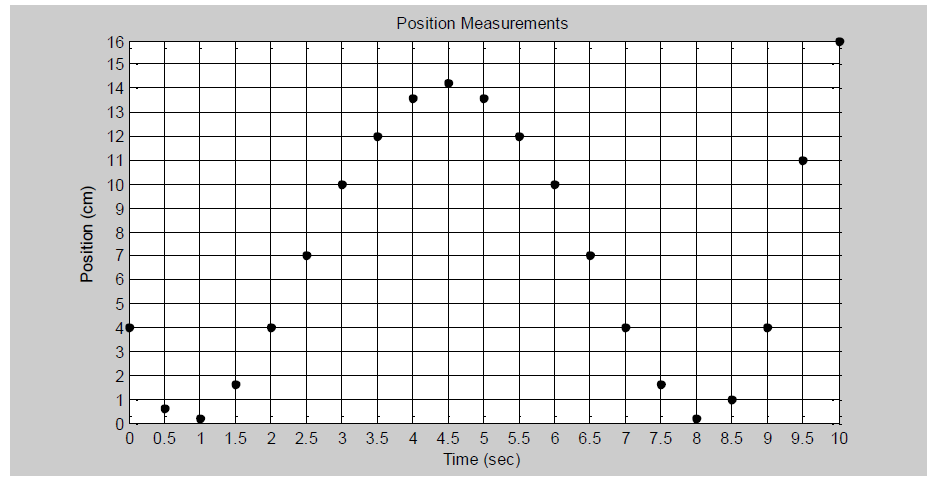
**ENED 1091 HW#3**

**Due Week of February 22nd**

**Problem 1:**  The graph below shows position measurements (in cm) collected every 0.5 seconds over a 10 second interval of time.



1. Using ***linear interpolation***, estimate the position of the object at 6.15 seconds. Do this by hand (no ***interp1***) and show your calculations.

**Calculation and Result (include units):**

slope = (10 – 7)/(6 – 6.5) = -6

y = 10 – 6(x – 6)

y(6.15) = 10 - 6(6.15 - 6) = 9.1 cm

1. Now use ***interp1*** and linear interpolation to estimate the position of the object at 6.1, 6.2, 6.3, and 6.4 seconds.

**MATLAB Command and Results (include units):**

x =

6.0000 sec 6.5000 sec

>> y = [10 7]

y =

10 cm 7 cm

>> interp\_points = 6.1:.1:6.4

interp\_points =

6.1000 sec 6.2000 sec 6.3000 sec 6.4000 sec

>> interp = interp1(x,y,interp\_points,'linear')

interp =

9.4000 cm 8.8000 cm 8.2000 cm 7.6000 cm

**Problem 2:** For this problem, you need the data file, HW3P2.mat, posted on the Blackboard metasite with Homework #3. The file has a vector of times, t, which starts at 0 increments by 0.04 and ends at 0.8 seconds. It also has a vector of voltage measurements, V (volts), corresponding to the given times. Download the data file and save it in your current MATLAB folder. The command: load HW3P2 will load the two vectors into your workspace.

1. Plot the original data points (don’t connect the points with lines). Add title and axis labels (with units).

**Plot:**

****

1. Use ***interp1*** with a method of ***nearest*** to estimate the voltage every 0.005 seconds between 0 and 0.8 seconds. On a single plot, plot the original data points as red stars and the interpolated data points as black circles. Add title, axis labels (with units), and a legend.

**MATLAB Commands and Plot**

interp\_points = 0:.005:.8;

interp\_near = interp1(t,V,interp\_points,'nearest');

figure(1)

plot(t,V,'r\*',interp\_points,interp\_near,'ko');

xlabel('Time (sec)');

ylabel('Voltage (V)');

title('Voltage vs. Time Scatter');

legend('Original Data','Interpolated (Nearest) Data');

****

1. Repeat part (b) using ***linear*** interpolation.

**MATLAB Commands and Plot**

interp\_points = 0:.005:.8;

interp\_linear = interp1(t,V,interp\_points,'linear');

figure(1)

plot(t,V,'r\*',interp\_points,interp\_linear,'ko');

xlabel('Time (sec)');

ylabel('Voltage (V)');

title('Voltage vs. Time Scatter');

legend('Original Data','Interpolated (Linear) Data');

****

1. Repeat part (b) using ***spline*** interpolation.

**MATLAB Commands and Plot**

interp\_points = 0:.005:.8;

interp\_spline = interp1(t,V,interp\_points,'spline');

figure(1)

plot(t,V,'r\*',interp\_points,interp\_spline,'ko');

xlabel('Time (sec)');

ylabel('Voltage (V)');

title('Voltage vs. Time Scatter');

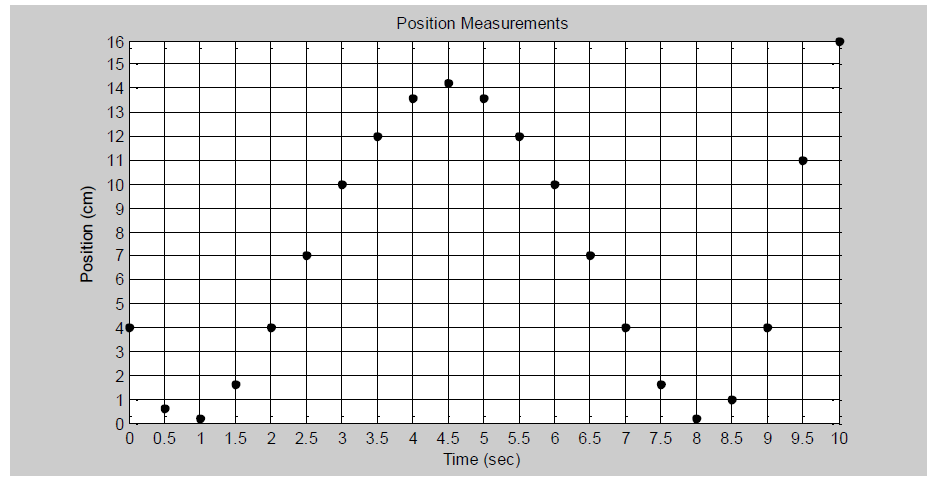
legend('Original Data','Interpolated (Spline) Data');

****

1. What kind of waveform does your plot in part (d) look like? Could you possibly have picked this up from looking at the original data points?

It looks like a sinusoid which is noticeable from the original points due to their periodic nature.

**Problem 3:** The graph below shows position measurements (in cm) collected every 0.5 seconds over a 10 second interval of time.

****

1. Using a Δt = 0.5 sec, estimate the velocity at t =  sec using the 2-point estimate and the 3-point estimate for the derivative. **Be sure to show your work and include units!**

(f(6 + .5) – f(6))/.5 = (7 – 10)/.5 = -6 cm/sec

2-point Estimate of Velocity at t = 6 sec: -6 cm/sec

(f(6 + .5) – f(6 - .5))/1 = 7 – 12 = -5 cm/sec

3-point Estimate of Velocity at t = 6 sec: -5 cm/sec

1. Using the estimate for 2nd derivative and a Δt = 0.5 sec, estimate the acceleration at t = 6 sec. **Again, show work and include units!**

(f(6 + .5) – 2\*f(6) + f(6 - .5))/.52 = (7 – 20 + 12)/.25 = -4 cm/sec2

Estimate of Acceleration at t = 6 sec: -4 cm/sec2

1. What could be changed to improve the accuracy of the derivative estimates?

The accuracy could be improved by having a smaller Δt (closer points).

**Problem 4:** For this problem, you need the data file, HW3P4.mat, posted on the Blackboard metasite with Homework #3. The MATLAB command: load HW3P4 will load the data into the MATLAB workspace. The data file has 4 vectors:

* t is a vector of times (seconds) starting at 0, incrementing by 0.1, and ending at 4
* pos is a vector of position measurements (mm) taken at the times specified in vector t
* t\_act is a vector of times (seconds) starting at 0, incrementing by 0.005, and ending at 4
* v\_act is a vector of velocity measurements (mm/s) corresponding to the times in t\_act

1. Plot the position measurements over time (don’t connect the individual data points with lines). Add a title and axis labels (with units) to the plot.

**Plot:**

****

1. Use the position measurements and the 2-pt reverse estimate for derivative to estimate the velocity at each time in vector t. Assume the initial velocity is 0.

**MATLAB Code for 2-PT Estimate of Velocity:**

v\_est = zeros(1,41);

for k = 2:41

v\_est(k) = (pos(k)-pos(k-1))/(t(k)-t(k-1));

end

1. On a single graph, plot the velocity estimates over time, t, as individual data points and the actual velocity (v\_act) over t\_act. Add a title, axis labels (with units), and a legend.

**Plot and MATLAB Code for Plotting:**

****

figure(2)

plot(t,v\_est,'ko',t\_act,v\_act,'r-');

xlabel('Time (sec)');

ylabel('Velocity (mm/sec)');

title('Velocity vs. Time');

legend('Estimated Velocity','Actual Velocity');

1. How could the accuracy of the velocity estimates be improved?

If there were more position data points such that they were closer together then the accuracy would be improved.