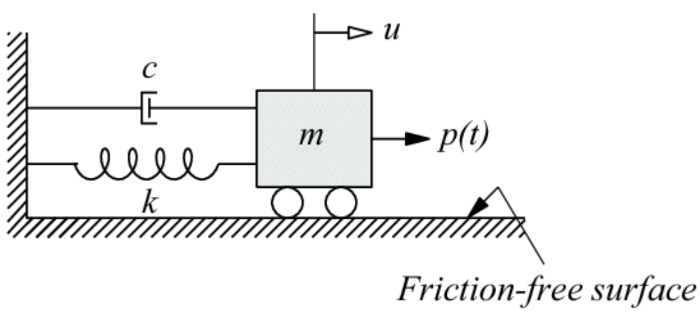
Andrew D'Onofrio
October 29th, 2023
Cornell DEBUT
Phase 2B Documentation

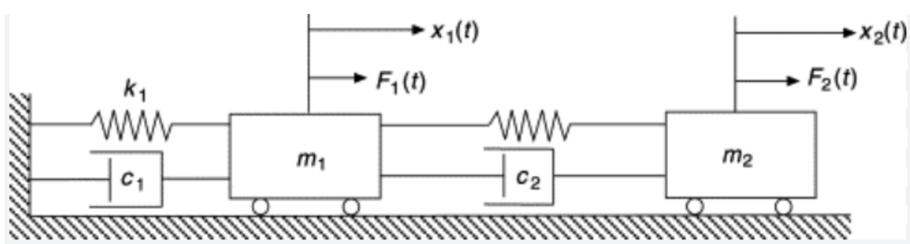
Considering the majority of the semester I've focused on finishing the mathematical model for the tuned-mass damper system, I thought it'd be effective to provide clear documentation for the MATLAB script in place of the deliverable I am most proud of. Especially since there may still be confusion about the necessity of the MATLAB script (MATLAB model) in comparison to the single-degree-of-freedom model developed (Chalkboard model).

Originally, I thought it'd be effective to create a 3D plot comparing the residual of the cane's motion concerning a range of spring and mass values for the tuned-mass damper. By analyzing this plot, we would be able to determine the most effective parameters for the cane. To do this, I wanted to solve for a singular motion equation that had inputs for the parameters of the whole cane (system 1) and the tuned-mass damper system by itself (system 2). I first developed the chalkboard model to solve this problem.

When developing the chalkboard model, I interpreted that the oscillation of the cane equally transferred to the oscillation of the tuned-mass damper, and by critically damping the tuned-mass damper, we would critically damp the excitation of the cane. This is presented in the following model:



Although intuitive, the above is an incorrect interpretation of the actual principles dictating the motion of an object with a tuned mass damper. Altogether, because the mass between the cane and the tuned-mass damper oscillate independently of one another (i.e. both move with different frequencies due to different mass ratios), and the driving force purely acts on system 1, the chalkboard model does not provide enough degree-of-freedom for the design. Instead of a single DOF system, a two DOF system is necessary to properly describe the relationship between the position of system 1 and system 2 during oscillation. The following is a diagram of a two-degree-of-freedom system:



**Important Note: Typically, TMDs include a damper that increases the rate of damping of the system. With the omission of damping in our design, there are additional changes in the Purdue equations that omitted damping coefficient (c)*

However, describing this trend is incredibly complex by hand, requiring a 4th-order differential equation (as seen in the Purdue document*) that is incredibly difficult to solve and graph by hand. This is specifically where the necessity of MATLAB comes in. Despite the complexity of the problem, the software is able to do the difficult 4th-order differential calculations and plot the motion results despite parameter changes.

The MATLAB script (currently) computes the motion of the two systems independently of one other, in addition to the implementation of the trend-mass damper with the oscillating cane. Variables "tspan" and "F0" are initialized to set the range of motion computed and the amplitude of the driving force wave. Although there is no spring for the cane, since the cane has an oscillation, we treat the first system as a mass connected to a spring with a driving force. As such, to set the natural frequency (omega_1) of the hand tremor, set up the equation w = sqrt(k1/m1) with a known natural frequency value, solving for k1 and m1. With this, the oscillating driving force (P) that acts as the tremor on the cane is set with the equation P = F0 * sin(wt + phi). With the parameters set, MATLAB computes the simpler single-degree-of-freedom calculations for the individual systems' motion independently, followed by a two-degree-of-freedom calculation for the two systems' motion together. In terms of the differential equation part, long story short, the script can solve the Purdue equations by the following: creating four "state variables" that represent the position and velocity of the first system, in addition to the position and velocity of the second system, creating four "state equations" that essentially represent the relation between the differential formulas and the "state variables," and finally solving everything to produce the motion graph.

**Important Note: Typically, TMDs include a damper that increases the rate of damping of the system. With the omission of damping in our design, there are additional changes in the Purdue equations that omitted damping coefficient (c)*