AMATH582 Hw1: Tracking Submarine

Xiangyu Gao

January 27, 2020

Abstract

We are hunting for a submarine in the Puget Sound, which emits an unknown acoustic frequency that we need to detect. The submarine is recorded using noisy acoustic data obtained over a 24-hour period in half-hour increments. In this report, we try to locate the moving submarine and find its trajectory from the noisy acoustic data.

1 Introduction and Overview

We try to locate the submarine and find its trajectory using the acoustic signature, and also identify the acoustic admissions of this new class of submarine. The recorded acoustic data contains 49 columns of data for measurements over a 24-hour span at half-hour increments in time. With this data, we did three things as below:

- Through averaging of the spectrum, we determined the frequency signature (center frequency) generated by the submarine.
- We filtered the data around the center frequency determined above in order to de-noise the data and determine the path of the submarine. We used plot3 to plot the path once have it.
- We determined where to send the P-8 Orion sub-tracking aircraft by giving the x and y coordinates to follow the submarine.

2 Theoretical Background

The Fourier Transform is an integral transform defined over the entire line $x \in [-\infty, \infty]$. The Fourier transform and its inverse are defined as

$$F(k) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-ikx} f(x) dx \tag{1}$$

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{ikx} F(k) dk \tag{2}$$

Gaussian filter is one of the simplest filters,

$$\mathcal{F}(k) = \exp\left(-\tau \left(k - k_0\right)^2\right) \tag{3}$$

where τ measures the bandwidth of the filter, and k is the wavenumber. The generic filter function $\mathcal{F}(k)$ in this case acts as a low-pass filter since it eliminates high-frequency components in the system of interest. Note that these are highfrequencies in relation to the center-frequency ($k = k_0$) of the desired signal field [1].

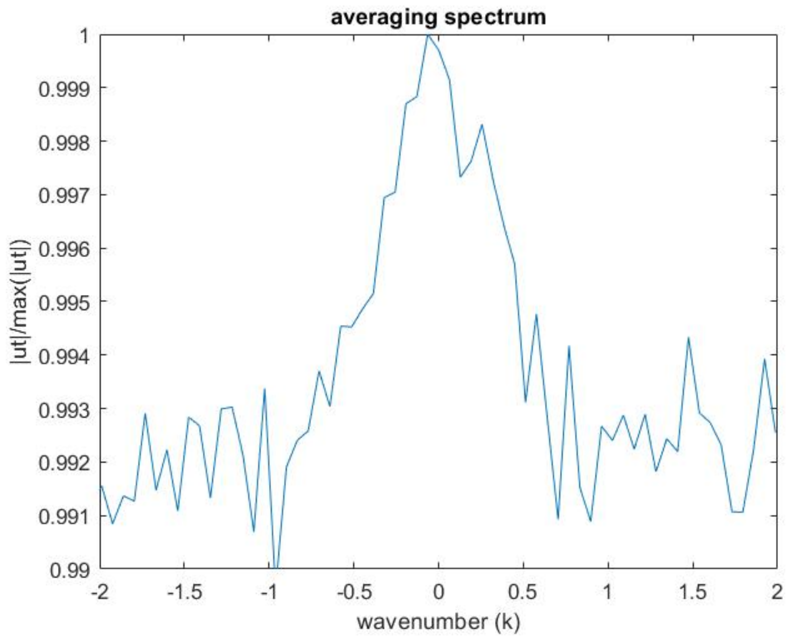

Figure 1: The averaged spectrum of submarine.

3 Algorithm Implementation and Development

The algorithm implementation details are listed below and corresponding MATLAB code is attached in Appendix B:

1. Imports the data as the 262144x49 (space by time) matrix. Reshape imported data as a 4D matrix.
2. Perform FFT on reshaped data along the time domain.
3. Average the obtained spectrums. Plot the averaged spectrum.
4. Find the center frequency generated by submarine.
5. Create Gaussian filter using found center frequency. Filter out the FFT results.
6. Perform IFFT to the FFT result, and find the location of submarine at each time slot.
7. Plot the path of submarine using plot3.
8. Get the numerical x-, y-coordinates to track the submarine according the spatial domain and resolution.

4 Computational Results

The averaged spectrum of the FFT results of the submarine data is shown in Fig. 1, where we could easily find the center frequency at wavenumber $k_0 = -0.1282$. Based on k_0, we create a Gaussian filter and use it to remove the noisy component from the obtained FFT spectrum. The new spectrum with noise removal is presented in Fig. 2.

We perform the IFFT on the noise-filtered FFT results to find the location of submarine for each time slot. We show the 3D submarine moving path in Fig. 3. Note that the location is represented by its index in original data matrix.

To track the submarine using the sub-tracking aircraft, we obtain the x and y coordinate of the submarine location of each timestamp and transform the index representation to spatial domain representation. The suggested path for aircraft is presented in Fig. 4.

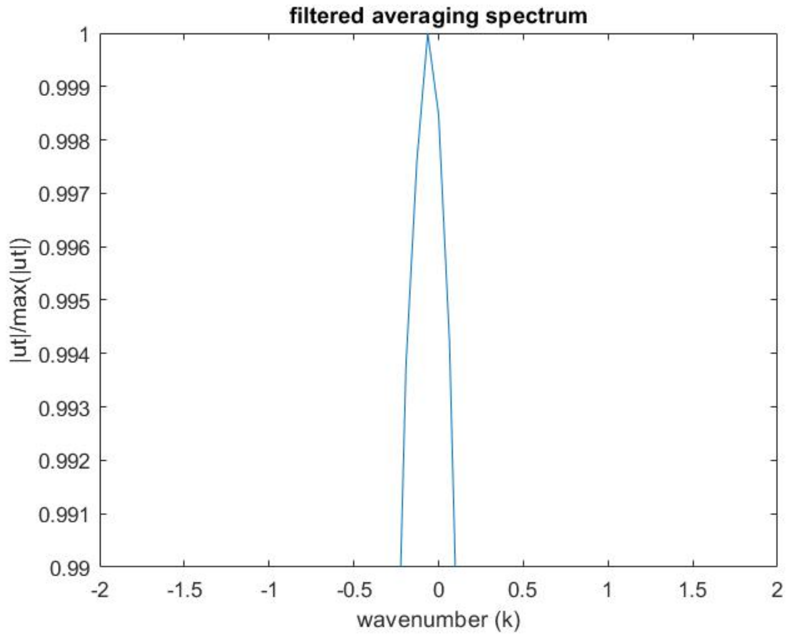

Figure 2: The averaged spectrum with Gaussian filter removing the noisy frequency components.

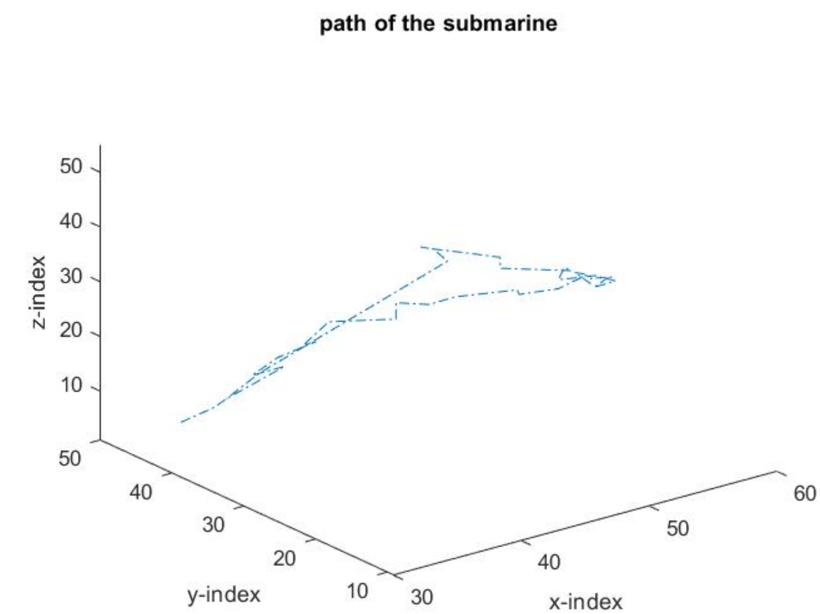

Figure 3: The 3D path of the submarine. The location of submarine is represented by index of the initial data martix.

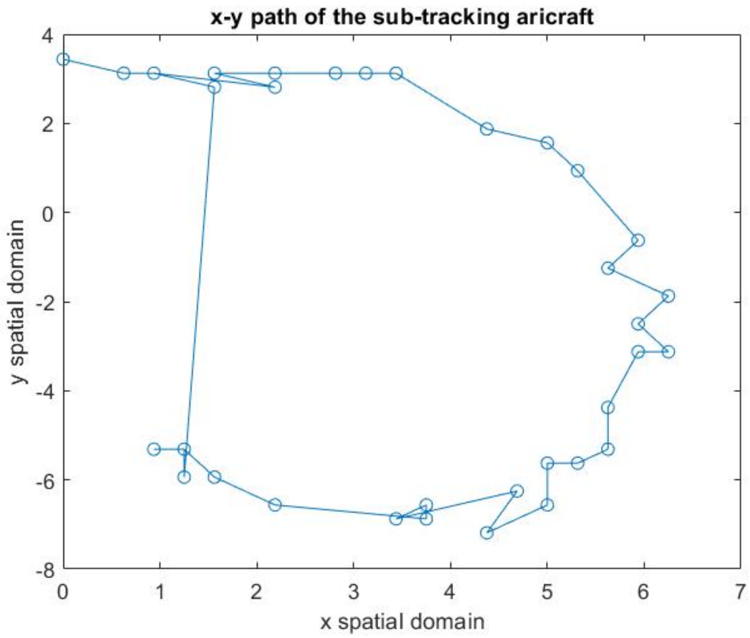

Figure 4: The tracking path of the aircraft. We present the x and y coordinates by spatial domain.

5 Summary and Conclusions

In this report, the location and path of a submarine is successfully found by determining its center frequency and de-noising the data in frequency domain. We also suggest the tracking path for aircraft at the end.

References

[1] Jose Nathan Kutz. *Data-driven modeling & scientific computation: methods for complex systems & big data*. Oxford University Press, 2013.

Appendix A MATLAB Functions

The important MATLAB functions and thier implementation explanations are listed here:

- `y = fft(X,N,DIM)` applies the fft operation across the dimension DIM.
- `y = ifft(X,N,DIM)` returns the inverse discrete Fourier transform of X across the dimension DIM.
- `[I,J] = ind2sub(SIZ,IND)` returns the arrays I and J containing the equivalent row and column subscripts corresponding to the index matrix IND for a matrix of size SIZ.
- `plot3(X,Y,Z)` where X, Y and Z are three matrices of the same size, plots several lines obtained from the columns of X, Y and Z.
- `y = linspace(x1,x2,n)` returns a row vector of `n` evenly spaced points between `x1` and `x2`.
- `[X,Y] = meshgrid(x,y)` returns 2-D grid coordinates based on the coordinates contained in the vectors `x` and `y`. X is a matrix where each row is a copy of `x`, and `Y` is a matrix where each column is a copy of `y`. The grid represented by the coordinates `X` and `Y` has `length(y)` rows and `length(x)` columns.

Appendix B MATLAB Code

The MATLAB codes for Problem 1 3 are shown here.

```
% Clean workspace
clc
clear
close all

% Imports the data as the 262144x49 (space by time) matrix called subdata
load 'C:\Users\Xiangyu Gao\Downloads\subdata\subdata.mat';

%% Problem 1
L = 10; % spatial domain
n = 64; % Fourier modes

% reshape subdata
Un = zeros(n, n, n, 49);
for j=1:49
    Un(:,:,:,j)=reshape(subdata(:,j),n,n,n);
end

% % Visualization
% x2 = linspace(-L,L,n+1); x = x2(1:n); y =x; z = x;
% [X,Y,Z]=meshgrid(x,y,z);
% for j=1:49
%     M = max(abs(Un(:, :, :, j)),[],'all');
%     close all
%     isosurface(X,Y,Z,abs(squeeze(abs(Un(:, :, :, j))))/M,0.7);
%     axis([-10 10 -10 10 -10 10])
%     grid on
%     drawnow
%     pause(1)
% end

% FFT on reshaped data
Un_fft = fftshift(fft(Un, 64, 4));
Un_fft_avg = squeeze(sum(abs(Un_fft), [1 2 3]));
Un_fft_avg = Un_fft_avg / max(Un_fft_avg);

% Plot averaged spectrum
ks = (2*pi/49)*[0:(n/2-1) -n/2:-1];
ks = fftshift(ks);
figure(1)
plot(ks, Un_fft_avg)
axis([-4, 4, 0.99, 1])
xlabel('wavenumber (k)')
ylabel('|ut|/max(|ut|)')
title('averaging spectrum')

% Find the center frequency generated by submarine
[~, I] = max(Un_fft_avg)
ks(I)
```

Listing 1: MATLAB code for Problem 1.

```matlab
%% Problem 2
% Gaussian filter
filter = exp(-0.3*(ks-ks(I)).^2);
filter = reshape(filter, 1, 1, 1, n);
Un_fft_filt = filter .* Un_fft;

% plot filtered data
Un_fft_filt_avg = squeeze(sum(abs(Un_fft_filt), [1 2 3]));
Un_fft_filt_avg = Un_fft_filt_avg / max(Un_fft_filt_avg);

figure(2)
plot(ks, Un_fft_filt_avg)
axis([-4, 4, 0.99, 1])
xlabel('wavenumber (k)')
ylabel('|ut|/max(|ut|)')
title('filtered averaging spectrum')

% ifftshift, ifft
Un_filt = ifft(ifftshift(Un_fft_filt), 49, 4);

% determine the path of submarine
P_vec = [];
for i = 1:49
    [~, P] = max(abs(squeeze(Un_filt(:,:,:,i))), [], 'all', 'linear');
    P_vec = [P_vec, P];
end

% tranform liner indices to subscripts
sz = [n, n, n];
[I1, I2, I3] = ind2sub(sz, P_vec);

% % Visualization
% x2 = linspace(-L,L,n+1); x = x2(1:n); y =x; z = x;
% [X,Y,Z]=meshgrid(x,y,z);
% for j=1:49
%     M = max(abs(Un_filt(:, :, :, j)),[],'all');
%     close all
%     isosurface(X,Y,Z,abs(squeeze(abs(Un_filt(:, :, :, j))))/M,0.7);
%     axis([-10 10 -10 10 -10 10])
%     grid on
%     drawnow
%     pause(1)
% end

% plot the path of the submarine
figure(3)
plot3(I1, I2, I3, '-.')
xlabel('x-index'), ylabel('y-index'), zlabel('z-index')
axis([35, 55, 9, 45, 5, 55])
title('path of the submarine')
```

Listing 2: MATLAB code for Problem 2.

```
%% Problem 3
% Get the numerical x-, y-coordinates to track the submarine
x2 = linspace(-L,L,n+1); x = x2(1:n); y =x; z = x;
figure(4)
plot(x(I1), x(I2), '-o')
xlabel('x spatial domain'), ylabel('y spatial domain')
title('x-y path of the sub-tracking aricraft')
```

Listing 3: MATLAB code for Problem 3.