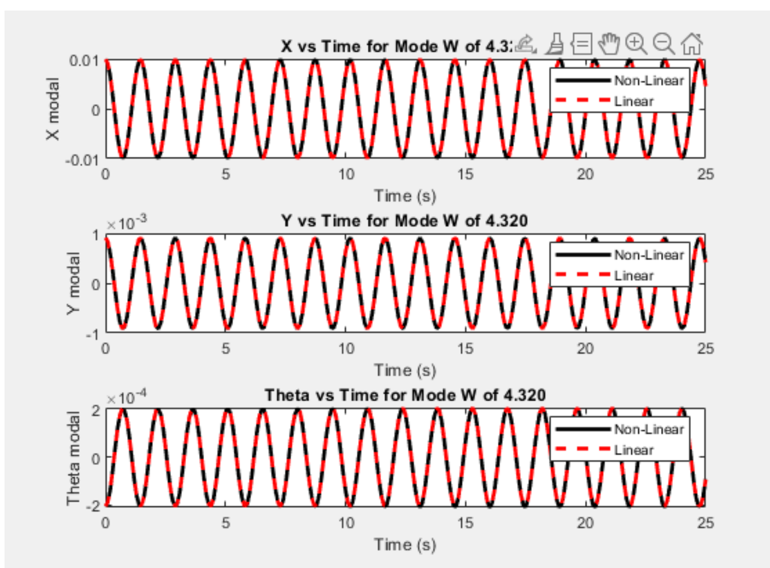

Figure 9: X, Y and theta Modal for mode = 4.32.

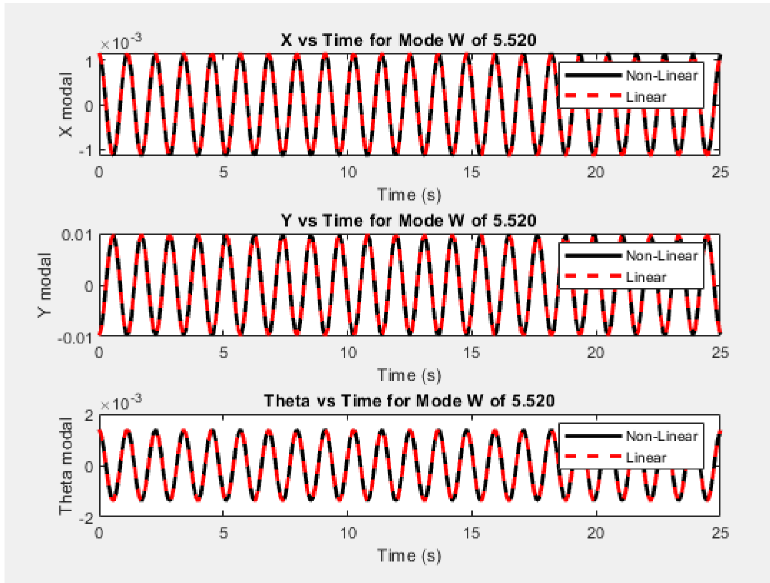

Figure 10: X, Y and theta Modal for mode = 5.52.

These plots show the mode shapes at X, Y and Theta for 2 different natural frequencies within the system. Since oscillations maintain their amplitude, energy dissipation will most likely be minimal

4 Reflection

4.1 Process Reflection

- If we started over, we would coordinate more time besides dedicated class time to work on the project together. This would benefit us by allowing us to be all on the same page regarding any questions we have about the report, or if we all have the same understanding of our processes detailed in the project, for example. We could, in the future, go as a group to office hours to work together there, so it's less of one person waiting for the other to finish their respective section.
- What we would keep the same is, especially considering a code-intensive project, dividing and conquering different sections of the work. Splitting the assignment up into different days, plots/code, and explanations helped us take accountability to complete a different part of the report to be well ahead of the deadline!
- What we would do differently is better coordinate time to work as a team to 1) allow for opportunities for us to ask each other questions as a team to deepen everyone's understanding.

4.2 Lessons Learned

- Three things we learned from this project:
1. Stay ahead of deadlines to not stress hours before the project deadline!
2. Understanding the purpose of each function. We ran into debugging errors when we didn't realize the function was supposed to take in position for calculating acceleration, so instead we put the input as velocity.
3. Being present and engaged while in class is our most productive working time, since we are all in one place and also have the teaching team to assist us. Taking advantage of this time is key in getting ahead of the daily assignments that are part of the larger project.

5 Deliverables

5.1 Video Submission

- Video showing the nonlinear system vibrating at a single frequency: [https://www.youtube.com/watch?v=dbWjrv4dnFw&t=1s]

5.2 Code and Additional Materials

- Link to code: [https://drive.google.com/drive/folders/1AMcSPVBj6yo5s6WNiUvnC8TlAk8HE6DV?usp=sharing]