Levi Schulze

Final Project

Wave Graph Report

5/2/2021

Topic was approved by Laurel Hilger.

The idea for this project came from a physics class that I am currently in where we are discussing simple harmonic motion. This is when an object is moved from equilibrium and then a restoring force moves it back and forth from that equilibrium point. An example of this would be a mass that is attached to a string or a pendulum. In both cases, the object would oscillate around the equilibrium position. By using frequency, amplitude, and wavelength, a graph of the wave can be constructed. This is the basic idea for the project.

My idea for the project was to create a user interface that contained a graph of the motion of an object. There would be two inputs that are edit boxes. One would be for the wavelength of the desired graph while the other would be the speed of the wave. These two values could then be used to calculate and display the frequency of the wave in a text box using the equation Frequency \* Wavelength = Speed. Additionally, I planned to add a slider to be able to adjust the graph as well as utilize more GUI features. Finally, I planned to change the x-axis scale to better view the graph.

For the completed project I changed my idea for the x-axis scale to change the x-axis variable instead. I came to this conclusion after researching the topic more where I found two separate equations that could be used to graph the displacement vs. distance of the wave as well as the displacement vs. time interval of the wave. The equation for distance is Amplitude \* cos( (2pi / wavelength)\*x). Alternatively, the equation of the graph vs. time interval is Amplitude \* cos(2pi \* frequency \* x). This change gave the project even more use and added complexity to the code to challenge myself more.

I felt that this was a good way to utilize the GUI functionality of Matlab since it would utilize the graphing capabilities of matlab as well as using the edit, text, slider, and radio button GUI elements. I started off the project by creating a plot for the graph and adding the button group and the two radio buttons in the group. Next, I added the slider element with a callback function to change the amplitude value and plot the data. Additionally, I added the edit boxes into the main function and created callback functions for them. In the callback functions I needed to convert the imputed string to a number. Then I check to see if the other data box has a value in it. If it does, it will graph the data. This is so it does not try to graph the data before all of the required data is entered. Additionally, I added another if statement to make sure that the value entered is a number and send back an error message if it is not. Finally, I created a plot function to plot all of the data. I used an if statement to determine which radio button is selected and to create the correct graph accordingly.

I did run into difficulties when I was trying to update the text box to display the calculated frequency. At first, I was trying to make a separate callback function that would change the value of the text box but this did not work as I expected since the callback would never execute. After trying different ways to do it, I moved the code to change the text box inside of the plot function. This ended up working well and the code will display the correct frequency everytime the graph is plotted.

Through this project, I have learned different ways to utilize GUI in Matlab. I could see myself using this functionality often when writing code that I end up using very often to make it easier and more efficient to use. It would also be useful in the future when writing code that people without programing knowledge will be using.