EE430
TERM PROJECT
PART II

Arda Yazıcı 2232981
Kutay Uğurlu 2232841

Kutay Uğurlu 2232841
Arda Yazıcı 2232981

Content

Kutay Uğurlu 2232841
Arda Yazıcı 2232981

EE430 Term Project Part 2

In the second part of our term project, we have studied a speed and range estimation problem by utilizing the Short-Time Fourier Transform (STFT) of a time-domain signal. STFT is some kind of a Fourier Transform which is utilized to specify the sinusoidal frequency content of local parts of signals. Firstly, we have determined the range of the source by time-of-arrival estimation. Besides, we have determined the speed of the source by frequency estimation. In addition, because of the Doppler effect, the signal which is emitted by the moving source is received in a modified form. The aforementioned modification is dependent on the velocity vector of the source with respect to the receiver.

For the issue, which is explained above, we have implemented signal processing methods in MATLAB. There are five separate parts in this project. These parts are Signals Emitted by a Moving Source, Generating the Signals, Methods/Algorithms, Results/Discussion, and Conclusion respectively. In this report document, the aforementioned parts are explained in a detailed fashion.

Signals Emitted by a Moving Source:

In the second part of the project, we have considered the following scenario. The source emits short pulse with already known starting time. Besides, this emitting source has a nonzero speed with respect to the sensor. By taking the Doppler effect into account, this sensor investigates the delayed and warped version of the transmitted signal with additive noise. Furthermore, we are assuming that clocks of the sensor and the source are synchronized. We are also assuming that the signal travels at a constant speed of c = 340 m/s in the medium.

Generating the Signals:

There are a number of different MATLAB scripts that generate time-domain samples representing the required functions. The functions return both the signals and vectors for time axes. The vector that represents the time axis is generated by sampling the time axis of length total_length uniformly by the number of F_s*total_length, where the total_length represents the length of the signal in seconds and F_s represents the sampling frequency in Hz.

The explanation and the code for these functions are presented below as follows.

Kutay Uğurlu 2232841
Arda Yazıcı 2232981

Sinusoidal signal:

```
function [signal, t] = get_cosine(total_length, F_s, frequency, amplitude, phase)
t = linspace(0,total_length,total_length*F_s);
signal = amplitude * cos(2*pi*frequency*t+phase);
end
```

Rectangle windowed linear chirp:

```
function [signal, t] = get_linear_chirp(duration, F_s, frequency_init, amplitude, phase, bandwidth,
shift)
t = linspace(0,duration,duration*F_s);
t = t + shift; % Assumed user inputs t0 to produce s(t-t0)
signal = amplitude * cos(2*pi*(frequency_init*t + bandwidth/(2*duration)*t.^2) + phase);
end
```

Windowed sinusoidal:

```
function [signal, t, w] = get_windowed_s(total_length, F_s, frequency, amplitude, phase,
window_name, starting_time, window_length)

%Built-in window uses sample number. To convert seconds to sample number we
%must use
T =  1 / F_s;
t = [0 : T : total_length];
t_w = [starting_time : T : starting_time + window_length];
w = transpose(window(str2func(window_name), window_length*F_s + 1)); %To maintain the
given sampling period, needed to consider the outermost samples
signal = amplitude * cos(2*pi*frequency*t + phase);
intersection = intersect(t,t_w);
idx_start_signal = find(t == intersection(1));
idx_end_signal = find(t == intersection(end));
idx_start_window = find(t_w == intersection(1));
idx_end_window = find(t_w == intersection(end));
signal = signal(idx_start_signal:idx_end_signal) .* w(idx_start_window:idx_end_window);
w = w(idx_start_window:idx_end_window);
t = t(idx_start_signal:idx_end_signal);
end
```

Kutay Uğurlu 2232841
Arda Yazıcı 2232981

The Short-Time Fourier transform (STFT):

In this part of the term project, to realize the STFT operation, we first determined the number of time slices for the STFT window to iterate through. Until the STFT window attains the last slice, the function takes the FFT of the element-wise product of that slice of the signal with the window. When the STFT window arrives at the end of the signal, we have developed our MATLAB code such that it takes the FFT of the product of the window with the last window-length samples of the function.

```
function [STFT, number_of_slices] = st_ft(F_s ,signal, window, stride_in_sec)
window = transpose(window);
window_length_in_sample = length(window);
stride_in_sample = stride_in_sec * F_s;
N  = length(signal);
number_of_slices = ceil((N - window_length_in_sample)/stride_in_sample) + 1; % total number of
slides including semi-full one
STFT = zeros(window_length_in_sample, number_of_slices);
for i = 1:number_of_slices-1
    STFT(:,i) = transpose(fftshift(fft(signal((i-1)*stride_in_sample+1:(i-1)*
stride_in_sample+window_length_in_sample).*window)));
end
STFT(:,number_of_slices) = transpose(fftshift(fft(signal(end-window_length_in_sample+1:end))));
end
```

Up to this point, we have demonstrated the MATLAB scripts that generate the required signals and we have also shown the script for the STFT operation. In the following parts of the report, we will be providing information about the spectrogram of the signals and the methods/algorithms we have utilized in the second part of the term project.

Spectrogram:

In the previous part, we have fulfilled the STFT operation for the generated signals. The output of the above STFT function is a matrix whose columns store the FFT values of the element-wise product of the window and the respective time slice. In order to determine the frequency intervals where we want to compute the frequency content, we have specified an input parameter for the frequency range. The Spectrogram function iterates through the columns of the STFT matrix and takes the sum of the absolute square of the frequency spectrum slices generated based on the frequency range input. The symmetry around the 0 Hz axis is due to the fact that the magnitudes of FFTs of the real signals, that we generated, are even. Besides, it needs to be mentioned that we could have truncated the rows corresponding negative frequencies but we have decided to keep it as it is.

Kutay Uğurlu 2232841
Arda Yazıcı 2232981

Determined ranges for the parameters to be used in estimation:

In this part of our term project, we have been expected to determine our own ranges for the values of Δ, Ω_0, m, t_0, v.

The first parameter we have decided to set is duration to be used for the utilization for the estimation of the speed and range by the visual inspection of the spectrogram. Afterwards, the sampling frequency was chosen to be 40 kHz. Besides, taking the extreme cases into account, we have determined the ranges for the parameters as follows:

- First of all, we have determined the range for Δ as follows: $\Delta = (0, 10)$ s
- Then, the maximum speed that we wanted to estimate is $v = (0, 0.75\mathrm{c}) = (0, 255)$ m/s
- After that, in the duration that we computed the STFT of the received signal, the maximum delay that we expect is $t_0 = (0, 5)$ s. Hence, the maximum range that we claim to estimate turns out to be $5\mathrm{c} = 1700$ m.

Taking these above values into account, we should prevent the maximum power carrying portion of the signal's frequency from being greater than 20 kHz which corresponds to Nyquist Rate, in other words, the maximum frequency that we are able to detect in the signal using STFT.

- For the sinusoidal signal, $\Omega_0 = (0, 5\text{x}10^3)$ Hz

The reason how above limit was set is by considering the extreme cases. One of such extreme case is maximum speed. A signal source moving with 0.75c results in a Doppler shifted signal which is 4 times compressed of the original (transmitted) one. Also, we expect the detected frequency to be of maximum 20 kHz. That is why we have designed the transmitted signal to be of maximum 4 kHz.

In addition, for the linear chirp signal, we should consider both the frequency shift due to the definition of the linear chirp signal and the frequency shift because of the Doppler phenomenon. Furthermore, again considering the extreme cases, we have attained the following boundary for m.

- $m = (0, \sim 40\text{x}10^3)$

Using the frequency shift amount formula $f_0 + t\frac{m}{2\Delta} = \frac{F_s}{2} = 20kHz$ regarding the linear chirp signal and inserting t=5 seconds in the formula, we arrived at the following conclusions:

For the zero relative speed case, if we were to receive a 5 seconds signal, that would also imply a duration of 5 seconds, corresponding bandwidth value depends on the initial frequency f_0 of the linear chirp signal.

Presuming the initial frequency as 20 kHz, we have approximated the bandwidth to be around 39.8 kHz.

For the maximum speed case, the duration corresponding to 5 seconds received signal is 20 seconds. In order for the received signal to carry a maximum frequency of 20 kHz, the transmitted signal should carry a maximum frequency of 5 kHz. Inserting these previously asserted values to the above equation, the bandwidth was approximated to be around 38.4 kHz.

The approach in the above texts justifies the design choice for the parameter m.

Methods/Algorithms:

In this part of the report, our main purpose is to explain the methodology we have developed in terms of the estimation of the speed and range for sinusoidal and linear chirp signals.

In the system we have designed, the signals to be transmitted are amplified before the transmission so that we can have a situation such that the disregarding of the noise becomes easier. At the aforementioned case, we have also considered that the signals are constant in amplitude.

Speed Estimation for Sinusoidal Signal:

We know surely that frequency spectrum of the Doppler-shifted sinusoidal signal is composed of only one distinct frequency when the noise is disregarded. By using the last time instance portion of the STFT of the signal, we approximate the fundamental frequency of Doppler-shifted signal via computing the frequency range carrying the highest power. Furthermore, even if we are still able to compute the spectrum of the transmitted signal, we already know that which frequency of sinusoidal we have transmitted. Utilizing the formula $\frac{c-v}{c} = \frac{f_{transmitted}}{f_{doppler}}$, we are able to estimate the speed of the signal source in terms of the constant speed of sound in the medium.

Range Estimation for Sinusoidal Signal:

In terms of estimating the range for the signal, we have computed the delay using the first nonzero sample of noise disregarded signal.

Kutay Uğurlu 2232841
Arda Yazıcı 2232981

Speed Estimation for Linear Chirp Signal:

In the speed estimation for the linear chirp signal case, instead of using the frequency carrying maximum power content in the latest time portion, this time we have utilized the earliest time portion right after the detection of the Doppler shifted signal. In the case of speed estimation for the sinusoidal signal, we have used the formula $\frac{c-v}{c} = \frac{f_{transmitted}}{f_{doppler}}$. On the other hand, for the linear chirp signal, we have fulfilled the speed estimation via the formula $\frac{c-v}{c} = \frac{f_{transmitted}+\frac{m}{5}}{f_{doppler}}$, which was obtained by the observations made on the spectrogram data. The detailed steps of the algorithm can be investigated from the appendices.

Range Estimation for Linear Chirp Signal:

First of all, it can be said that linear chirp signal can be considered as a sinusoidal signal whose frequency is a linear function of time. Moreover, being constant amplitude, we can again utilize the method we have previously made usage of for the sinusoidal delay estimation case. The approach we have benefited utilizes the last sample of the block that consists of continuum of zero samples.

Test Procedure:

In order to test the performance of the algorithm that we have developed, for constant speed and range values, we simulated realizations with random errors for multiple times. Besides, we did utilize the mode as an evaluation metric. Using this value, we have evaluated the performance of the estimation.

Another Supplementary Approach for Estimation:

In this section, we want to mention an approach that we have previously developed in the design phase for the method for estimation. This approach is based on the assumption such that received signals are exposed to zero-mean Gaussian noise that results in constant signal to noise ratio (SNR). In this case, each of the realizations has the form $r(t) = s(t) + n$ where $r(t)$ denotes the received signal, $s(t)$ denotes the transmitted signal, and n denotes the noise. By adding up the simulated realizations of $r(t) + n$ for N trials, we have $N.s(t)$ and $\sum_{x=1}^{N} n_x$, where n_x denotes the realization for the xth trial. For large N, the latter term starts to vanish. Dividing this sum by N, we are left with the approximation of $s(t)$. Besides, we have conducted simulations to test this approach and the resulting estimation was satisfactory. However, presuming that the noise always has the aforesaid form is not sensical. Hence, we have decided not to utilize this approach in the estimation part of the project.

Kutay Uğurlu 2232841
Arda Yazıcı 2232981

Results/Discussion:

In this part of the report, we want to address the questions that are asked in the last part of our project description document.

First of all, we will start with explaining the way to estimate range and speed by human inspection using spectrograms. At this point, it needs to be mentioned that the algorithms for the estimation of range and speed are already demonstrated in the above parts of Methods/Algorithms section. At this section, our major purpose is to provide insight on these algorithms using spectrograms given in below figures.

For the transmitted sinusoidal signal, the frequency band carrying maximum power content is the same during the whole transmission time. In addition, the same can also be said for the received signal for the whole receiving time. Therefore, we can focus on any time portion we want. For the sake of simplicity, we have chosen the last time instance of the STFT to utilize in the algorithm that we have constructed. After finding the frequencies that carry maximum power content for both of the signals, we used the ratio of them in order to find the relative speed. For instance, in Figure 1, the frequency of the transmitted sinusoidal signal can be observed to be 5 kHz. The received signal has a fundamental frequency of 10 kHz. The ratio between them yields a compression factor of two.

For the transmitted linear chirp signal, we have a slightly different approach. In that case, we have tried to catch the initial frequency of both transmitted and received Doppler shifted signals. The algorithm that we have used so as to catch the initial frequencies is very similar to that was utilized to estimate the delay. Once the initial frequencies are found, we again used the ratio to estimate the speed by utilizing the similar approach that was used in the case of sinusoidal signal. This approach is explained on Figure 3 as follows:

- The initial frequency of transmitted signal is 500 Hz.
- The initial frequency of Doppler shifted signal is 6 kHz.
- The bandwidth value with which these experiments are conducted is 5 kHz.
- Using the formula $\frac{c-v}{c} = \frac{f_{transmitted}+\frac{m}{5}}{f_{doppler}}$, we have arrived at the ratio of $\frac{500+\frac{1000}{5}}{6000} = \frac{1}{4}$.
- This yields $v = 0.75c$.

Now, we want to present the numerical results of estimation for 100 Monte Carlo trials.

Signal Type	Real Speed (c)	Real Delay (sec)	Estimated Speed (c)	Estimated Delay (sec)
Linear Chirp	0.75	3.00	0.73	3.00
Linear Chirp	0.50	2.00	0.51	2.01
Sinusoidal	0.50	3.00	0.49	3.00

The above-mentioned approach can also be verified on the following figures.

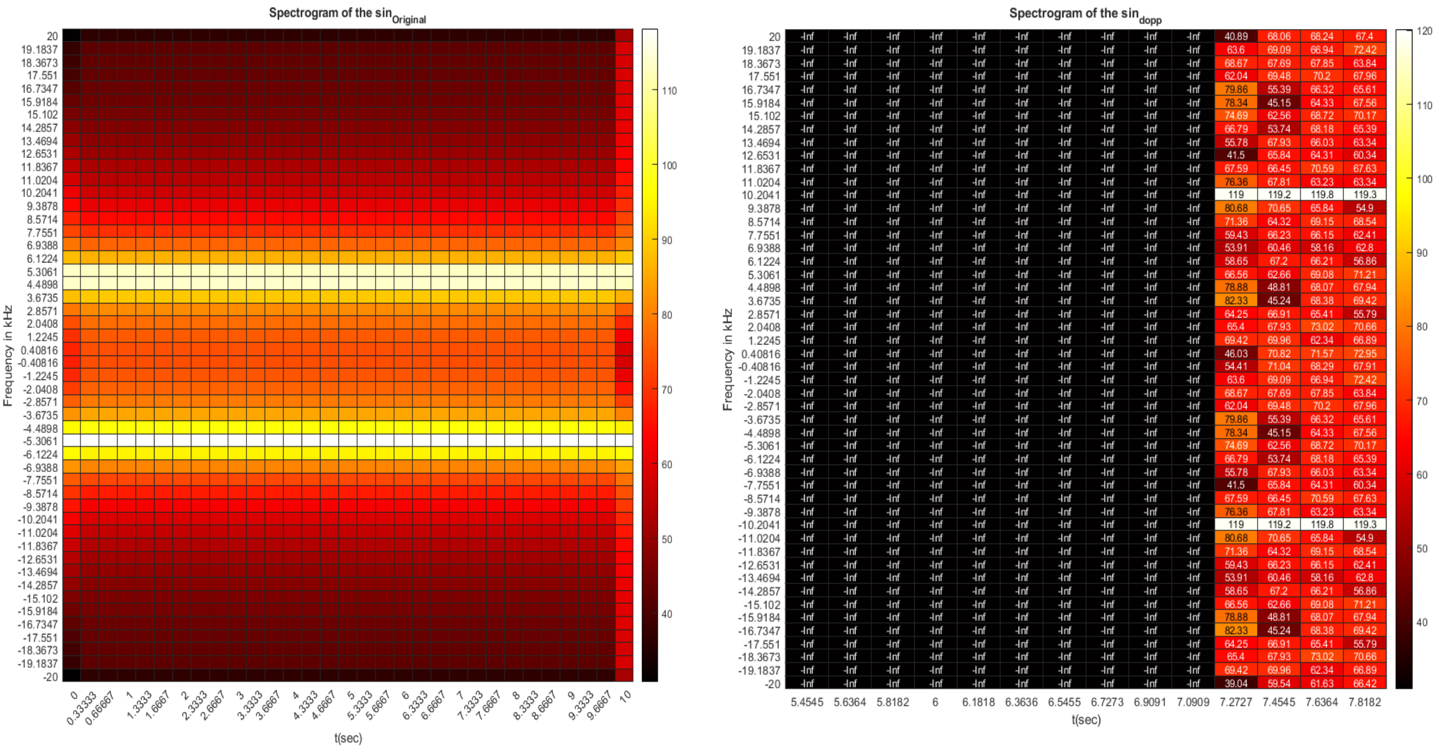

Figure 1. 5 kHz Sinusoidal Signal transmitted from a source traveling with 0.5c speed

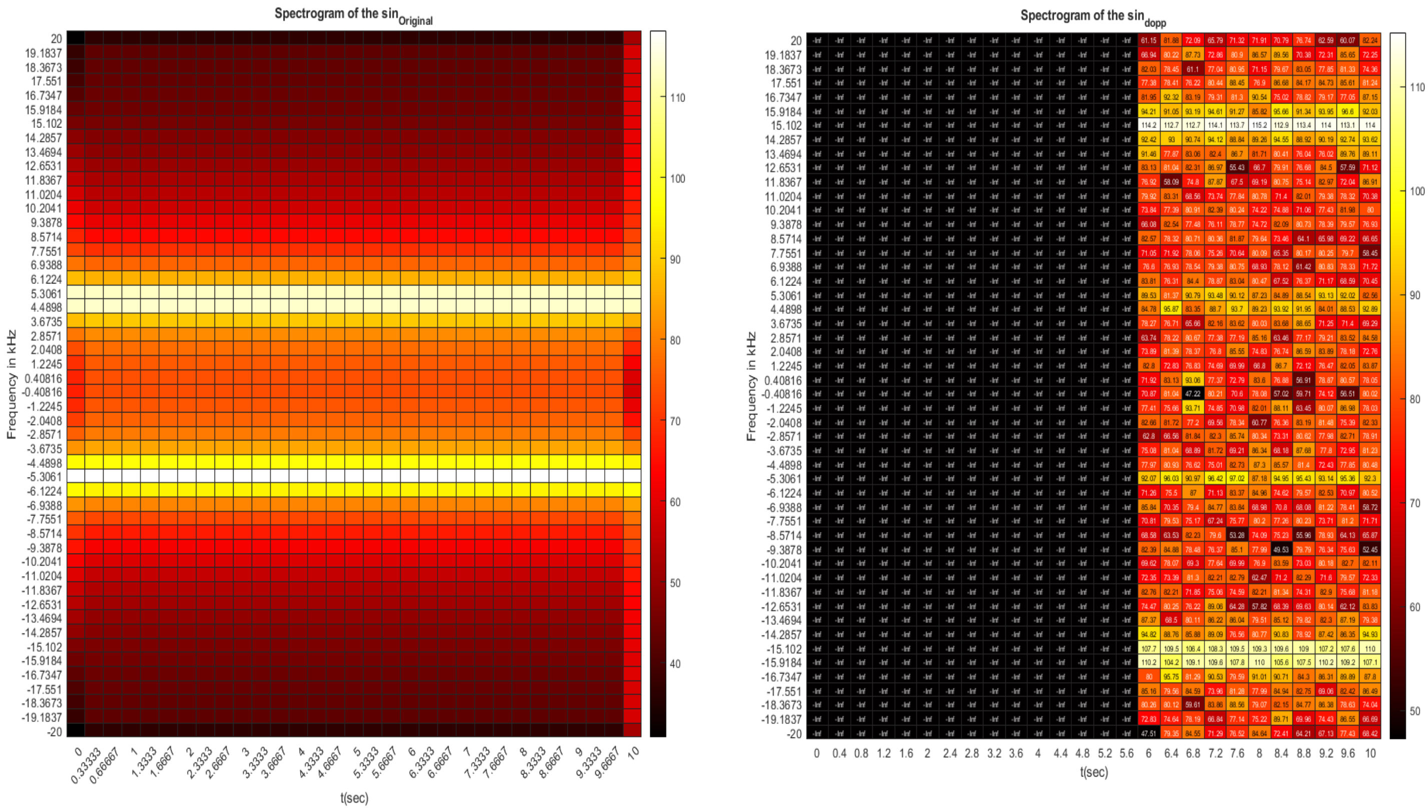

Figure 2. 5 kHz Sinusoidal Signal transmitted from a source traveling with 0.66c speed

Kutay Uğurlu 2232841
Arda Yazıcı 2232981

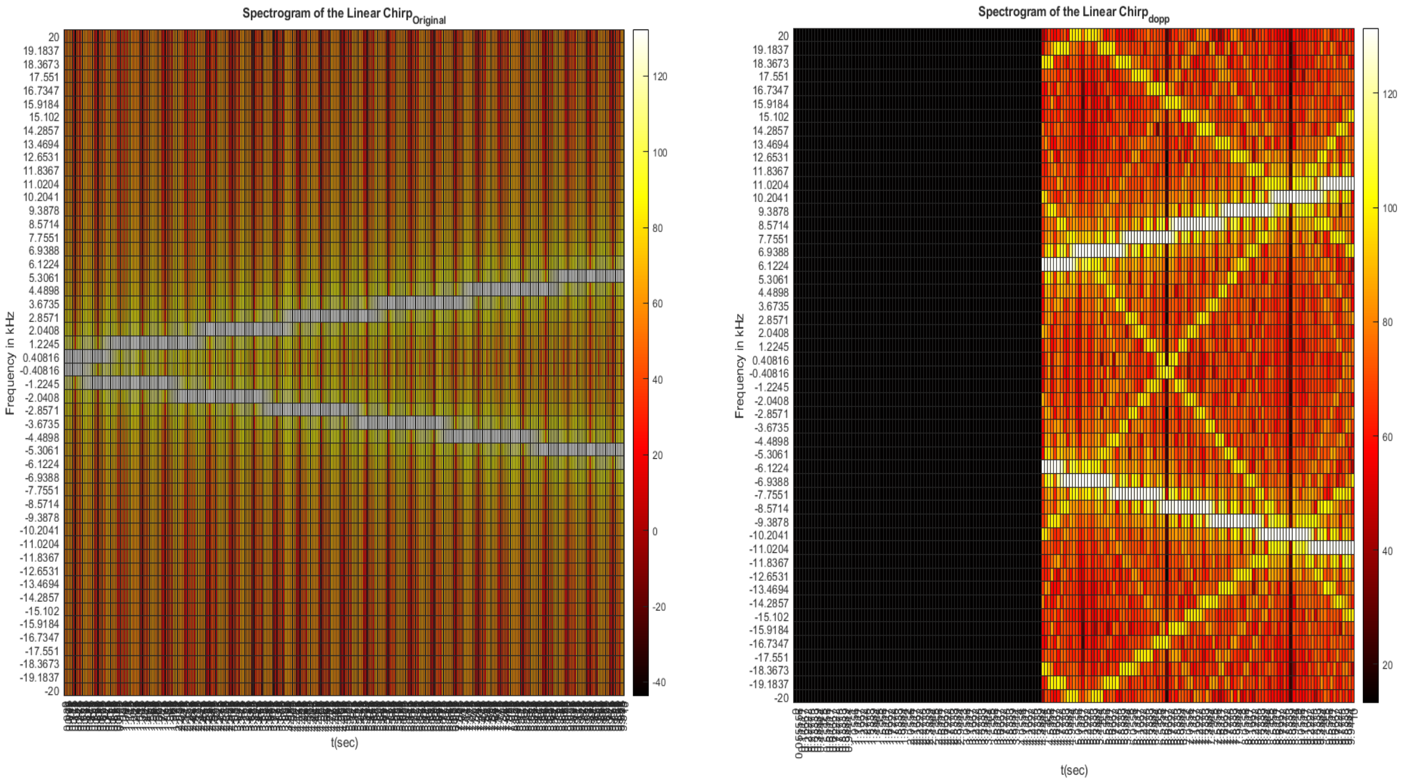

Figure 3. Linear Chirp Signal transmitted from a source traveling with 0.75c speed

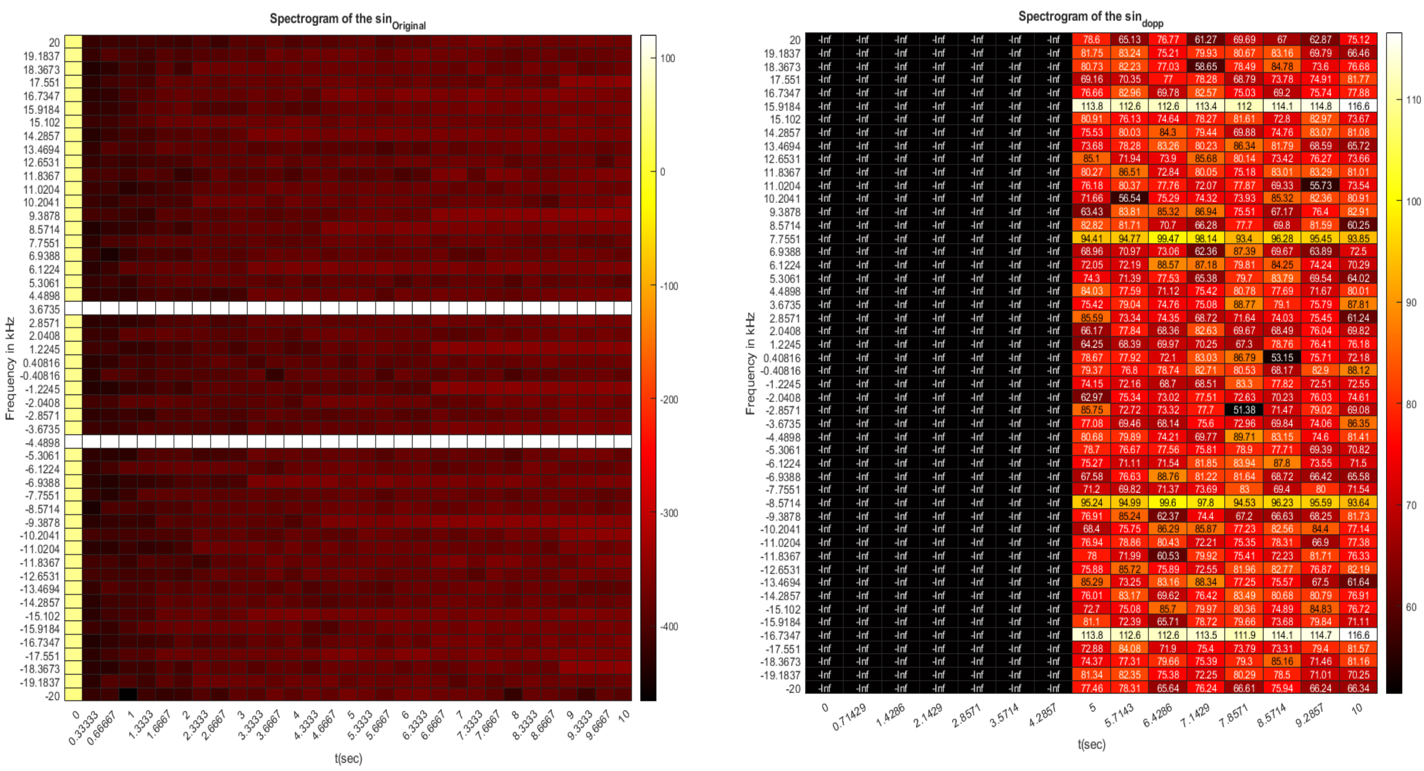

Figure 4. Sinusoidal Signal transmitted from a source traveling with 0.75c speed

Conclusion:

In the second part of the term project, we have coded up signal generator functions, a MATLAB script that computes Short Time Fourier Transform of a given signal, and another MATLAB script that generates the spectrogram for the generated signals. Besides, we have implemented the estimation algorithm script and visualization of the results in a comparative manner.

If the grader of this report wants to investigate different scenarios, the scripts containing the algorithm, visualization and simulation routines are already provided in the submission folder with some of the figures used in this report.