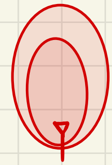
Beamforming and Angle of Arrival (AOA)

- Omnidirectional antennas: radiate signals equally in all directions

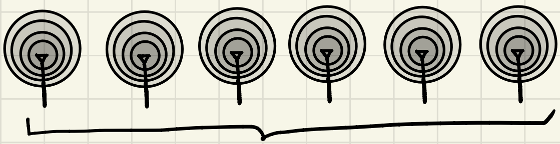
  Directional antennas : Direct the radiation more in certain directions and less in others.



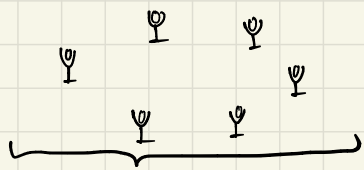
- creating such non-circular radiation patterns ⇒ Beamforming → Spatial Filter

  How ?

- Let's consider an ARRAY of omni-directional antennas (or even microphones)



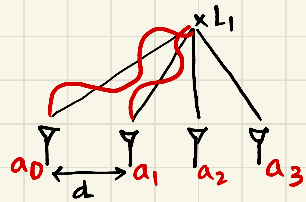
Linear antenna array



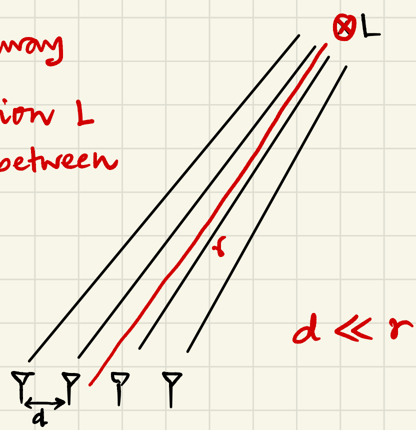
circular mic. array (Alexa)

- say, these antennas transmit all at the same time?
  - ↳ what signals will you receive from different locations?

- consider nearby locations first :
  - ↳ The aggregate signals at these nearby locations vary based on the location.
  - ↳ No pattern is visible as you move.
  - ↳ This is called " NEAR FIELD ".

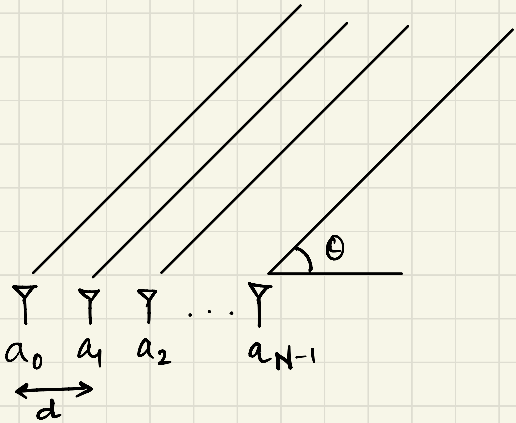


- Now, consider locations that are far away
  - ↳ When distance from antennas to location L becomes ≫ than seperation 'd' between the antennas, then the signal paths almost become PARALLEL
  - ↳ called "FAR FIELD"
  - ↳ Let's analyze far field effects



$d \ll r$

(→)



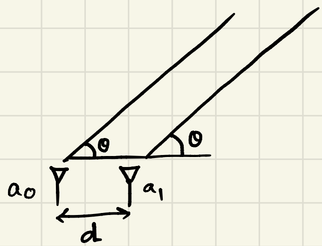
- All antennas transmit
- Say Rx receives $s_0(t)$ from antenna a_0 … and $s_i(t)$ from antenna a_i
- Received signal $y(t)$

$y(t) =$

(→) Now, assume (no echo or multipath).

↳ Then what is the difference between $s_0(t)$ and $s_1(t)$?

Ans :



$s_1(t)$ travels distance than .

⇓

How much phase shift ϕ does this cause?

λ distance causes 2π phase shift

$\therefore$ $d\cos\theta$

⇓

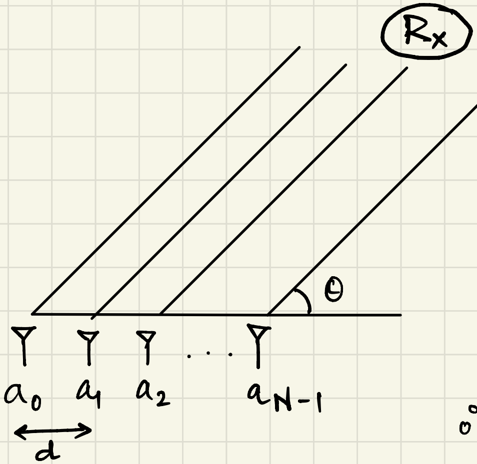
How can we mathematically write that $s_1(t) = s_0(t)$ phase shifted by $\phi = \frac{2\pi}{\lambda} d\cos\theta$

↳ Recall

Thus :

$s_0(t) = \cos(2\pi f_1 t)$ $s_1(t) =$

$\therefore$ $s_1(f) =$



$s_1 =$, $\phi =$

$s_2 =$

$\vdots$

$s_N =$

$\therefore \; Y =$

$$\cong s_0\left(\frac{1 - e^{jN\phi}}{1 - e^{j\phi}}\right), \quad \phi = \frac{2\pi}{\lambda} d\cos\theta$$

→ Plot Y_f or y_t against θ

→ So the beams look like :

→ Observe, the natural beam is pointing towards

⊕ Beam Rotation

Now I want the main lobe to point towards

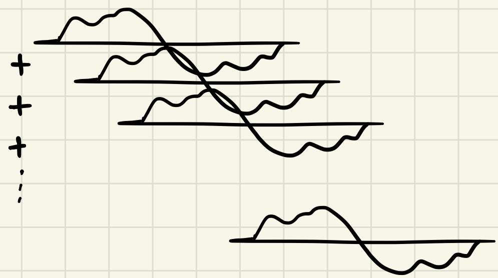
↳ i.e., towards θ.

↳ How? By making signals from all antennas

So, first let's see how signals add up along θ

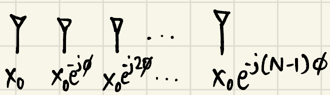
Recall $Y =$

This is like



⊕ For max SNR at R_x,

i.e.,



$\therefore \; Y = \sum_{k=0}^{N-1}$

$\therefore \; Y =$

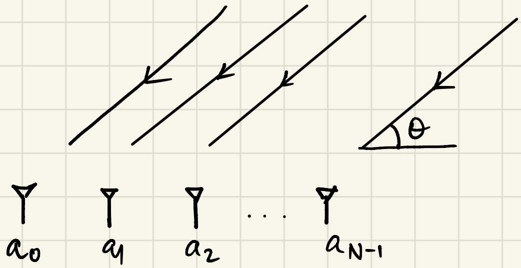
This is called

⊕ Analogy:

Stagger runners at the starting line

to ensure they all run the same distance

ANGLE OF ARRIVAL (AOA)



Signal arriving from

Antenna array needs to figure out the

How can you estimate AoA ? Well, similar concepts as beamforming

- Say received signal is now

y_N = [ ] = [ ] $\xrightarrow{\text{Freq.}}$ $X(f)$ [ ]

- From this received vector, how do you detect θ ?
  - Answer:
  - Algorithm :

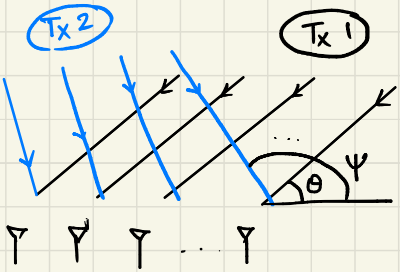
```
for θ_i =                    // search over all AoA θ
{
    α_i =                    // calculate phase shift

    C_θi = [          ][   ] //

}
Plot                         // Plot the AoA spectrum
```

AoA =

- Now, let's assume sending in parallel.
  - ↳ can we still decode the AoAs ?



say $\phi_1 =$

$\phi_2 =$

$\bar{Y} =$

- Now how can you decode
  - ↳ Answer : Looking for a certain phase pattern ⟶

$$[\quad]^{*} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_{N-1} \end{bmatrix} =$$

Perform this for all values of

Hope dot product large when

→ Modelling noise $\bar{Y} = A\bar{s} +$

$$\begin{bmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{N-1} \end{bmatrix} = \begin{bmatrix} \quad \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}$$

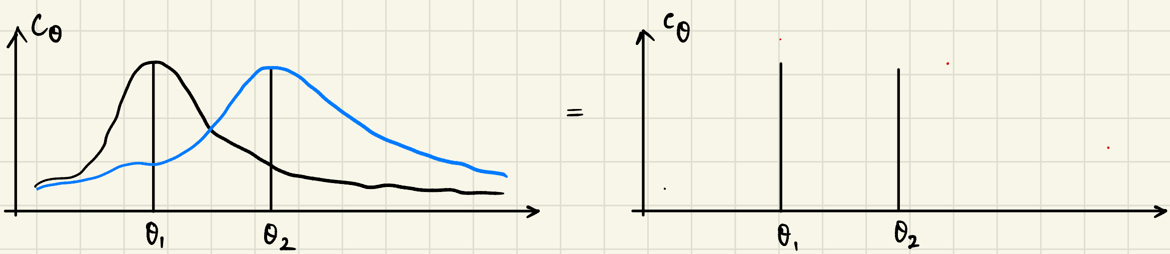
$$\begin{bmatrix} \quad \end{bmatrix}^* \begin{bmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{N-1} \end{bmatrix} = \begin{bmatrix} \quad \end{bmatrix}^* \begin{bmatrix} \quad \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} + \begin{bmatrix} \quad \end{bmatrix} \begin{bmatrix} n_0 \\ n_1 \\ \vdots \\ n_{N-1} \end{bmatrix}$$

correlating for

$$= \quad x_1 + \quad x_j +$$

→ By correlating along all directions a_i, we get an

→ Problem is



Peak is

Especially when signals are