MegaMIMO: Scaling Wireless Throughput with the Number of Users

Hariharan Rahul, Swarun Kumar and Dina Katabi
There is a Looming Wireless Capacity Crunch

Given the trends in the growth of wireless demand, and based on current technology, the FCC projects that the US will face a spectrum shortfall in 2013.

The iPhone 4 demo failed at Steve Jobs’s keynote due to wireless congestion. Jobs’s reaction: “If you want to see the demos, shut off your laptops, turn off all these MiFi base stations, and put them on the floor, please.”
MegaMIMO alleviates the capacity crunch by transmitting more bits per unit of spectrum.
Today’s Wireless Networks

Access Point 1

Access Point 2

Access Point 3

User 1

User 2

User 3

Interference!

Access Points Can’t Transmit Together in the Same Channel
Interference from $x_2 + x_3 \approx 0$

Data: $x_1$ survives

User 1

Interference from $x_1 + x_3 \approx 0$

Data: $x_2$ survives

User 2

Interference from $x_1 + x_2 \approx 0$

Data: $x_3$ survives

User 3

All Access Points Can Transmit Simultaneously in the Same Channel
Interference from $x_2 + x_3 \approx 0$ Data: $x_2$ survives

Interference from $x_1 + x_2 \approx 0$ Data: $x_3$ survives

MegaMIMO

Access Point 1

Access Point 2

Access Point 3

User 1

User 2

User 3

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
Diving Into The Details
Transmitting Without Interference

Wants $x_1$
Receives $y_1$

Wants $x_2$
Receives $y_2$

\[ y_1 = d_1 x_1 \]
\[ y_2 = d_2 x_2 \]

\[
\begin{bmatrix}
  y_1 \\
  y_2
\end{bmatrix} =
\begin{bmatrix}
  d_1 & 0 \\
  0 & d_2
\end{bmatrix}
\begin{bmatrix}
  x_1 \\
  x_2
\end{bmatrix}
\]
Transmitting Without Interference

Wants $x_1$
Receives $y_1$

Wants $x_2$
Receives $y_2$

\[
\begin{align*}
y_1 &= d_1 x_1 + 0 \cdot x_2 \\
y_2 &= 0 \cdot x_1 + d_2 x_2
\end{align*}
\]

\[
\begin{bmatrix}
y_1 \\
y_2
\end{bmatrix} = \begin{bmatrix}
x_1 \\
x_2
\end{bmatrix}
\]

Diagonal
Transmitting Without Interference

**Goal:** Make the effective channel matrix diagonal

**Diagonal Matrix → Non-Interference**
On-Chip MIMO

• All nodes are synchronized in time to within nanoseconds of each other.

• Oscillators at all nodes have exactly the same frequency, i.e., no frequency offset.
On-Chip MIMO

Sends $x_1$  

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

Sends $x_2$

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

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

Non-diagonal Matrix $\rightarrow$ Interference
On-Chip MIMO

Sends $x_1$ Sends $x_2$

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

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

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

Sends $s_1$ 

\[ y_1 = h_{11} s_1 + h_{12} s_2 \]

\[ y_2 = h_{21} s_1 + h_{22} s_2 \]

\[
\begin{bmatrix}
  y_1 \\
  y_2
\end{bmatrix} = \begin{bmatrix}
  s_1 \\
  s_2
\end{bmatrix}
\]
Making Effective Channel Matrix Diagonal

\[
y_1 = h_{11} s_1 + h_{12} s_2 \\
y_2 = h_{21} s_1 + h_{22} s_2
\]

\[
\begin{bmatrix}
y_1 \\
y_2
\end{bmatrix} = H \begin{bmatrix}s_1 \\
s_2\end{bmatrix}
\]
Making Effective Channel Matrix Diagonal

\[
y_1 = h_{11}s_1 + h_{12}s_2 \\
y_2 = h_{21}s_1 + h_{22}s_2
\]

\[
\begin{bmatrix}
y_1 \\
y_2
\end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} s_1 \\
    s_2
\end{bmatrix} = H \begin{bmatrix} s_1 \\
    s_2
\end{bmatrix}
\]
Making Effective Channel Matrix Diagonal

\[ y_1 = h_{11}s_1 + h_{12}s_2 \]
\[ y_2 = h_{21}s_1 + h_{22}s_2 \]

\[
\begin{bmatrix}
    y_1 \\
    y_2
\end{bmatrix}
= HH^{-1}
\begin{bmatrix}
    x_1 \\
    x_2
\end{bmatrix}
\]

Effective channel is diagonal
Beamforming System Description

Channel Measurement:
- API and AP2 measure channels to clients
- Clients report measured channels back to APs

Data Transmission:
- Packets are forwarded to both APs
- Each AP computes its beamformed signal $s_i$ using the equation $\bar{\mathbf{s}} = \mathbf{H}^{-1} \bar{\mathbf{x}}$
- Clients 1 and 2 decode $x_1$ and $x_2$ independently
Distributed Transmitters

• Nodes are not synchronized in time.
  – We use SourceSync to synchronize senders within 10s of ns (SIGCOMM 2010)
  – Works for OFDM based systems like Wi-Fi, LTE etc.

• Oscillators are not synchronized and have frequency offsets relative to each other.
MegaMIMO

• First wireless network that can scale network throughput with the number of transmitters
• Algorithm for phase synchronization across multiple independent transmitters
• Demonstrated in a wireless testbed implementation
What Happens with Independent Oscillators?

\[
\begin{bmatrix}
  h_{11} & h_{12} \\
  h_{21} & h_{22}
\end{bmatrix}
\]
What Happens with Independent Oscillators?

\[ h_{11} e^{j(\omega_{T1} - \omega_{R1})t} \]

\[
\begin{bmatrix}
  h_{11} & h_{12} \\
  h_{21} & h_{22}
\end{bmatrix}
\]
What Happens with Independent Oscillators?

\[ \begin{bmatrix} h_{11} e^{(\omega_{T1} - \omega_{R1})t} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \]
What Happens with Independent Oscillators?

\[
\begin{bmatrix}
    h_{11} e^{j(\omega_{T1} - \omega_{R1})t} & h_{12} e^{j(\omega_{T2} - \omega_{R1})t} \\
    h_{21} & h_{22}
\end{bmatrix}
\]
What Happens with Independent Oscillators?

\[
\begin{bmatrix}
    h_{11} e^{j(\omega_{T1} - \omega_{R1})t} & h_{12} e^{j(\omega_{T2} - \omega_{R1})t} \\
    h_{21} & h_{22}
\end{bmatrix}
\]
What Happens with Independent Oscillators?

\[
\begin{bmatrix}
 h_{11} e^{j(\omega T_1 - \omega R_1) t} & h_{12} e^{j(\omega T_2 - \omega R_1) t} \\
 h_{21} e^{j(\omega T_1 - \omega R_2) t} & h_{22} e^{j(\omega T_2 - \omega R_2) t}
\end{bmatrix}
\]
What Happens with Independent Oscillators?

\[
\begin{bmatrix}
  h_{11} e^{j(\omega_{T1} - \omega_{R1})t} & h_{12} e^{j(\omega_{T2} - \omega_{R1})t} \\
  h_{21} e^{j(\omega_{T1} - \omega_{R2})t} & h_{22} e^{j(\omega_{T2} - \omega_{R2})t}
\end{bmatrix}
\]

Time Varying
Channel is Time Varying

\[ H(t) \]
Does Traditional Beamforming Still Work?

\[
\begin{bmatrix}
y_1(t) \\
y_2(t)
\end{bmatrix} = H(t) \begin{bmatrix}
s_1(t) \\
s_2(t)
\end{bmatrix}
\]
Does Traditional Beamforming Still Work?

\[
\begin{bmatrix}
    y_1(t) \\
    y_2(t)
\end{bmatrix}
= 
\begin{bmatrix}
    H(t) \\
    H^{-1}
\end{bmatrix}
\begin{bmatrix}
    x_1(t) \\
    x_2(t)
\end{bmatrix}
\]

Not Diagonal
Does Traditional Beamforming Still Work?

Beamforming does not work

\[
\begin{bmatrix}
  y_1(t) \\
  y_2(t)
\end{bmatrix} = H(t)H^{-1}\begin{bmatrix}
  x_1(t) \\
  x_2(t)
\end{bmatrix}
\]

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 slaves
  - Imitate the behavior of the lead AP by fixing the rotation of their oscillator relative to the lead.
Decomposing $H(t)$

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

$$
\begin{bmatrix}
    e^{-j\omega_{R1}t} & 0 \\
    0 & e^{-j\omega_{R2}t}
\end{bmatrix}
\begin{bmatrix}
    h_{11} e^{j(\omega_{T1})t} & h_{12} e^{j(\omega_{T2})t} \\
    h_{21} e^{j(\omega_{T1})t} & h_{22} e^{j(\omega_{T2})t}
\end{bmatrix}
$$
Decomposing $H(t)$

$$
\begin{bmatrix}
  e^{-j\omega_{R1}t} & 0 \\
  0 & e^{-j\omega_{R2}t}
\end{bmatrix}
\begin{bmatrix}
  h_{11} e^{j(\omega_{T1})t} & h_{12} e^{j(\omega_{T2})t} \\
  h_{21} e^{j(\omega_{T1})t} & h_{22} e^{j(\omega_{T2})t}
\end{bmatrix}
$$
Decomposing $H(t)$

$$
\begin{align*}
\begin{bmatrix}
\exp(-j\omega_{R1}t) & 0 \\
0 & \exp(-j\omega_{R2}t)
\end{bmatrix}
\begin{bmatrix}
h_{11} & h_{12} \\
h_{21} & h_{22}
\end{bmatrix}
\begin{bmatrix}
\exp(j\omega_{T1}t) & 0 \\
0 & \exp(j\omega_{T2}t)
\end{bmatrix}
\end{align*}
$$
Decomposing $H(t)$

$$
\begin{bmatrix}
    e^{-j\omega_{R1}t} & 0 \\
    0 & e^{-j\omega_{R2}t}
\end{bmatrix}
\begin{bmatrix}
    h_{11} & h_{12} \\
    h_{21} & h_{22}
\end{bmatrix}
\begin{bmatrix}
    e^{j\omega_{T1}t} & 0 \\
    0 & e^{j\omega_{T2}t}
\end{bmatrix}
$$
Decomposing $H(t)$

$$H = \begin{bmatrix} e^{-j\omega_{R1}t} & 0 \\ 0 & e^{-j\omega_{R2}t} \end{bmatrix} \quad \begin{bmatrix} e^{j\omega_{T1}t} & 0 \\ 0 & e^{j\omega_{T2}t} \end{bmatrix}$$

Diagonal

Devices cannot track their own oscillator phases...
Decomposing $H(t)$

$$H = \begin{bmatrix} e^{-j\omega_{R1}t} & 0 \\ 0 & e^{-j\omega_{R2}t} \end{bmatrix} \quad e^{j\omega_{T1}t}$$

$$e^{j\omega_{T1}t}$$
Decomposing $H(t)$

$$H(t) = R(t) T(t)$$

$R(t)$:
$$\begin{bmatrix}
  e^{j(\omega_{T1} - \omega_{R1})t} & 0 \\
  0 & e^{j(\omega_{T1} - \omega_{R2})t}
\end{bmatrix}$$

$T(t)$:
$$\begin{bmatrix}
  1 & 0 \\
  0 & e^{j(\omega_{T2} - \omega_{T1})t}
\end{bmatrix}$$

Depends only on transmitters
Decomposing $H(t)$

$$H(t) = R(t) \cdot H \cdot T(t)$$

where

$$H = \begin{bmatrix} e^{j(\omega_{T1} - \omega_{R1})t} & 0 \\ 0 & e^{j(\omega_{T1} - \omega_{R2})t} \end{bmatrix}$$

and

$$R(t) = \begin{bmatrix} 1 & 0 \\ 0 & e^{j(\omega_{T2} - \omega_{T1})t} \end{bmatrix}$$

$$T(t)$$
Beamforming with Different Oscillators

\[
\begin{bmatrix}
  y_1(t) \\
  y_2(t)
\end{bmatrix} = R(t) \cdot H \cdot T(t) \begin{bmatrix}
  s_1(t) \\
  s_2(t)
\end{bmatrix}
\]

\[
\begin{bmatrix}
  s_1(t) \\
  s_2(t)
\end{bmatrix} = T(t)^{-1} H^{-1} \begin{bmatrix}
  x_1(t) \\
  x_2(t)
\end{bmatrix}
\]
Beamforming with Different Oscillators

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

Diagonal

\[
\begin{bmatrix}
s_1(t) \\
s_2(t)
\end{bmatrix} = T(t)^{-1} H^{-1} \begin{bmatrix} x_1(t) \\
x_2(t)
\end{bmatrix}
\]
Transmitter Compensation

\[ T(t) = \begin{bmatrix} 1 & 0 \\ 0 & e^{j(\omega_{T2} - \omega_{T1})t} \end{bmatrix} \]
Transmitter Compensation

\[ T(t)^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & e^{-j(\omega_{T2} - \omega_{T1})t} \end{bmatrix} \]

Slave AP imitates lead by multiplying each sample by oscillator rotation relative to lead.

Requires only local information \(\rightarrow\) 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

Directly compute phase at each slave by measuring channel from lead

\[ h_{2}^{\text{lead}}(t) = h_{2}^{\text{lead}} e^{j(\omega_{T2} - \omega_{T1})t} \]
Resynchronization

Lead AP:

– Prefixes data transmission with synchronization header
Slave AP:

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

\[
\begin{bmatrix}
  y_1(t) \\
  y_2(t)
\end{bmatrix} = R(t).H.T(t)
\]

\[
\begin{bmatrix}
  s_1(t) \\
  s_2(t)
\end{bmatrix}
\]
Receiver Compensation

\[
\begin{bmatrix}
y_1(t) \\
y_2(t)
\end{bmatrix} = R(t) \cdot H \cdot T(t) \cdot T(t)^{-1} \cdot H^{-1} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}
\]

\[
\begin{bmatrix}
y_1(t) \\
y_2(t)
\end{bmatrix} = R(t)^{-1} \cdot R(t) \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}
\]
Receiver Compensation

\[ R(t) = \begin{bmatrix} e^{j(\omega_{T1} - \omega_{R1})t} & 0 \\ 0 & e^{j(\omega_{T1} - \omega_{R2})t} \end{bmatrix} \]
Receiver Compensation

\[ R(t)^{-1} = \begin{bmatrix} e^{-j(\omega_{T1} - \omega_{R1})t} & 0 \\ 0 & e^{-j(\omega_{T1} - \omega_{R2})t} \end{bmatrix} \]

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

• Decoupling channel measurement across different clients
• Using MegaMIMO for diversity
• Compatibility with off-the-shelf 802.11n cards

Described in paper
Performance
Implementation

- Implemented in USRP2
- 2.4 GHz center frequency
- OFDM with 10 MHz bandwidth
- 10 software radios acting as APs, all in the same frequency
- 10 software radios acting as clients
Testbed
Does MegaMIMO Scale Throughput with the Number of Users?

• Fix a number of users, say $N$
• Pick $N$ AP locations
• Pick $N$ client locations
• Vary $N$ from 1 to 10

• Compared Schemes:
  – 802.11
  – MegaMIMO
Does MegaMIMO Scale Throughput with the Number of Users?
Does MegaMIMO Scale Throughput with the Number of Users?

![Graph showing total throughput in Mb/s against the number of APs on the same channel. The graph indicates that total throughput remains relatively constant as the number of APs increases.]
Does MegaMIMO Scale Throughput with the Number of Users?

MegaMIMO

10x throughput gain over existing Wi-Fi

802.11
What are MegaMIMO’s Scaling Limits?

- Theoretical
  \[ \log \text{SNR} \]
  Can Scale Indefinitely with \( N \)

- Practical
  Errors in \( H \) and phase synchronization affect accuracy of beamforming
What are MegaMIMO’s Scaling Limits?

- $N$ APs transmit to $N$ users, $N = 2 \ldots 10$.
- Perform MegaMIMO as before, but with a zero signal for some client (i.e. null at that client)

  Phase Alignment is Accurate
  $\rightarrow$ Received signal at noise floor (0 dB)

  Inaccuracy in Phase Alignment
  $\rightarrow$ Received signal higher than noise floor (>0 dB)
What are MegaMIMO’s Scaling Limits?

Interference to Noise Ratio (dB)

Number of APs on Same Channel
What are MegaMIMO’s Scaling Limits?

Interference to Noise Ratio $\approx 1.5$ dB even at 10 users.
Compatibility with 802.11n

- 802.11n over 20 MHz Bandwidth
- Two 2-antenna USRPs acting as APs
- Two 2-antenna 802.11n clients
- Throughtput gain of MegaMIMO over 802.11n
- Demonstrates
  - Compatibility with 802.11n clients
  - Compatibility with MIMO APs and clients
Compatibility with 802.11n

Fraction of Runs vs. Throughput Gain
Compatibility with 802.11n

Fraction of Runs vs. Throughput Gain

Throughput Gain

1.8x
Compatibility with 802.11n

Median Gain of 1.8x with 2 receivers
Related Work

• Theoretical
  – Aeron et al., Simeone et al., Ozgur et al. (Distributed and Virtual MIMO)
• Empirical
  – DIDO, Fraunhofer
  – Network MIMO
Conclusion

• Wireless networks are facing a spectrum crunch
• MegaMIMO enables multiple independent transmitters to transmit to independent receivers in the same frequency bands
• Distributed and accurate phase synchronization
• Can enable wide body of theoretical work
  – lattice coding, noisy network coding, distributed superposition coding, dirty paper coding