

AMATH 482 HW1: Tracking a Submarine

Tiffany Tang

Department of Applied Mathematics

University of Washington

leqitang@uw.edu

Abstract

In this paper, Fast Fourier Transform(FFT) is used to find the center frequency and location of a moving submarine by analyzing noisy acoustic data. Each recorded time's frequency spectrum is averaged to reduce the white noise and obtain the center frequency by applying FFT. I obtained the path of the submarine in time dimension after denoise the data with a Gaussian filter.

1 Introduction

Acoustic data detection has been widely used in security and monitoring. A new submarine technology also utilizes acoustic data to emit an unknown acoustic frequency. By applying FFT, the acoustic data is analyzed to determine a submarine's center frequency through averaging the spectrum. Besides, the acoustic data is denoised by a Gaussian filter to locate its path. Once figuring out the trajectory of the submarine by receiving the acoustic data, P-8 Poseidon subtracking aircrafts would be sent to track submarines.

2 Theoretical Background

Fourier Series:

Any function in the interval $x \in [-L, L]$ can be expressed in terms of linear combination of harmonically sinusoidal functions $\sin(2\pi fx + \phi)$, $\cos(2\pi fx + \phi)$ of different amplitudes(A), frequencies(f) and phases(ϕ), known as Fourier series. We write the periodic function as the infinite sum of these sines and cosines:

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^n (a_n \cos(kx) + b_n \sin(kx)), x \in [-L, L] \quad (1)$$

Where the coefficient a_0 , a_n , and b_n are given by the formulae

$$a_0 = \frac{1}{2L} \int_{-L}^L f(x) dx \quad a_n = \frac{1}{L} \int_{-L}^L f(x) \cos \frac{n\pi}{L} x dx \quad b_n = \frac{1}{L} \int_{-L}^L f(x) \sin \frac{n\pi}{L} x dx$$

Using exponential form of the $\cos(x) = \frac{e^{ix} + e^{-ix}}{2}$ and $\sin(x) = \frac{e^{ix} - e^{-ix}}{2}$ in equation

(1) and coefficient formulae, we can express $f(x)$ in complex exponential form.

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{\frac{(ik\pi)x}{L}}, x \in [-L, L] \quad \text{where } c_k = \frac{1}{2L} \int_{-L}^L f(x) e^{-\frac{(ik\pi)x}{L}} dx, k \in \mathbb{Z}$$

Fourier Transform:

The Fourier Transform is an integral transform defined over the interval $x \in [-L, L]$.

Fourier transform and its inverse are defined as

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

From the definition, we set that the transform is over the area $x \in [-\infty, \infty]$ and our special domain is in the definite main $x \in [-L, L]$.

Fast Fourier Transform:

The Fast Fourier Transform (FFT) is the algorithm that transform a function of time into a function of frequency. Comparing to Discrete Fourier Transform (DFT) which has a computing complexity $O(N^2)$, FFT has its advantage that it has relatively low operation complexity of $O(N \log N)$.

The FFT algorithm performs more efficiently with large data. Also, MATLAB has a built-in function to perform the FFT for data analysis.

Averaging:

The signal data are filled with large amount of white noise which prevent us from figuring out the signals. Since the white noise could be modeled in a normal distribution and added to each Fourier component of the spectrum. We could average over many realizations in frequency space. Therefore, the noise from each realization will cancel out and we could get the central frequency.

Filtering:

There are many ways to filter the signal data in order to denoise and better detect the signal. A spectral filter around the central frequency is used in this project to remove undesired frequencies and most of the white noise. The Gaussian filter is used here:

$$F(k) = \exp(-\tau(k - k_0)^2)$$

This is a Gaussian function with τ determining the width of the filter and the constant k_0 is the target signal.

3 Algorithm Implementation and Development

In order to find the center frequency generated by the submarine and determine the path of the submarine's movement, four components have been conducted with algorithm implemented.

1. Data Acquisition and Setup

After the acoustic data within 49 time points is loaded, the frequency components have been setup. Since the Fast Fourier Transform assumes 2π periodic signals and $[-L, L]$ is the domain of input data, the frequencies are rescaled by $\frac{2\pi}{2L}$. The 3-

D grid $[X, Y, Z]$ in spatial-time discretized domain and the 3-D grid $[Kx, Ky, Kz]$ in frequency domain have been setup.

2. Spectral averaging

Averaging method with 49 realizations has been applied to average the acoustic data. I applied the 3-D Fourier transform (fft3 function) to transform the data from spatial domain to frequency domain and rearrange the data by shifting the zero-frequency component to the center of the array (fftshift function). I got the sum of the data from 49 times of the loop and then average the sum to offset noise. I took the absolute

value of the averaged data, since the original results in the matrix contain both real and imaginary parts.

3. Filtering and denoising

First, the Gaussian filter is set with replacing the k_0 with the center frequency values in the 3-D. A 3×49 matrix is created to record the 3-D locations in the x, y, and z-axis in each time's measurement. For each time point, I applied the 3-D Fourier transform (fft3 function) on the acoustic data and rearrange (fftshift function) it to the center at zero frequency. Then I applied the filter to the signal data in the frequency domain. Inverse 3-D Fourier transform (ifft3 function) is used to acquire the signal data in the time domain. Then I found the indexes for peaks and obtained the 3-D locations. Lastly, I used the plot3 function to plot all of the coordinates recorded to figure out the submarine trajectory.

4. Obtaining the Final Locations

By finding the last column of the locations that record all 49 positions of the submarine, I get the final location of the submarine. Since the subtracking aircraft cannot go underwater, I give the final x and y coordinates for the submarine.

4 Computational Results

The center frequency $K_0 = [5.3407, -6.9115, 2.1991]$

The coordinates of the submarine at each instance of time were obtained and showed in Table 1. The figure 1 shows the trajectory of the submarine.

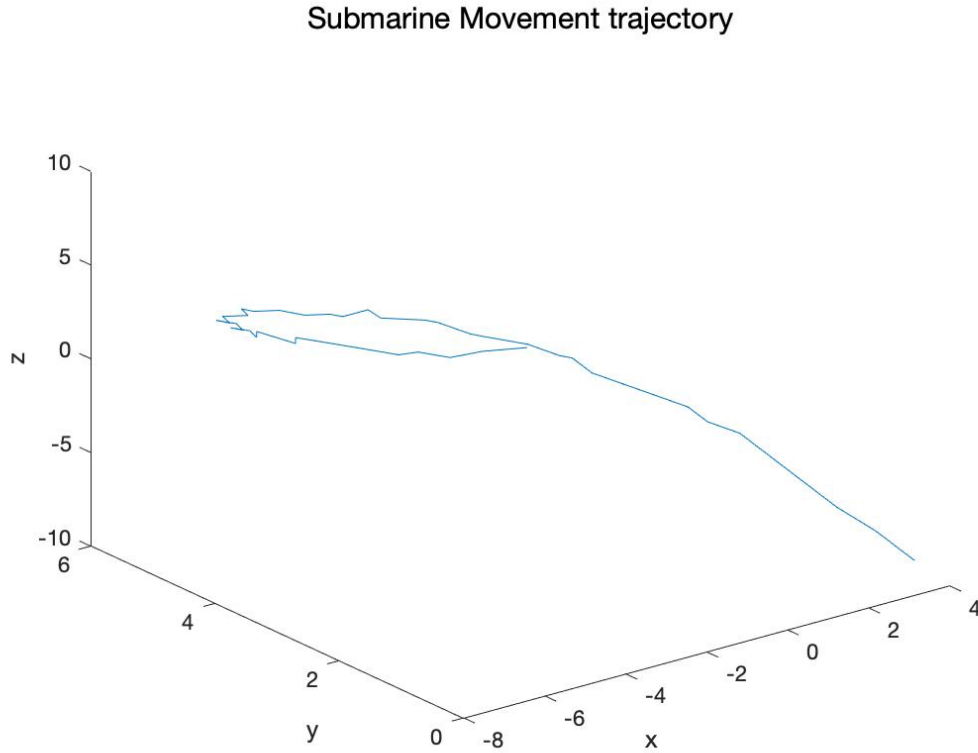


Figure 1: The trajectory of the submarine over a 24-hour period in half-hour increments.

Time	x	y	z	Time	x	y	z
1	3.125	0	-8.125	26	-2.8125	5.9375	-0.625
2	3.125	0.3125	-7.8125	27	-3.125	5.9375	-0.3125
3	3.125	0.625	-7.5	28	-3.4375	5.9375	0
4	3.125	1.25	-7.1875	29	-4.0625	5.9375	0.3125
5	3.125	1.5625	-6.875	30	-4.375	5.9375	0.625
6	3.125	1.875	-6.5625	31	-4.6875	5.625	0.9375
7	3.125	2.1875	-6.25	32	-5.3125	5.625	1.25
8	3.125	2.5	-5.9375	33	-5.625	5.3125	1.5625
9	3.125	2.8125	-5.625	34	-5.9375	5.3125	1.875
10	2.8125	3.125	-5.3125	35	-5.9375	5	2.1875
11	2.8125	3.4375	-5	36	-6.25	5	2.5
12	2.5	3.75	-4.6875	37	-6.5625	4.6875	2.8125
13	2.1875	4.0625	-4.375	38	-6.5625	4.375	3.125
14	1.875	4.375	-4.0625	39	-6.875	4.0625	3.4375
15	1.875	4.6875	-3.75	40	-6.875	3.75	3.75
16	1.5625	5	-3.4375	41	-6.875	3.4375	4.0625
17	1.25	5	-3.125	42	-6.875	3.4375	4.375
18	0.625	5.3125	-2.8125	43	-6.875	2.8125	4.6875
19	0.3125	5.3125	-2.5	44	-6.5625	2.5	5
20	0	5.625	-2.1875	45	-6.25	2.1875	5
21	-0.625	5.625	-1.875	46	-6.25	1.875	5.625
22	-0.9375	5.9375	-1.875	47	-5.9375	1.5625	5.625
23	-1.25	5.9375	-1.25	48	-5.3125	1.25	5.9375
24	-1.875	5.9375	-1.25	49	-5	0.9375	6.5625
25	-2.1875	5.9375	-0.9375				

Table 1: the x, y and z coordinates at each of 49 time points over a 24-hour span

In order to send my P-8 Poseidon subtracking aircraft, the last location of the submarine traveling over the 24-hour period is needed to obtain. The location at the 49th instance was found at

$$[x, y] = [-5.0, 0.9375]$$

5 Summary and Conclusions

To locate the movement of submarine, we use a broad spectrum recording of acoustics to keep track of the location of submarine in 49 time points. Applying the averaging method to average the raw acoustic data which contain white noise, I find the center frequency, which is the max frequency of where the wave hits the submarine. Apart from averaging the data, I used a Gaussian filter to denoise the data, since it is not necessary to assume the mean of noise throughout time is zero within this method. When we know the location of the central frequency, I could locate the trajectory of the moving submarine over a time period.

Depending on different raw data we get, either averaging or various filters could be applied to denoise the noisy data. The denoising methods with the implication of Fourier transform in the data analysis could be used in many natural science research analyses.

Appendix A. MATLAB Functions

1. `y = linspace(x1, x2, n)` returns a row vector of n evenly spaced points between $x1$ and $x2$.
2. `[X, Y, Z] = meshgrid(x, y, z)` returns 3-D grid coordinates defined by the vectors x , y , and z .
3. `B = reshape(A, sz)` reshapes A using size vector sz into B
4. `Y = fftn(X)` returns the multidimensional Fourier transform of an N -D array using a fast Fourier transform algorithm
5. `Y = ifftn(X)` returns the multidimensional discrete inverse Fourier transform of an N -D array using a fast Fourier transform algorithm
6. `Y = fftshift(X)` rearranges a Fourier transform X by shifting the zero-frequency component to the center of the array
7. `[row, col] = ind2sub(sz, ind)` returns the array row and col containing the equivalent row and column subscripts corresponding to the linear indices ind for a matrix of size sz .

Appendix B. Matlab Code

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

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

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); % grid in frequency domain

%% problem 1 - center frequency
Utave = zeros(64, 64, 64)

for j = 1:49
    Un(:,:,j)=reshape(subdata(:,j),n,n,n);
    Utn = fftshift(fftn(Un));
    Utave = Utave + Utn;
end
Utave = abs(Utave./49);
[maxVals, indices1] = max(Utave(:));
[kx0, ky0, kz0] = ind2sub([n,n,n], indices1);
Kx0 = Kx(kx0, ky0, kz0)
Ky0 = Ky(kx0, ky0, kz0)
Kz0 = Kz(kx0, ky0, kz0)
%% problem 2 - filter
locations = zeros(3, 49);
tau = 0.2;
filter = exp(-tau*((Kx-Kx0).^2+ (Ky-Ky0).^2+ (Kz-
Kz0).^2)); % Define the filter
for j = 1:49
    Un(:,:,j)=reshape(subdata(:,j),n,n,n);
    Utn = fftshift(fftn(Un));
    Unft = filter.*Utn; % Apply the filter to the signal
in frequency space
Unf = ifftn(Unft); % the signal in time space

M = max(abs(Unf),[],'all');
indices2 = find(abs(Unf)==M);
```

```

[a, b, c] = ind2sub([n, n, n], indices2);
locations(1, j) = X(a,b,c);
locations(2, j) = Y(a,b,c);
locations(3, j) = Z(a,b,c);
end
figure(1)
plot3(locations(1, :), locations(2, :), locations(3,:),
'ko-', 'Linewidth', 1)
title('Submarine Movement trajectory', 'FontSize', 15)
xlabel('x')
ylabel('y')
zlabel('z')
axis([-10 10 -10 10 -10 10]), grid on
x49 = locations(1, 49);
y49 = locations(2, 49);
z49 = locations(3, 49);
L = sprintf('The 49th location of the submarine is
at %s %d %f.', x49, y49, z49);
%% Problem 3 - get the table of the 49 2-D positions in
each time point
num = [1:49]
l = [locations(1,:);locations(2,:)]
XY_locations = table(num, l)
writetable(XY_locations, 'xy.csv')
type 'xy.csv'

```