PHY407 Lab-05 Report

PATRICK SANDOVAL[1,2] AND CHIN CHONG LEONG[3,2]

[1] *Q1 - Revisiting Relativistic Spring*
[2] *Q3 - Analysis of Sea Level Pressure*
[3] *Q2 - Audio Filtering*

1. Q1 - REVISITING RELATIVISTIC SPRING

For the pseudo-code and code for this and all is consecutive sub-questions please refer to the python script `Lab05_Q1.py`. The function used to perform the Euler-Cromer methods is in the python script labeled `MyFunctions.py`.

1.1. *Q1a - Simulating a Relativistic Spring*

From our previous work on `Lab03` we know that the equation of motion for a relativistic particle on an ideal spring is the following

$$m\ddot{x} = -kx\left(1 - \frac{\dot{x}^2}{c^2}\right)^{3/2} \tag{1}$$

However, if we want to integrate this system numerically we need to perform the following change of variable $u_1 \equiv x$ and $u_2 \equiv \dot{x}$ such that we can create an equivalent first order ODE system which is simpler to work with.

$$\begin{aligned} \dot{u}_1 &= u_2 \\ \dot{u}_2 &= -\frac{k}{m}u_1\left(1 - \frac{u_2^2}{c^2}\right)^{3/2} \end{aligned} \tag{2}$$

Lastly we recall the critical displacement and the period to amplitude relation describes in `Lab03` where $x_c = c\sqrt{m/k}$ and for relativistic particles where $v \to c$ the period to amplitude relation is linear $T \propto x_0$ and for small amplitudes the period is independent of amplitude. For the purposes of this lab we will simulate three relativistic particles on springs with three different initial displacements $x_0 = 1\,\text{m}$, $x_0 = x_c$, and $x_0 = 10x_c$. In order to obtain figure 1 we

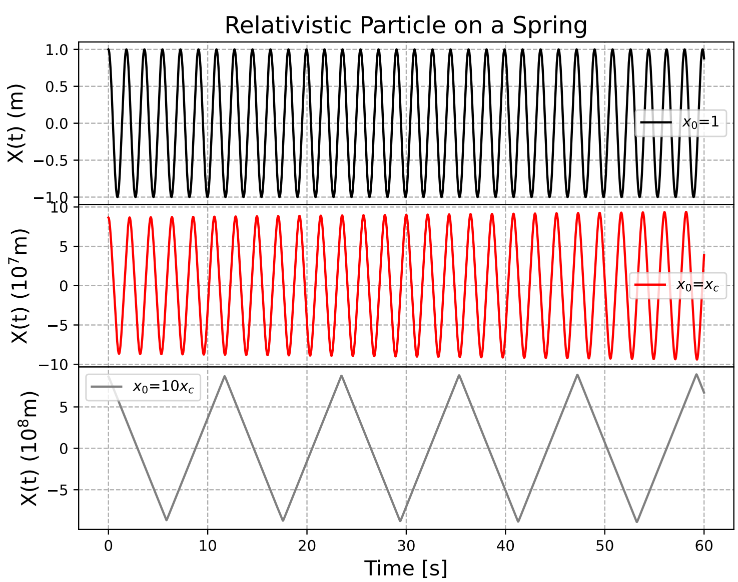

Figure 1: Simulated relativistic particles on a spring for three different initial displacements $x_0 = 1\,\text{m}$, $x_0 = x_c$, and $x_0 = 10x_c$ from top to bottom respectively.

used the Euler-Cromer method on eq.2 with different initial positions to solve for the different cases. We chose a large time domain because this will improve the spectral resolution for our Fourier decomposition later.

1.2. *Q1b - Fourier Transform of Relativistic Spring*

With the time series generated in subsection 1.1 we can take the Fourier transform of the signal and accurately find the oscillating frequency of each simulation. We plot the angular frequency of oscillation against the Fourier coefficients because the lab tells us to but for it is more useful to plot the linear frequency instead of the angular frequency becasue its makes our comparison more intutive. Figure 2 is consistent with our understanding of the relativistic particle on

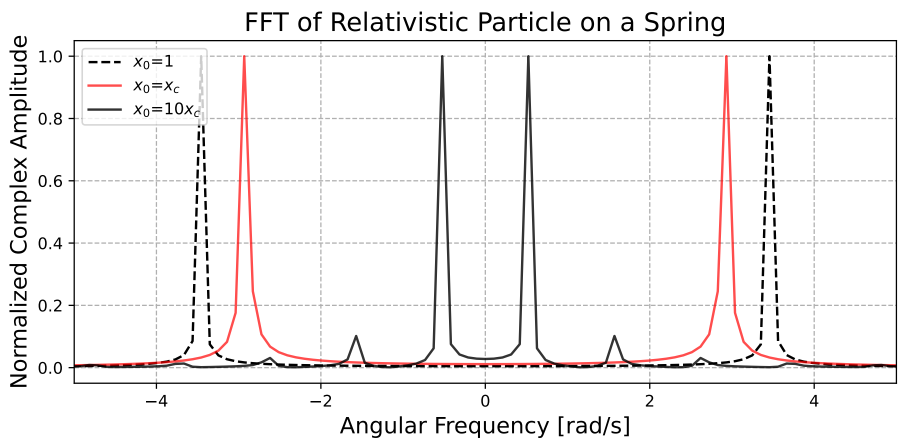

Figure 2: Fourier transform for all three time series of figure 1. The complex amplitudes have been previously normalized for easier comparison.

a spring because we see that larger amplitudes make angular frequency smaller values which implies longer periods. This is exactly the behaviour we saw in Lab03 through the Gaussian quadrature method.

1.3. *Q1c - Comparing Results against Gaussian Quadrature*

Here we continue our comparative study from subsection 1.2 and plot the corresponding angular frequencies predicted by the Gaussian quadrature method on the same axis as figure 2. From figure 3 We see that both methods are consistent

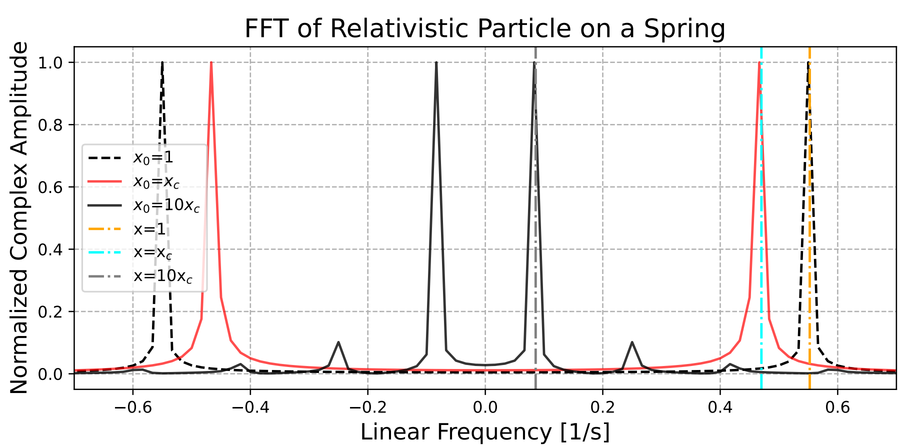

Figure 3: Comparison of harmonic frequencies for relativistic particle on a spring through Fourier analysis and Gaussian quadrature scheme. Vertical lines are the calculated frequencies from the Gaussian quadrature method.

with the periods for the corresponding initial displacements.

2. Q2 - AUDIO FILTERING

For the pseudo-code of this question please refer to the python script titled `Lab05_Q2.py`. The original sound file used in this question is `GraviteaTime.wav`

2.1. *Q2b - Original Time Series of Audio*

Using the code provided in the background material, the data in the sound file are read into 2 channels, and the time series of both channels are plotted in Figure 4. Specifically, we chose to look at the series at between 0.02 and 0.05 seconds (as suggested by in Question Q2.c).

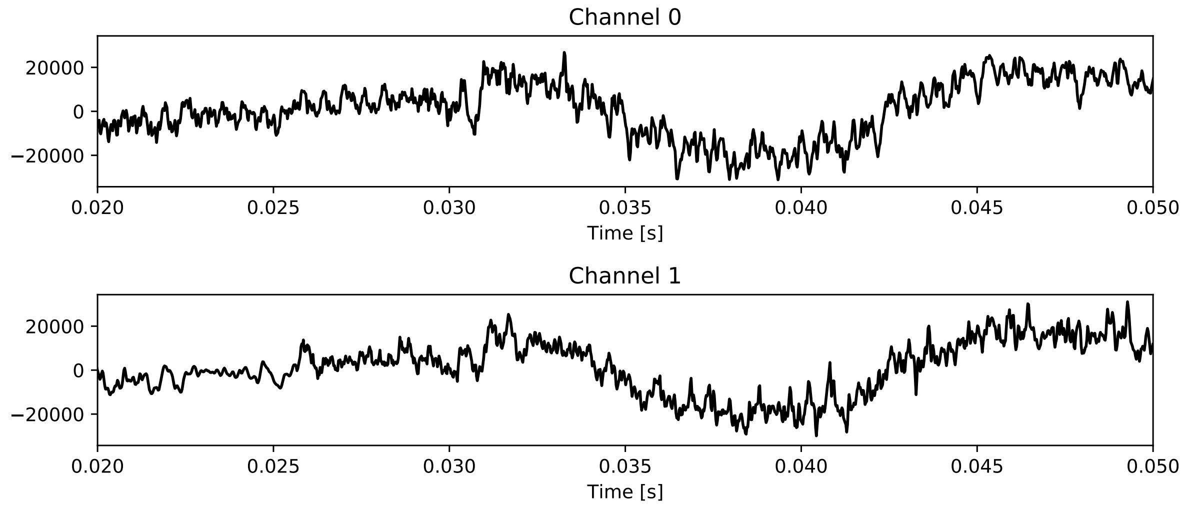

Figure 4: Data (amplitude) in the sound file on channel 0 and channel 1 on the time interval $t \in [0.02s, 0.05s]$.

2.2. *Q2d - Filter Time Series with Fourier Transform*

In order to implement filter frequencies greater than 880 Hz to zero, we first perform Fourier transforms to transform the signal on each channel to the frequency domain, where the amplitude of the Fourier coefficients are plotted in Figure 5.

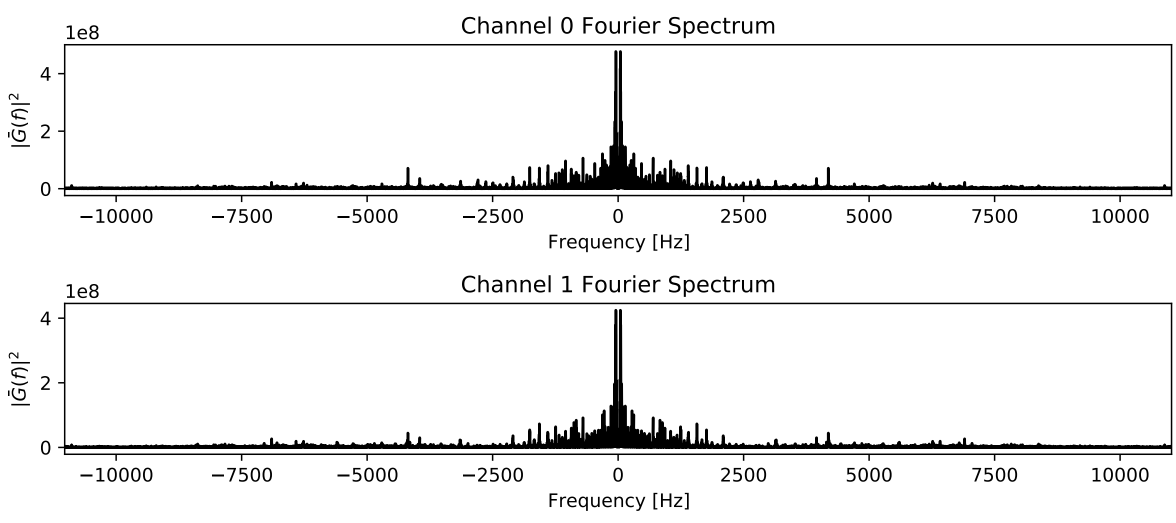

Figure 5: The amplitude of the original Fourier coefficients.

Then based on this frequency domain, we applied a cut such that the amplitude of Fourier coefficient whose (absolute) frequency is greater than 880 Hz would be set to zero. Figure 6 shows the amplitude of the Fourier coefficients after filtering.

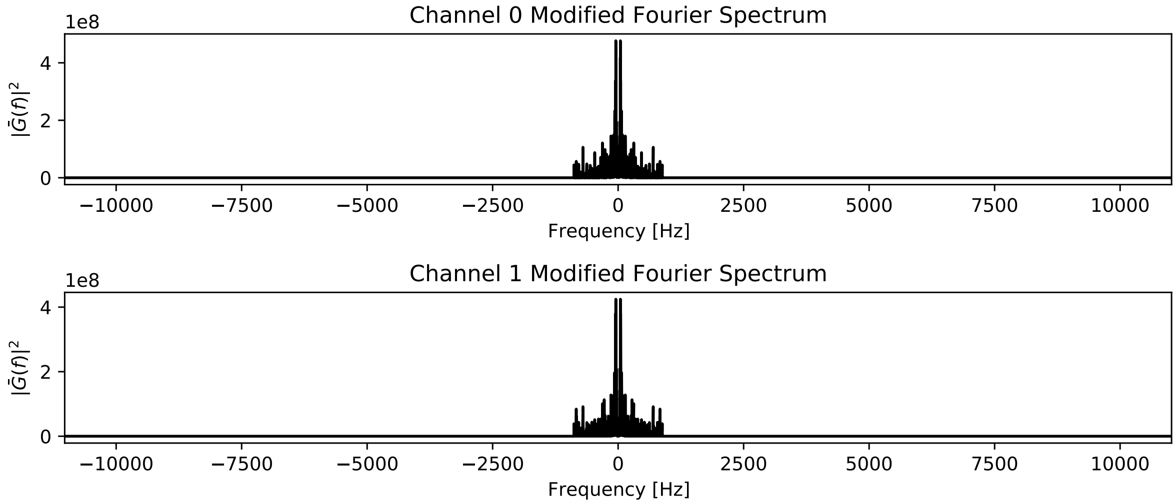

Figure 6: The amplitude of the Fourier coefficients after filtering those greater than 880 Hz.

Finally, we perform inverse Fourier transforms the signal on each channel back to the time domain. The filtered time series is plotted in Figure 7.

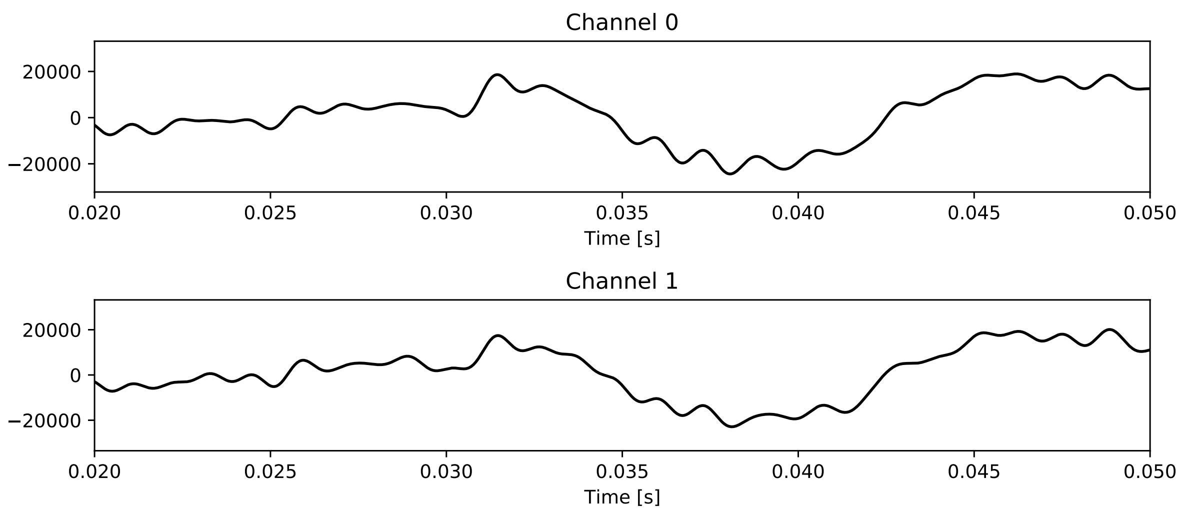

Figure 7: The filtered time series on channel 0 and 1 on the same time interval as in Figure 4.

As we can see, the filtered time series looks like a smoother version of the original time series. The filtered time series on both channel are written into a new sound file `GraviteaTime lpf.wav` attached in submission.

3. Q3 - ANALYSIS OF SEA LEVEL PRESSURE

For the pseudo-code and code for this and all is consecutive sub-questions please refer to the python script `Lab05_Q3.py`.

3.1. *Q3a - 2-Dimensional Fourier Transform of Sea Level Pressure*

We will Fourier decompose in the longitudinal direction the sea level pressure at 50°S, start from January 1^{st}, 2015 throughout 120 days. Figure 8 is read from bottom up as the temporal axis is on the y-axis and the spatial axis is

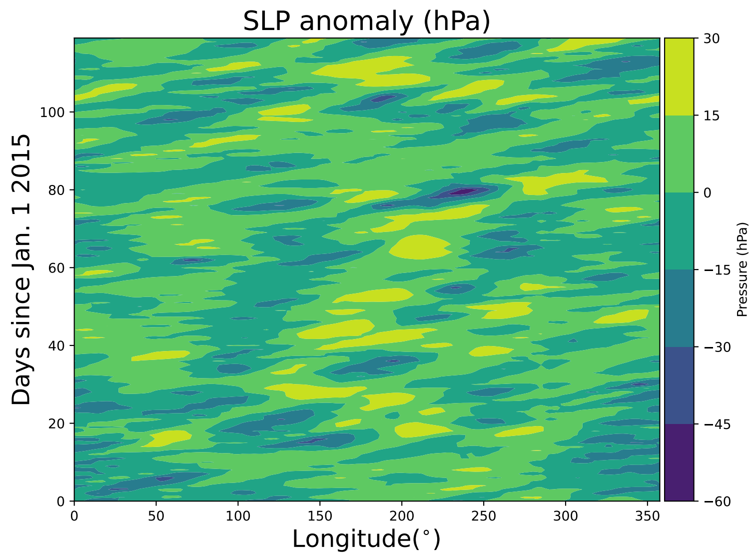

Figure 8: Sea level pressures at 50°S measured longitudinally throughout 120 days

on the x-axis. So we can see the pressure evolution of a patch by fixing a longitude and moving upward. Now we can begin to take a 2-dimensional Fourier transform of the SLP and plot our results in the spectral space. We must

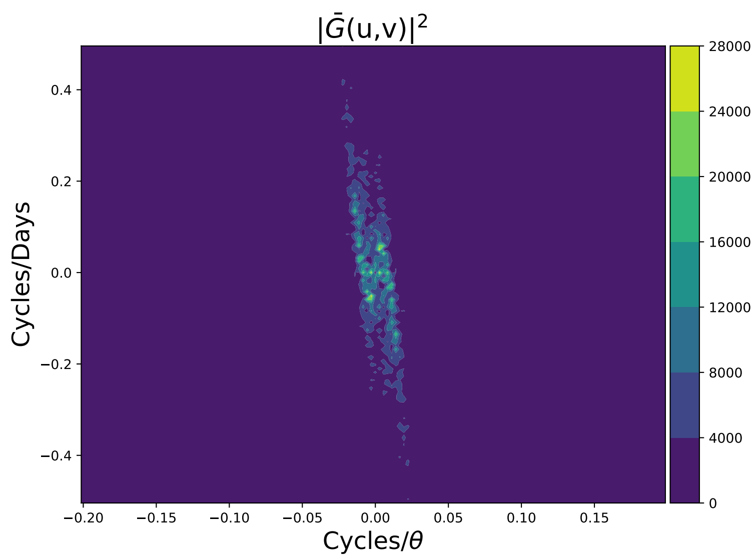

Figure 9: Two-dimensional Fourier transform of sea level pressure measurements. Plotted values are the complex magnitude of the transform. Spectrum has been shifted such that zeroth frequency is at the center of the spectrum.

recognize that the wave number is inversely related to the wavelength therefore, it follows a in inverse relation with the spatial dimension. Therefore, the columns of our Fourier spectrum are the related wave numbers we are looking for, and because wave number can star from zero the index of the column matches the wave number we are interested in looking at. So if we want to isolate the wave number m=3 and m=5 we just mask all other columns with zeros leaving those two columns untouched. We then take a 2-dimensional inverses Fourier on these masked arrays and get the following two results.

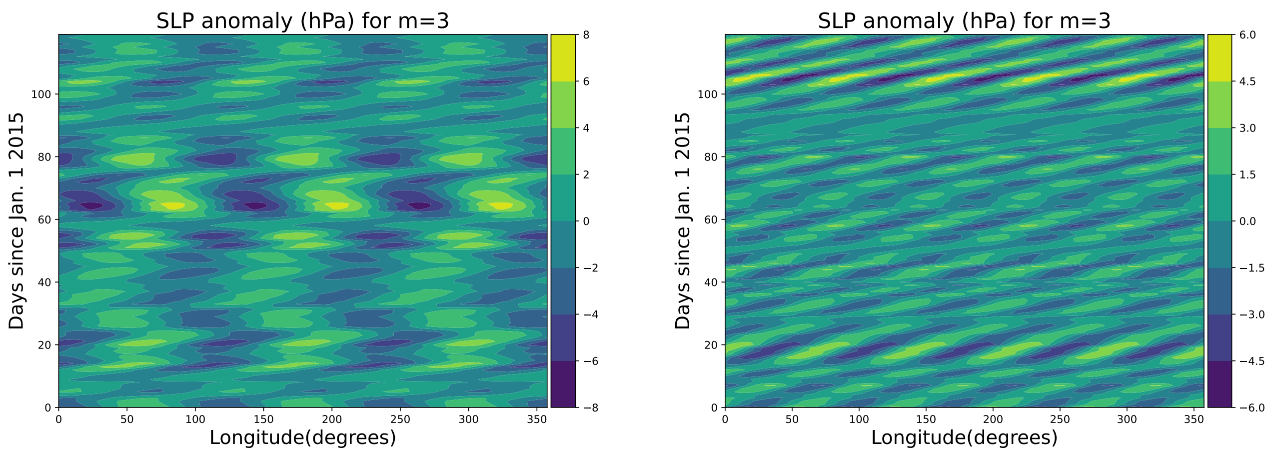

Figure 10: (Left) Sea level pressures for wave number $m = 3$. (Right) Sea level pressures for wave number $m = 5$. Both figures plot the real values of the inverse Fourier because we are looking to observe a real quantity such as the pressure.

3.2. *Q3b - Physical Interpretation*

We begin by noting that wave number is inversely proportional to the wavelength therefore, the SLP measurement for $m = 5$ are for waves with shorter wavelengths than for for $m = 3$. We notice in figure 10 that the pressure measurements have a steeper and clearer pattern from left to right. This indicates the wave is travelling eastward with some velocity proportional to the slope of the pattern. As opposed to the plot for $m = 3$ where such a clear pattern is not so obvious and a positive slope becomes noticeable around 90 days onward. However, the slope of this pattern is not as steep as the one for $m = 5$ which leads us to conclude that these longer wavelength waves travel eastward slower than the shorter waves. Which is consistent with our theory of atmospheric wave propagation.