Chapter 11

Fourier Analysis

Brief intro: still in progress

Computing Fourier transforms using the Fast Fourier Transform (FFT) algorithm is a standard tool in physics. There are many books and articles on Fourier theory and how the FFT works. For brevity I am simply going to focus on what you need to know to use an FFT program and leave the theory for you to pick up elsewhere.

11.1 FFT and time series

We start with a "time series," which is a series of measurements that are made at some regular interval. These measurments are normally stored on the computer in an array. We also need to know the sample interval (the time between sample points, the reciprocal of the digitization rate) and the number of points in the time series. In most cases the time series consists of real (i.e., not complex) numbers and we will assume this is true for the examples shown.

The purpose of the FFT is to examine the different frequency waves that may be present in the data. The FFT is a linear transformation that allows us to look at the data in the "frequency domain" rather than the "time domain" of the original data. In fact, the original time series can be expressed exactly as a sum of harmonic waves of different frequency content and phase. The FFT allows us to readily compute what the frequencies and phases of these waves are.

The simplest FFT algorithms require that the number of points in the original time series be a power of two. We will assume that this is true here, but you should be aware that alternative FFT methods exist for non-powers of two.

Thus, the input to a typical FFT algorithm will consist of an array of data values, the number of time points (n), and the sample interval (dt). For a real time series, the FFT algorithm will then return a complex spectra that will give the amplitude and phase at a

series of equally spaced frequency points. There will be $\frac{n}{2} + 1$ frequency points and the frequency spacing will be $df = \frac{1}{ndt}$.

- The first frequency point is at $f = 0.0$
- The last frequency point is at $f = \frac{1}{2dt}$, and is called the **Nyquist frequency**.

11.1.1 Filters and the Nyquist frequency

The **Nyquist frequency** is the highest frequency that one can resolve in the time series. At the Nyquist frequency, there are two data points in the time series per wavelength. If higher frequencies are present in the data, they will be "aliased" so that they are seen at lower frequencies. From the time series alone, there is no way to discriminate between correctly resolved frequencies and aliased data from higher frequencies. Thus, sometimes filters (anti-aliasing filters) are applied to the data prior to digitization to remove the frequencies above the Nyquist so that this will not be a problem.

11.2 The spectra

The FFT algorithm provides the spectra at $\frac{n}{2} + 1$ frequency points which may be represented as $\frac{n}{2} + 1$ complex numbers[1]. (complex numbers are necessary to represent both the amplitude and phase of each harmonic component).

Once we have the frequency domain points we can make an amplitude spectra plot by plotting the absolute values (Cabs function) of the points versus frequency. If we want a power spectrum, then we square the amplitudes; in many cases, however, we are only interested in the position of the peaks or the shape of the spectrum so we don't need to worry about this.

11.3 Inverse FFT

We can also transform back to the time domain by computing an inverse FFT. If your code is working correctly, you should get back exactly the same time series as you started with.

[1] We started with n real points in the time domain and now we have n+2 real points in the frequency domain, if we count each complex number as two reals. Have we gained information compared to the original time series? The resolution to this apparent problem comes from the fact that the first and last frequency points always have zero imaginary parts. Thus the total number of independent data points is the same in both the time and frequency domains. In fact, most FFT algorithms return only $n/2$ frequency points by packing the real part of the Nyquist point into the imaginary part of the zero frequency point.

11.4 Appendix

11.4.1 Lab exercise

Question 31 *Example 1*

A properly tuned guitar will have strings with fundamental mode frequencies as follows:

$$\begin{aligned} E &= 82Hz \\ A &= 110Hz \\ D &= 147Hz \\ G &= 196Hz \\ B &= 247Hz \\ E &= 330Hz \end{aligned} \tag{11.1}$$

There will be overtones (harmonics) at frequencies of $n \times f_0$ where f_0 is the fundamental (lowest) frequency. Thus, for the A string, we should expect to see peaks at 110 Hz, 220 Hz, 330 Hz, 440 Hz, etc. The relative power between the different harmonics is what determines the tone of the guitar.

Digitization: First, find a way of digitizing the sound of the A string. Save the result to a file which we are going to FFTed and the resulting spectrum plotted.

Do the following comparisons:

- string plucked near center vs. near one end
- sound sampled just after pluck vs. a few seconds later
- cheap vs. expensive guitar