POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

HomeWork #1- Radar Imaging

TESI DI LAUREA MAGISTRALE IN
TELECOMMUNICATION ENGINEERING

Author: **Pouya Aminaie**

Professor: Andrea Virgilio Monti-Guarnieri
Academic Year: 2023-24

Contents

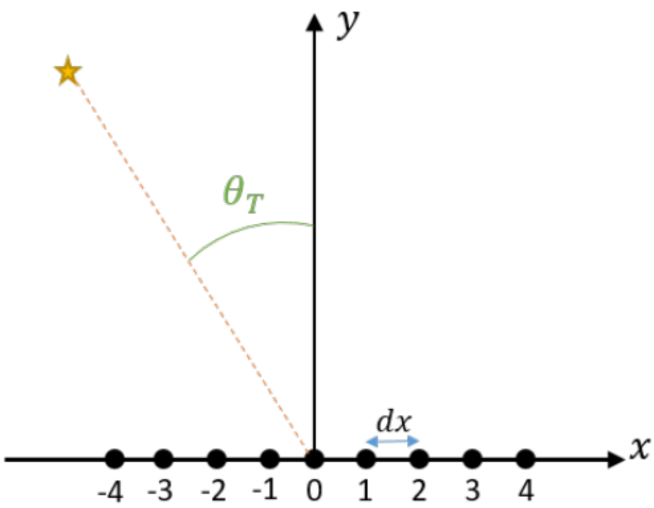
1. Introduction

In this assignment, our primary objective is to ascertain the angular position of a target. This involves each component transmitting a radar signal and capturing the reflected echo from the intended target. Through the analysis of these received echoes, we can precisely determine the angular location of the target.

The system geometry is defined as follows:

In this task, our initial focus is on estimating the angular position of a target using a Uniform Linear Array (ULA). To achieve this, we must design the ULA, comprising nine antennas with appropriate spacing between the elements, positioned around a single target situated at an angular position θ. Each antenna is responsible for transmitting a signal and receiving the echo back with specified characteristics.

$$\text{Sinusoidalsignal } g(t) = e^{j2\pi f_0 t}$$
$$\text{Carrier frequency} = 77 \text{ GHz}$$
resolution of 2 degrees.



The figure above shows the structure of the system.

2. System Model

In this section, our aim is to formulate the model for the signal received by the *n^th^* antenna. Given that the signal undergoes perfect demodulation at the receiver, our objective is to linearize the phase term of the received signal with respect to variable x around the central element. In accordance with the provided question, the signal transmitted by the *n^th^* antenna is:

$$S_{Tx} = g(t) = e^{2j\pi f_0 t}$$

received signal nth antenna with delay (only delayed version at receiver):

$$S_{Rx} = g(t - t_0)$$

Then we apply the delay to the main formula:

$$S_{Rx} = g(t - t_0) = e^{2j\pi f_0 (t - t_0)} = e^{2j\pi f_0 t} \cdot e^{-2j\pi f_0 t_0}$$

However, since we possess the demodulated version of the signal at the receiver, denoted as $(t - t_0)$ due to its origin from the target, it is necessary to multiply it by $e^{-2j\pi f_0 t}$ to eliminate the $e^{2j\pi f_0 t}$ component.

$$S_{Rx} = e^{2j\pi f_0 t} \cdot e^{-2j\pi f_0 t_0} \times e^{2j\pi f_0 t}$$

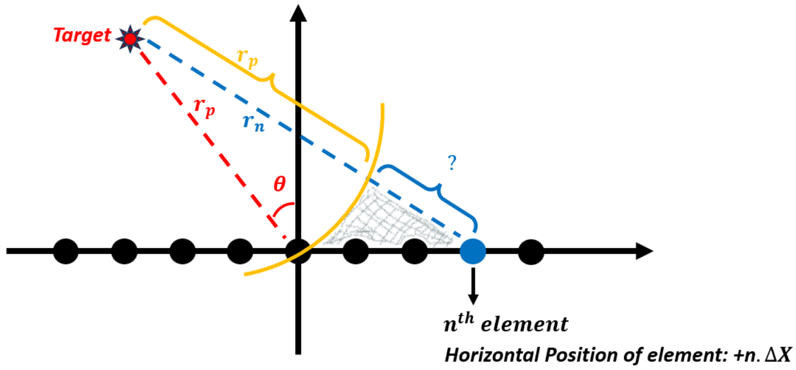
$$e^{2j\pi f_0 t} \cdot e^{-2j\pi f_0 t_0} \times e^{-2j\pi f_0 t} = e^{-2j\pi f_0 t_0}$$

$$\boxed{\rightarrow S_{Rx} = e^{-2j\pi f_0 t_0}}$$

As we know, t_0 is the time. We have calculated in the next section.

3. Array Design and DoA Estimation

3.1 Method



$\boldsymbol{r_p = r_n - X.\sin(\theta)}$

As we know :

$$\boxed{S_{Rx} = e^{-2j\pi f_0 t_0}}$$

On the other hand:

$$t_0 = \frac{r_n}{c}$$

t_0 : Corresponding delay of n^{th} antenna.

Now by substituting these formulas we have:

$$t_0 = \frac{r_n - x\sin\theta}{c}$$

$$S_{RX}(t) = e^{4j\pi f_0 \frac{r_n + x\sin\theta}{c}} = e^{4j\frac{\pi}{\lambda}(r_n + x\sin\theta)}$$

$$\begin{cases} |S_{RX}(x)| = 1 \\ \measuredangle S_{RX}(x) = 4j\dfrac{\pi}{\lambda}(r_n - x\sin\theta) \end{cases}$$

Now we can detect DoA by the following method:

One of the ways to detect the direction of arrival is Phase Comparison Technique which involves measuring the phase differences between signals received by different antennas in an array. By analyzing these differences, we can calculate angles of arrival for incoming signals.

Signal A => n=5 => $\frac{2\pi}{\lambda}(-\sin\theta.5\Delta_x)$

Signal B => n=6 => $\frac{2\pi}{\lambda}(-\sin\theta.6\Delta_x)$

By subtracting A-B:

$$\frac{2\pi}{\lambda}(3\Delta_x \cdot \sin\theta - 2\Delta_x \cdot \sin\theta)$$

$$\varphi = \frac{2\pi}{\lambda}\Delta x \sin\theta_t$$

$$\sin\theta_T = \frac{\lambda\varphi}{2\pi\Delta_x}$$

$$DOA = \theta_T = \sin^{-1}\left(\frac{\lambda\varphi}{2\pi\Delta_x}\right)$$

The calculation of the length of the antenna is as follow:

$$F = \frac{C}{\lambda} \longrightarrow \lambda = \frac{C}{F}$$

$$\lambda = \frac{3\times10^8}{77\times10^9} = 0.00389$$

In addition, we have a limitation on the spacing of antennas on any array:

$$\mathrm{d} \leq \frac{\lambda}{2} \rightarrow \mathrm{d} \leq \frac{0.00389}{2} = 0.00195\ \mathrm{m} = 19\ \mathrm{mm}$$

$$\mathrm{L}= \lambda\ /(((2^*\mathrm{pi})/180)^*\mathrm{resolution})$$

In this assignment, the resolution is defined as 2 degree. We can get:

Length of array = L= 0.0558

3.2 MATLAB Implementation

After designing our array, and designing the parameters in MATLAB, we need to define the following vectors:

1. Antenna's Position
2. Target Position
3. The vector contains the phase of arrived signal

We have tested our proposed method for different kind of target positions which are as follows:

1. Test#1 :
    - X= 5 km
    - Y= 5 km
    - $\theta = \tan^{-1}\left(\frac{y}{x}\right) = 45°$

2. Test#2 :
    - X= 10 km
    - Y= 5 km
    - $\theta = \tan^{-1}\left(\frac{y}{x}\right) = 26.56°$

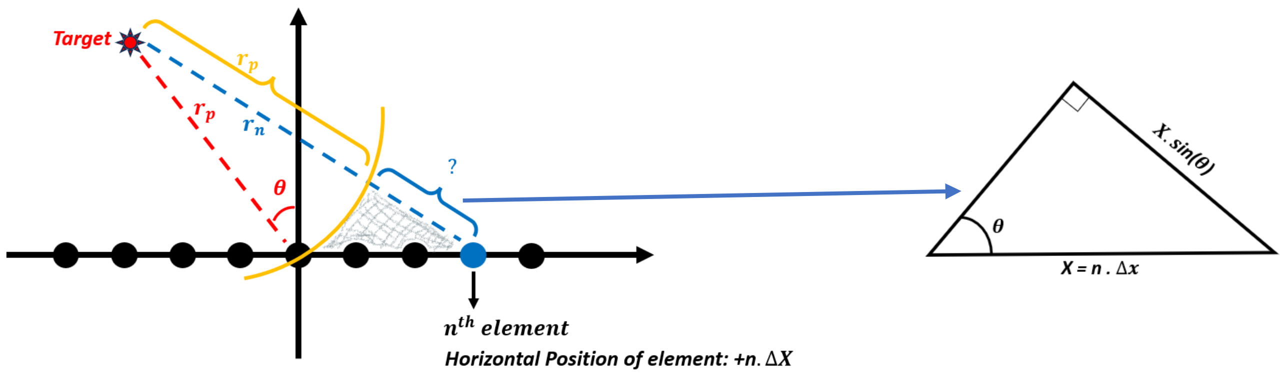
3. Test#3 :
   - X= 18 km
   - Y= 25 km
   - $\theta = \tan^{-1}\left(\frac{y}{x}\right) = 54.24°$

Then. We estimate the DoA for each test and calculate the errors which are as follows:

	Test 1	Test 2	Test 3
Estimated DoA	45.0167	26.5717	54.2505
Real DoA	45	26.56	54.24
Error	0.0167	0.0067	0.0044

Table 1

As is shown in the table above, the estimated DoA and the real DoA are not the same. The reason the approximation in the calculation of the straight line the triangle as illustrated below:



In fact, we approximate the shadowed shape with a triangle. Therefore:

$$r_n = r_p + X.\sin(\theta)$$

However, the shadowed shape is not a pure triangle and caused an error in our estimation. Through applying geometric approximation, we precisely estimate the arc length of a circle by considering it equivalent to the chord length of a triangle.

As detailed in Table 1, after examining multiple tests and calculating the errors in DoA, it is evident that the resolution of 2 degrees is maintained. Since the computed error is less than 2 degrees, we can confidently state that the DoA estimation aligns with the defined resolution of the array. Furthermore, it is clear from the above result that the estimated value is always more or less the same as the real one.

```
real_DoA =

    45

estimated_DoA =

   45.0167

error =

    0.0167

fx >>
```

Figure 3: Test#1

```
real_DoA =

   26.5651

estimated_DoA =

   26.5717

error =

    0.0067

fx >>
```

Figure 2: Test#2

```
real_DoA =

   54.2461

estimated_DoA =

   54.2505

error =

    0.0044

fx >>
```

Figure 1: Test#3

Then, spacing between the element's changes for several values in order to see the behavior of error whether if it increased or decreased. For the simulation with different spacing *dx*, we considered the following assumptions:

- Using the same length of array *L = 5.58 [cm]*
- Using the same position of target in Test#1: *(x,y) = (5,5) [km]*
- Only changing the spacing *dx*

By the default value of dx and the above assumption, we got error = 0.0167. Now we increase it for the following values of dx and the summary of the result mentioned in the table below:

dx [m]	Error [deg]
0.0019	0.0167
0.01	0.0861
0.015	0.1292
0.035	0.3025
0.045	0.3895
0.055	0.4768
0.075	0.6523
0.085	0.7404

Table 2

The behavior of Error in DoA estimation represented in the figure below:

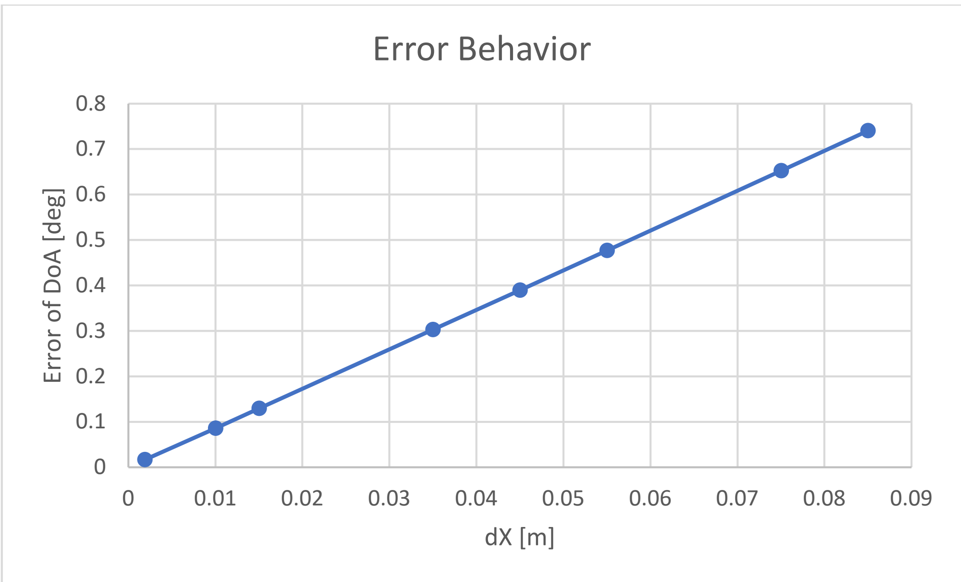


Figure 4: Error of DoA

As it is shown in the figure above, by increasing the space between the elements, the error of DoA estimation increases. The more spacing *dx* we have, the more error in estimation we get.

In the next step, we add gaussian noise to the received signal in order to see the standard deviation with respect to different levels of noise power. For this purpose, we used Monte-Carlo simulation with 500 iterations. We used the previous parameter already defined in Test#1.

```
Target_x= 5;                        % Horizoontal Distance[m]
Target_y= 5;                        % Vertical Distance[m]
dx=lambda/2;                        % space between two consequential antennas
```

We set the number of simulations to 500. In each iteration, we used a unique noise power as a baseline multiplied by complex random gaussian variables. For the calculation of the standard deviation, we have to use the following formula. N is the number of simulation and it is defined N=500 in MATLAB.

$$\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{N}}$$

Equation 1

After 500 simulations, the result is standard deviation with respect to 500 different noise floor is represented below:

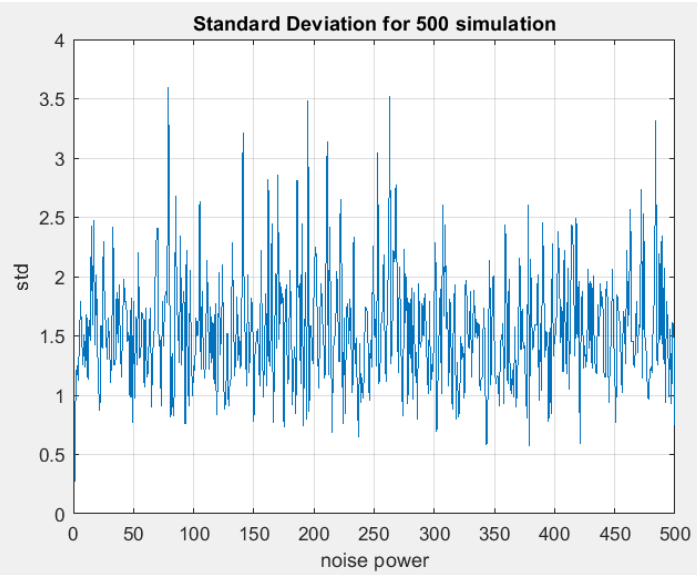


Figure 5: Standard Deviation vs. Noise Power

The error is also plotted for 500 number of noise power in the figure below:

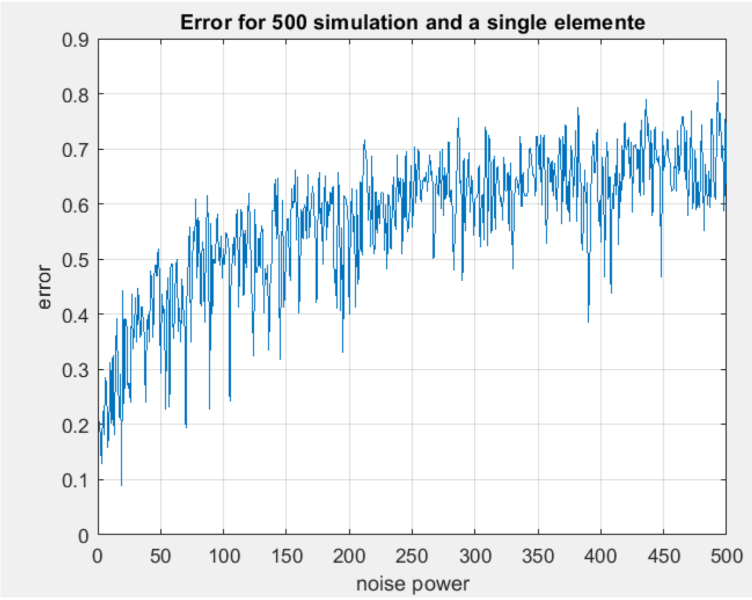


Figure 6: Error vs. Noise Power

It is clear from the above figure that by increasing the power of noise, the error will increase. The more noise power we have, the more errors we get in the DoA estimation.

4. 2D position estimation

4.1 System Model

In this part, the transmit signal is multiplied by the cardinal sine and we have a bandwidth of 1 GHz.

As we know:

$$Band\ width = 1\ GHz$$

By following the Nyquist's theorem we have:

$$Sampling\ frequency \geq\ 2\ Band\ width$$
$$or$$
$$f_s\ \geq 2B$$

Let's define the Sampling rate in order to avoid aliasing:

$$\boxed{f_s\ =\ 4\ GHz}$$

Therefore

$$Time\ resolution = T_s\ = \frac{1}{f_s} = 0.25\ ns$$

And the space resolution is as follow:

$$\rho_r = \frac{C}{2BW} = \frac{3 \times 10^8}{2 \times\ 10^9} = 0.15\ m$$

For calculating $S_{RX}(t)$ at the receiver and $S_{TX}(t)$ at the transmitter:

$$S_{TX}(t) = g(t) = sinc(Bt)e^{2j\pi f_0 t}$$
$$S_{RX}(t) = g(t - t_0) \cdot S_{TX}^*$$

1) *Define function* $S(t) = sinc(Bt)e^{2j\pi f_0 t}$

2) *Sampling* S @ $f_s = 4\ GHz$ *it should be complex signal.*

3) *We have the value of* r'_{pn} *then* $\frac{r_n}{C} = t_0$ *, also we have the vector t so, we can find the value of vactor t' by* $t' = t - t_o$.

4) *we should* $find\ S_{RX}(t') = g(t') \times g^*(t')$ *(We calculated the(t') vector in the 3*[th] *step).*

5) *Plot FFT → at the max point and then read* f_x*:*

$$f_x = \frac{Sin\,\psi}{\lambda} \rightarrow \lambda \cdot f_x = Sin\,\psi$$

Then

$$\psi = \sin^{-1}(\lambda \cdot f_x)$$