Quiz 2: Digital Signal Processing

Prof. Jordan C. Hanson

April 9, 2025

1. (a) Derive the Fourier series of a square wave, listing which coefficients are non-zero. (b) Write an `octave` script that creates a time series corresponding to the Fourier series, with a fundamental frequency of 1.4 MHz and a maximum time T_{max} corresponding to about 10 periods. (c) Compute the DFT of the time series. (d) Using the `octave` routines `figure()` and `subplot()`, plot the time series in one graph, and in the same figure, plot the magnitude of the DFT in another graph.

2. Do you observe the **Gibbs effect** in your Fourier series in the previous exercise? Why or why not?

3. (a) Define an $N = 1000$ sample $\delta[n]$ signal in an `octave` script, with the non-zero value in the first sample. (b) Compute the magnitude and phase of the DFT, and graph them versus frequency. (c) Advance the non-zero sample in the $\delta[n]$ by 100 samples, and recompute the graphs. (d) What is happening to the phase? Use the `unwrap()` function in `octave` to graph the linear relationship between phase and frequency. (e) Use the slope of the phase versus frequency to measure the group delay of the $\delta[n]$. Do you obtain the right result? (f) **Bonus:** what happens to the group delay measurement if the $\delta[n]$ signal has noise?

4. **Clipping** in DSP data. (a) Using `octave`, create a sine wave with the following properties: a sampling rate of 10 MHz, a frequency of 100 kHz, a T_{max} of 6 ms, and an amplitude of 1.0. The data should have more than 10^4 samples, so there is no need to graph it. (b) Using the `find()` function in `octave`, set all samples *greater* than 0.75 to 0.75. Set all samples *less* than -0.75 to -0.75. Here is a clue as to how this works:

   ```
   x(find(x>=0.75)) = 0.75
   ```

   The resulting signal is now **clipped**. Clipping often results when DSP data falls outside a nominal digital range. (c) Plot the magnitude of the DFT, and locate the frequency spike at 100 kHz. (d) Do you observe harmonics? In your own words, explain the spectrum of harmonics given that the signal is clipped.

5. Download the "NASDAQ closing prices 2024-2025" file from the course Moodle page. The left column has units of days (starting with April 10th, 2024), and the right column is the value of the NASDAQ stock index in US dollars. (a) Plot the data, and in the same plot, apply a 1-week moving average filter to smooth the data. (b) Note where the announcement of US tariffs marks a sharp drop in value. (c) Note where the value rises sharply after the US President announced a 90 day pause in tariffs during April 2025. (d) Does this moving average capture rapid shifts in economic policy? Why or why not? (e) **Bonus**: measure the *lag* of the moving average filter. Lag represents the time it takes the filter to respond to the data.

6. Perform each of the following in `octave`. (a) Assume our task is to isolate an AM radio station with a carrier frequency of 740 kHz. Create a low-pass, windowed-sinc filter designed to filter noise above 745 kHz. The number of samples M in the filter kernel is your choice. (b) Using *spectral inversion*, create a high-pass windowed-sinc filter designed to filter noise below 735 kHz. (c) Combine these filters to create a band-pass filter, and plot the frequency response. (d) Mix a 740 kHz carrier with a 2.5 kHz audio signal *plus noise*, and plot the magnitude of the DFT. (e) Use your band-pass filter on the data, and plot the filtered spectrum.

7. **FFT convolution.** Perform the following with FFT convolution. (a) Show that the convolution of two square pulses is a triangle wave. (b) Show that the convolution of one period of a sawtooth wave with itself is a "quadratic wave." **Bonus:** Generate code to play these signals as audio to hear the differences.

8. (a) Create a step pulse in `octave`, and run it through a recursive LP filter. (b) Plot the *phase* of the output versus frequency. (c) Are the results *linear* or non-linear?

9. (a) Reverse the order of the step pulse samples in the previous exercise. (b) Run the reversed step through the recursive LP filter, and plot the phase response. (c) Run the step through the recursive LP filter, reverse the output, and run it through again, to show the phase response is linear (or zero). *For a clue, see Fig. 19-8 in the course textbook.*

10. **Bonus: chirping signals**. Suppose we have a signal with a frequency that depends on time:

    $$f(t) = f_0 - \beta t \tag{1}$$

    The start frequency is f_0 in MHz, and the "chirp rate" is β, with units of MHz/μs. The chirp signal is

    $$s(t) = A\cos\left(2\pi f(t) t\right) \tag{2}$$

    If we delay the signal by a time t_{d}, this corresponds to $s(t - t_{\text{d}})$. (a) Show that mixing $s(t)$ and $s(t - t_{\text{d}})$ results in two frequency components, and the lower one is βt_{d}. (b) Using `octave`, create a chirping signal with $\beta = 2$ MHz/μs, $f_0 = 5$ MHz, and $t_{\text{d}} = 0.5$ μs. Add plenty of noise to the signal. (c) Using the filter of your choice, isolate the low-frequency component. Show that your code produces the correct value for the "downconverted" low-frequency, βt_{d}. DSP for chirping signals can arise in the analysis of radar echos for fast-moving objects.