Dangerous Toy Findings

Matthew Johnson

College of Engineering

University of Oklahoma

Norman, OK

*Abstract*—This document is to show that a toy manufacturer’s toy did not oscillate at .5 Hz, thereby causing a family cat to run out in front of a CART bus.

# Introduction

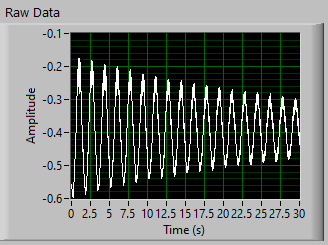
Studies by the American Feline Medical Association show that when cats are subjected to oscillations of half a hertz (+/- 10%) that they tend to flip out and head for wheels of CART buses. To show that the manufacturer of a toy being blamed for the loss of a prominent OU family’s cat was not at fault, data was collected via an accelerometer to find the frequency of the toy’s oscillation. As documentation for usage of the accelerometer was nonexistent, data that was received in had to be corrected and mathematically manipulated to find the frequency for said toy.

# Collection of Data

## Accelerometer Setup

First, the accelerometer was attached to the toy and the accelerometer data lines were connected to a data acquisition device to collect the voltage readings from the accelerometer.

## Data Collection

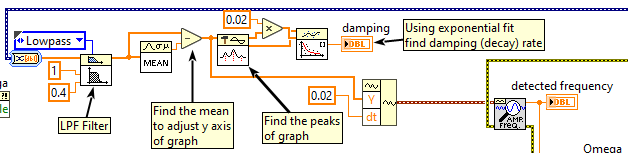


A LabVIEW program was then created to take collected data in for analysis. The data was sampled at 50 Hz, 10 samples at a time via the DAQ Assistant VI, and 1500 samples were collected via the Collector VI. Using the Write to Measurement File VI, raw data was outputted to a measurement file for backup and results verification purposes.

# Preperation of Data

After data has been collected it must first be multiplied by 0.0254 to convert inches to meters, then it must be filtered to remove any noise that was inherent during the collection process. The data was passed through the Butterworth Filter VI lowpass filter with a low cutoff frequency of .4 Hz.

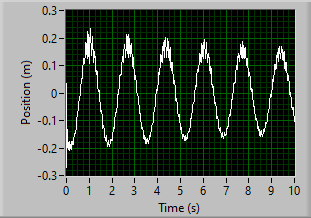
To center the data at zero the mean of the data was found by using the Mean VI and then subtracted from all values of the data.



With the data centered it was passed into the Build Waveform VI with a Δt of .02. The resulting wave was passed into the Extract Single Tone vi and the resulting frequency was outputted:

The resulting frequency was then multiplied by 2π to get the resulting value of omega:

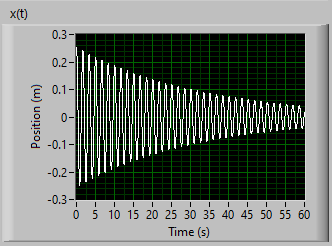
We can also do graphical analysis to come to the same conclusion for frequency. If we take a 10 second window of data, seeing that the wave repeats 6 times, we can find the period. Then with the period we can divide it from one to come up with an approximate frequency:



On a split path from after we had centered up the data, we also found the damping rate. This was found by using the Peak Detector VI to find peak locations and amplitudes. The outputted sample locations were then multiplied by the rate of time for samples, .02 seconds (50 Hz), to adjust damping rate for time. At this point peaks and locations (in time) were fed into the Exponential Fit VI, then the outputted damping rate was multiplied by -1 to correct for time scale and is:

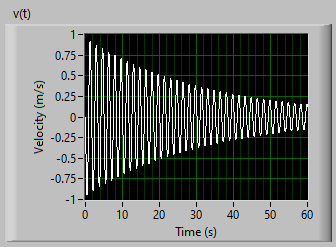
# Calculating Values

To calculate the resulting x(t), v(t), and a(t) graphs we use the user inputted distance or amplitude of toy pulled on spring, the damping rate alpha, and the value of omega into the Mathscript Node VI to perform calculations. The first value we want to find is position of the toy over time. Using the formula:



(1)

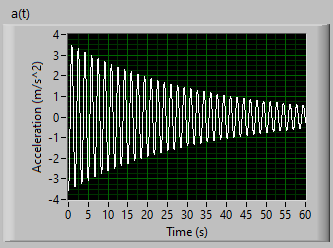
By multiplying the amplitude *(A)* by the exponential decay *(α)* of the toy’s oscillation over time (*t* ), multiplied by a cosine wave of omega *(ω)* multiplied by time (*t* ), an x(t) is calculated and graphed over a sixty second interval of the toys position in meters as it oscillates back and forth.



To then find the toy’s velocity over the same time period we take the first derivation of x(t) which gives us the following formula and graph:

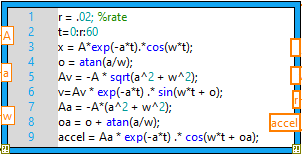
(2)

We find the acceleration of the toy over time by taking the first derivative of v(t) or the 2nd derivative of x(t) giving us the following formula and graph:



(3)

The processing of x(t), a(t), and v(t) was all completed in the Mathscript Node VI by inputting the variables omega *(ω)*, amplitude *(A)*, and alpha *(α)* to output x(t), v(t), and a(t).



# Display of Formulas As Strings

To display the formulas in the user interface with calculated values, the amplitude, damping, and omega were inputted into the Stringbuilder VI, a powerful sub VI that does lots of concatenations and multiplying of values to output string indicators to display the formulas for x(t), v(t), and a(t) along with the frequency *()* and the damping *(α)*.

# Conclusions

Therefore as a result of finding that the toy oscillates at 0.596 Hz it could not have caused the cat that short circuits at +/- ½ Hz to run out into the wheels of the oncoming cart bus.

# Appendix

