Tracking of A Submarine

AMATH 482 HW1

Xuanhui Chen

Abstract

This report aims to detect a submarine and find its trajectory by analyzing the noisy acoustic data. Fast Fourier Transform (FFT) is used heavily as a fundamental mathematical tool throughout this paper. The frequency signature generated by the submarine can be found by averaging the frequency spectrum using the FFT. A Gaussian filter is then used to filter out the white noise generated by the reflection from other geographical objects, and we can easily track the path of the submarine and find its location.

I. Introduction and Overview

We are tracing the track of a submarine in Puget Sound that has new technology of emitting unknown acoustic frequencies. The submarine is moving, and its location and trajectory are to be determined. We obtained the spatial data that has 49 columns for measurements over a 24-hour period in half-hour increments, meaning we have 49 signals with respect to time. In order to detect the moving submarine surrounded by other white noise, we need to remove these noise and determine the path and location of this submarine.

We will use the FFT approach in this report. First, the frequency signature (central frequency) generated by the submarine is determined by averaging in the frequency space. Second, the data is denoised by filtering out the distracting white noise away from the center frequency. Lastly, we will determine the location and path of the submarine.

II. Theoretical Background

Fourier Transform and Fourier Series

Given a function $f(x)$ for $\mathrm{x} \in [-\mathrm{L}, \mathrm{L}]$. Since the frequencies are periodic, we can write the discontinuous function $f(x)$ as a infinite sum of the combination of $\sin(kx)$ and $\cos(kx)$, given that $\mathrm{k} \in \mathbb{Z}^+$:

$$\mathrm{f(x)} = \frac{a_0}{2} + \sum_{k=1}^{\infty} \left(a_k \cos\left(\frac{\pi k x}{L}\right) + \mathrm{b}_k \sin\left(\frac{\pi k x}{L}\right) \right), \quad x \in [-L, L]. \quad (1)$$

Fourier Coefficients

The Fourier Coefficients can be derived from the formula of the Fourier Series:

$$a_k = \frac{1}{L} \int_{-L}^{L} f(x) \cos\left(\frac{\pi k x}{L}\right) dx, \quad (2)$$

$$b_k = \frac{1}{L}\int_{-L}^{L} f(x)\sin\left(\frac{\pi kx}{L}\right)dx, \quad (3)$$

$$a_0 = \frac{1}{2L}\int_{-L}^{L} f(x)\,dx. \quad (4)$$

Fourier Transform

The *Fourier Transform* and its inverse of $f(x)$ are:

$$\hat{f}(k) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} f(x)e^{-ikx}\,dx, \quad (5)$$

$$f(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} \hat{f}(k)e^{ikx}\,dk, \quad (6)$$

respectively, where e^{ikx} can be written in the form of $\sin(kx)$ and $\cos(kx)$. The *Fourier Transform* takes a spatial function of x, and coverts it to a function of frequencies, k.

Gaussian Filter

The Gaussian function works as a filter with τ, which determines the width of the filter, and k_0 determines the center of the filter:

$$F(k) = e^{-\tau(k-k_0)^2}, \quad (7)$$

which is used to filter out the white noise.

III. Algorithm Implementation and Development

The algorithm implemented can be separated into four parts:

- Loading raw data and forming the initial spectral plot:
  - The original spatial and frequency domain were set up. Spatial and spectral meshgrid is created, where the reshaped into $64\times64\times64$ points. (See **Appendix B** Part 1a)
  - The isosurface plot of raw spatial data is created and shown in Figure 1. (See **Appendix B** Part 1b)
- Averaging the spectrum and find the center frequency:
  - Use the built-in function *fftn()* and *fftshift()* of MATLAB to find the average of the spectrum. (See **Appendix A** and **Appendix B** Part 2a)
  - The center frequency is found at k_0. (See **Appendix B** Part 2b)
- Filter the data:
  - The Gaussian filter is applied with the chosen $\tau = 0.2$ to eliminate the

white noise around the center frequency. (See **Appendix B** Part 3)

- Generate the path and find the final location of the submarine:
    - The filtered data then applies the FFT and IFFT back and forth, which are built-in functions of MATLAB. The peak of each signal is then recorded in a matrix, and the final location coordinate of the submarine is found in the last row of that matrix. The trajectory is then plotted. (See **Appendix B** Part 4)

IV. Computational Results

For the center frequency, it is found at:

$$k_c = [5.3407, -6.9115, 2.1991].$$

After the Gaussian filter has been applied based on the center frequency, we found the peaks of each signal and recorded them in a matrix, and the trajectory (see **Table 1**) of the submarine is plotted as Figure 2 shows. The final location of the submarine is then found in the last row of the matrix, which is shown as the coordinates of (x,y,z):

$$[x, y, z] = [-5, 0.9375, 6.5625].$$

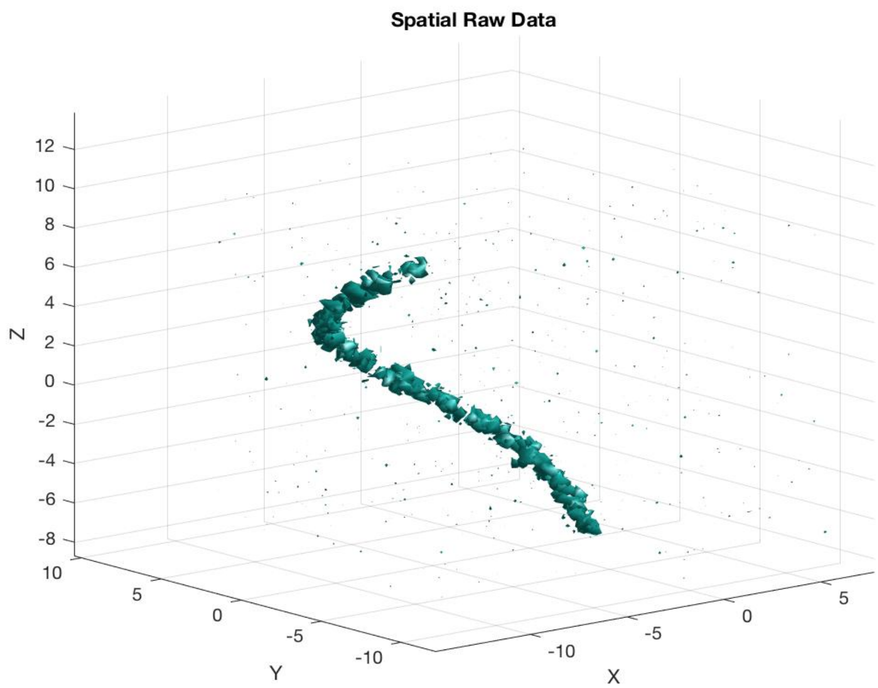

Figure 1: Isosurface plot of the spatial raw data (unshifted and not yet averaged).

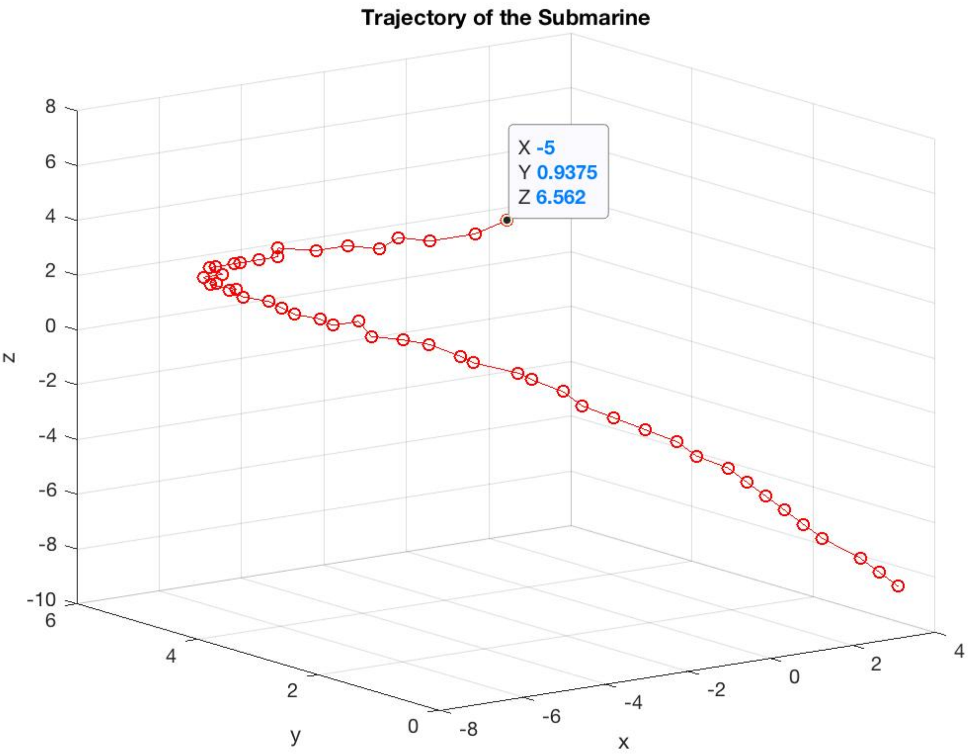

Figure 2: Trajectory of the submarine, the final location is on the top shown as a black point with coordinates.

1	3.1250	0	-8.1250
2	3.1250	0.3125	-7.8125
3	3.1250	0.6250	-7.5000
4	3.1250	1.2500	-7.1875
5	3.1250	1.5625	-6.8750
6	3.1250	1.8750	-6.5625
7	3.1250	2.1875	-6.2500
8	3.1250	2.5000	-5.9375
9	3.1250	2.8125	-5.6250
10	2.8125	3.1250	-5.3125
11	2.8125	3.4375	-5
12	2.5000	3.7500	-4.6875
13	2.1875	4.0625	-4.3750
14	1.8750	4.3750	-4.0625
15	1.8750	4.6875	-3.7500
16	1.5625	5	-3.4375
17	1.2500	5	-3.1250
18	0.6250	5.3125	-2.8125
19	0.3125	5.3125	-2.5000
20	0	5.6250	-2.1875
21	-0.6250	5.6250	-1.8750
22	-0.9375	5.9375	-1.8750
23	-1.2500	5.9375	-1.2500
24	-1.8750	5.9375	-1.2500
25	-2.1875	5.9375	-0.9375
26	-2.8125	5.9375	-0.6250
27	-3.1250	5.9375	-0.3125
28	-3.4375	5.9375	0
29	-4.0625	5.9375	0.3125
30	-4.3750	5.9375	0.6250
31	-4.6875	5.6250	0.9375
32	-5.3125	5.6250	1.2500
33	-5.6250	5.3125	1.5625
34	-5.9375	5.3125	1.8750
35	-5.9375	5	2.1875
36	-6.2500	5	2.5000
37	-6.5625	4.6875	2.8125
38	-6.5625	4.3750	3.1250
39	-6.8750	4.0625	3.4375
40	-6.8750	3.7500	3.7500
41	-6.8750	3.4375	4.0625
42	-6.8750	3.4375	4.3750
43	-6.8750	2.8125	4.6875
44	-6.5625	2.5000	5
45	-6.2500	2.1875	5
46	-6.2500	1.8750	5.6250
47	-5.9375	1.5625	5.6250
48	-5.3125	1.2500	5.9375
49	-5	0.9375	6.5625

Table 1: Table of the 49 coordinates (x,y,z) to follow the submarine

V. Summary and Conclusion

By using the FFT approach, we firstly found the center frequency through averaging of the 49 spectra. Then the Gaussian filter was applied to eliminating the white noise around the center frequency in this spatial domain. The trajectory and location of the submarine is finally obtained by searching for the peak of each filtered signal. So we should send our P-8 Poseidon subtracking aircraft to the coordinate:

$$[-5, 0.9375, 6.5625].$$

Appendix A: MATLAB functions used and brief implementation explanation

- **linspace(X1, X2, N)**: generate N points between X1 and X2.
- **fftshift(X)**: shift zero-frequency component to center of spectrum.
- **[X,Y,Z] = meshgrid(xgv,ygv,zgv):** replicates the grid vectors xgv, ygv, zgv to produce the coordinates of a 3D rectangular grid (X, Y, Z).
- **reshape(X,M,N,P,...)**: returns an N-D array with the same elements as X but reshaped to have the size M-by-N-by-P-by-....
- **isosurface(X,Y,Z,V,ISOVALUE)**: computes isosurface geometry fordata V at isosurface value ISOVALUE.
- **fftn(X)**: returns the N-dimensional discrete Fourier transform of the N-D array X.
- **[I1,I2,I3] = ind2sub(SIZ,IND)**: returns N subscript arrays I1,I2,I3.
- **ifftn(F)**: returns the N-dimensional inverse discrete Fourier transform of the N-D array F.

Appendix A: MATLAB Codes

```
% Clean workspace
clear all; close all; clc

load subdata.mat % Imports the data as the 262144x49 (space by time) matrix called↙
subdata 5

% Part 1a
L = 10; % spatial domain
n = 64; % Fourier modes
x2 = linspace(-L,L,n+1);
x = x2(1:n);
y = x;
z = x;
k = (2*pi/(2*L))*[0:(n/2 - 1) -n/2:-1];
ks = fftshift(k);

[X,Y,Z]=meshgrid(x,y,z);
[Kx,Ky,Kz]=meshgrid(ks,ks,ks);

% Part 1b: Original Spatial and Spectral Resolution
figure(1)
for j=1:49
    Un(:,:,:)=reshape(subdata(:,j),n,n,n);
    M = max(abs(Un),[],'all');
    %close all;
    isosurface(X,Y,Z,abs(Un)/M,0.7)
    axis([-20 20 -20 20 -20 20]), grid on, drawnow
    pause(0.00001)
end
title("Spatial Raw Data");
xlabel('X');ylabel('Y');zlabel('Z');

% Part 2a: Average of the spectrum
Uave = zeros(n,n,n);
for j = 1:49
    Un(:,:,:)=reshape(subdata(:,j),n,n,n);
    Utn = fftn(Un);
    Uave = Uave + Utn;
end
Uave = fftshift(Uave)/49;

% Part 2b: Find the center frequency
[Y,index] = max(Uave(:));
[a,b,c] = ind2sub(size(Uave), index);
center_f = [ks(b), ks(a), ks(c)]; % center frequency

% draw the denoised data
% isosurface(Kx,Ky,Kz,abs(Uave)/max(abs(Uave(:))),0.7)

% Part 3: Filter
tau = 0.2;
filter = exp(-tau*((Kx-center_f(1)).^2 + (Ky-center_f(2)).^2 + ...
    (Kz-center_f(3)).^2));
track = zeros(49,3);

% Part 4: Path
for j = 1:49
```

```
    Un(:,:,:)=reshape(subdata(:,j),n,n,n);
    Unt = fftn(Un);
    Unft = filter .* fftshift(Unt); % filter out the center
    Unf = ifftn(Unft);

    [Y,index] = max(Unf(:));
    [a,b,c] = ind2sub(size(Unf), index);
    coordinates = [x(b), y(a), z(c)];
    track(j,:) = coordinates;
end
figure(2)
plot3(track(:,1), track(:,2), track(:,3), 'r-o'), grid on
title("Trajectory of the Submarine");
xlabel('x');ylabel('y');zlabel('z');
```

Reference

1. Course note.
2. MATLAB help section.