Code Repository: `https://github.com/alexbeattie42/Comp-Methods-Exercise-5`

1. Multibody Simuation Solver This solution begins with the sample code from the course git repository for creating a multi-body system simulations, and builds upon it to provide values for the position or displacement, displacement derivative (velocity) and θ and $\dot{\theta}$ angles. This solution utilizes guidance from the recorded lectures and materials as well as prior experience from the 2 multi-body simulation courses in the mechatronic system design masters program. This is a very interesting problem because it involves manually implementing a solution for the solver system that Matlab and Simulink implement automatically for Simscape multi-body. The accompanying code can be found in the github repository above as well as the solution for the first portion of this assignment (1 of 2).

   The constraint equations are provided as follows:

$$\begin{bmatrix} a * cos(\phi) + b * cos(\theta) - d \\ a * sin(\phi) - b * sin(\theta) \end{bmatrix} \tag{1}$$

   The partial derivatives are computed manually to form the Jacobian matrix as shown below

$$\begin{bmatrix} b * -sin(\theta) & -1 \\ -b * cos(\theta) & 0 \end{bmatrix} \tag{2}$$

   The Jacobian matrix represents all possible permutations of the partial derivatives of specified variables. Since the partial derivatives do not contain ϕ which is the only variant reliant on time, the first derivative can be computed by directly plugging in the solved θ value into the left hand terms of the Jacobian matrix. The values for θ and displacement are calculated using the Newton-Raphson method and their derivatives are calculated using the method described above for each time step.

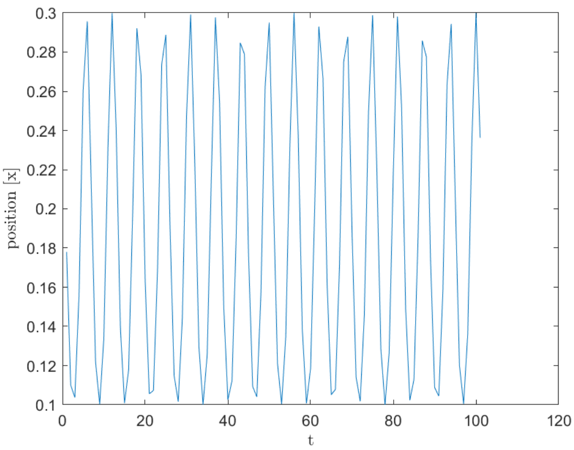

Figure 1: position vs. time

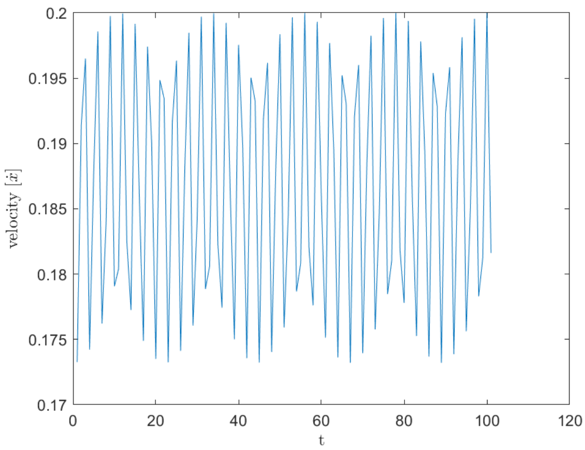

Figure 2: velocity vs. time

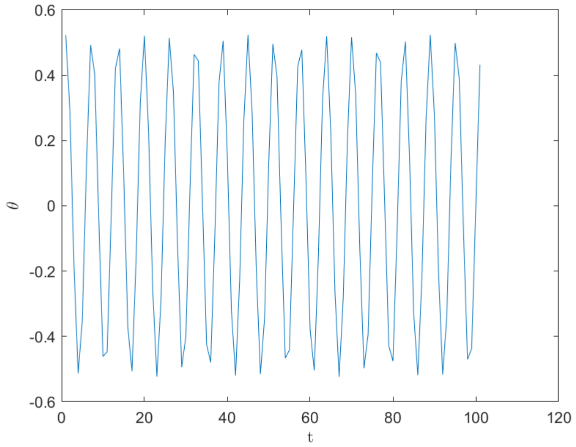

Figure 3: θ vs. time

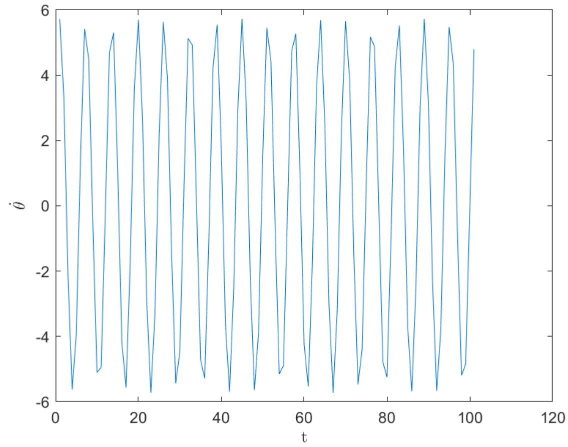

Figure 4: $\dot{\theta}$ vs. time

Figure 1 illustrates the position of the oscillating system over time and figure 2 represents its velocity (time derivative. Figure 3 represents the angle of the system and figure 4 represents its time derivative. When examining the velocity of the system, an overall larger sinusoidal pattern emerges on top of the sinusoidal pattern of the displacement which is not immediately evident by viewing the displacement. When comparing the angle θ and its time derivative the sinusoidal pattern remains consistent but the amplitude is increased in the time derivative.

Creating a solver from scratch in this method is interesting to implement because it sheds insight into the preliminary working and algorithms of simulation tools (like Simscape) without requiring intense knowledge and study into the optimizations and computational methods that power modern solvers. This could be used widely in multi body system dynamics value and time derivative calculation.