ECE 148 Homework 4

Sanjot Bains

May 8, 2025

Problem 1: Fourier Series Expansion

The periodic signal $f(t)$ with period $T = 16$ is defined as:

$$f(t) = \begin{cases} -5, & -8 \leq t < -5 \\ 3, & -5 \leq t < 5 \\ -5, & 5 \leq t < 8 \end{cases}$$

The plot of $f(t)$ over one period is shown below:

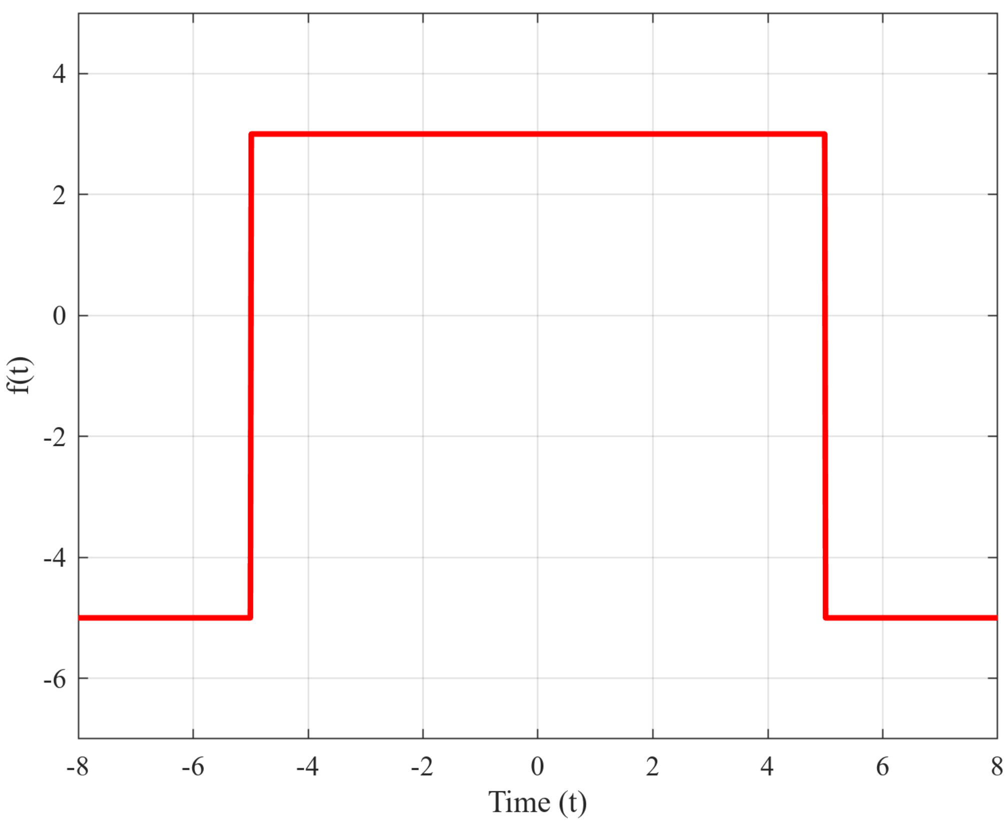

Figure 1: Periodic Signal $f(t)$

The Fourier series expansion of $f(t)$ in complex form is given by:

$$F_n = \frac{1}{T}\int_{-T/2}^{T/2} f(t)e^{-jn\omega_0 t}dt$$

where $\omega_0 = \frac{2\pi}{T}$ is the fundamental frequency. Of course, in the MATLAB code we approximate the integral using a Riemann sum:

```
w_0 = 2*pi/T;

N = 7; % Number of harmonics

k = -N:N;

f_t = f(t);
dt = t(2) - t(1);

F_n = zeros(size(k));
for n = 1:length(k)
    integrand = f_t .* exp(-1j * k(n) * w_0 * t);
    F_n(n) = (1/T) * sum(integrand) * dt;
end
```

Listing 1: MATLAB Script to Compute Fourier Series coefficients

Problem 2: Signal Reconstruction

Using 7 harmonics ($n = \pm 1, \pm 2, \ldots, \pm 7$), the signal is reconstructed as:

$$f_{\text{reconstructed}}(t) = \sum_{n=-7}^{7} F_n e^{jn\omega_0 t}$$

```
f_reconstructed = zeros(size(t));
for n = 1:length(k)
    f_reconstructed = f_reconstructed + F_n(n) * exp(1j * k(n) * w_0 * t);
end
```

Listing 2: MATLAB Script to Reconstruct Signal

The reconstructed signal is compared with the original signal below:

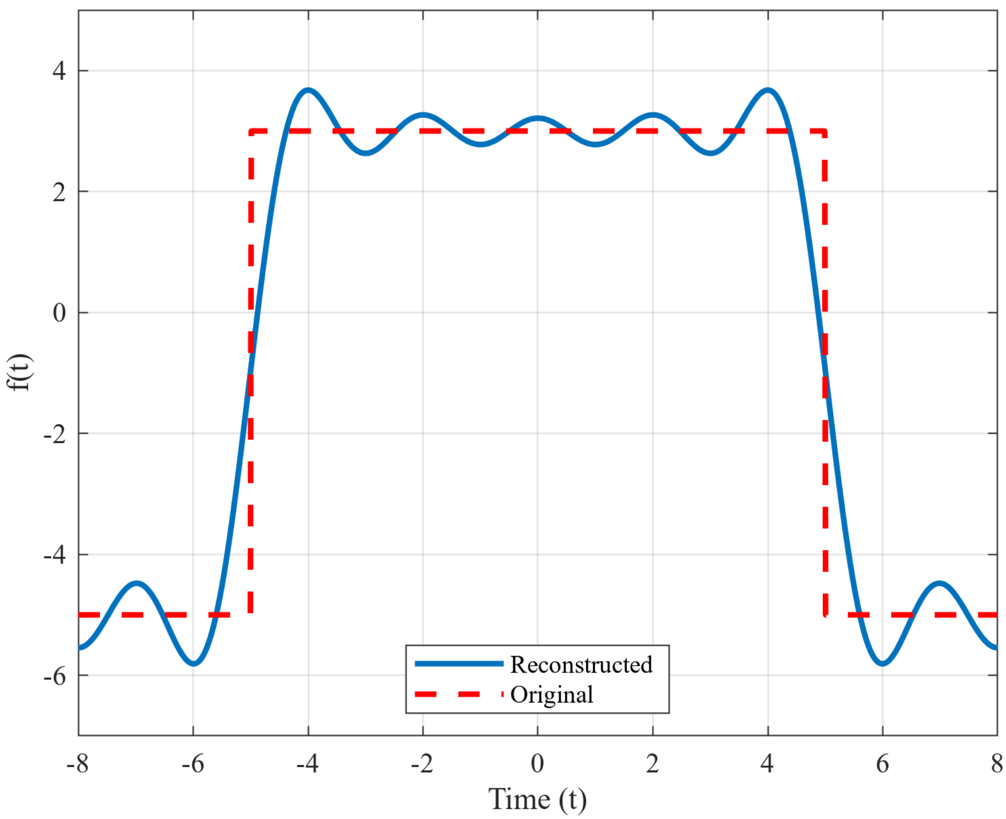

Figure 2: Reconstructed Signal vs Original Signal

Problem 3: DTFT of Fourier Coefficients

The 15 Fourier coefficients $\{F_n\}$ are used to compute the DTFT:

$$F(\omega) = \sum_{n=-7}^{7} F_n e^{-j\omega n}$$

```
omega = linspace(-pi, pi, 1000);

F_omega = zeros(size(omega));
for k = 1:length(omega)
    F_omega(k) = sum(F_n .* exp(-1j * omega(k) * (-N:N)));
end
```

Listing 3: MATLAB Script to Compute DTFT

The DTFT is plotted over the interval $(-\pi, \pi)$:

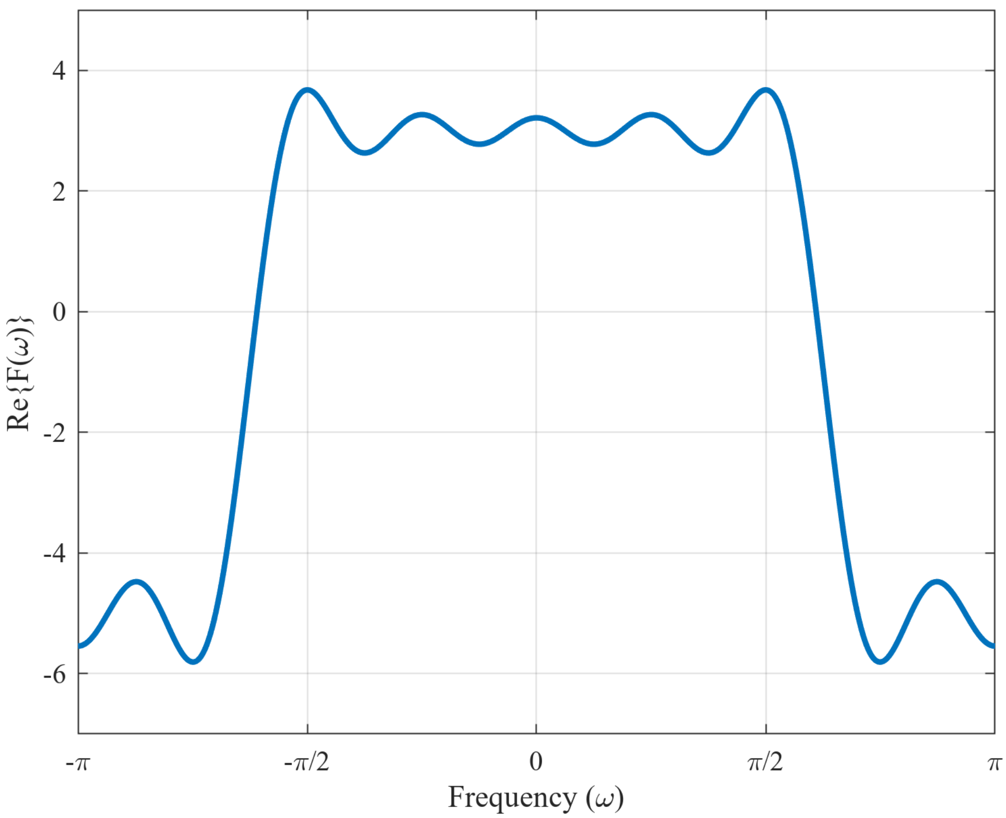

Figure 3: DTFT of Fourier Coefficients

Problem 4: 32-point DFT

The 15-point sequence $\{F_n\}$ is padded with zeros to form a 32-point sequence. The 32-point DFT is computed and plotted below:

```
F_32 = zeros(1, 32);
F_32(2:8) = F_n(9:15); % Positive Fourier coefficients (F_1 to F_7)
F_32(26:32) = F_n(1:7); % Negative coefficients (F_-7 to F_-1)

F_fft_32 = fft(F_32);
F_fft_32_shifted = fftshift(F_fft_32);
```

Listing 4: MATLAB Script to Compute 32-point DFT

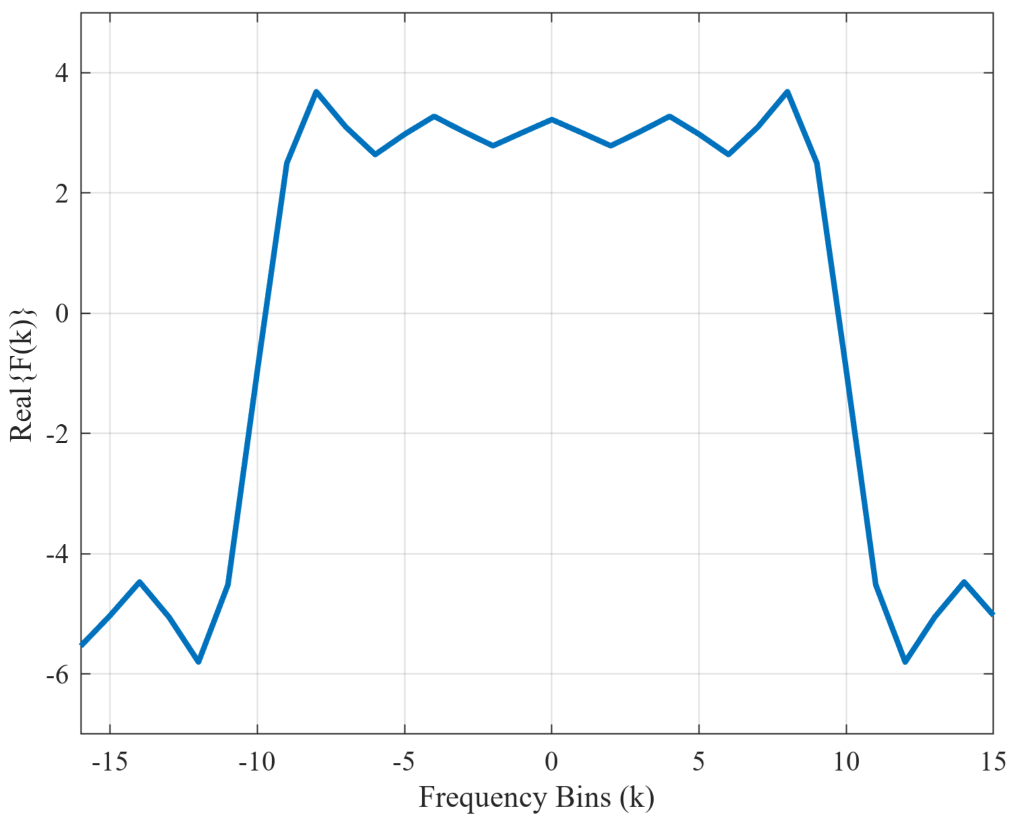

Figure 4: 32-point DFT of Fourier Coefficients

Problem 5: 64-point DFT

Similarly, the 15-point sequence $\{F_n\}$ is padded with zeros to form a 64-point sequence. The 64-point DFT is computed and plotted below:

```
F_64 = zeros(1, 64);
F_64(2:8) = F_n(9:15);
F_64(58:64) = F_n(1:7);

F_fft_64 = fft(F_64);
F_fft_64_shifted = fftshift(F_fft_64);
```

Listing 5: MATLAB Script to Compute 64-point DFT

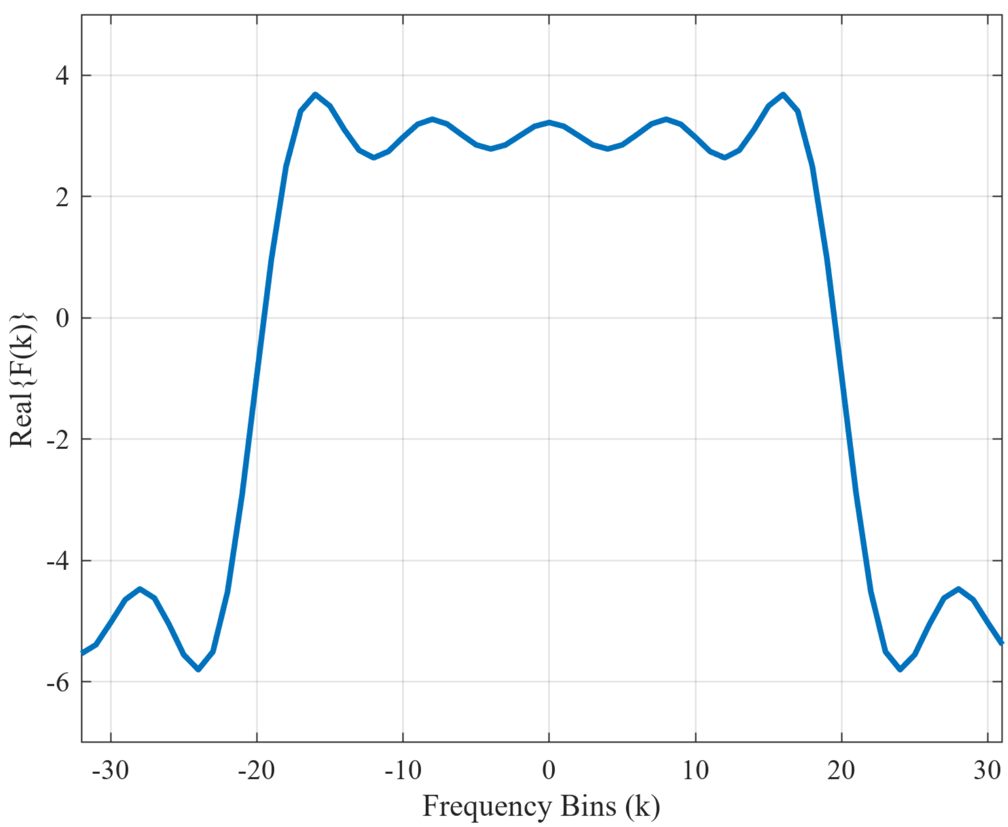

Figure 5: 64-point DFT of Fourier Coefficients

Problem 6: Summary

- The Fourier series expansion resembles the periodic signal $f(t)$, but does not contain the discontinuities. To "match" the original signal, we would need to use more harmonics.
- The DTFT contains the same information as the Fourier series expansion, but is not periodic and is plotted on the interval $(-\pi, \pi)$.
- The 32-point and 64-point DFTs are similar to the DTFT, but are sampled at discrete intervals.
- The 32-point DFT has a lower frequency resolution than the 64-point DFT. I initially plotted these as bar graphs as the FFT is classically represented, but I kept the line plots for consistency with the DTFT.
- The 32-point and 64-point DFTs demonstrate the effect of zero-padding on frequency resolution. The input to both contains the same information, so long as one does not conside the zeros as additional info.
- The 64-point DFT has a higher resolution and is better able to represent the component frequencies.
- I spent more time on the LaTeX than the actual homework, but I had fun and that's what matters.