Paul Buttles

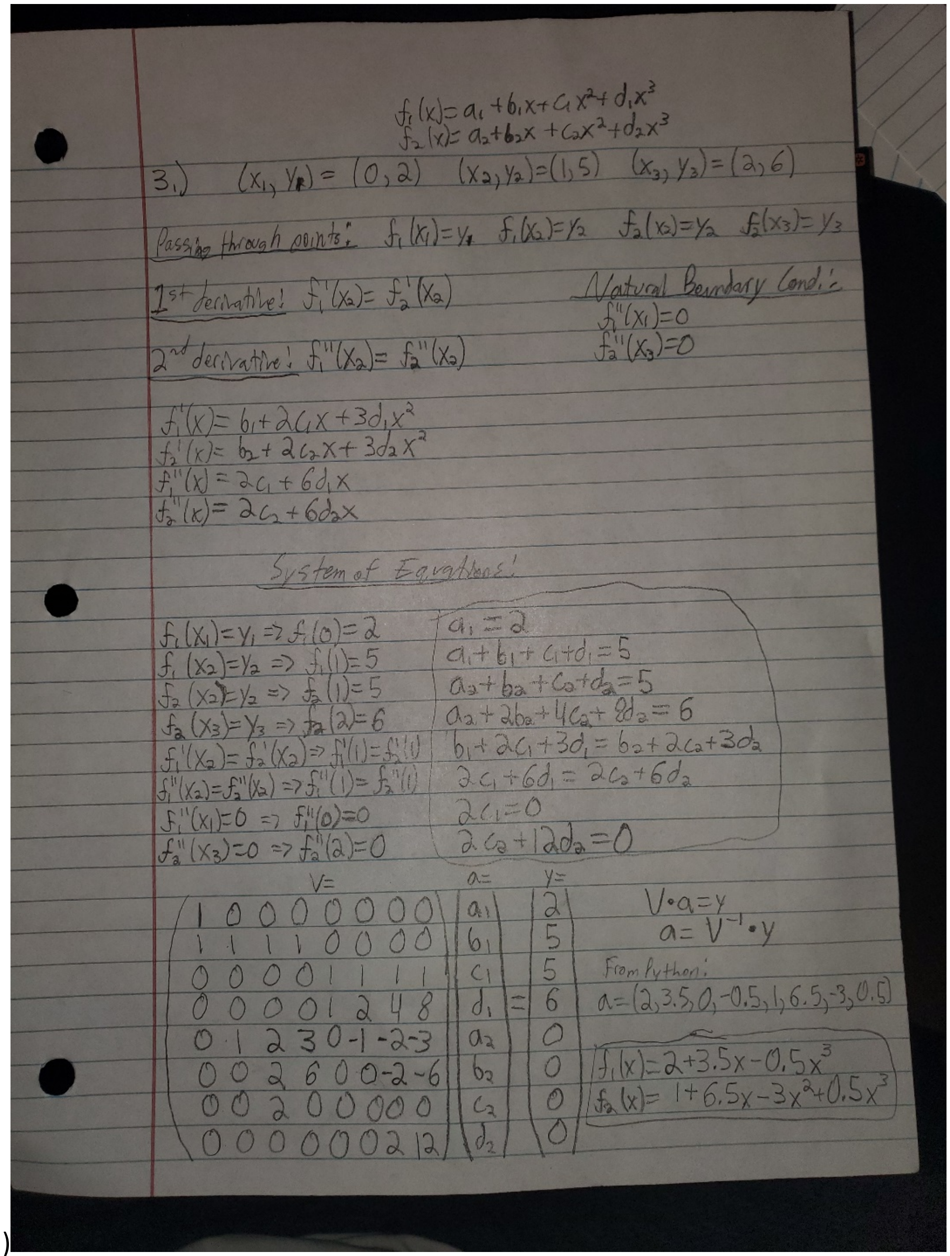
11/13/2020

PHYS 331, HW11

1.) The plots of F(t) make intuitive sense, as the force delivered over a short time (tau = 1s) has a very narrow peak (high quality factor), corresponding to the abrupt force, whereas the force delivered over a longer time (tau = 10s) has a wider peak (lower quality factor) as it is delivered over a longer period of time. Fourier transforming this yields plot 2, where the Fourier transform of the tau = 1s curve has a lower amplitude but continues over a longer range of frequencies, whereas the tau = 10s curve is quite high but quickly approaches zero. The tau = 3s curve has a decently high amplitude and also decays somewhat quickly to zero, but not quite as quickly as tau = 10s. The resonance frequency, sqrt(k/m), was initially calculated to be 1.46 rad/s. Dividing by 2pi converts this into 0.23 Hz, which is the resonant frequency as it is seen in the frequency domain of the Fourier Transforms. The FFT of tau = 1s still has a significant magnitude at 0.23 Hz, where as tau = 10s is very near 0 at this frequency and tau = 3s is approaching zero and has some slight magnitude at this frequency. All of these then translate into X(w). For tau = 10s, X(w) looks very similar to F(w), peaking at a high magnitude at w=0 and quickly decaying to 0. For tau = 1s, however, it looks different. It appears similar to the plot in F(w), except once it reaches the resonant frequency, it peaks substantially before decaying back down. Similarly, for tau = 3s, it slopes down at first like in F(w), and once it hits the resonant frequency, it has a small little bump, proportional to the small but non-zero magnitude of F(w) at the resonant frequency. In short, the magnitude of driving force frequency F(w) at the resonant frequency corresponds to the size of the peak of the motion frequency X(w) at this same resonant frequency. Finally, these motion frequencies translate into motion through time. For tau = 10s, it never reached the resonant frequency, so there is no resonant motion. Instead, there is simply motion around t=0 as the force is gradually delivered, and it tapers off to zero with no oscillation. For tau = 1s, the quickly delivered force allows the motion to reach its resonant frequency, translating to strongly oscillatory motion that gradually decays. For tau = 3s, it begins by decaying gradually like tau = 10s, but resolves to some small oscillations as it tapers off. This corresponds to the small peak in X(w) as its F(w) had a small but not insignificant driving force frequency magnitude at the resonant frequency.

2.) First, the explicit solution. For values of h=1, 0.1, and 0.01, the explicit solution ceases to function well. Rather than fit roughly alongside the exact solution, it blows up on the right tail, reaching values on the order of 10^12, 10^79, and 10^307 respectively, and thus is not a valid model. At h=0.001, it begins to function properly, and although it lacks the dip inward on the left-hand side that is observed in the exact solution, it otherwise follows alongside the exact solution for the rest of the interval. At h=0.0001 and 0.00001, it begins to conform to the dip inwards on the left-hand side and matches the exact solution closely for the entire interval. Secondly, the implicit solution. For all values of h chosen, from 1 to 0.00001, the implicit solution followed closely alongside the exact solution. While the exact solution itself was quite rough and not usable for h=1, the implicit still mimics the roughly plotted exact solution and doesn't blow up. For h=0.1, both the exact solution and implicit solution begin to take shape, and the implicit solution follows closely along, with only a sliver of visible difference seen in the dip that takes place just right of the center of the graph. As the left-hand dip becomes evident in the exact solution at h=0.01, the implicit solution does not quite match, but catches up to approximately model this dip at h=0.001. At values of h=0.001 and beyond, the implicit solution fits perfectly. It can then be seen that the implicit solution works at all h, while the explicit solution fails at h=1, 0.1, and 0.01, and that the implicit solution more consistently matches the exact solution and can model the solution well without increasingly fine values for h (and hence is more computationally efficient).

3 and 3EC.)



$f_1(x) = a_1 + b_1 x + c_1 x^2 + d_1 x^3$
$f_2(x) = a_2 + b_2 x + c_2 x^2 + d_2 x^3$

3.) $(x_1, y_1) = (0, 2)$ $(x_2, y_2) = (1, 5)$ $(x_3, y_3) = (2, 6)$

Passing through points: $f_1(x_1) = y_1$ $f_1(x_2) = y_2$ $f_2(x_2) = y_2$ $f_2(x_3) = y_3$

1st derivative: $f_1'(x_2) = f_2'(x_2)$

2nd derivative: $f_1''(x_2) = f_2''(x_2)$

Natural Boundary Cond.:
$f_1''(x_1) = 0$
$f_2''(x_3) = 0$

$f_1'(x) = b_1 + 2c_1 x + 3d_1 x^2$
$f_2'(x) = b_2 + 2c_2 x + 3d_2 x^2$
$f_1''(x) = 2c_1 + 6d_1 x$
$f_2''(x) = 2c_2 + 6d_2 x$

System of Equations:

$f_1(x_1) = y_1 \Rightarrow f_1(0) = 2$	$a_1 = 2$
$f_1(x_2) = y_2 \Rightarrow f_1(1) = 5$	$a_1 + b_1 + c_1 + d_1 = 5$
$f_2(x_2) = y_2 \Rightarrow f_2(1) = 5$	$a_2 + b_2 + c_2 + d_2 = 5$
$f_2(x_3) = y_3 \Rightarrow f_2(2) = 6$	$a_2 + 2b_2 + 4c_2 + 8d_2 = 6$
$f_1'(x_2) = f_2'(x_2) \Rightarrow f_1'(1) = f_2'(1)$	$b_1 + 2c_1 + 3d_1 = b_2 + 2c_2 + 3d_2$
$f_1''(x_2) = f_2''(x_2) \Rightarrow f_1''(1) = f_2''(1)$	$2c_1 + 6d_1 = 2c_2 + 6d_2$
$f_1''(x_1) = 0 \Rightarrow f_1''(0) = 0$	$2c_1 = 0$
$f_2''(x_3) = 0 \Rightarrow f_2''(2) = 0$	$2c_2 + 12d_2 = 0$

$$V = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 2 & 4 & 8 \\ 0 & 1 & 2 & 3 & 0 & -1 & -2 & -3 \\ 0 & 0 & 2 & 6 & 0 & 0 & -2 & -6 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 12 \end{pmatrix} \quad a = \begin{pmatrix} a_1 \\ b_1 \\ c_1 \\ d_1 \\ a_2 \\ b_2 \\ c_2 \\ d_2 \end{pmatrix} = y = \begin{pmatrix} 2 \\ 5 \\ 5 \\ 6 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$V \cdot a = y$
$a = V^{-1} \cdot y$

From Python:
$a = (2, 3.5, 0, -0.5, 1, 6.5, -3, 0.5)$

$f_1(x) = 2 + 3.5x - 0.5x^3$
$f_2(x) = 1 + 6.5x - 3x^2 + 0.5x^3$

4a.) This will be particularly difficult wherever sin(x) = 0. Normally, this is where sin(x) changes signs, but because f(x)=|sin(x)|, the function bounces here and forms a cusp. Cusps are not differentiable, so any boundary conditions that require differentiation will fail at this point.

4b.) At x=0, the function hits y=0, but the other lowest support points for each interval get close to 0 without hitting it. This is because the samples are taken over the interval -10 to 10 with step size 0.1. It would continue to hit 0 precisely if this were sampled over an interval fit to pi, such as -2pi to 2pi with step size pi/12. It should hit 0 at npi where n in an integer, but the sampling interval instead plots f(x) at 3.1 and 3.2, straddling 3.1415, or 6.2 and 6.3, straddling 6.2832, etc.

4c.) At this cusp, the linear interpolation fit better than nearest, quadratic, or cubic. The nearest neighbor interpolation produced a step function-like graph that stepped around the true function but didn’t actually align with it. The quadratic and cubic interpolations handled the left and right arms of the function well, but failed around the cusp. Both of these assume the function is differentiable on the interval being extrapolated, but cusps are by definition not differentiable, so it fails here and creates a rounded edge that is not reflected in the true function. The linear interpolation aligns the best as the curving over this interval is quite minimal, and the cusp is sharp, thus straight lines between points fit quite well to the true function at this interval.

4d.) In the previous question, we saw that the linear interpolation handled non-differentiable functions and functions with minimal curving quite well, while cubic interpolation needs differentiability. Linear interpolation does not work well where the function curves greatly between support points, while cubic interpolation is designed to handle this areas well. For this function, cubic interpolation will perform better than linear interpolation at the peak of each function, as this area is differentiable throughout, and requires smooth curving that linear interpolation is not capable of. As seen on the plot produced, the cubic interpolation appears to align nearly perfectly with the true function, while the linear interpolation is quite jagged.