

C.7 Spectrum Solutions

C.7.1 In-lab section

1. The magnitudes of the discrete Fourier series coefficients are plotted in the top plot of figure C.13. This is calculated using the following Matlab code:

```
t = [0:1/8000:1-1/8000];
frequencies = [0:4000];

% First waveform, a sinusoid.
x = sin(2*pi*800*t);
[Ax, Px] = fourierSeries(x);
subplot(3,1,1);
plot(frequencies, Ax);
title('FS coefficients of a sine wave');
ylabel('amplitude');
```

The horizontal axis is the frequency in Hertz of the sinusoidal component corresponding to each coefficient. Note that only one of the coefficients is non-zero, the one at 800 Hz. It indicates an amplitude of 1 for that component.

2. The chirp is generated and plotted by the following code, which assumes `t` and `frequencies` have been defined as above:

```
% Second waveform, a chirp.
y = sin(2*pi*800*(t.*t));
[Ay, Py] = fourierSeries(y);
subplot(3,1,2);
plot(frequencies, Ay);
title('FS coefficients of a chirp');
ylabel('amplitude');
```

The magnitudes of the discrete Fourier series coefficients are plotted in the middle plot of figure C.13. That plot shows frequency components ranging from 0 to 1600 Hz, which is exactly the range of instantaneous frequencies predicted by

$$\omega(t) = \frac{d}{dt}2\pi 800t^2 = 4\pi 800t$$

(divide by 2π to get frequencies in Hz). The odd ringing near 1600 Hz is difficult to explain. It is, in fact, a sort of Gibbs' phenomenon.

3. The reverse chirp is calculated and plotted using:

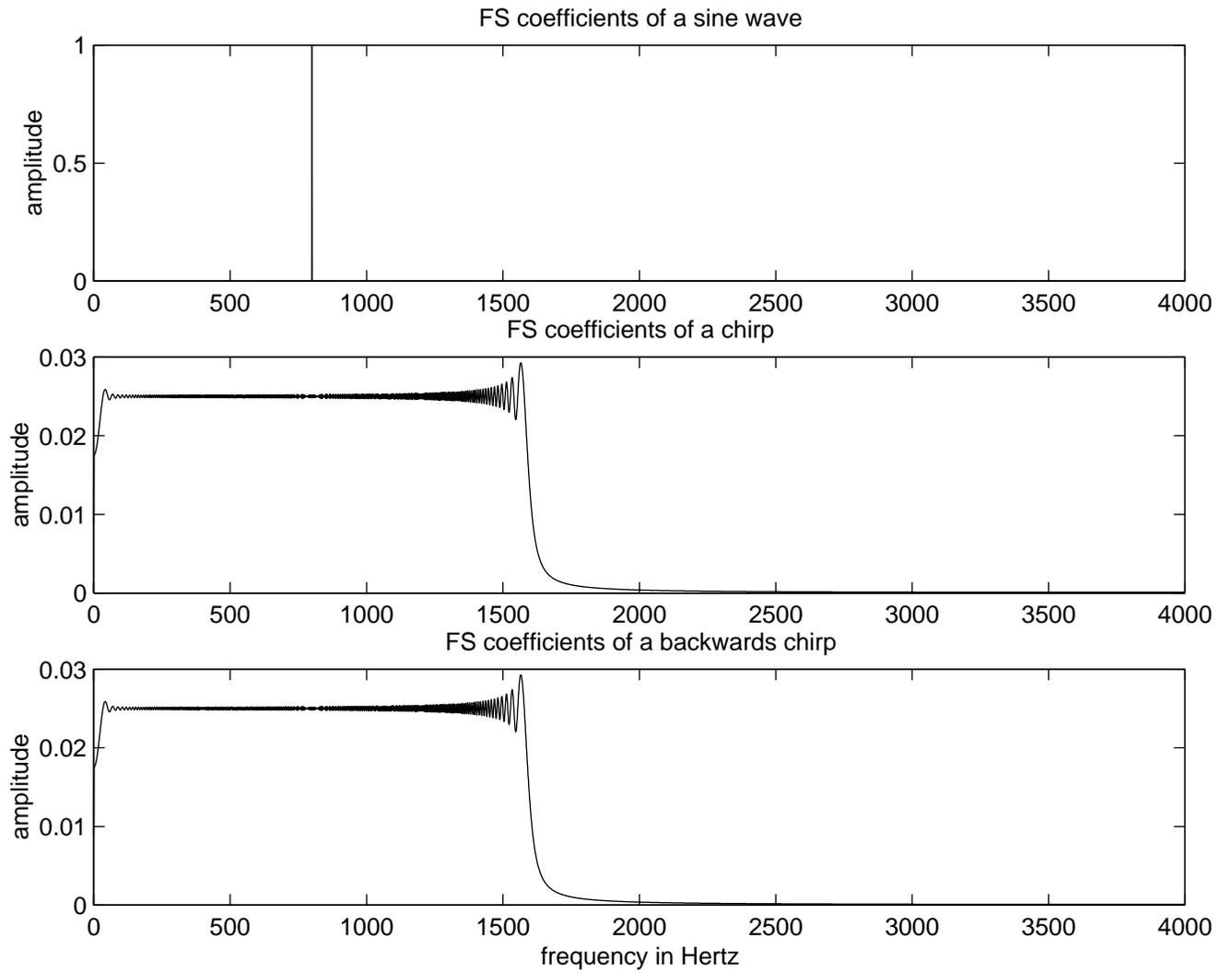


Figure C.13: Discrete Fourier series coefficients of a sine, chirp, and reverse chirp.

```

% Third waveform, a reverse chirp.
z = y(8000:-1:1);
[Az, Pz] = fourierSeries(z);
subplot(3,1,3);
plot(frequencies, Az);
title('FS coefficients of a backwards chirp');
ylabel('amplitude');
xlabel('frequency in Hertz');

```

This is shown in the bottom plot of figure C.13. It is evidently identical to the discrete Fourier series coefficients of the chirp, although it certainly does not sound the same. This suggests that the differences in the two waveforms must be entirely in the phases of their sinusoidal components, which we are not plotting. Moreover, these phase differences are large enough to be perceptible as time shifting of the signal. In fact, this signal is a reverse chirp, where the highest frequency appears first, and the lowest last.

4. In the given plot, the first set of DFS coefficients shows most of the frequency content near zero. Each successive plot shows the frequency content at higher frequencies. This corresponds well with the psychoacoustic perception of the sound, which is that of a sinusoid shifting in frequency from 0 to 1600 Hz.

The plot for the reverse chirp is shown in figure C.14. That plot shows a spectrum that starts with most of the frequency content at 1600 Hz. It then shifts down to zero. The sound is similar to the chirp, but going from high to low frequencies rather than low to high. The Matlab code for generating the sets of DFS coefficients of the reverse chirp is:

```
waterfallSpectrogram(z, 8000, 400, 30);
```

5. The spectrogram for the reverse chirp is generated by the following commands

```

specgram(z, 512, 8000);
title('spectrogram of a reverse chirp');

```

The result is shown in figure C.15.

6. The texttttrain.au file yields the spectrogram and plot shown in figure C.16.

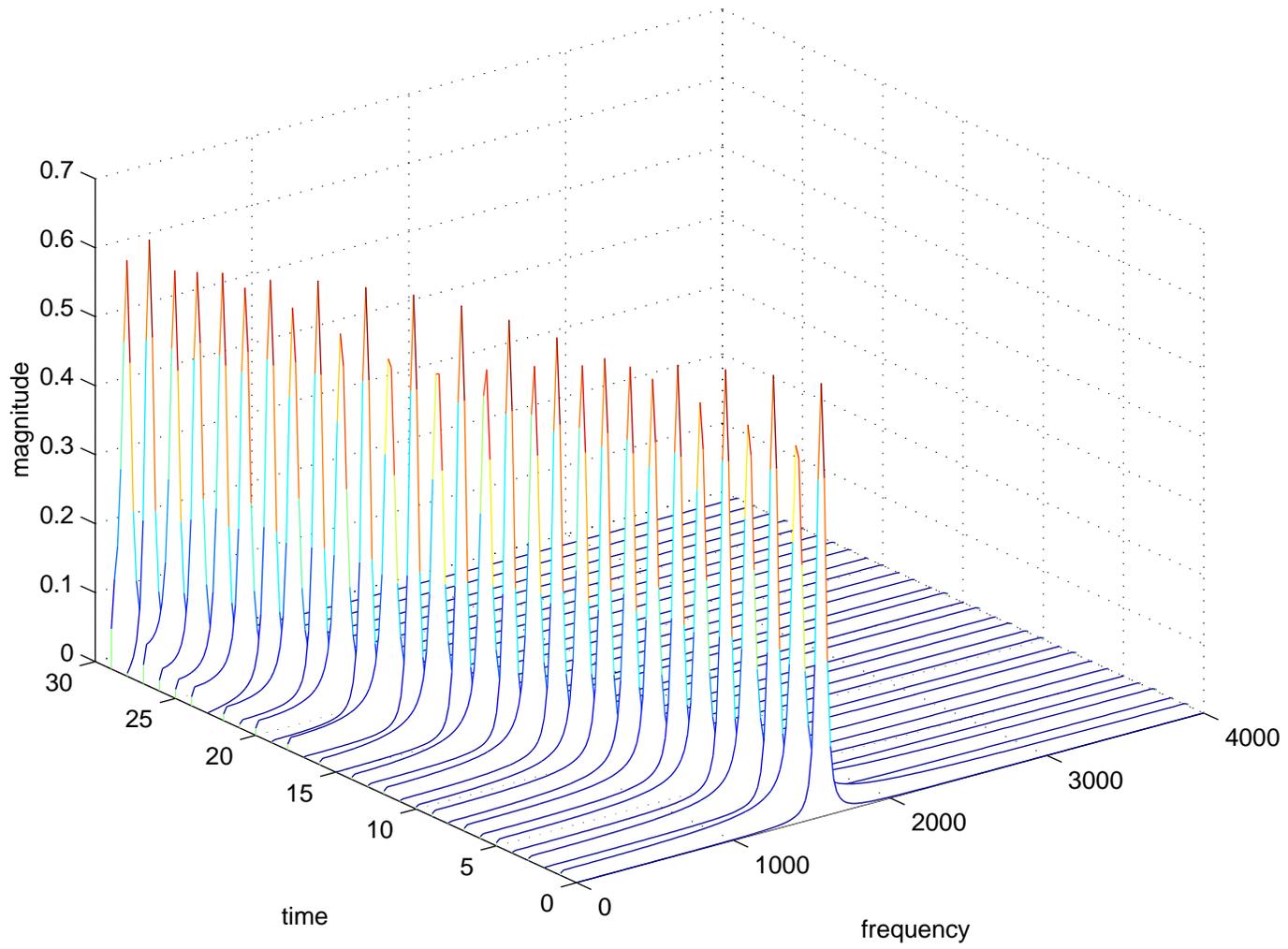


Figure C.14: Time varying discrete Fourier series analysis of a reverse chirp.

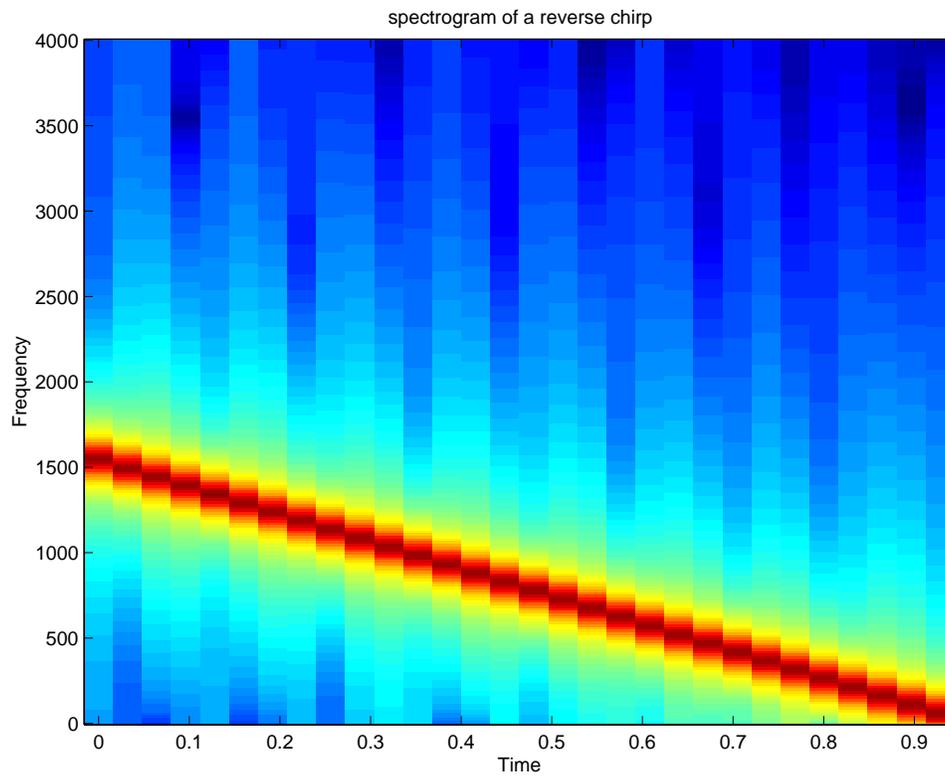


Figure C.15: Spectrogram of the time reversed chirp.

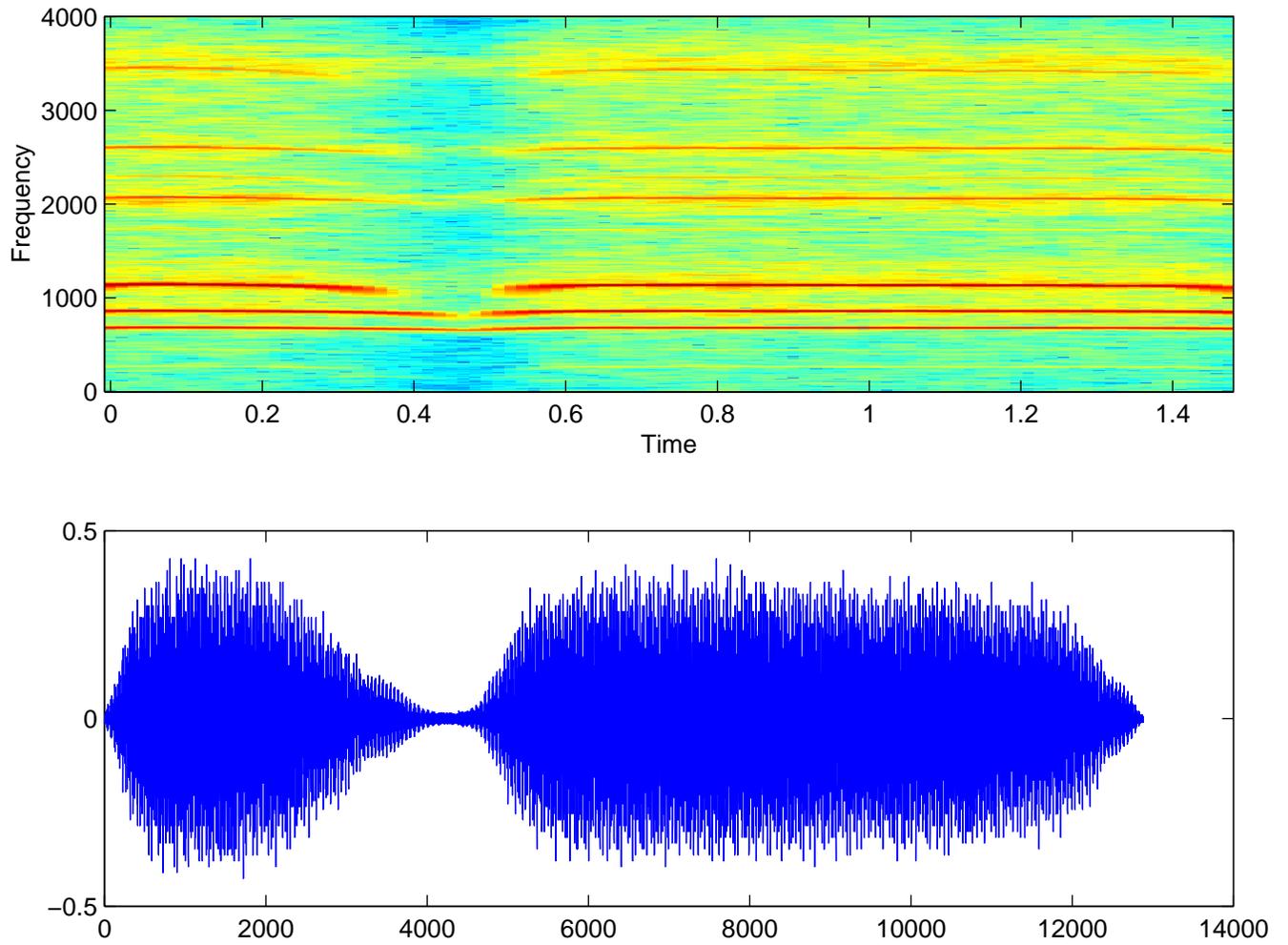


Figure C.16: Spectrogram and waveform for a train whistle.