ECE 598HH: Advanced Wireless Networks and Sensing Systems

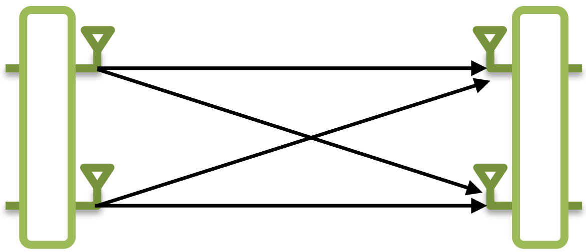
Lecture 9: MIMO Part 2
Haitham Hassanieh

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

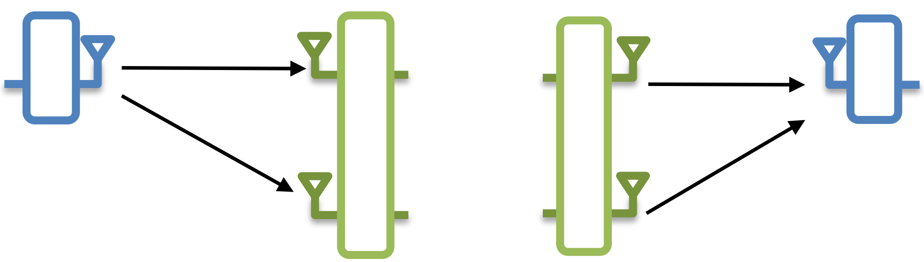
*Some of the slides in this lecture are courtesy of on Kate Lin, Hariharan Rahul, & Omid Abari

Last Lecture

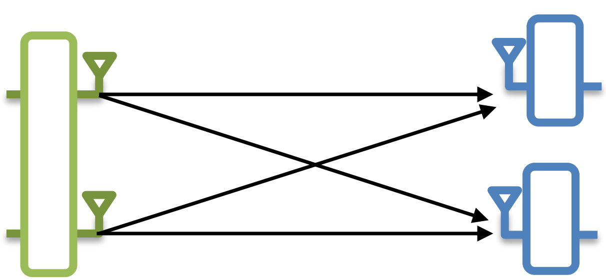
MIMO Multiplexing Gain



MIMO TX/RX Diversity Gain

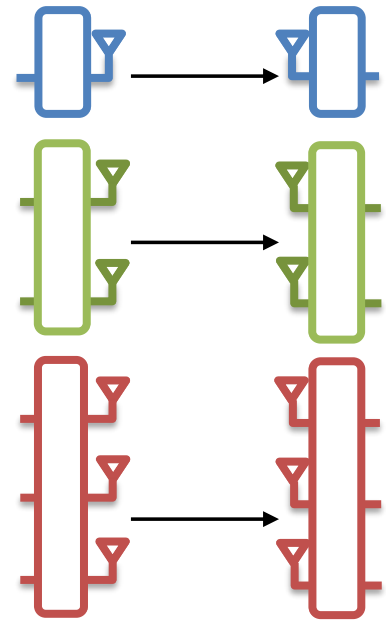


MU-MIMO Beamforming

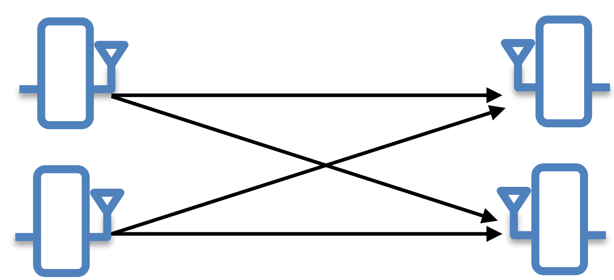


This Lecture

Heterogeneous # of antennas (Interference Nulling/Alignment)



Distributed MIMO



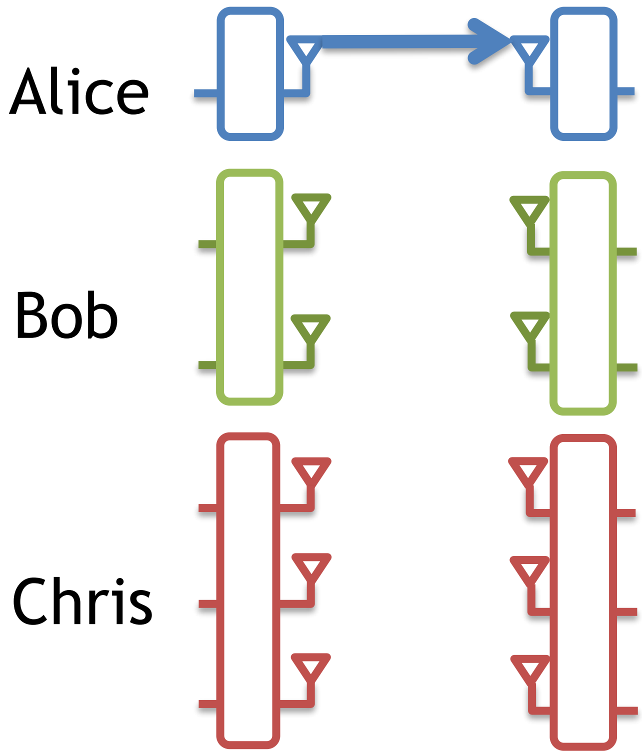
1-antenna devices

2-antenna devices

3-antenna devices

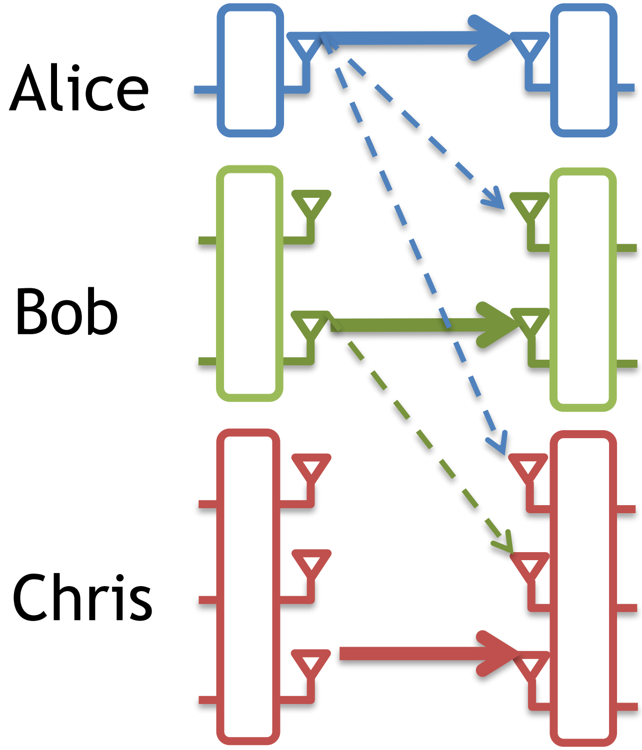
Wireless nodes increasingly have
heterogeneous numbers of antennas

Consider a scenario with Tx-Rx pairs that have a different number of antennas

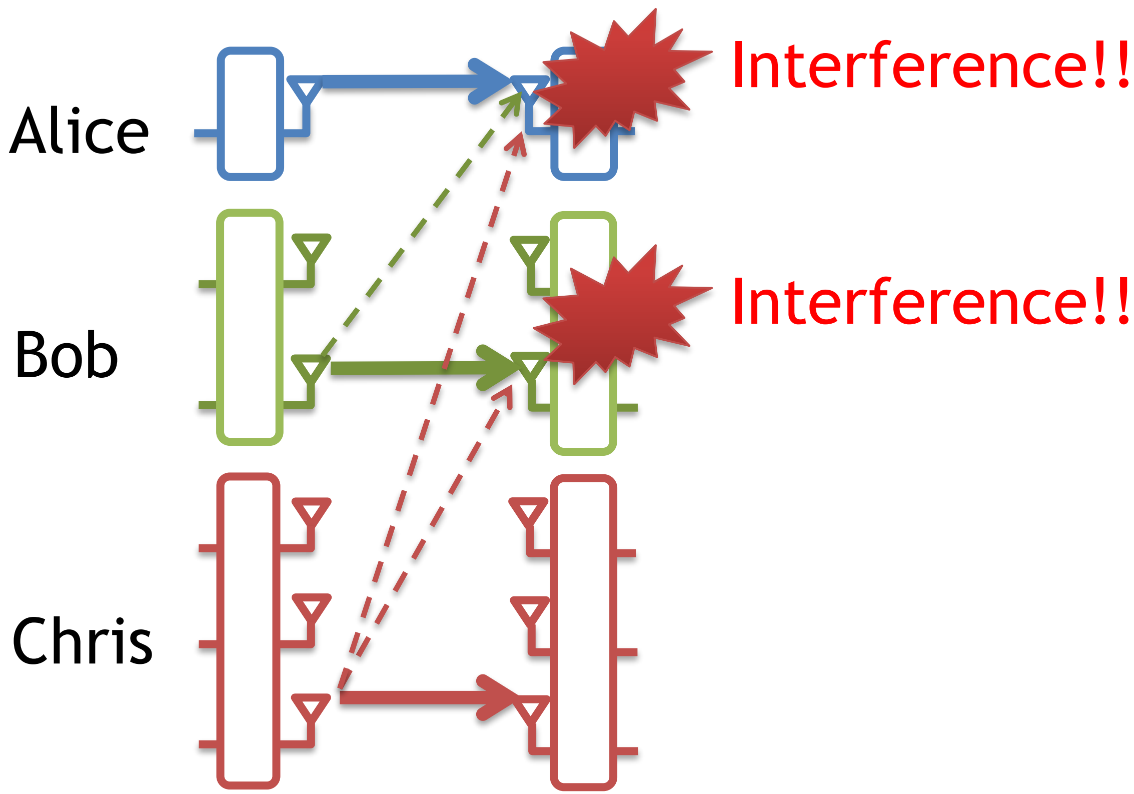


When a single-antenna node transmits, multi-antenna nodes refrain from transmitting

But, MIMO Nodes Can Receive Multiple Concurrent Streams

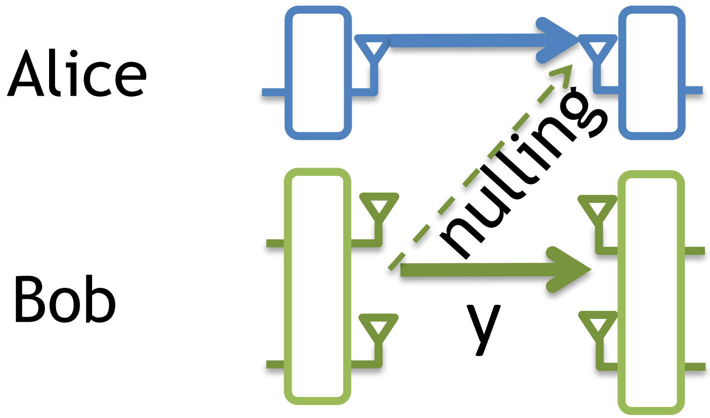


It’s Not That Simple



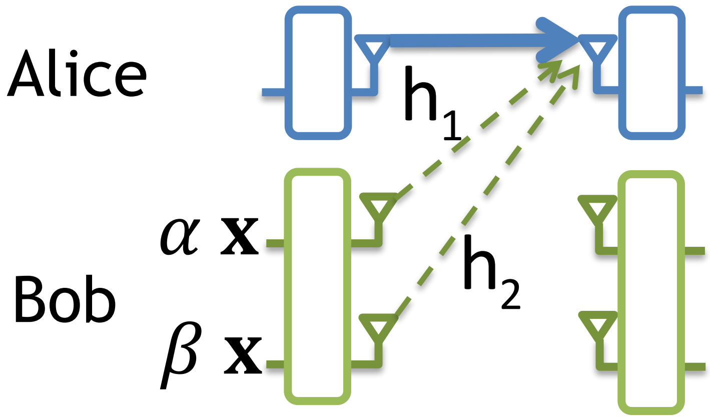
But, how do we transmit without interfering at receivers with fewer antennas?

Interference Nulling



- Signals cancel each other at Alice's receiver

Interference Nulling



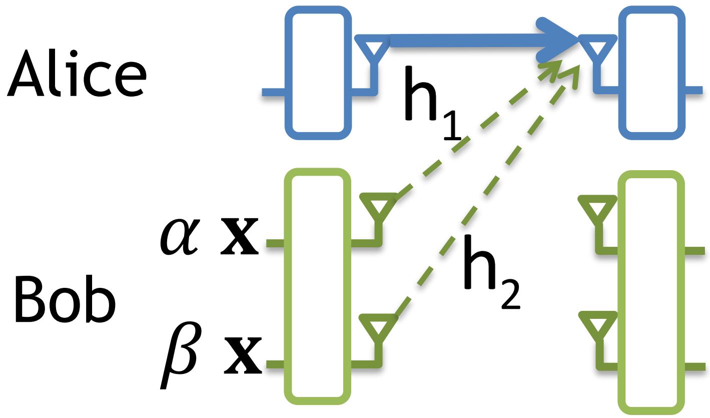
$$h_1 \alpha\, \mathbf{x} + \mathrm{h}_2 \beta\, \mathbf{x} = 0$$

➔ Nulling : $h_1 \alpha = -\mathrm{h}_2 \beta$

$\neq 0$

- Signals cancel each other at Alice’s receiver
- Signals don’t cancel each other at Bob’s receiver
  - Because channels are different
- Bob’s sender learns channels either by feedback from Alice’s receiver or via reciprocity

Interference Nulling



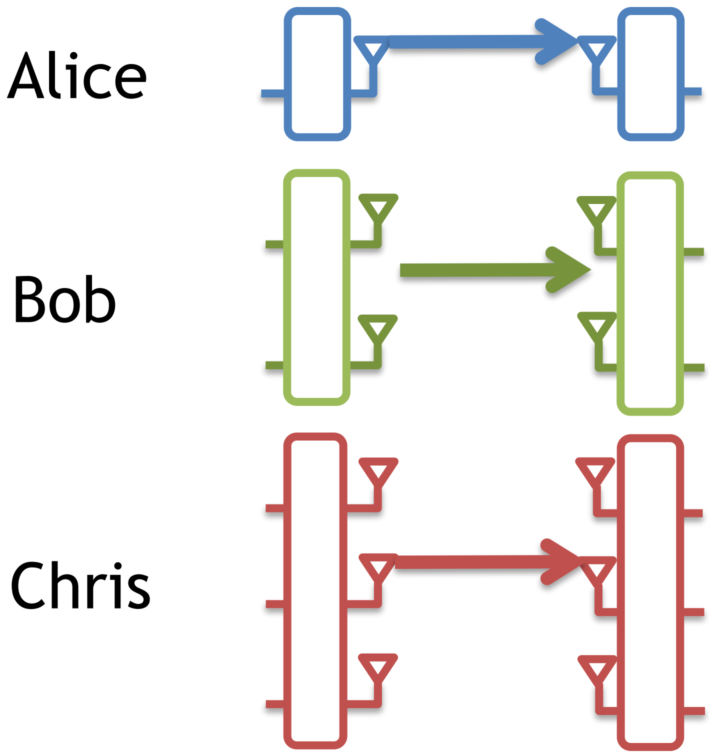
$h_1 \alpha \mathbf{x} + \mathrm{h}_2 \beta \mathbf{x} = 0$

➔ Nulling : $h_1 \alpha = -\mathrm{h}_2 \beta$

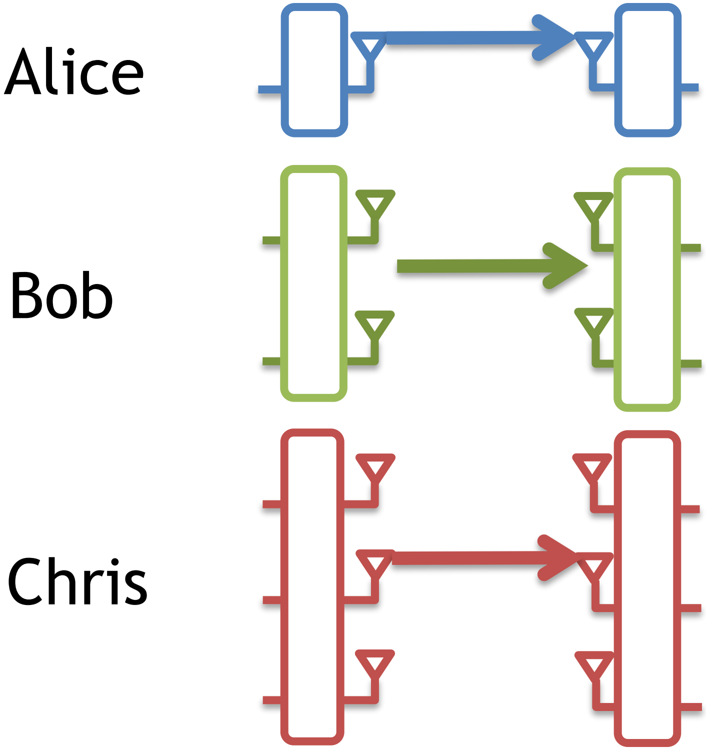
$\neq 0$

Q: How to transmit without interfering with receivers with fewer antennas?

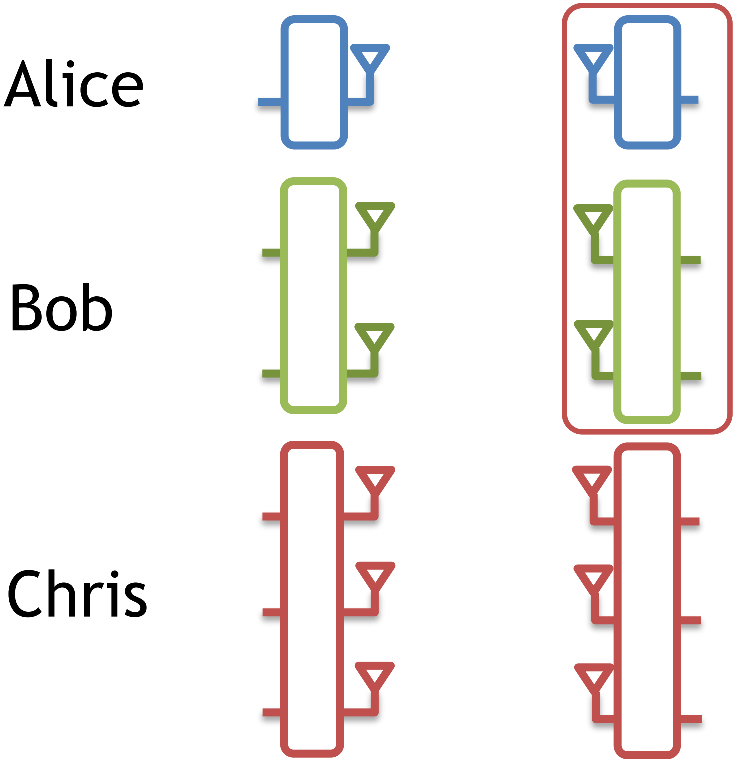
A: Nulling


Alice
Bob
Chris

Is Nulling Alone Enough?

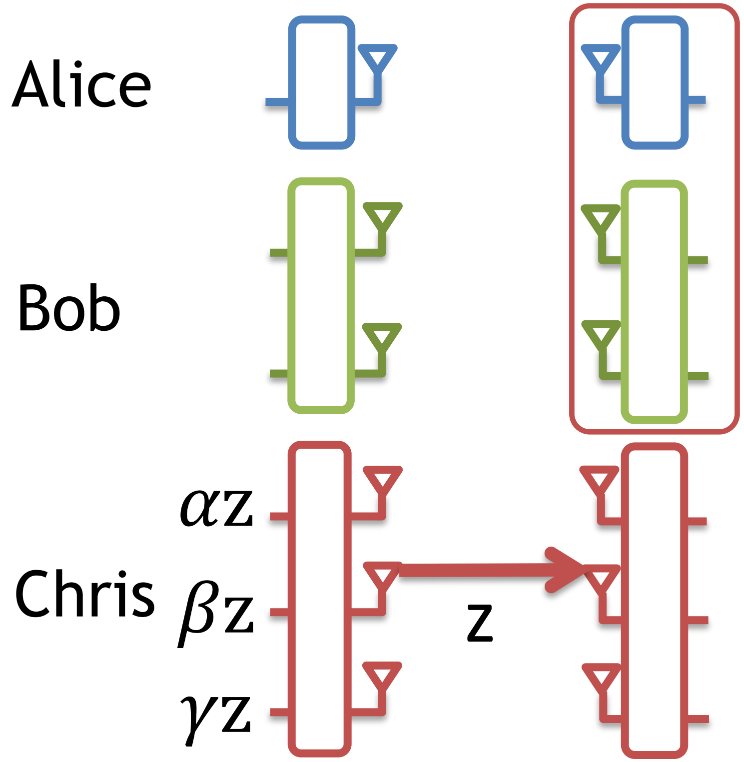


Is Nulling Alone Enough?



Chris needs to null at three antennas

Is Nulling Alone Enough? **NO!**



null

$$(h_{11}\alpha + h_{21}\beta + h_{31}\gamma)z = 0$$

$$(h_{12}\alpha + h_{22}\beta + h_{32}\gamma)z = 0$$

$$(h_{13}\alpha + h_{23}\beta + h_{33}\gamma)z = 0$$

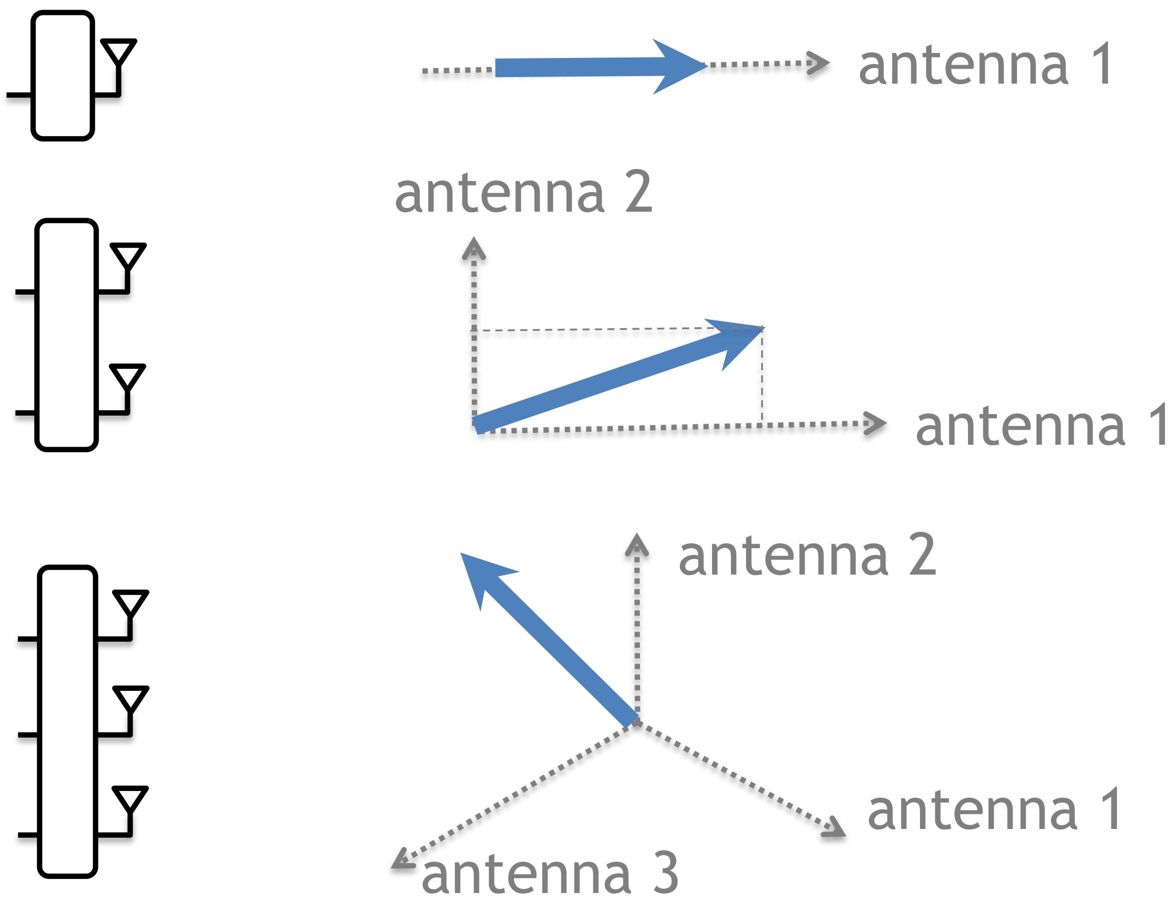
Only Solution: $\alpha = \beta = \gamma = 0$

Transmit Nothing!!!

Are we doomed?

MIMO Recap

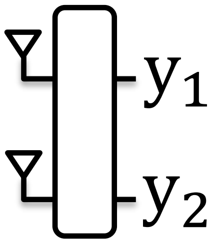
1. N-antenna node receives in N-dimensional space

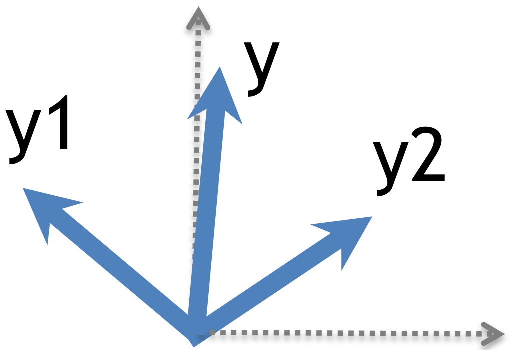


MIMO Recap

1. N-antenna node receives in N-dimensional space
2. N-antenna receiver can decode N signals

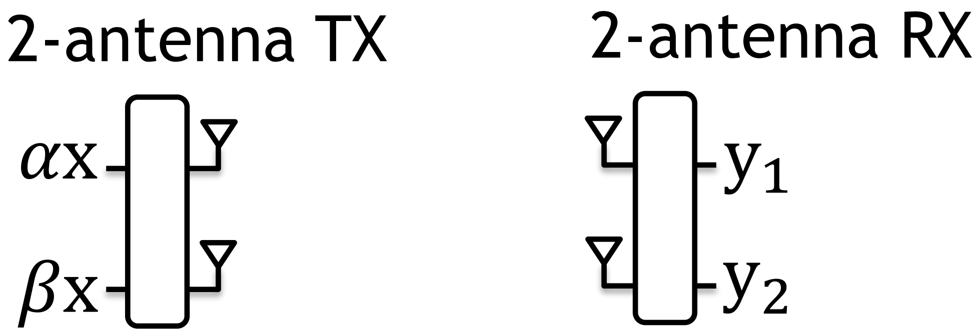
2-antenna RX

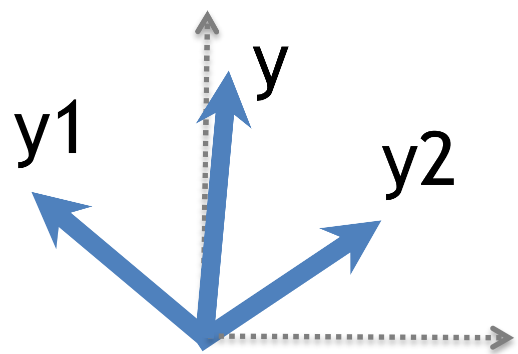




MIMO Recap

1. N-antenna node receives in N-dimensional space
2. N-antenna receiver can decode N signals
3. Transmitter can rotate the received signal



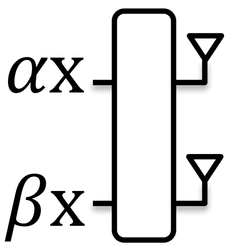


x is the sender's symbol, (y1,y2) is the received symbol, (α, β) is the pre-coding vector, and H is the channel matrix

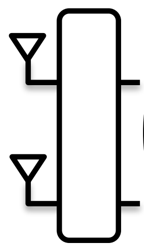
MIMO Recap

1. N-antenna node receives in N-dimensional space
2. N-antenna receiver can decode N signals
3. Transmitter can rotate the received signal

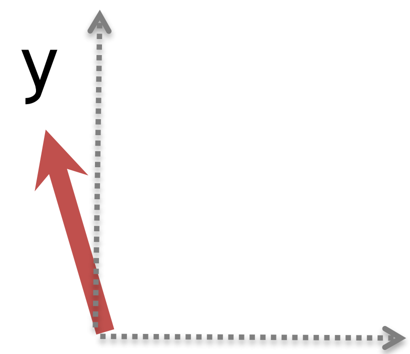
2-antenna TX



2-antenna RX



$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = H \begin{pmatrix} \alpha \\ \beta \end{pmatrix} x$$

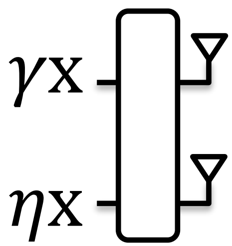


x is the sender's symbol, (y1,y2) is the received symbol, (α, β) is the pre-coding vector, and H is the channel matrix

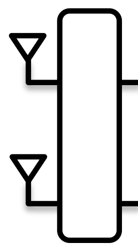
MIMO Recap

1. N-antenna node receives in N-dimensional space
2. N-antenna receiver can decode N signals
3. Transmitter can rotate the received signal

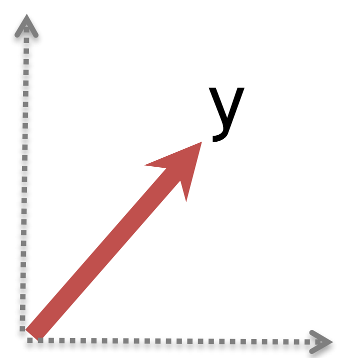
2-antenna TX



2-antenna RX



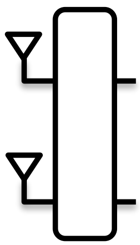
$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = H \begin{pmatrix} \gamma \\ \eta \end{pmatrix} x$$

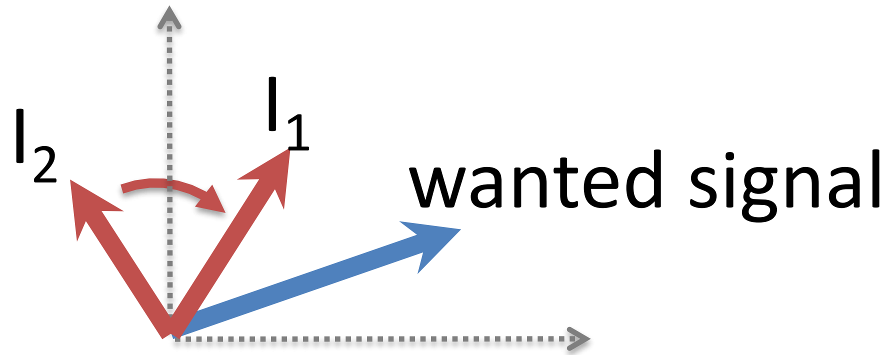


Sender can rotate the received symbol by multiplying with a different pre-coding vector; However to align along a particular direction, sender needs to know H

Interference Alignment

2-antenna receiver

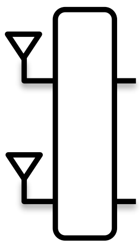


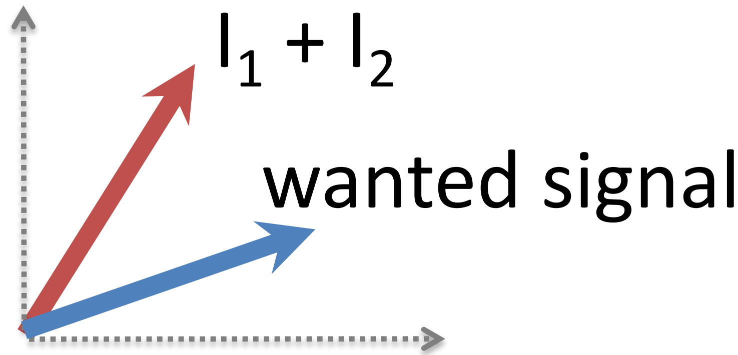


If I_1 and I_2 are aligned,

Interference Alignment

2-antenna receiver



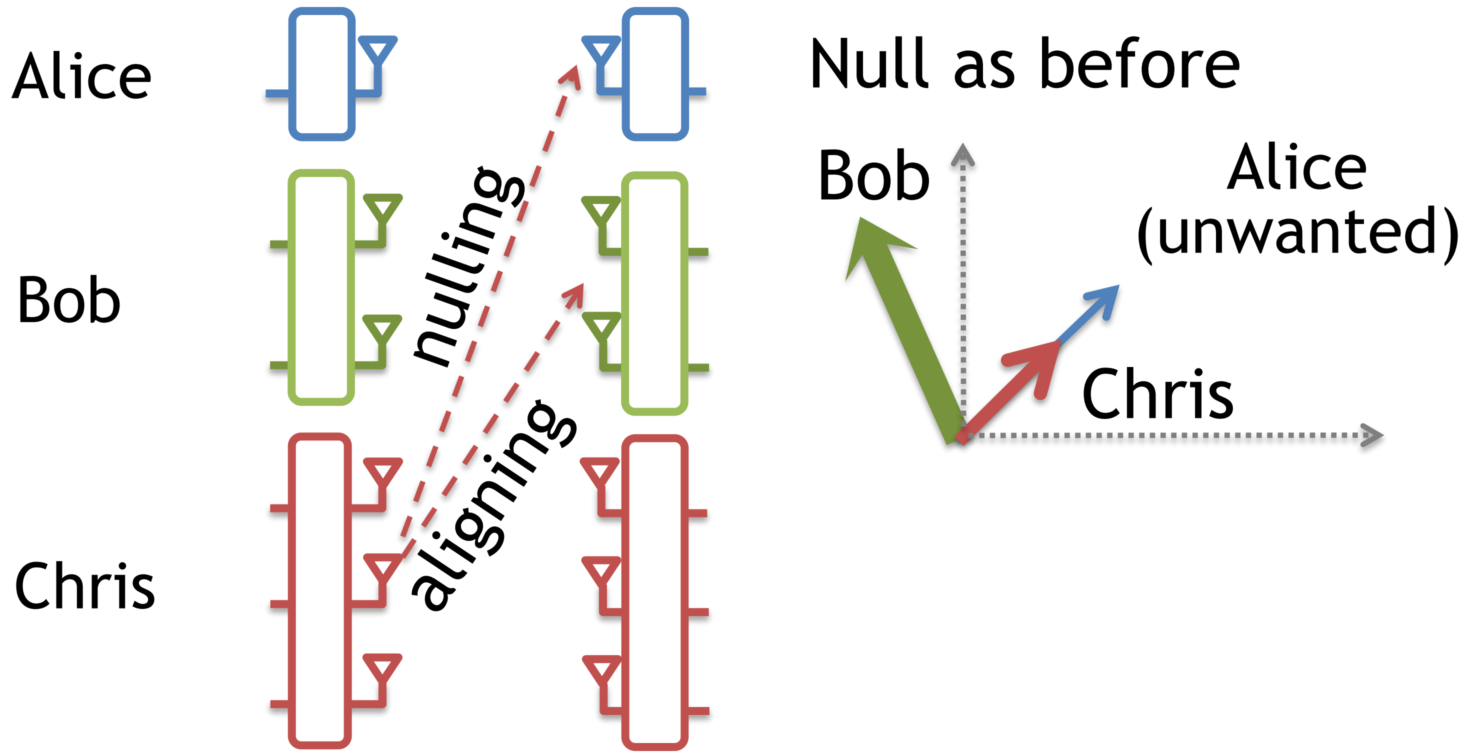


If I_1 and I_2 are aligned,

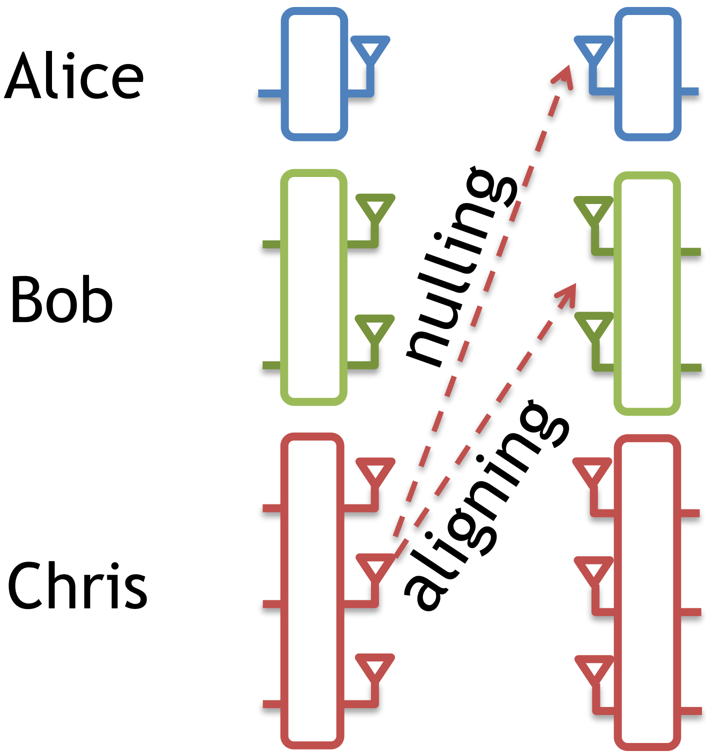
→ appear as one interferer

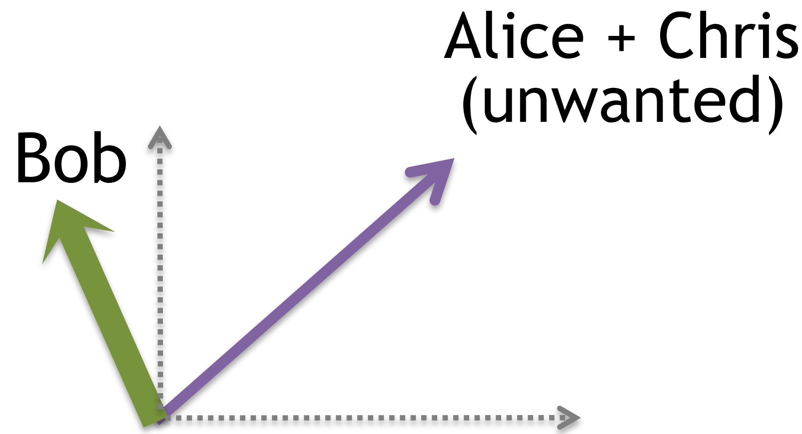
→ 2-antenna receiver can decode the wanted signal

Use Nulling and Alignment



Use Nulling and Alignment

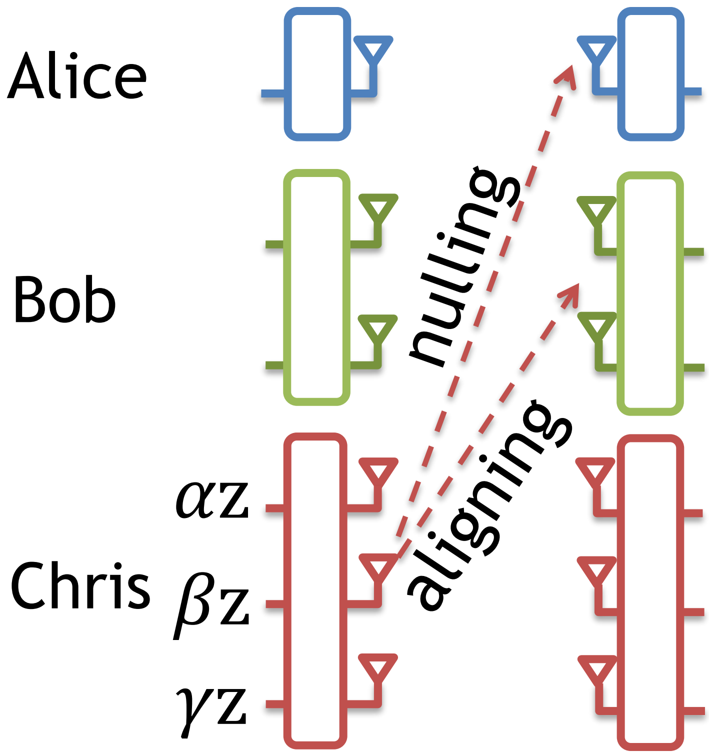




2-signals in 2D-space

➔Can decode Bob's signal

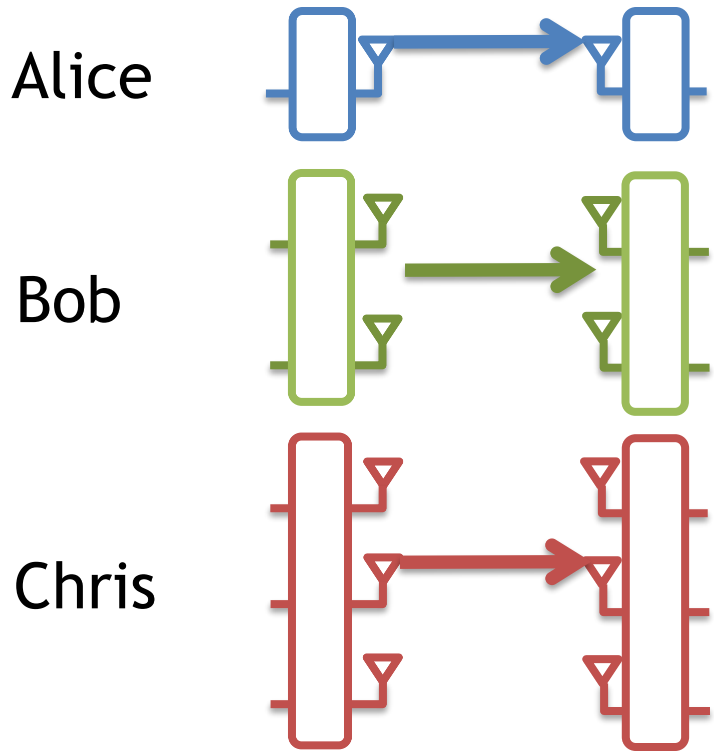
Use Nulling and Alignment



$$(h_{11}\alpha + h_{21}\beta + h_{31}\gamma)\text{z} = 0$$

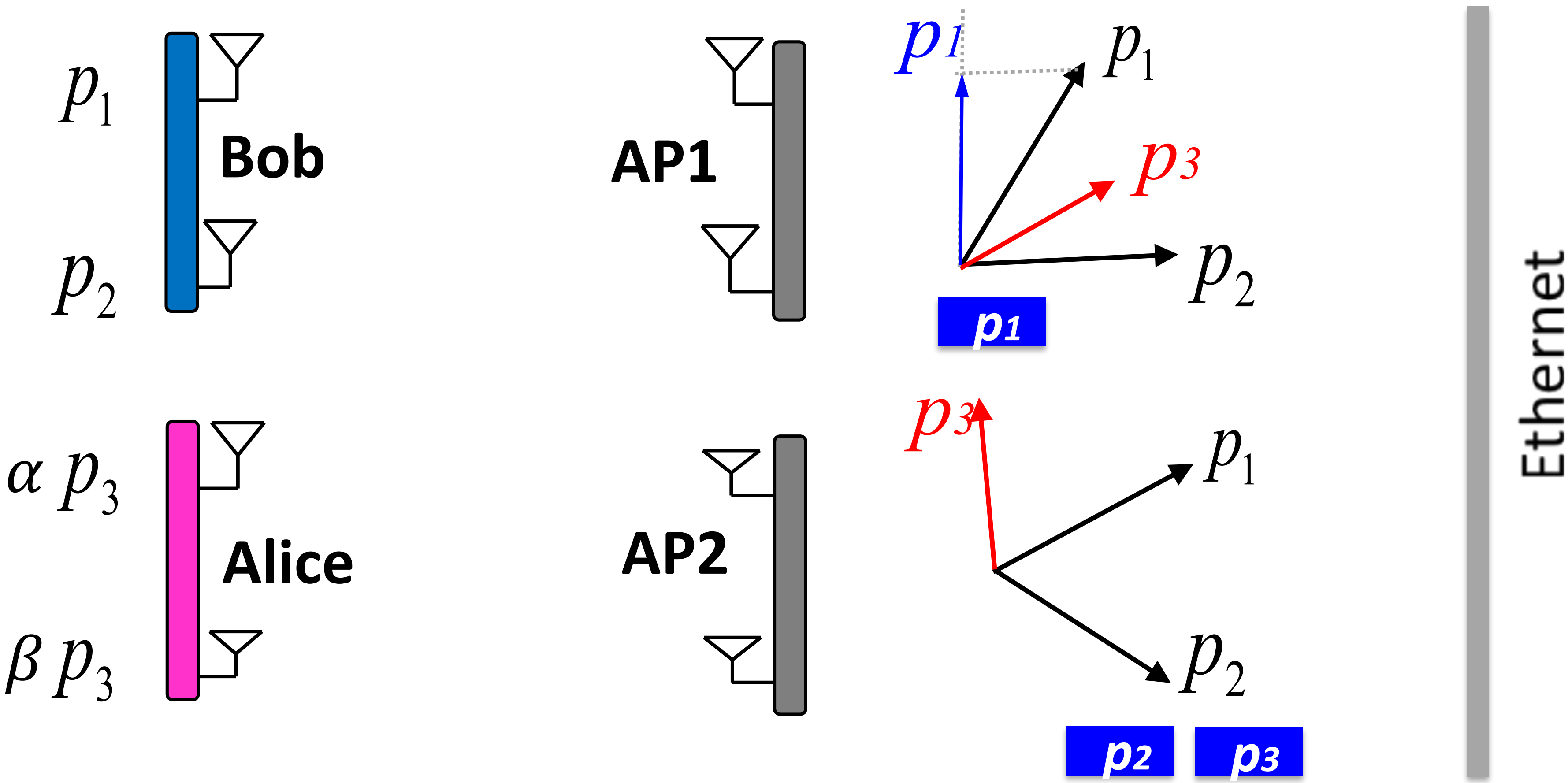
$$\begin{pmatrix} h_{12}\alpha + h_{22}\beta + h_{32}\gamma \\ h_{13}\alpha + h_{23}\beta + h_{33}\gamma \end{pmatrix} \propto \begin{pmatrix} h_{a1} \\ h_{a2} \end{pmatrix}$$

Use Nulling and Alignment



3 packets though receivers have fewer than 3 antennas

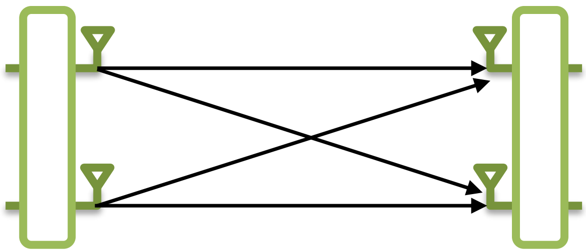
Interference Alignment and Cancellation (IAC)



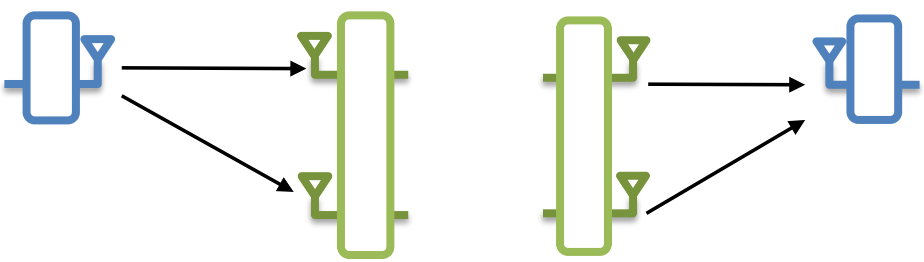
- Align P3 with P2 at AP1 → AP1 decodes P1 to its bits
- AP1 broadcasts P1 on Ethernet
- AP2 subtracts/cancels P1→ decodes P2, P3

Last Lecture

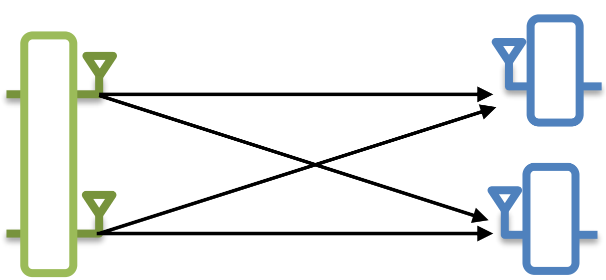
MIMO Multiplexing Gain



MIMO TX/RX Diversity Gain

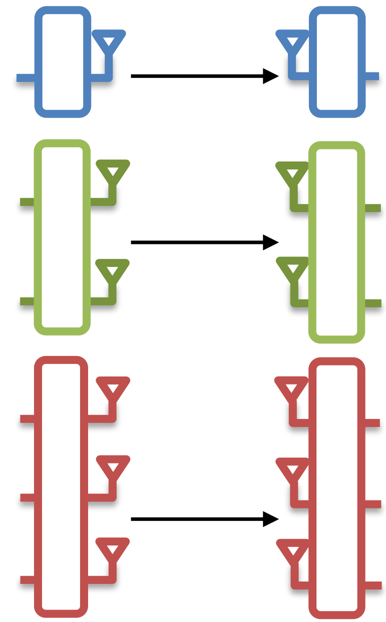


MU-MIMO Beamforming

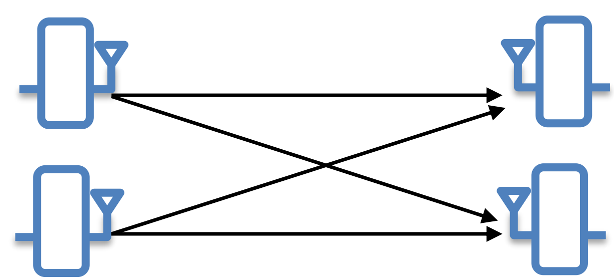


This Lecture

Heterogeneous # of antennas (Interference Nulling/Alignment)

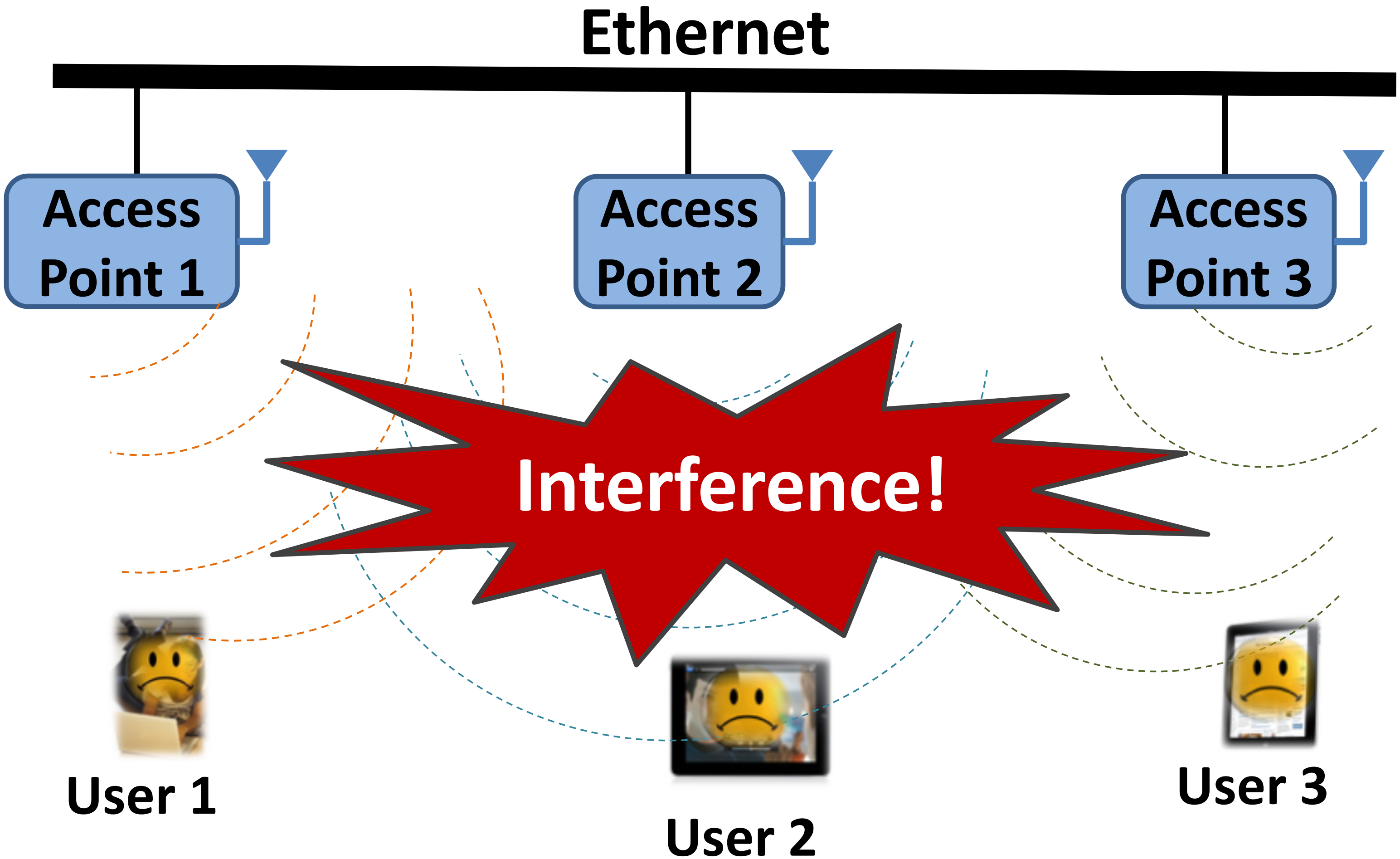


Distributed MIMO



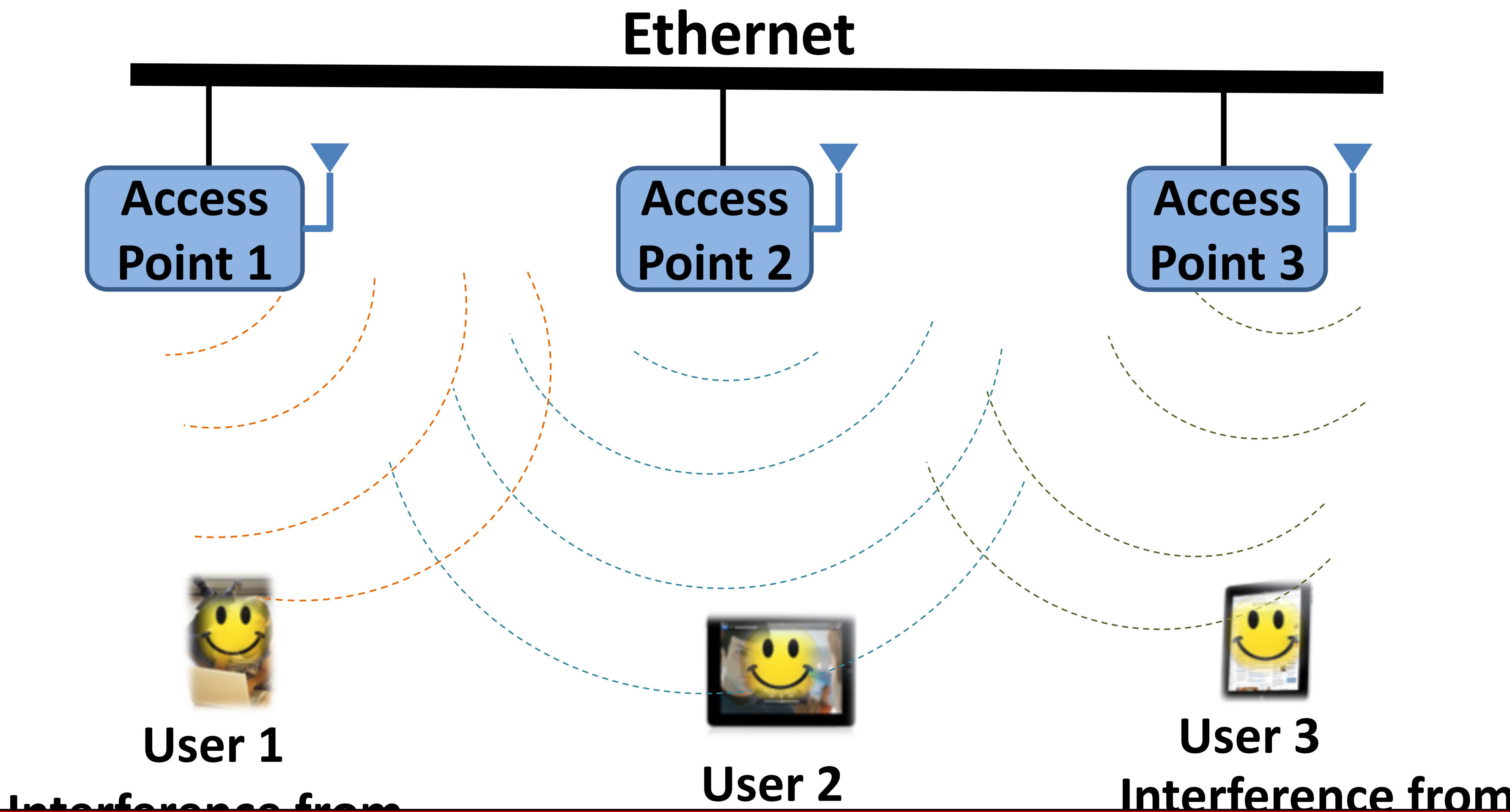
MegaMIMO:
Scaling Wireless Throughput with the Number of Users

Today's Wireless Networks



Access Points Can't Transmit Together in the Same Channel

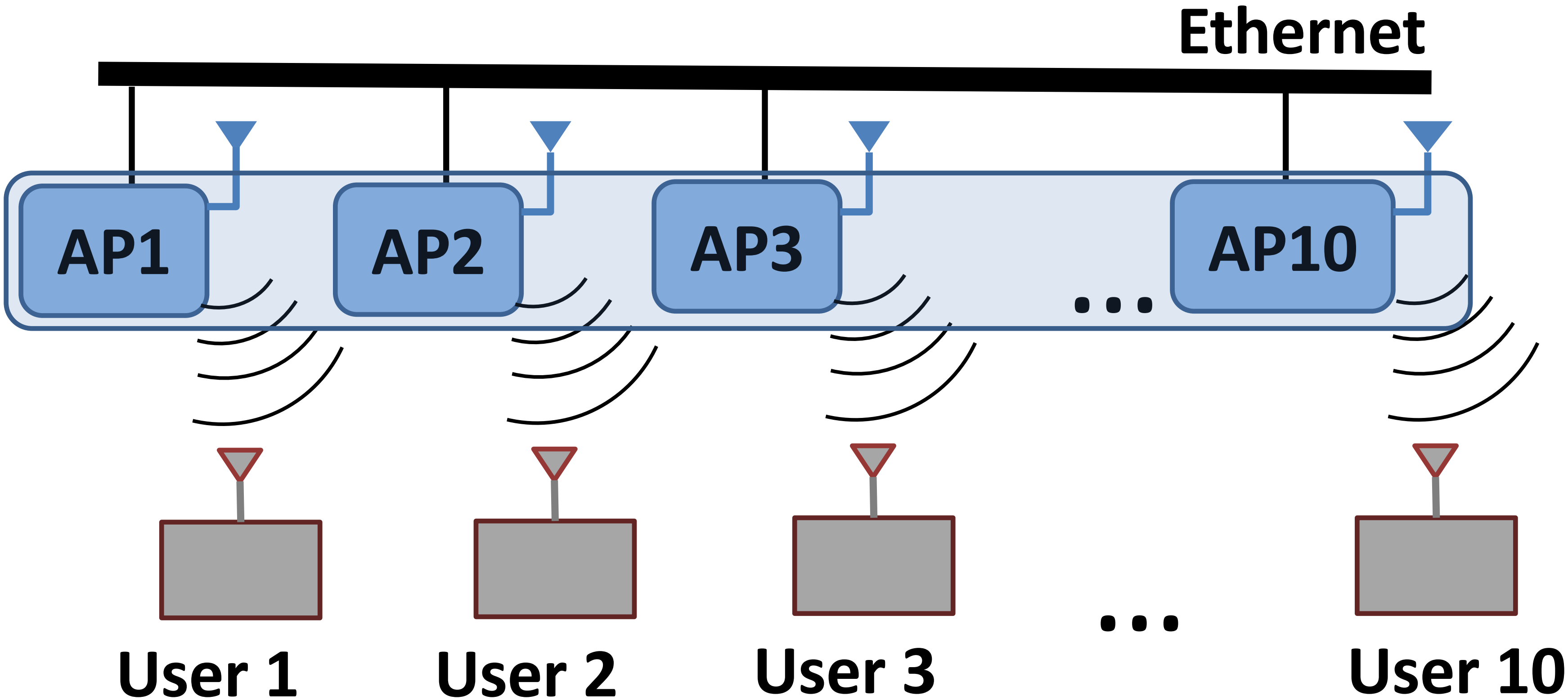
MegaMIMO



Enables senders to transmit together without interference

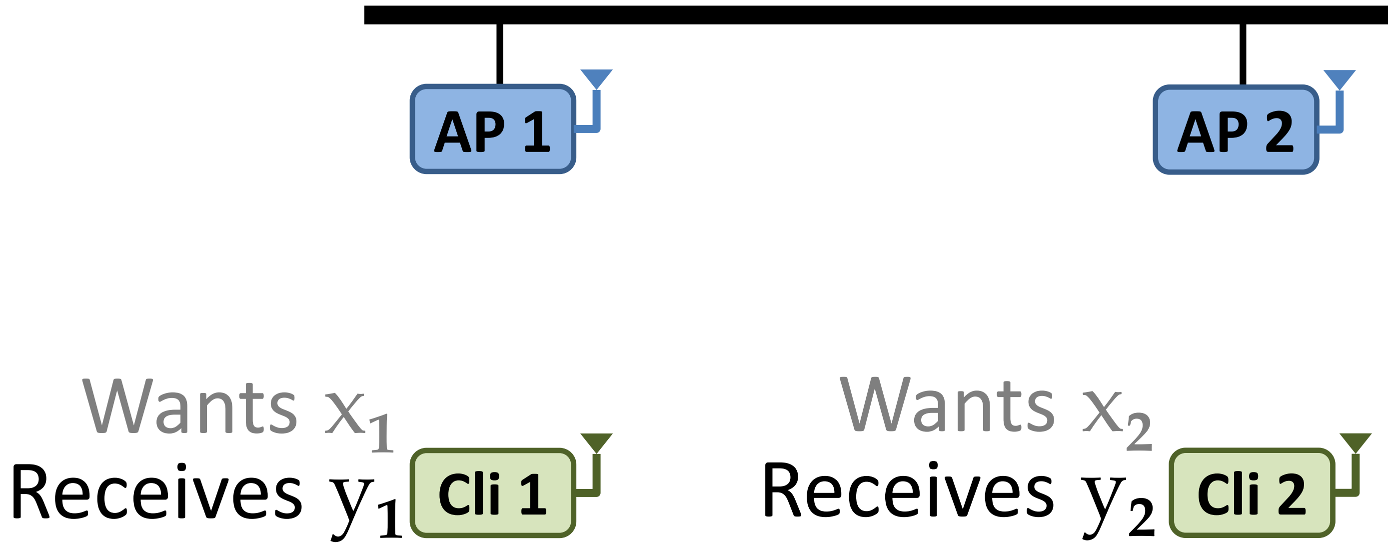
MegaMIMO = Distributed MIMO

Distributed protocol for APs to act as a huge MIMO transmitter with sum of antennas



10 APs → 10x higher throughput

Transmitting Without Interference

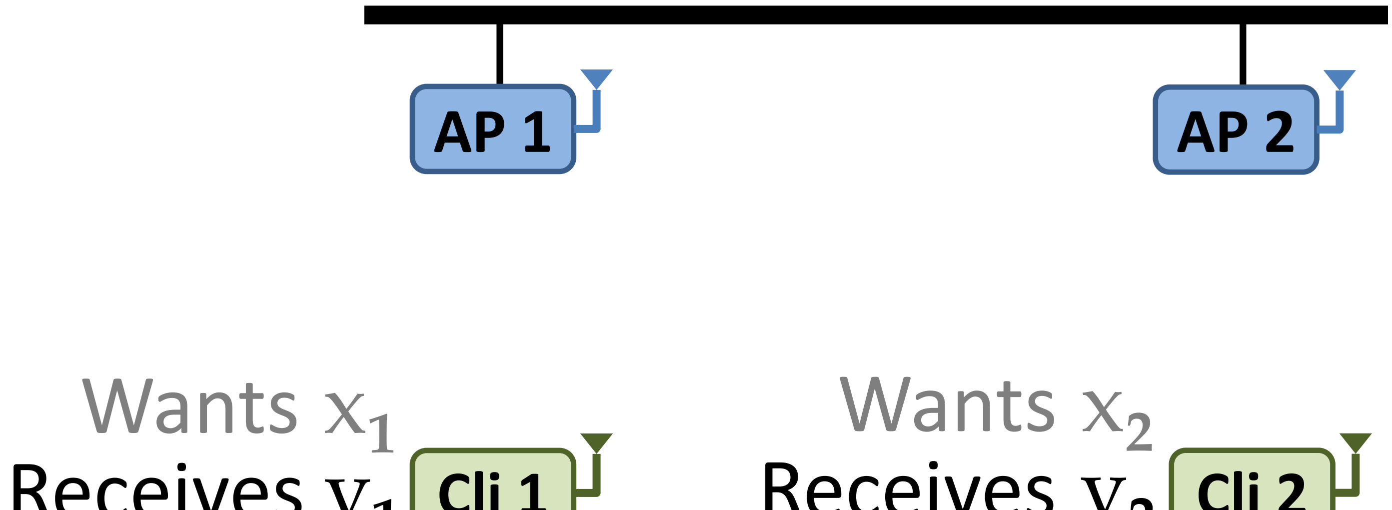


$$y_1 = d_1 x_1$$

$$y_2 = d_2 x_2$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Transmitting Without Interference



Diagonal Matrix → Non-Interference

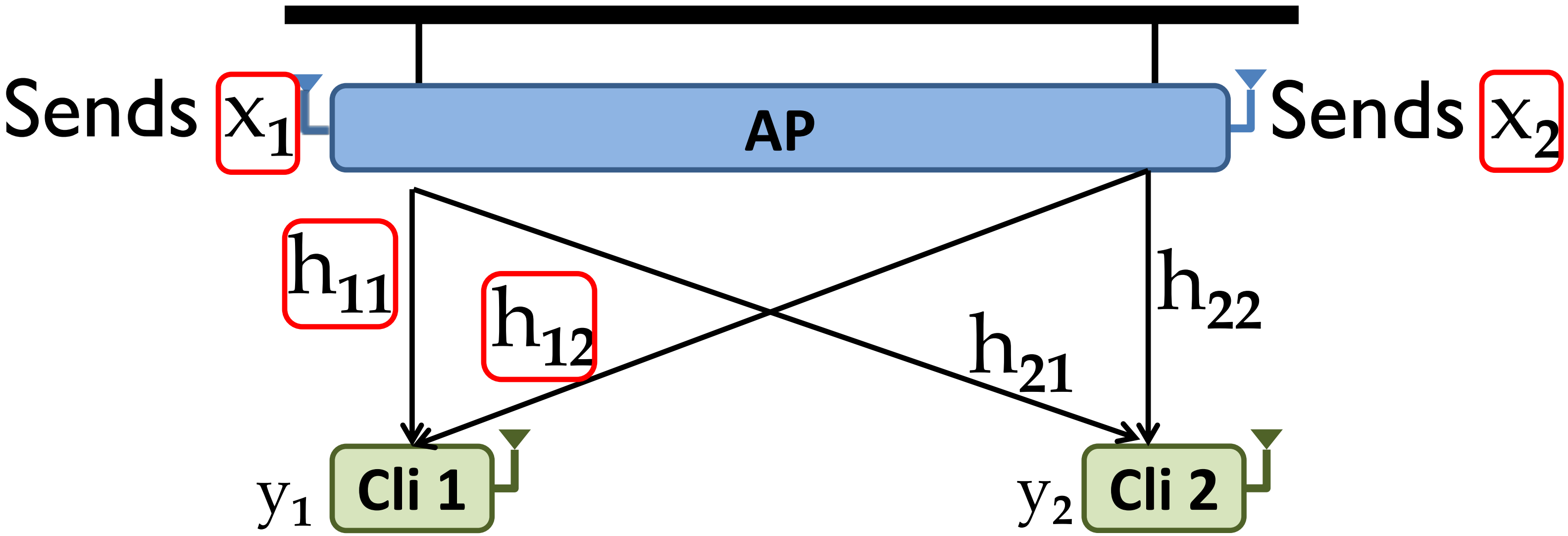
Goal: Make the effective channel matrix diagonal

On-Chip MIMO

- All antennas on the MIMO sender are synchronized in time to within nanoseconds of each other.

- All antennas on a MIMO sender have exactly the same oscillator, i.e., no frequency offset.

On-Chip MIMO



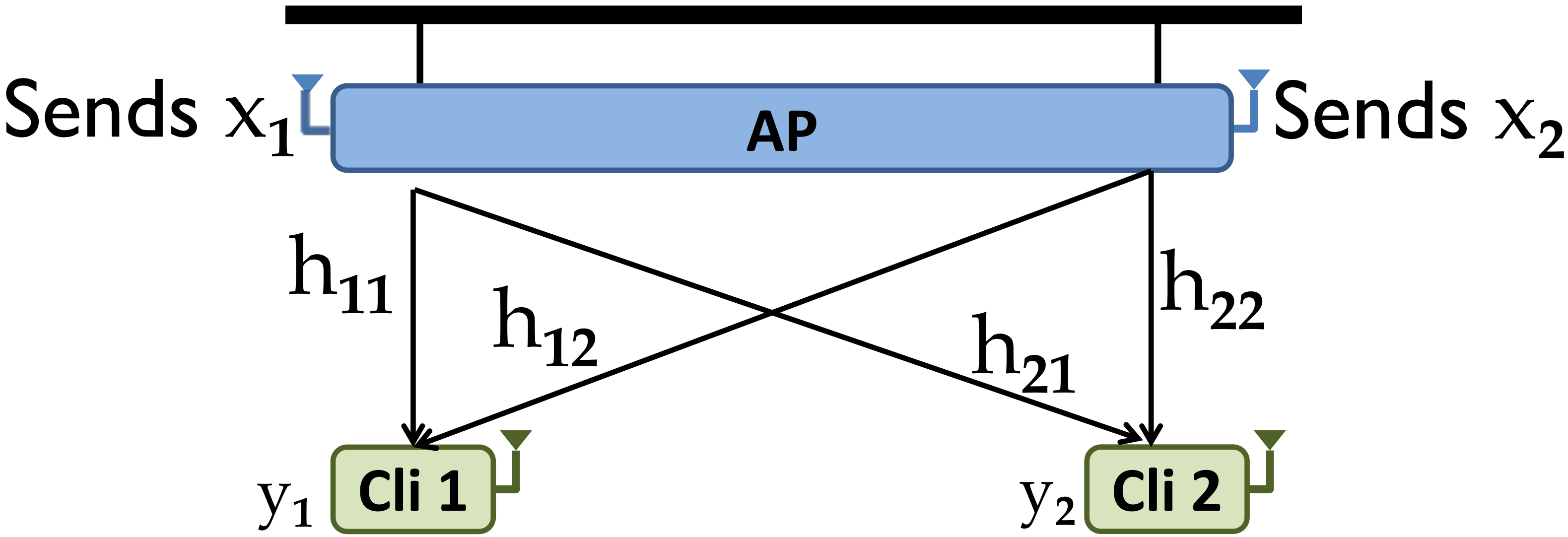
$$y_1 = h_{11} x_1 + h_{12} x_2$$

$$y_2 = h_{21} x_1 + h_{22} x_2$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Non-diagonal Matrix → Interference

On-Chip MIMO

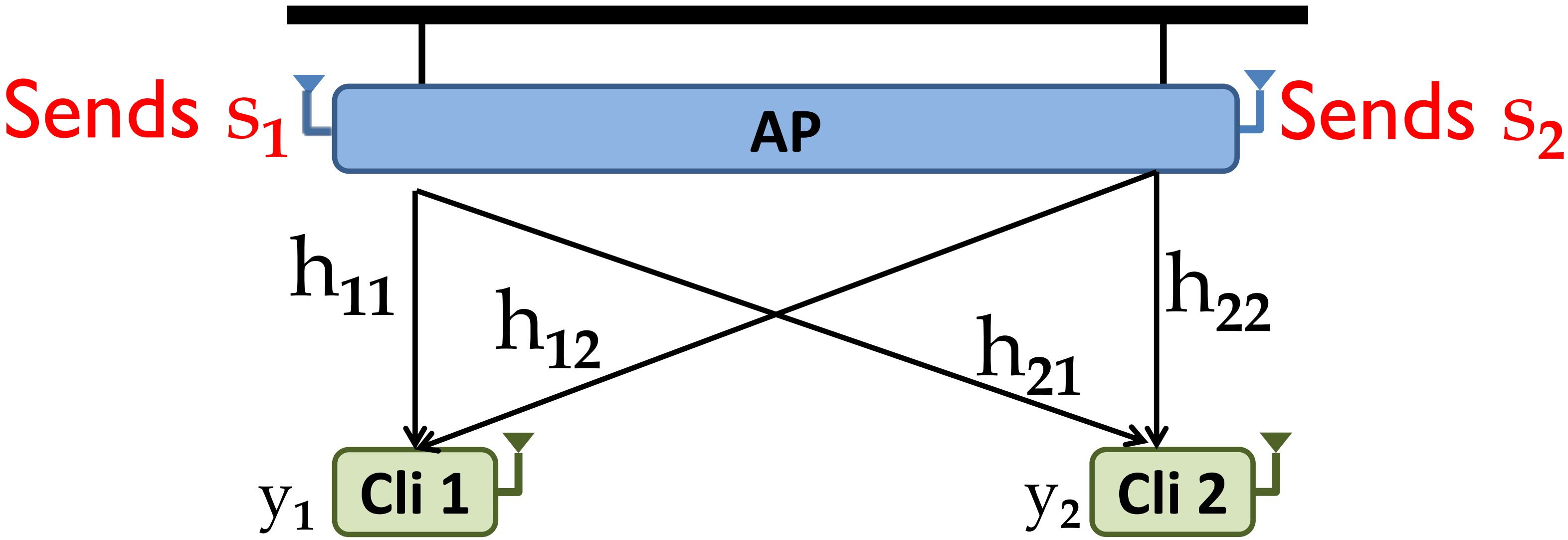


$$y_1 = h_{11} x_1 + h_{12} x_2$$

$$y_2 = h_{21} x_1 + h_{22} x_2$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

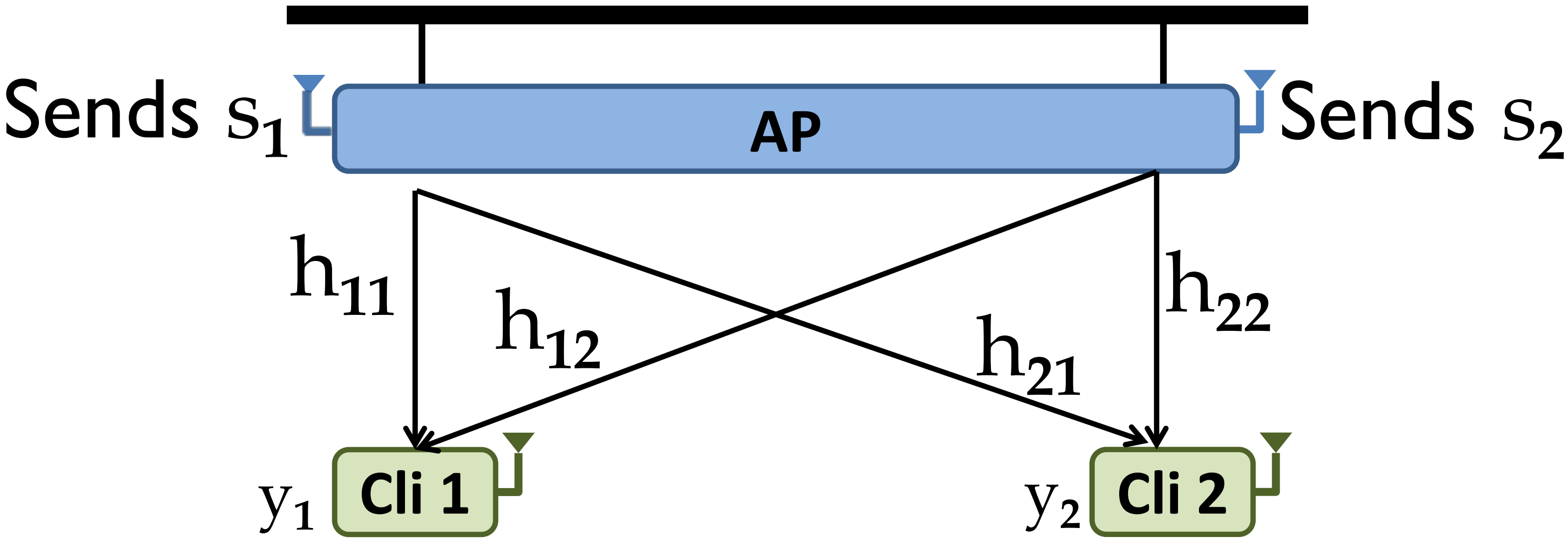
On-Chip MIMO



$$y_1 = h_{11} s_1 + h_{12} s_2$$
$$y_2 = h_{21} s_1 + h_{22} s_2$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}$$

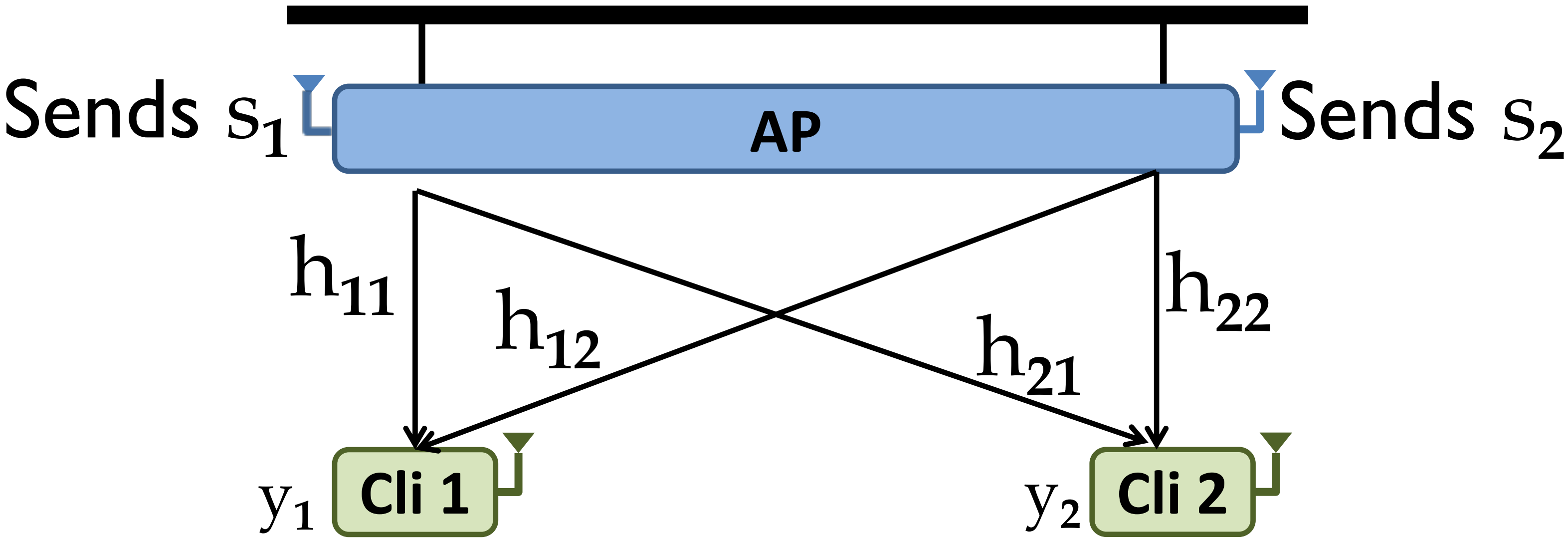
On-Chip MIMO



$$y_1 = h_{11} s_1 + h_{12} s_2$$
$$y_2 = h_{21} s_1 + h_{22} s_2$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \mathbf{H} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}$$

Making Effective Channel Matrix Diagonal

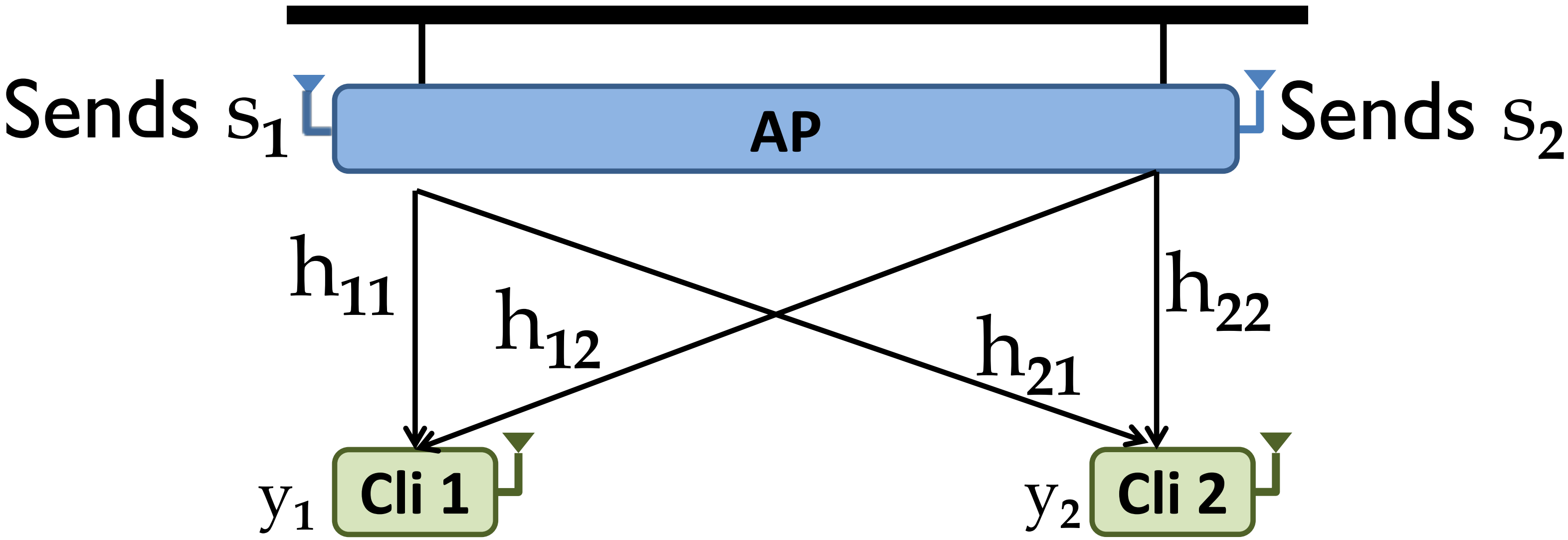


$$y_1 = h_{11} s_1 + h_{12} s_2$$

$$y_2 = h_{21} s_1 + h_{22} s_2$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \mathbf{H} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}$$

Making Effective Channel Matrix Diagonal



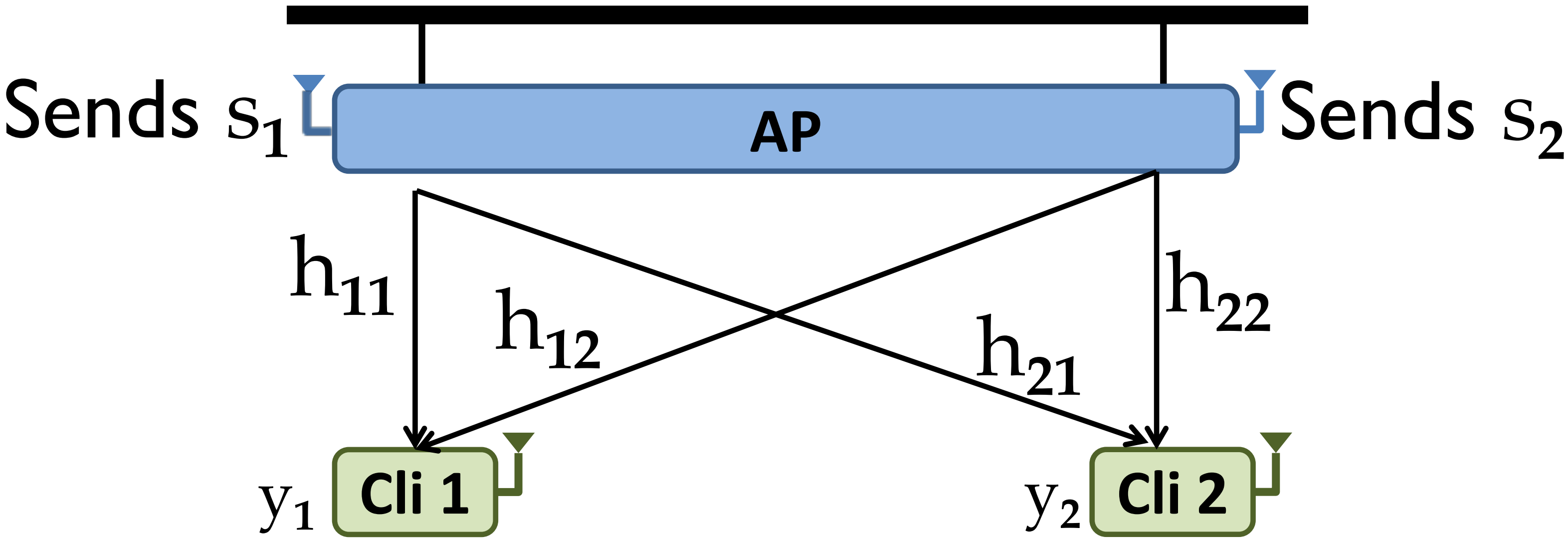
$$y_1 = h_{11} s_1 + h_{12} s_2$$

$$y_2 = h_{21} s_1 + h_{22} s_2$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \mathbf{H} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}$$

$$\mathbf{H}^{-1} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Making Effective Channel Matrix Diagonal



$$y_1 = h_{11} s_1 + h_{12} s_2$$

$$y_2 = h_{21} s_1 + h_{22} s_2$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \mathbf{H}\,\mathbf{H}^{-1} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Effective channel is diagonal

Beamforming System Description

Channel Measurement:

- Measure channels from sending antennas to clients
- Clients report measured channels back to APs

Data Transmission:

- MIMO sender computes its beamformed signal s_i using the equation $\vec{s} = H^{-1}\vec{x}$
- Clients 1 and 2 decode x_1 and x_2 independently

Distributed Transmitters Are Different

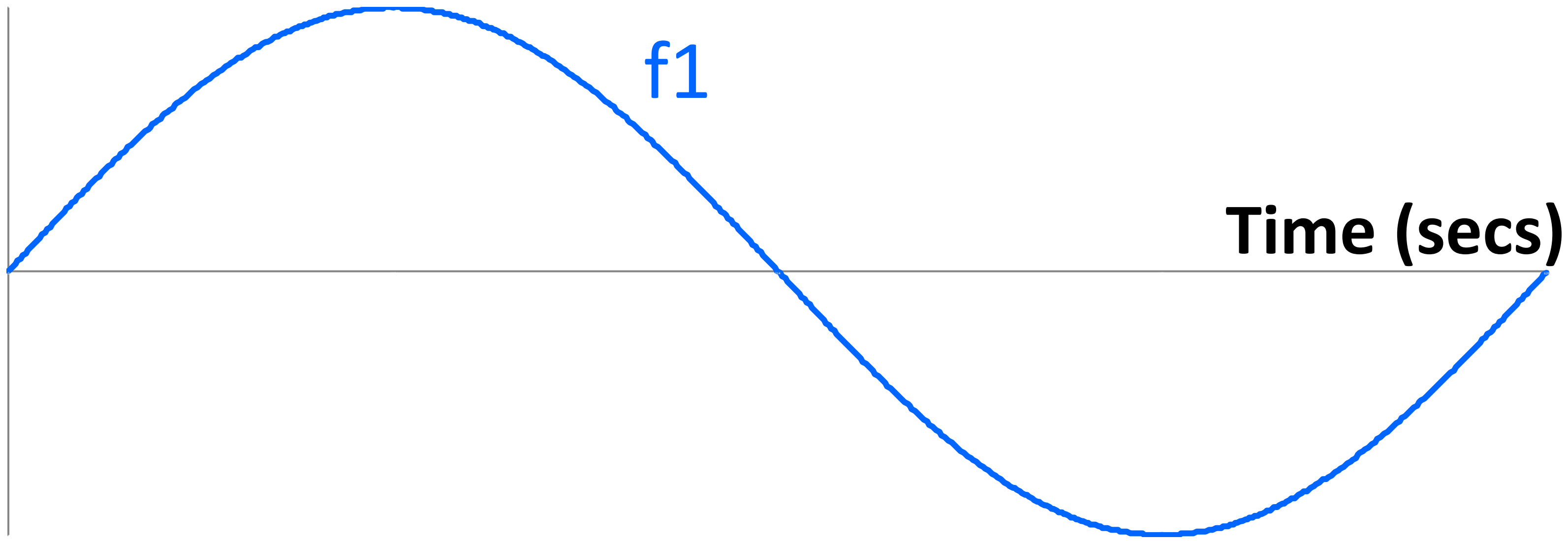
- Nodes are not synchronized in time.
  - We use SourceSync to synchronize senders within 10s of ns
  - Works for OFDM based systems like Wi-Fi, LTE etc.
- Oscillators are not synchronized and have frequency offsets relative to each other.

Synchronize Time Across APs

- Lead AP starts transmitting; Follower APs use the packet detection time to synchronize
- Challenge: Follower APs do not detect the packet at exactly the same time (difference in propagation delay, etc.)
- Solution: Each follower AP estimates the delay between its detection of the packet and the time the lead AP sent the first symbol in the packet… but How?

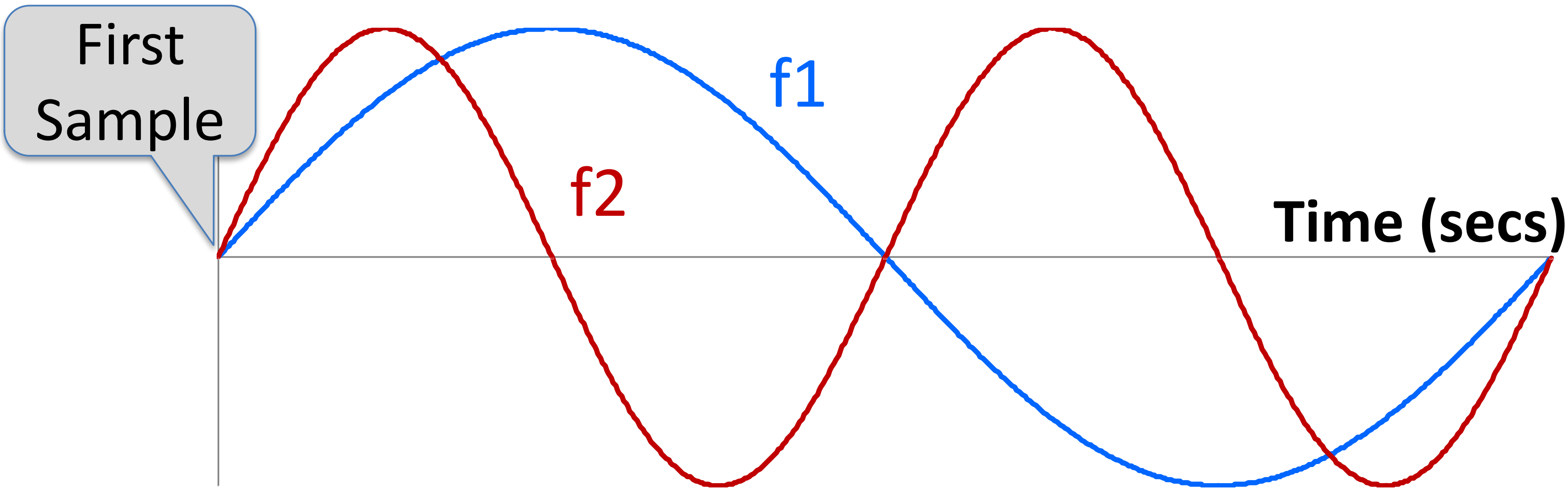
Estimating Packet Detection Delay in SourceSync

OFDM transmits signal over multiple frequencies



Estimating Packet Detection Delay in SourceSync

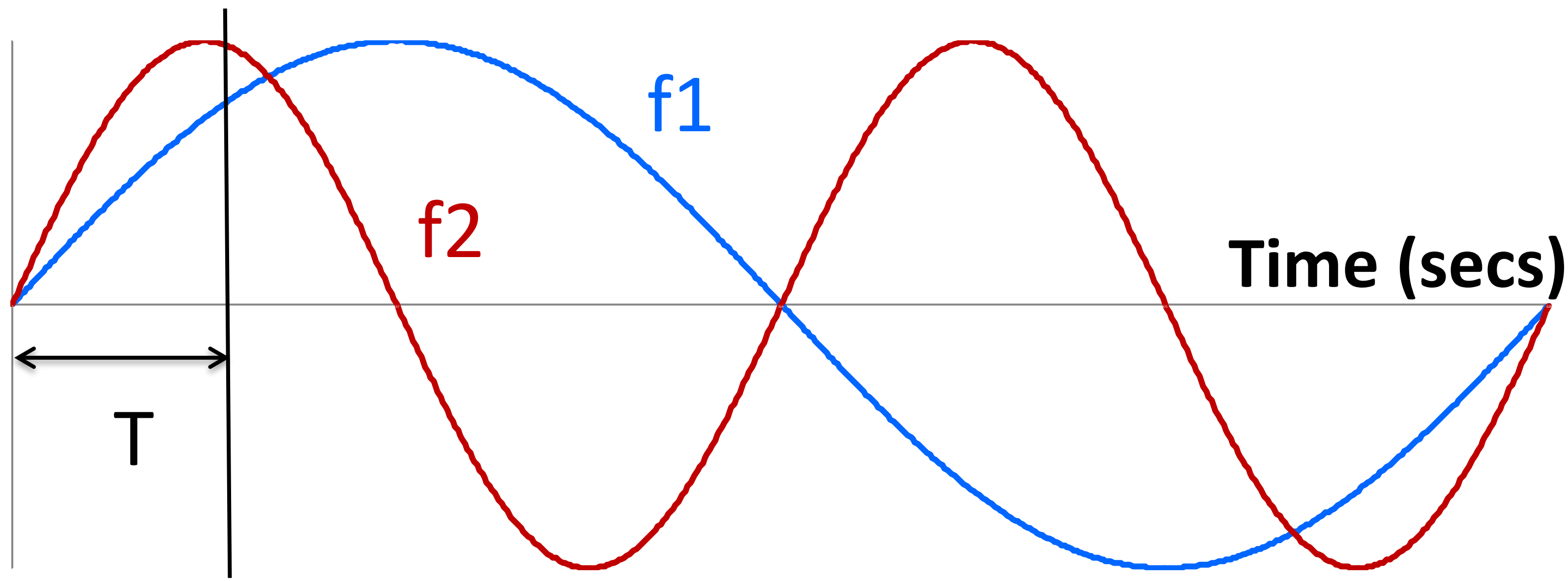
OFDM transmits signal over multiple frequencies



Detect on first sample → Same phase

Estimating Packet Detection Delay in SourceSync

OFDM transmits signal over multiple frequencies

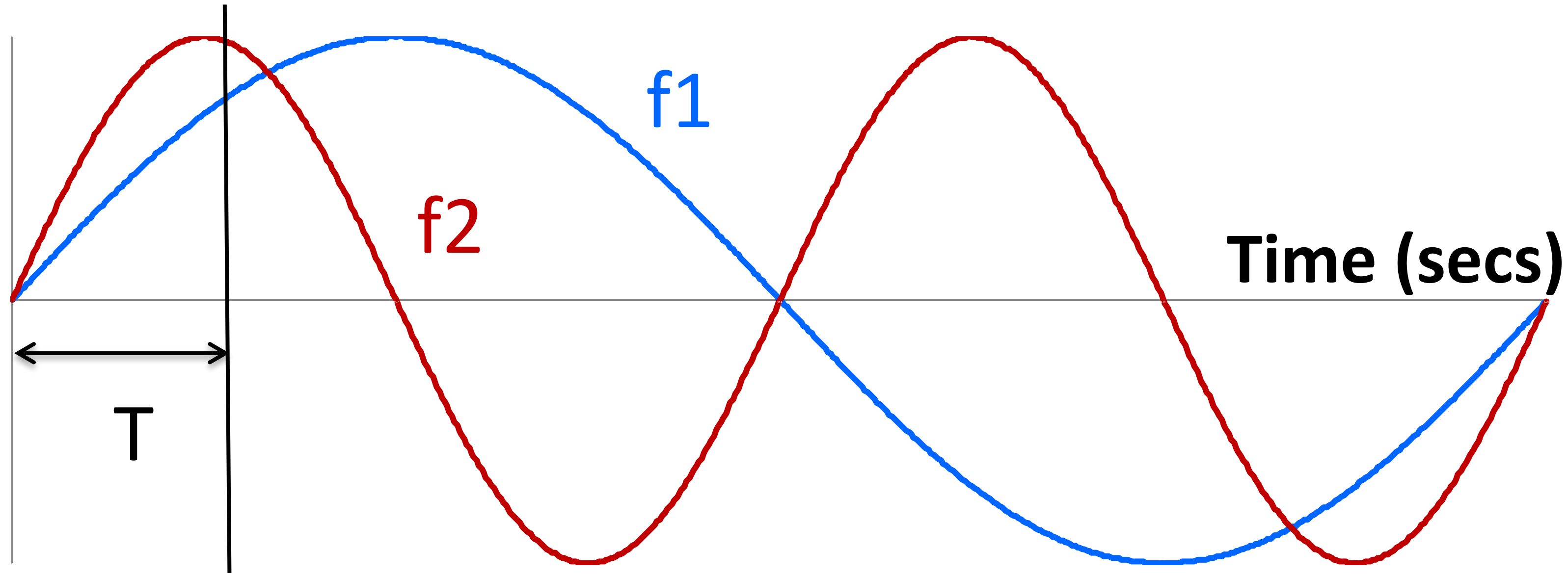


Detect after T

Frequencies rotate at different speeds

Estimating Packet Detection Delay in SourceSync

OFDM transmits signal over multiple frequencies

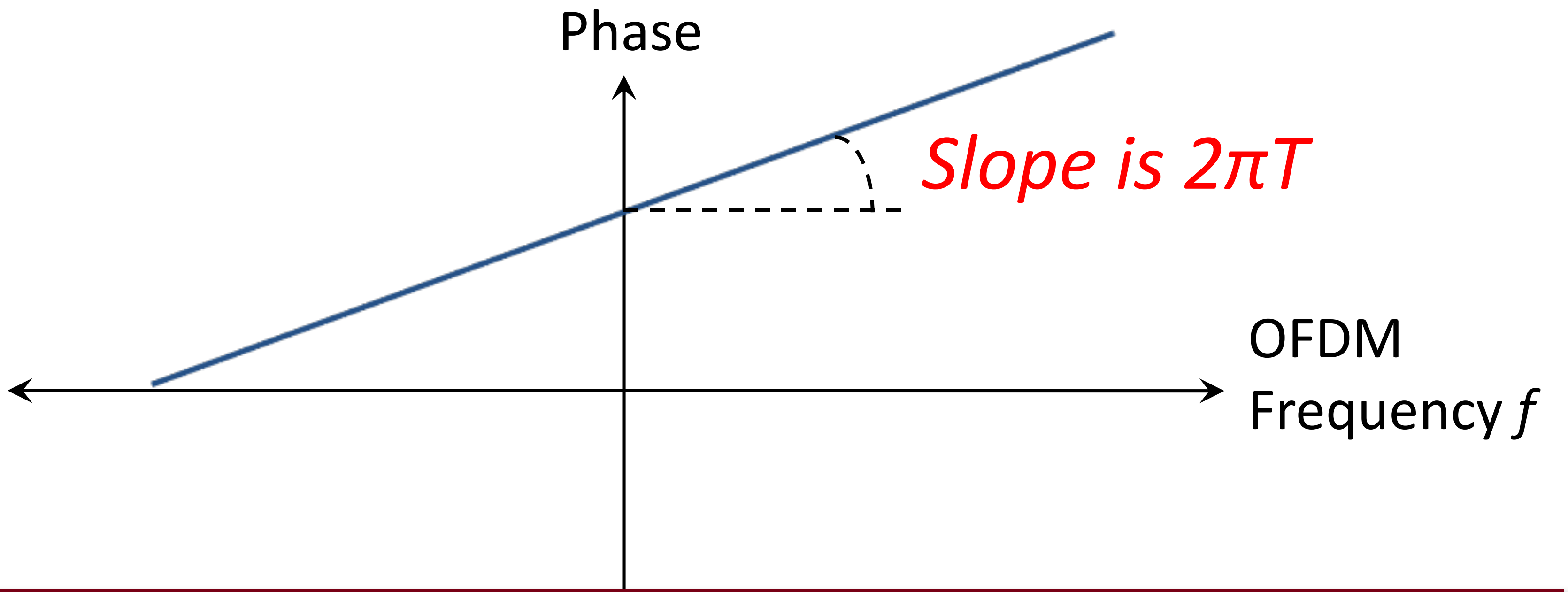


Detect after T

Different frequencies exhibit different phases

Phase = 2πfT

Estimating Packet Detection Delay in SourceSync

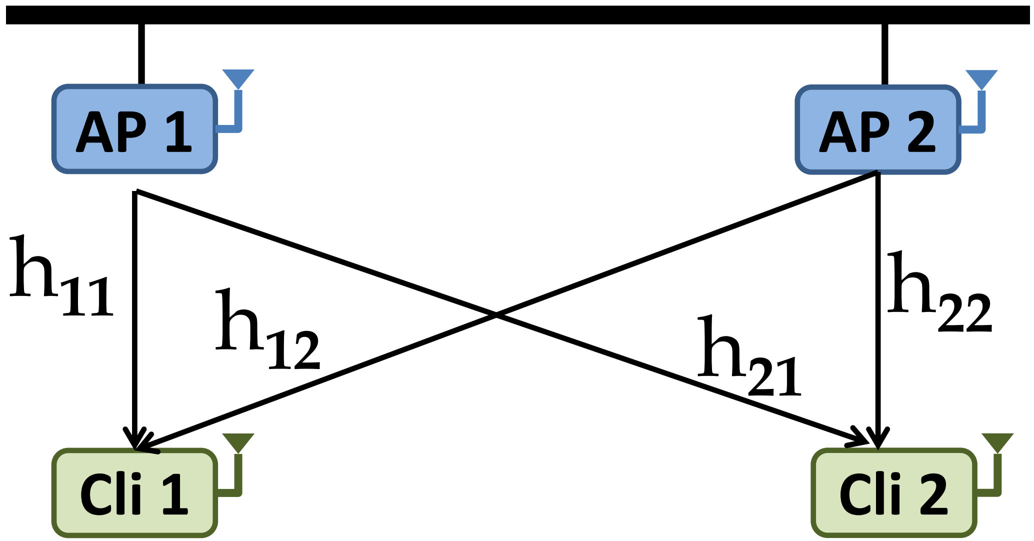


- Each AP estimates packet detection delay
- Estimate uses every symbol in packet → Robust to noise

Distributed Transmitters Are Different

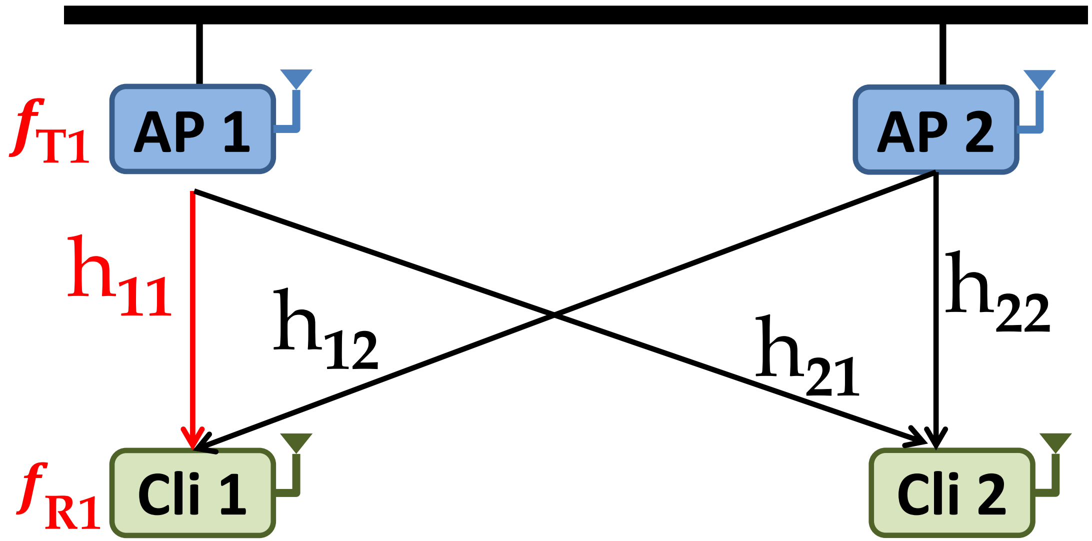
- Nodes are not synchronized in time.
  - Used SourceSync to synchronize senders within 10s of ns
  - Works for OFDM based systems like Wi-Fi, LTE etc.
- Oscillators are not synchronized and have frequency offsets relative to each other.

What Happens with Independent Oscillators?



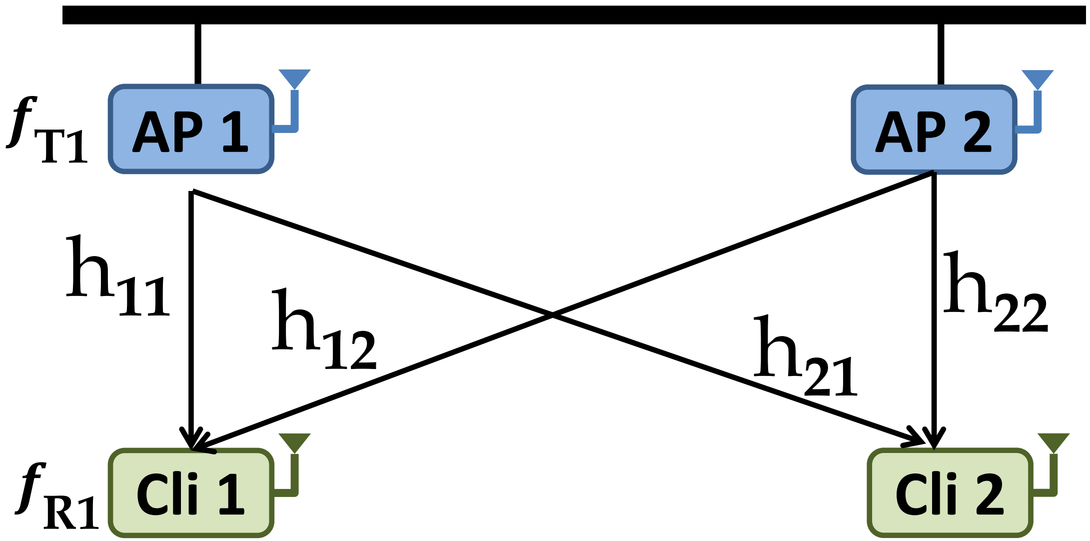
$$\begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}$$

What Happens with Independent Oscillators?



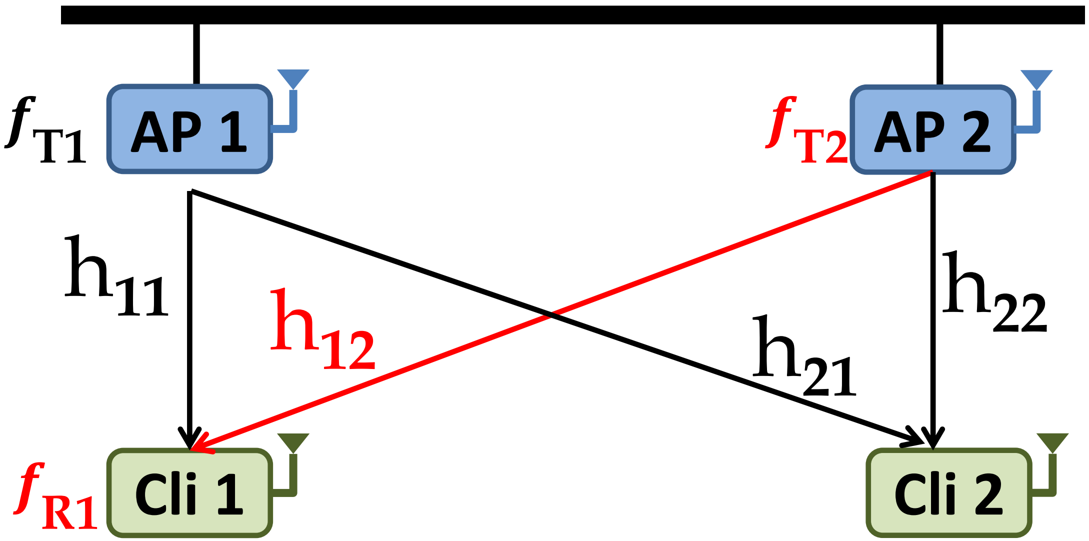
$$\begin{bmatrix} h_{11}\, e^{j2\pi(f_{T1} - f_{R1})t} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}$$

What Happens with Independent Oscillators?



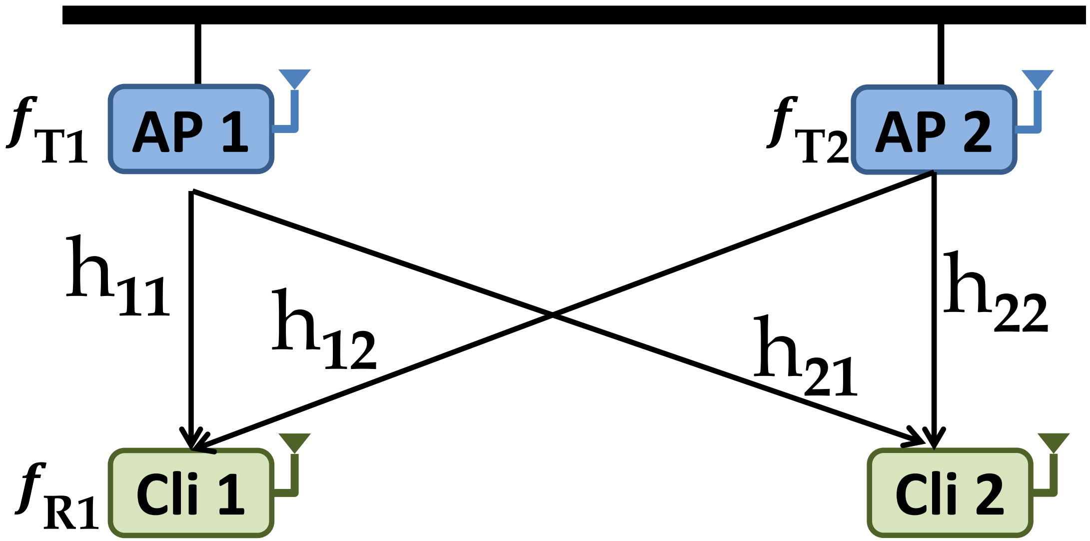
$$\begin{bmatrix} h_{11}\,e^{j2\pi(f_{T1} - f_{R1})t} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}$$

What Happens with Independent Oscillators?



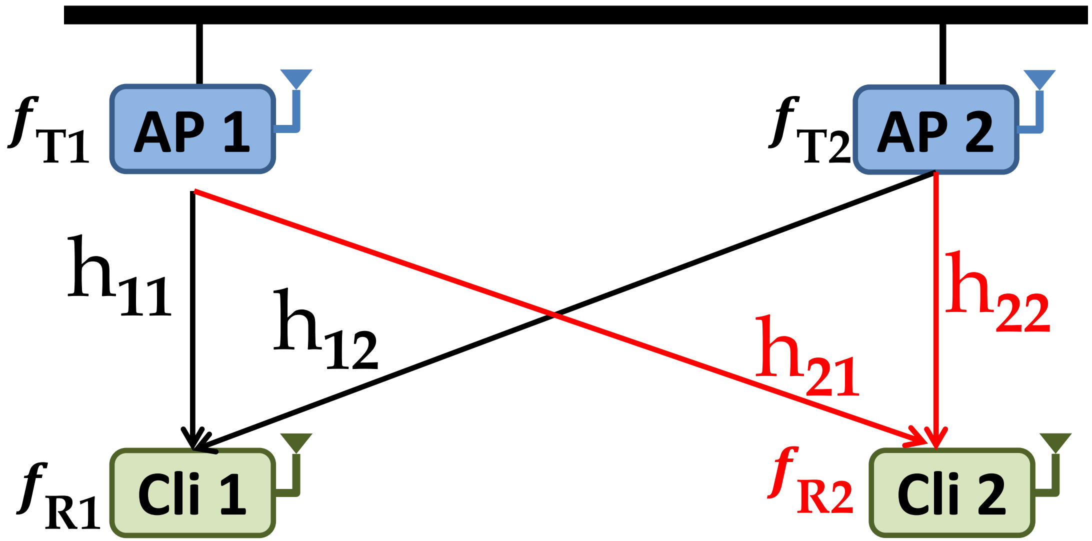
$$\begin{bmatrix} h_{11}\, e^{j2\pi (f_{T1} - f_{R1})t} & h_{12}\, e^{j2\pi (f_{T2} - f_{R1})t} \\ h_{21} & h_{22} \end{bmatrix}$$

What Happens with Independent Oscillators?



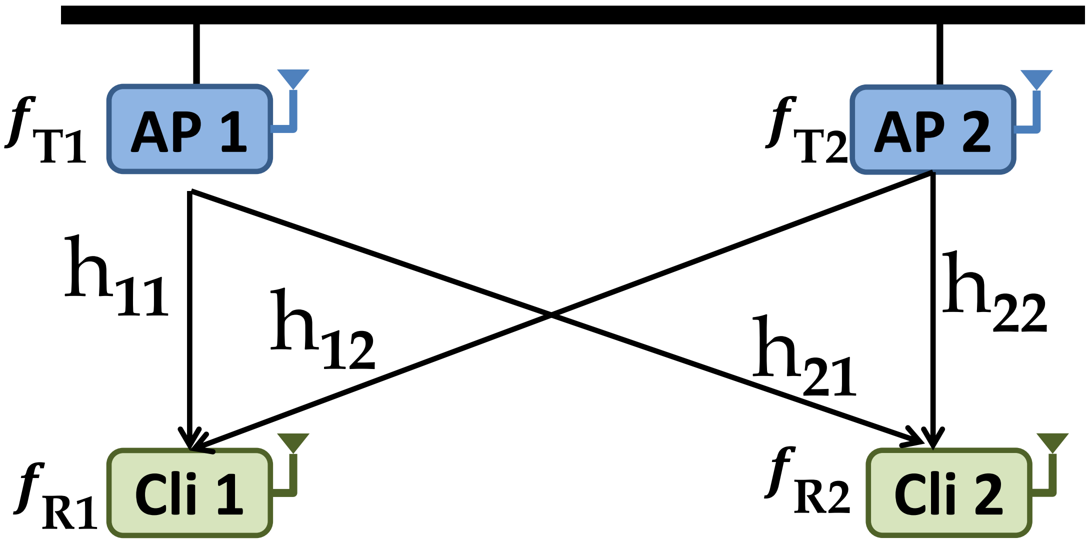
$$\begin{bmatrix} h_{11}\, e^{j2\pi\,(f_{T1}\,-\,f_{R1})t} & h_{12}\, e^{j2\pi(f_{T2}\,-\,f_{R1})t} \\ h_{21} & h_{22} \end{bmatrix}$$

What Happens with Independent Oscillators?



$$\begin{bmatrix} h_{11} e^{j2\pi (f_{T1} - f_{R1})t} & h_{12} e^{j2\pi (f_{T2} - f_{R1})t} \\ h_{21} e^{j2\pi (f_{T1} - f_{R2})t} & h_{22} e^{j2\pi (f_{T2} - f_{R2})t} \end{bmatrix}$$

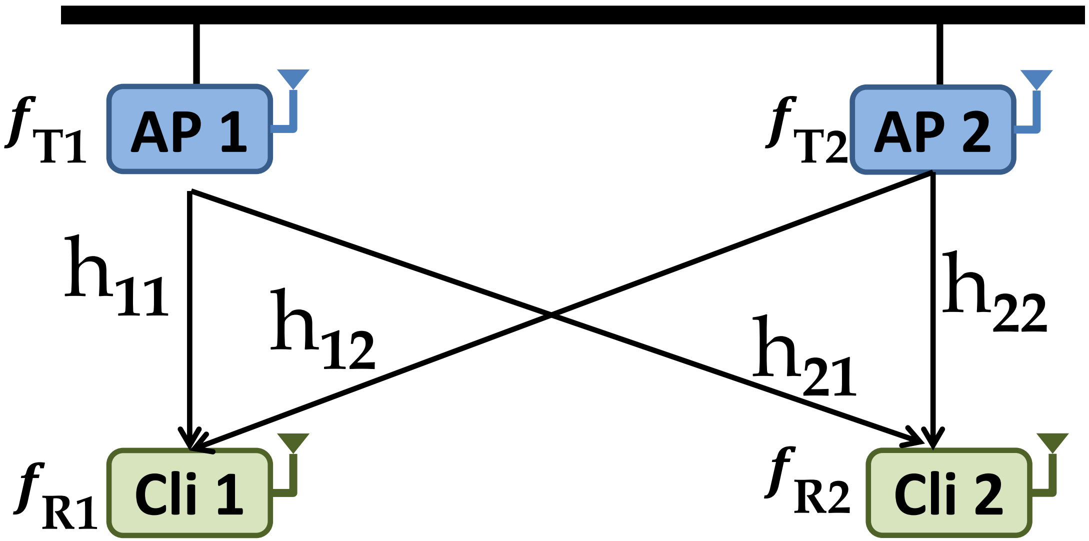
What Happens with Independent Oscillators?



$$\begin{bmatrix} h_{11}\, e^{j2\pi (f_{T1} - f_{R1})t} & h_{12}\, e^{j2\pi (f_{T2} - f_{R1})t} \\ h_{21}\, e^{j2\pi (f_{T1} - f_{R2})t} & h_{22}\, e^{j2\pi (f_{T2} - f_{R2})t} \end{bmatrix}$$

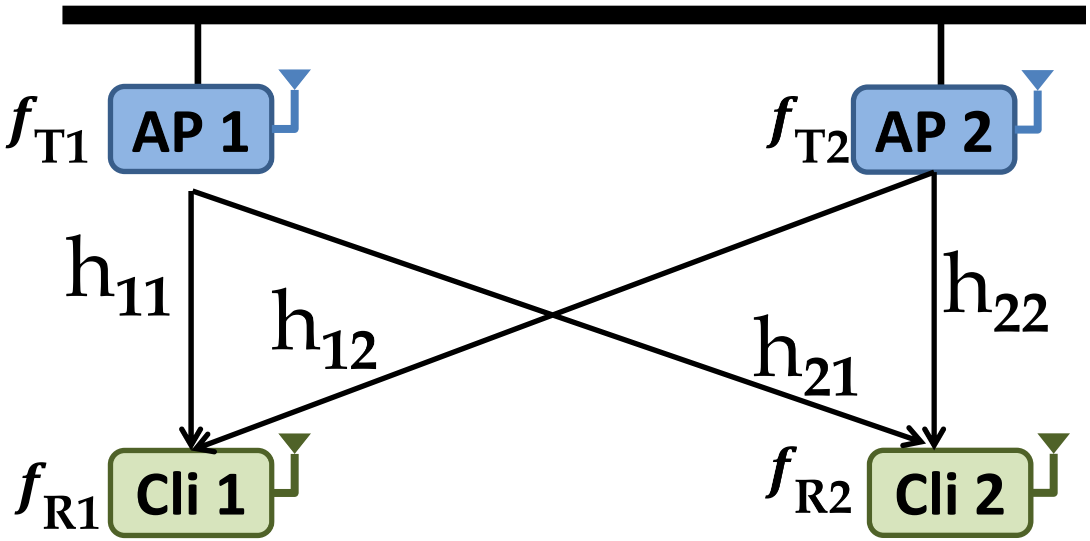
Time Varying

Channel is Time Varying



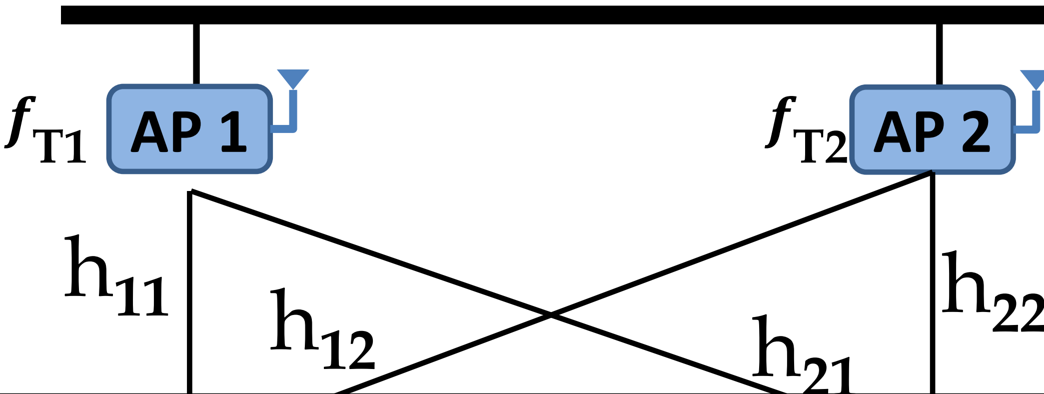
H(t)

Does Traditional Beamforming Still Work?



$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \mathbf{H(t)} \begin{bmatrix} s_1(t) \\ s_2(t) \end{bmatrix}$$

Does Traditional Beamforming Still Work?



Beamforming does not work

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \mathbf{H(t)}\,\mathbf{H^{-1}} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

Not Diagonal

Challenge

Channel is Rapidly Time Varying

↓

Relative Channel Phases of Transmitted Signals Changes Rapidly With Time

Prevents Beamforming

Distributed Phase Synchronization

High Level Intuition:

- Pick one AP as the lead
- All other APs are followers
  - Imitate the behavior of the lead AP by fixing the rotation of their oscillator relative to the lead.

Decomposing **H(t)**

$$\begin{bmatrix} h_{11}e^{j2\pi(f_{T1} - f_{R1})t} & h_{12}e^{j2\pi(f_{T2} - f_{R1})t} \\ h_{21}e^{j2\pi(f_{T1} - f_{R2})t} & h_{22}e^{j2\pi(f_{T2} - f_{R2})t} \end{bmatrix}$$

$$\begin{bmatrix} e^{-j2\pi f_{R1}t} & 0 \\ 0 & e^{-j2\pi f_{R2}t} \end{bmatrix} \begin{bmatrix} h_{11}\,e^{j2\pi(f_{T1})t} & h_{12}\,e^{j2\pi(f_{T2})t} \\ h_{21}\,e^{j2\pi(f_{T1})t} & h_{22}\,e^{j2\pi(f_{T2})t} \end{bmatrix}$$

Decomposing $\mathbf{H(t)}$

$$\begin{bmatrix} e^{-j2\pi f_{R1}t} & 0 \\ 0 & e^{-j2\pi f_{R2}t} \end{bmatrix} \begin{bmatrix} h_{11} e^{j2\pi(f_{T1})t} & h_{12} e^{j2\pi(f_{T2})t} \\ h_{21} e^{j2\pi(f_{T1})t} & h_{22} e^{j2\pi(f_{T2})t} \end{bmatrix}$$

Decomposing $\mathbf{H(t)}$

$$\begin{bmatrix} \mathbf{e}^{-j2\pi f_{R1}t} & 0 \\ 0 & \mathbf{e}^{-j2\pi f_{R2}t} \end{bmatrix} \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} \mathbf{e}^{j2\pi(f_{T1})t} & 0 \\ 0 & \mathbf{e}^{j2\pi(f_{T2})t} \end{bmatrix}$$

Decomposing $\mathbf{H(t)}$

$$\begin{bmatrix} e^{-j2\pi f_{R1} t} & 0 \\ 0 & e^{-j2\pi f_{R2} t} \end{bmatrix} \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} e^{j2\pi (f_{T1}) t} & 0 \\ 0 & e^{j2\pi (f_{T2}) t} \end{bmatrix}$$

Decomposing **H(t)**

$$\begin{bmatrix} \mathbf{e}^{-j2\pi f_{R1}t} & 0 \\ 0 & \mathbf{e}^{-j2\pi f_{R2}t} \end{bmatrix} \mathbf{H} \begin{bmatrix} \mathbf{e}^{j2\pi (f_{T1})t} & 0 \\ 0 & \mathbf{e}^{j2\pi (f_{T2})t} \end{bmatrix}$$

Diagonal

Devices cannot track their own oscillator phases…

Decomposing $\mathbf{H(t)}$

$$e^{j2\pi(f_{T1})t} \begin{bmatrix} e^{-j2\pi f_{R1} t} & 0 \\ 0 & e^{-j2\pi f_{R2} t} \end{bmatrix} \mathbf{H} \begin{bmatrix} e^{j2\pi(f_{T1})t} & 0 \\ 0 & e^{j2\pi(f_{T2})t} \end{bmatrix} e^{-j2\pi(f_{T1})t}$$

Decomposing $\mathbf{H(t)}$

$$\underbrace{\begin{bmatrix} e^{j2\pi(f_{T1}-f_{R1})t} & 0 \\ 0 & e^{j2\pi(f_{T1}-f_{R2})t} \end{bmatrix}}_{\mathbf{R(t)}} \mathbf{H} \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & e^{j2\pi(f_{T2}-f_{T1})t} \end{bmatrix}}_{\mathbf{T(t)}}$$

R(t)

T(t)

Depends only on transmitters

Decomposing **H(t)**

$$\underbrace{\begin{bmatrix} e^{j2\pi(f_{T1} - f_{R1})t} & 0 \\ 0 & e^{j2\pi(f_{T1} - f_{R2})t} \end{bmatrix}}_{\mathbf{R(t)}} \mathbf{H} \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & e^{j2\pi(f_{T2} - f_{T1})t} \end{bmatrix}}_{\mathbf{T(t)}}$$

R(t) **T(t)**

$$\mathbf{H(t) = R(t).H.T(t)}$$

Beamforming with Different Oscillators

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \mathbf{R(t).H.T(t)} \begin{bmatrix} s_1(t) \\ s_2(t) \end{bmatrix}$$

$$\begin{bmatrix} s_1(t) \\ s_2(t) \end{bmatrix} = \mathbf{T(t)^{-1}H^{-1}} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

Beamforming with Different Oscillators

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \mathbf{R(t)}.\mathbf{H}.\mathbf{T(t)}\,\mathbf{T(t)}^{-1}\mathbf{H}^{-1} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

Diagonal

$$\begin{bmatrix} s_1(t) \\ s_2(t) \end{bmatrix} = \mathbf{T(t)}^{-1}\mathbf{H}^{-1} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

Transmitter Compensation

$$\mathbf{T(t)} = \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{e}^{\mathbf{j2\pi(}f_{\mathrm{T2}} - f_{\mathrm{T1}}\mathbf{)t}} \end{bmatrix}$$

Transmitter Compensation

$$\mathbf{T(t)^{-1}} = \begin{bmatrix} 1 & 0 \\ 0 & e^{-j2\pi(f_{T2} - f_{T1})t} \end{bmatrix}$$

Follower AP imitates lead by multiplying each sample by oscillator rotation relative to lead

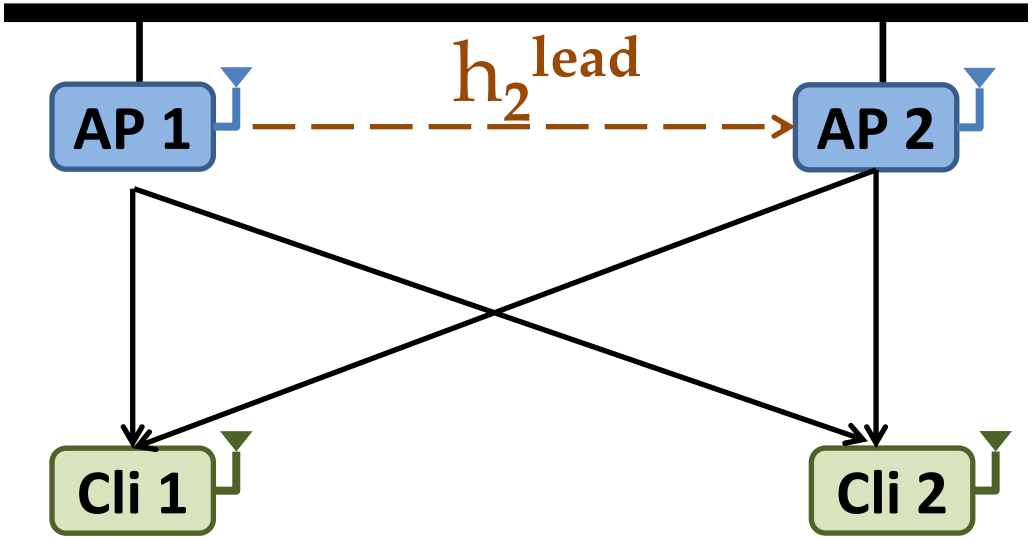
Requires only local information → Fully distributed

Measuring Phase Offset

- Multiply frequency offset by elapsed time
- Requires very accurate estimation of frequency offset
  - Error of 25 Hz (10 parts per BILLION) changes complete alignment to complete misalignment in 20 ms.

Need to keep resynchronizing to avoid error accumulation

Resynchronization



$$h_2^{lead}(t) = h_2^{lead}\, e^{j2\pi(f_{T2} - f_{T1})t}$$

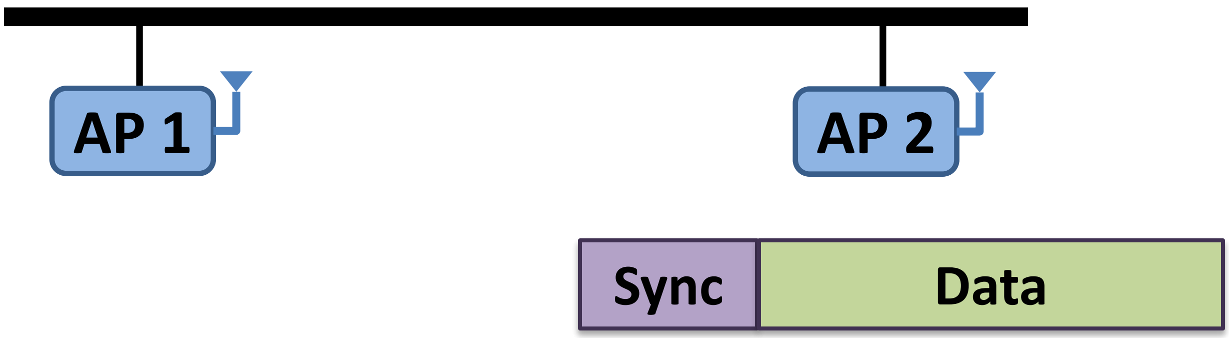
Directly compute phase at each follower AP by measuring channel from lead

Resynchronization

Lead AP:

– Prefixes data transmission with synchronization header

Resynchronization



Follower AP:

- Receives Synchronization Header
- Corrects for change in channel phase from lead
- Transmits data

Receiver Compensation

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \mathbf{R(t).H.T(t)} \qquad \begin{bmatrix} s_1(t) \\ s_2(t) \end{bmatrix}$$

Receiver Compensation

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \mathbf{R(t).H.T(t)\ T(t)^{-1}H^{-1}} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \mathbf{R(t)^{-1}\ R(t)} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

Receiver Compensation

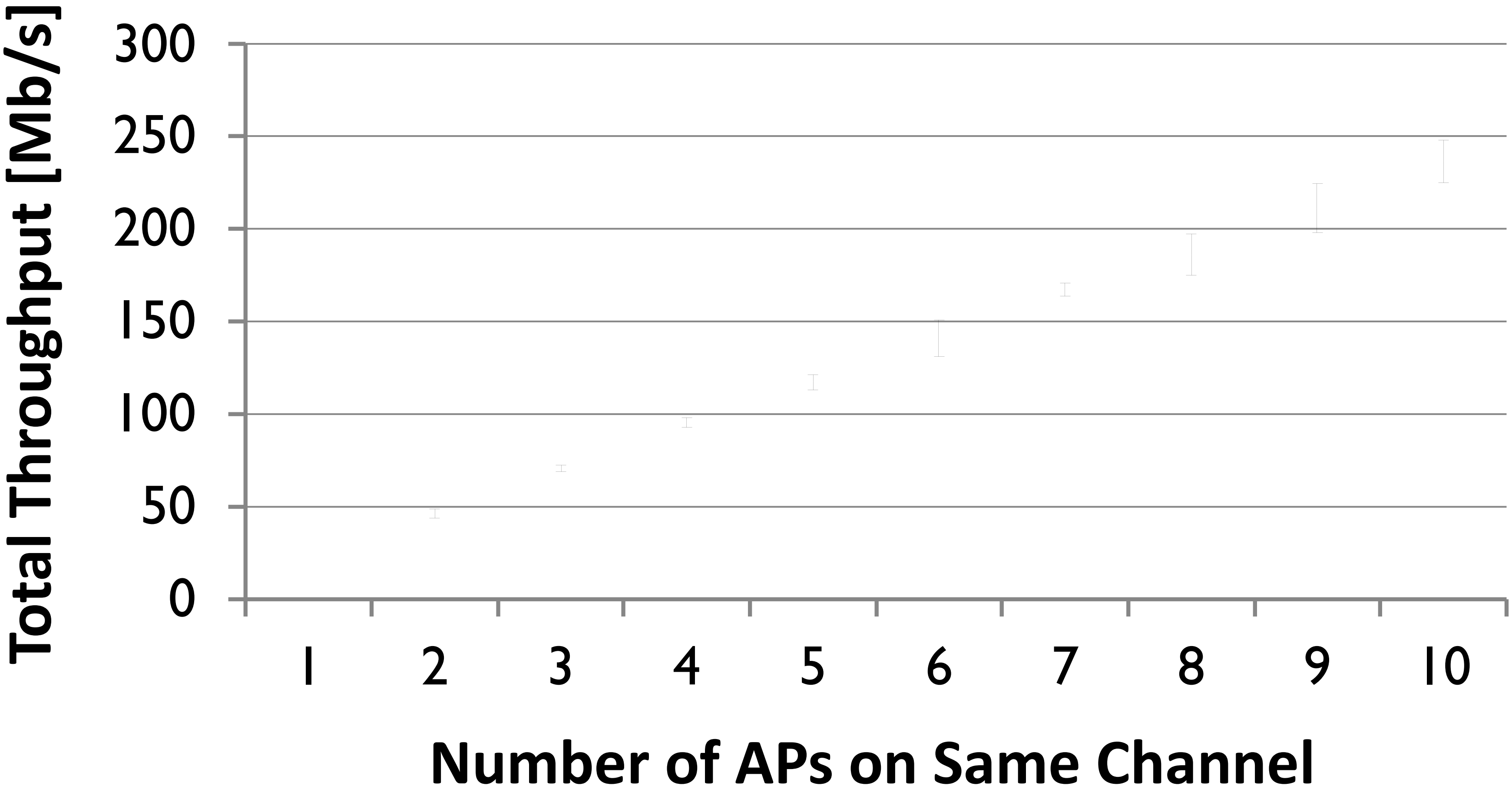
$$\mathbf{R(t)} = \begin{bmatrix} e^{j2\pi(f_{T1} - f_{R1})t} & 0 \\ 0 & e^{j2\pi(f_{T1} - f_{R2})t} \end{bmatrix}$$

Receiver Compensation

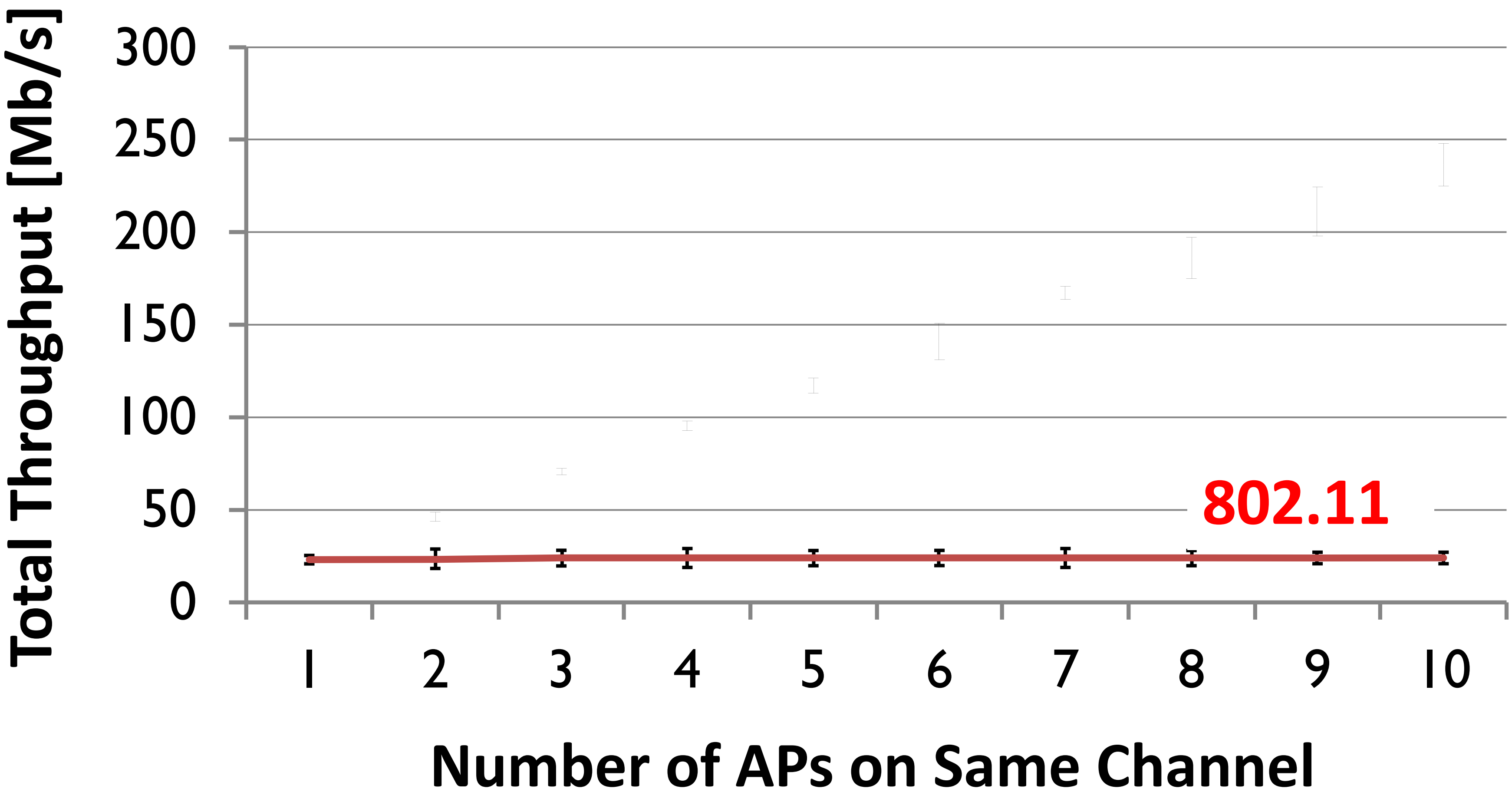
$$\mathbf{R(t)^{-1}} = \begin{bmatrix} e^{-j2\pi (f_{T1} - f_{R1})t} & 0 \\ 0 & e^{-j2\pi (f_{T1} - f_{R2})t} \end{bmatrix}$$

Receiver does what it does today – correct for oscillator offset from lead

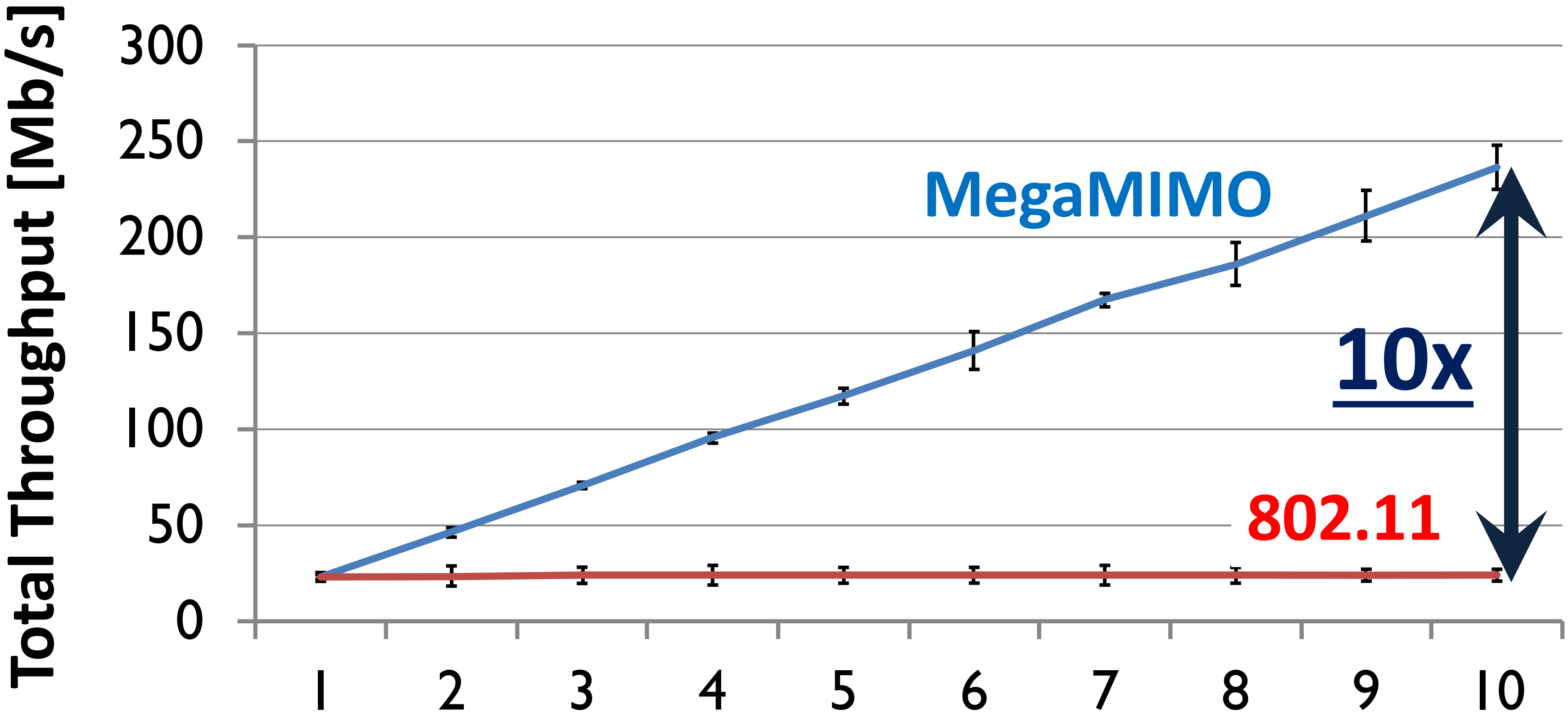
Does MegaMIMO Scale Throughput with the Number of Users?



Does MegaMIMO Scale Throughput with the Number of Users?



Does MegaMIMO Scale Throughput with the Number of Users?



10x throughput gain over existing Wi-Fi

AirShare

transmits the reference clock over the air

&

eliminates CFO

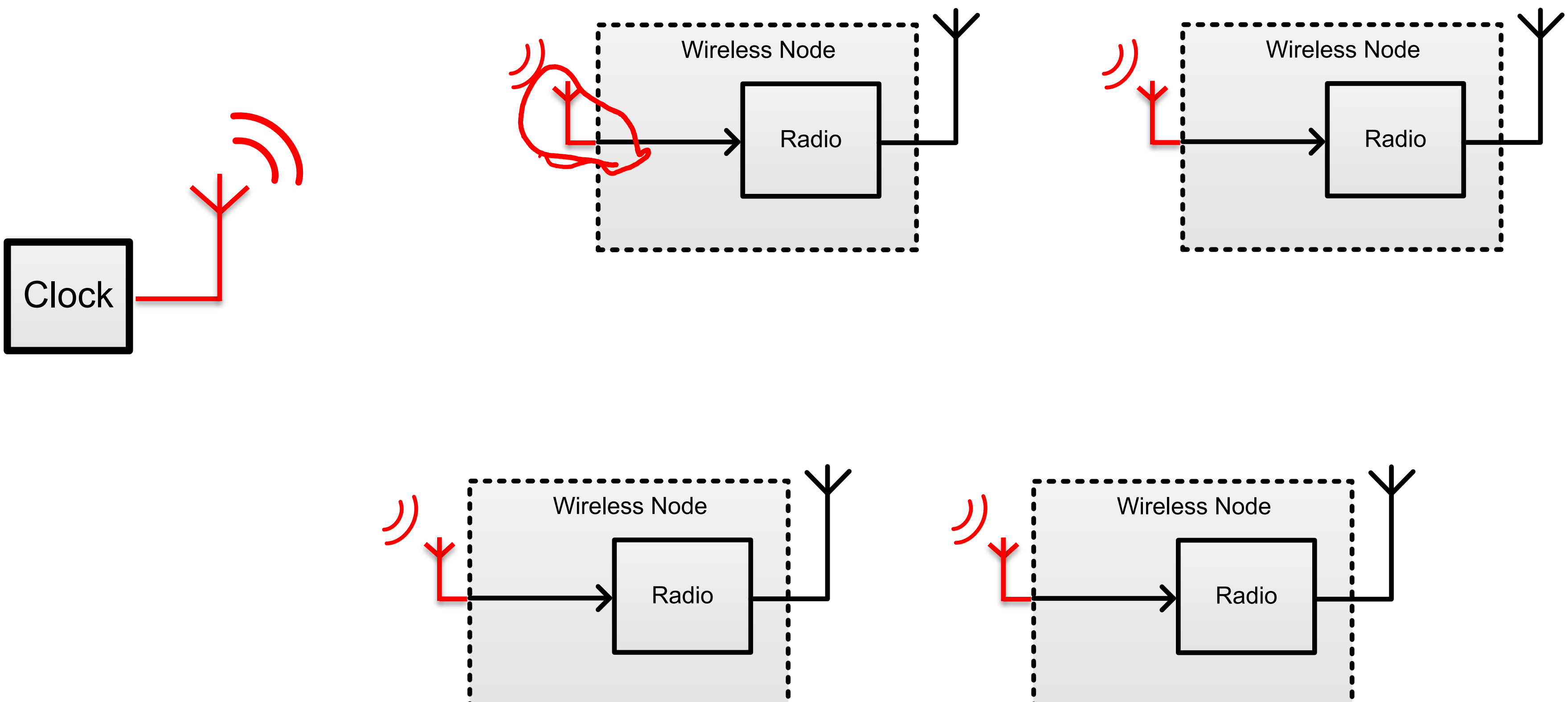
Protocol independent

Supports mobility

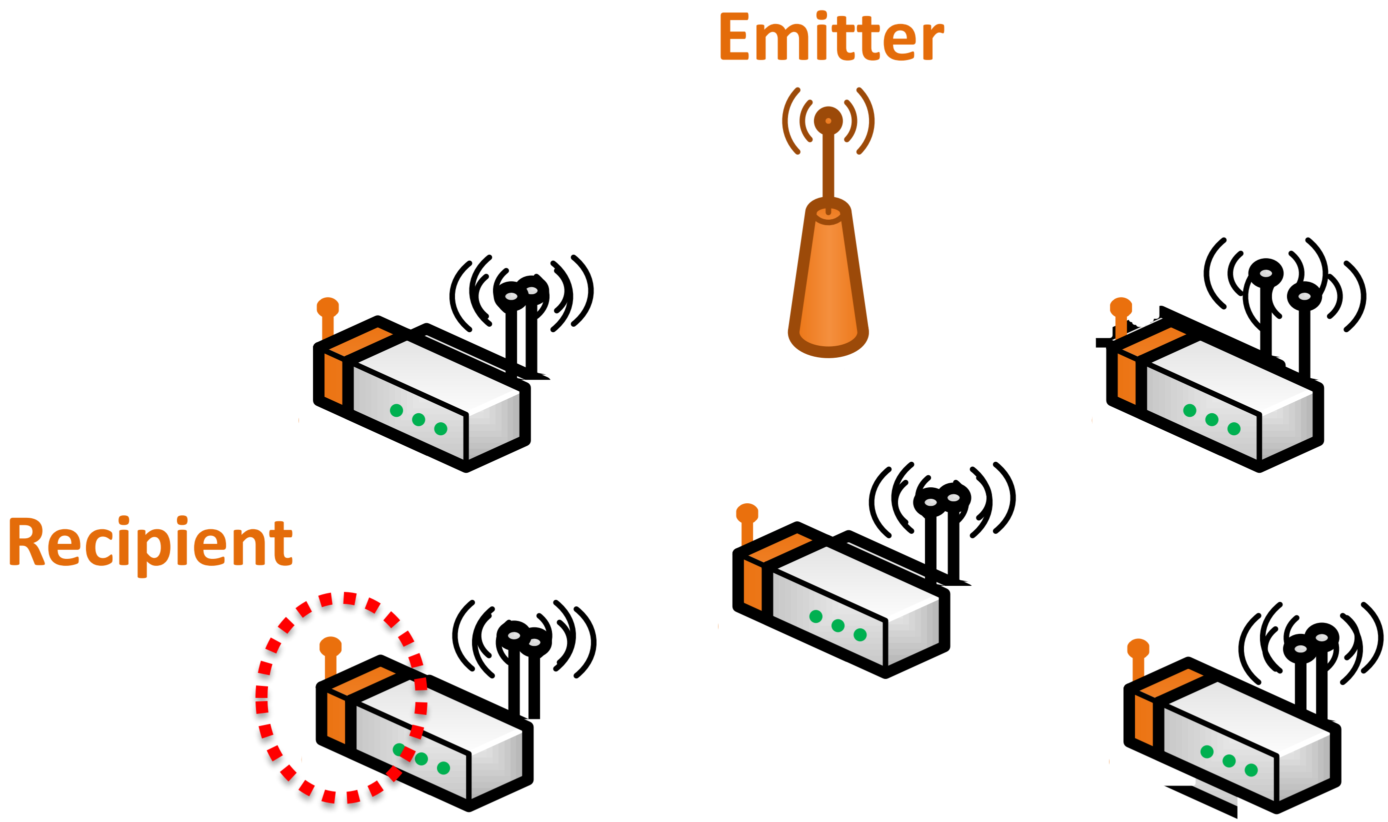
Cheap and Low-Power

Idea

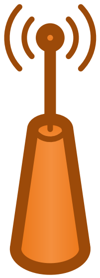
Transmit a reference over-the-air



AirShare Architecture



Emitter



How can emitter transmit a clock?

<u>Problem:</u> Reference clocks are typically 10-40 MHz

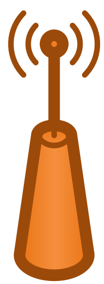
- FCC forbids transmitting such a low-frequency signal

- Requires large antennas

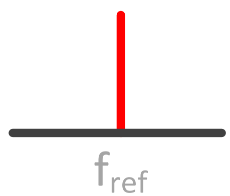
Transmit a Differential-reference

Instead of transmitting a signal at the clock frequency (10 MHz)

Transmits two signals separated by the clock frequency

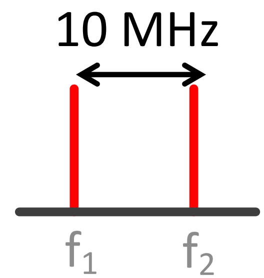


Emitter



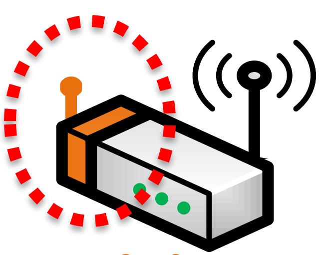
$$\sin(2\pi f_{ref} t)$$

f_{ref} = 10 MHz

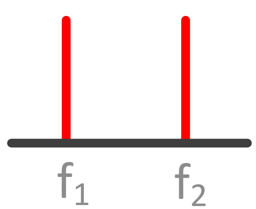


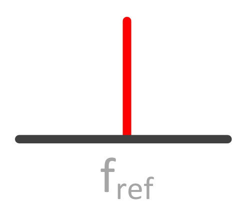
$$(\sin(2\pi f_1 t) + \sin(2\pi f_2 t))$$

f_2, f_1 = any frequency



Recipient

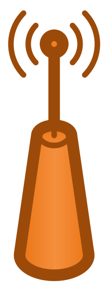




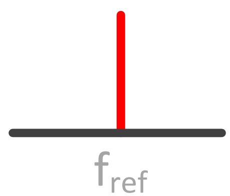
Transmit a Differential-reference

Instead of transmitting a signal at the clock frequency (10 MHz)

Transmits two signals separated by the clock frequency

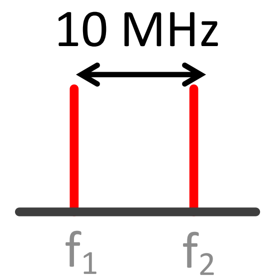


Emitter



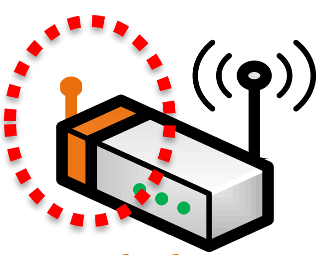
$$\sin(2\pi f_{ref} t)$$

f_{ref} = 10 MHz

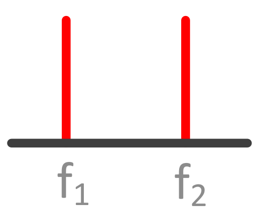


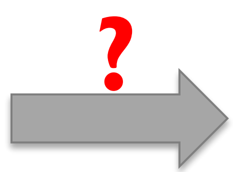
$$(\sin(2\pi f_1 t) + \sin(2\pi f_2 t))$$

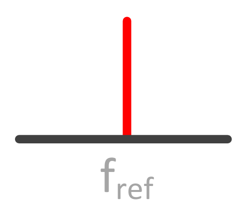
f_2, f_1 = any frequency

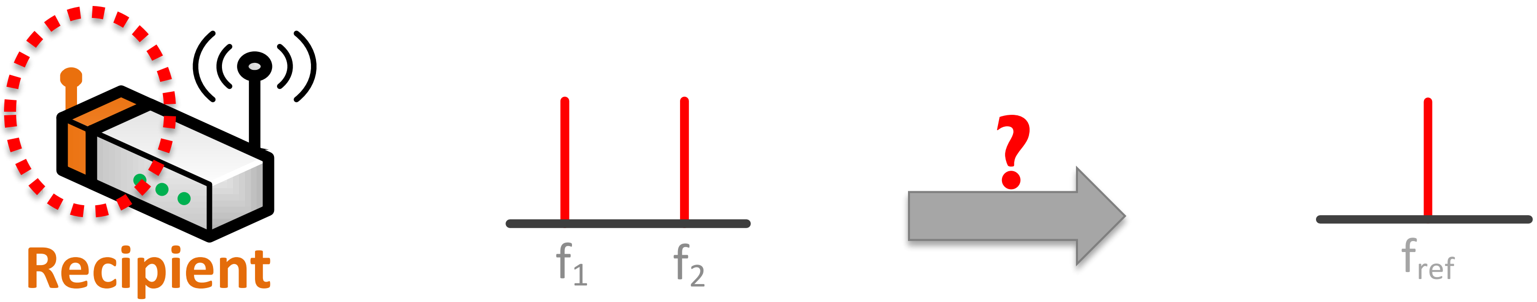


Recipient

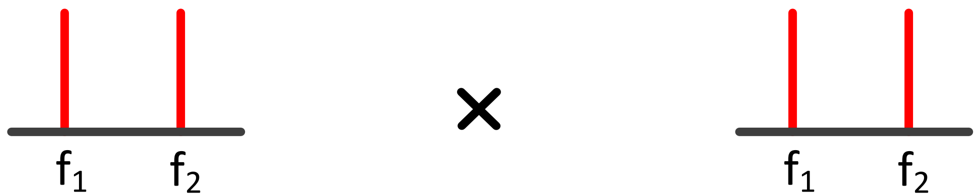








receives the signal and multiplies the signal by itself



$$(\sin(2\pi f_1 t) + \sin(2\pi f_2 t)) \times (\sin(2\pi f_1 t) + \sin(2\pi f_2 t))$$

Using trigonometric identities:

$$sin(\alpha) \times sin(\beta) = \frac{1}{2} cos(\alpha + \beta) + \frac{1}{2} cos(\alpha - \beta)$$

AirShare transmits the reference clock without violating FCC regulations

Conclusion

- Learned about Interference Nulling, IA, IAC and MegaMIMO
- In IA, the gains are lower, but
  - Transmitters need not be connected to the same Ethernet and exchange the packets
  - No need for phase synchronization
- IAC can bring additional gains in comparison to IA
  - Transmitters need not be connected to the same Ethernet and exchange the packets
  - No need for phase synchronization
- In MegaMIMO, the gains are linear with the total number of users
  - Need high speed Ethernet to connect the transmitters
  - Need tight phase synchronization between transmitters
- AirShare: Distributed phase synchronization