ABSTRACT

Multiple transmit and multiple receive antenna (MIMO) systems provide an additional spatial dimension that yields a degree of freedom gain. The degree of freedom gain in MIMO systems allows for spatial multiplexing of multiple data streams onto the channel and an increase in the channel’s capacity or improved reliability against multipath fading. An MIMO antenna system with \( N_t \) transmit antennas and \( N_r \) receive antennas, under suitable fading conditions will yield a \( \min(N_t, N_r) \) degree of freedom gain. This \( \min(N_t, N_r) \) degree of freedom gain can be employed by spatial multiplexing multiple data streams to increase the channel’s capacity by a factor of \( \min(N_t, N_r) \). In addition, the transmit signal at the transmit antenna array can be aligned in the direction of the transmit antenna array pattern so that the received signals add constructively at the receive antenna array. This transmit beamforming will provide a power gain in addition to the MIMO degree of freedom gain. However, in order to do transmit beamforming, the transmit antenna array must have knowledge of the channel’s state. Channel feedback in the conventional statistical MIMO channel is burdensome, due to the need to feed back information on each of the \( N_t N_r \) channel coefficients.

The angular domain provides a much more natural, compact representation of the MIMO wireless channel than does the conventional model. In the angular domain, the transmit and receive arrays are divided into \( N_t \) and \( N_r \) respective resolvable angles, and the angular domain channel model examines the channel between the transmit and receive angular cells. The angular domain approach takes advantage of the physical nature of the wireless channel and produces a sparse channel matrix, unlike the previous conventional statistical model. As a result, less information will need to be fed back to the transmitter. This thesis applies array processing techniques to the MIMO wireless channel modeled in the angular domain to demonstrate the reduced computation complexity and feedback rate of angular domain processing.
To my father and my mother, who have supported and encouraged me all through life and especially in the pursuit of my master's degree.
ACKNOWLEDGMENTS

Thanks to Ada Poon for all the support, patience, and understanding in the completion of this master's thesis.
# TABLE OF CONTENTS

LIST OF FIGURES ........................................................................................................ vi

1. PROPOSAL .................................................................................................................. 1
   1.1 System Model and Coordinate Change ................................................................. 2
   1.2 Estimation ............................................................................................................... 5
   1.3 Feedback Schemes ............................................................................................... 6
   1.4 Expected Performance ......................................................................................... 8

2. PROCEDURE .............................................................................................................. 9
   2.1 Environment .......................................................................................................... 9
   2.2 A Priori Processing .............................................................................................. 9
   2.3 Real-Time Processing .......................................................................................... 11
      2.3.1 Spectral Estimation ....................................................................................... 11
      2.3.2 Transmit Beamforming ................................................................................. 12
      2.3.3 Signal Space Estimation ............................................................................... 12
      2.3.4 Antenna Selection ......................................................................................... 13
   2.4 Simulation ........................................................................................................... 13

3. FINDINGS .................................................................................................................. 14

4. ANALYSIS ............................................................................................................... 16

5. CONCLUSION .......................................................................................................... 18

REFERENCE ................................................................................................................ 19

APPENDIX: MATLAB CODE ..................................................................................... 20
LIST OF FIGURES

Figure 1: Conventional MIMO Model.................................................................................. 1
Figure 2: Angular Domain Channel Model ........................................................................ 2
Figure 3: MIMO Wireless Channel with Finite Number of Scatterers............................ 4
Figure 4: Angular Domain Feedback Scheme Performance (N_t = 1)............................ 14
Figure 5: Angular Domain Feedback Scheme Performance (N_t = 5)............................. 15
1. PROPOSAL

In a MIMO wireless antenna system with $N_t$ transmit antennas and $N_r$ receive antennas, the conventional statistical baseband channel model is

$$y = Hx + w$$  \hspace{1cm} (1)

where $x$ is the $N_t$-dimensional vector of transmitted signals from the transmit antenna array, $H$ is the $N_r \times N_t$ complex channel matrix mapping between the transmit and receive antennas, $w$ is the additive complex normal white noise at the receive antenna array, and $y$ is the $N_r$-dimensional vector of received signals at the receive antenna array (Figure 1).

![Figure 1: Conventional MIMO Model](image.png)

In this example the channel is a flat-fading channel, so there is only one channel tap, and the channel is underspread, so the channel coefficients will not vary over the coherent time period.

However, taking advantage of the physical characteristics of the wireless channel, it is much more natural to look at the system in the angular domain. In the angular domain, instead of mapping between antennas, the angular channel matrix $H^a$ maps the complex channel gain between $N_t$ resolvable angular regions at the transmit antenna array and $N_r$
resolvable angular regions at the receive antenna array. The angular channel MIMO model is of the form

\[ y^a = H^a x^a + w^a \]  \hspace{1cm} (2)

where \( x^a \) is the \( N_r \)-dimensional angular domain vector of transmitted signals from the transmit antenna array, \( w^a \) is the additive complex noise rotated to angular domain coordinates, and \( y^a \) is the \( N_r \)-dimensional vector of received signals at the receive antenna array as seen in the angular domain (Figure 2).

![Figure 2: Angular Domain Channel Model](image)

1.1 System Model and Coordinate Change

In a MIMO wireless antenna system with \( N_l \) subcarriers the conventional statistical baseband channel model at the \( l^{th} \) subcarrier is

\[ y_l = H_l x_l + w_l \hspace{1cm} l = 1, \ldots, N_l \]  \hspace{1cm} (3)

where \( y_l \in C^{N_r}, x_l \in C^{N_t}, H_l \in C^{N_r \times N_l}, \) and \( w_l \sim CN(0, I_{N_l}) \). The MIMO channel matrix, \( H_l \), can be re-written from the angular domain perspective as

\[ H_l = \iint h(t, \mathbf{\kappa}, \mathbf{\hat{\kappa}}) e^{-j2\pi t\Delta f} a_r(\mathbf{\kappa}) a_t^+(\mathbf{\hat{\kappa}}) \, dt \, d\mathbf{\kappa} \, d\mathbf{\hat{\kappa}} \]  \hspace{1cm} (4)
where $\Delta f$ is the subcarrier spacing, $\mathbf{a}_t(\hat{k}) \in \mathbb{C}^{N_t}$ is the transmit array manifold, $\mathbf{a}_r(\hat{r}) \in \mathbb{C}^{N_r}$ is the receive antenna array manifold, and $h(t, \hat{r}, \hat{k})$ is the complex gain between transmit direction $\hat{k}$ and receive direction $\hat{r}$ at time $t$ due to an impulse at time zero. The array manifolds, $\mathbf{a}_t(\hat{k})$ and $\mathbf{a}_r(\hat{r})$, capture the characteristics of the antenna arrays, i.e. the details about the paths leaving or incoming on the antenna array. For example, the transmit antenna array manifold, $\mathbf{a}_t(\hat{k})$, for a uniformly spaced antenna array with $N_t$ transmit antennas, would take the form

$$
\mathbf{a}_t(\Omega_t) = \frac{1}{\sqrt{N_t}} \begin{bmatrix}
1 \\
e^{-j2\pi\Delta_t \Omega_t} \\
e^{-j2\pi(2)\Delta_t \Omega_t} \\
\vdots \\
e^{-j2\pi(N_t-1)\Delta_t \Omega_t}
\end{bmatrix}
$$

where $\Omega_t$ is the directional cosine of the signal departing the transmit antenna array and $\Delta_t$ is the spacing between array elements normalized to the carrier wavelength. The receive array manifold ($\mathbf{a}_r(\hat{r})$) will be of similar form.

If the MIMO wireless channel consists of a finite number of scatterers, then the channel can be modeled as having a finite number of paths connecting the transmit antenna array and receive antenna array (Figure 3) and the channel impulse response will take the form

$$
h(t, \hat{r}, \hat{k}) = \sum_i \rho_i \delta(t - \tau_i) \delta(\hat{r} - \hat{r}_i) \delta(\hat{k} - \hat{k}_i)
$$

where $\rho_i$ is the complex gain, $\tau_i$ is the path delay, $\hat{r}_i$ is the angle of arrival at the receive antenna array, and $\hat{k}_i$ is the angle of departure from the transmit antenna array for the $i^{th}$ physical path connecting the two antenna arrays.
Figure 3: MIMO Wireless Channel with Finite Number of Scatterers

With a finite number of paths, the advantage of the angular domain is apparent. The finite number of paths will result in a sparse angular channel matrix. The angular channel model will thus allow for compacted channel information to be sent back to the transmitter, reducing the feedback rate. In a typical urban environment, modeled by the Rayleigh fading channel, the angular channel matrix is relatively sparse. Thus we desire to describe the angular channel matrix in a statistical manner that will provide the necessary feedback to the transmitter for optimal angular channel use.

The channel power profile, the power of the channel path between a valid direction of arrival and a valid direction of departure with path delay less than the maximum delay spread, is denoted $P(t, \hat{\kappa}, \hat{k})$. With the new definition of the channel matrix and the channel power profile, the angular spectra of the channel, $G_{h}(\hat{\kappa}, \hat{k})$, transmit array, $G_{t}(\hat{k})$, and receive array, $G_{r}(\hat{\kappa})$, can be defined:

$$G_{h}(\hat{\kappa}, \hat{k}) := \int P(t, \hat{\kappa}, \hat{k}) \, dt \quad G_{t}(\hat{k}) := ||a_{t}(\hat{k})||^{2} \quad G_{r}(\hat{\kappa}) := ||a_{r}(\hat{\kappa})||^{2} \quad (7)$$
The transmit array and receive array spectra depend only on their respective array manifolds, not on the state of the physical channel.

The long-term correlation matrices for the transmit and receive arrays are as follows:

\[
E_{G_h,h}[H_i^\dagger H_i] = \tilde{G}_h \int G_r(\vec{k}) d\vec{k} \int a_t(\vec{k}) a_t^\dagger(\vec{k}) d\vec{k}
\] (8)

\[
E_{G_h,h}[H_i H_i^\dagger] = \tilde{G}_h \int G_t(\vec{k}) d\vec{k} \int a_r(\vec{k}) a_r^\dagger(\vec{k}) d\vec{k}
\] (9)

\[
\tilde{G}_h = E[G_h(\vec{k}, \vec{k})]
\] (10)

As can be seen, each matrix only depends on the array manifolds, so the array manifolds and thus the array spectra can be determined from these long-term correlations in setup prior to communication. With this a priori knowledge, the short-term transmit correlation matrix depends only on the angular spectra of the MIMO channel. The short-term transmit and receive correlation matrices are as follows:

\[
R_t := E_h[H_t^\dagger H_t] = \iint G_h(\vec{k}, \vec{k}) G_r(\vec{k}) a_t(\vec{k}) a_t^\dagger(\vec{k}) d\vec{d}\vec{k}
\] (11)

\[
R_r := E_h[H_r H_r^\dagger] = \iint G_h(\vec{k}, \vec{k}) G_t(\vec{k}) a_r(\vec{k}) a_r^\dagger(\vec{k}) d\vec{d}\vec{k}
\] (12)

\[
G_h^t(\vec{k}) := \int G_h(\vec{k}, \vec{k}) G_t(\vec{k}) d\vec{k}
\] (13)

The eigenvectors of the short-term transmit correlation matrix, \(R_t\), are the transmit beamforming vectors. Similarly, the eigenvectors of the short-term receive correlation matrix, \(R_r\), are the receive beamforming vectors.

### 1.2 Estimation

For the angular estimation of the channel spectra, the angular regions of the transmit and receive arrays will be represented discretely as the union of non-overlapping cells. The union of the \(N_t\) non-overlapping cells equals the entire angular region reachable by the transmit antenna array. The spectrum estimate at angle \(\theta_n\) in cell \(C_n\) is the average of the spectra in \(C_n\) for all \(\theta_n \in C_n\). Thus we define the reachable angular region for the transmit array and the receive array:
This allows for definition of the discrete forms of the transmit and receive array manifolds and angular spectra:

\[ \mathbf{A}_{t,j} = \mathbf{A}_t(C_{t,j}) \quad \text{and} \quad \mathbf{A}_{r,i} = \mathbf{A}_r(C_{r,i}) \]  

(16)

\[ \mathbf{A}_{t,j} = \int_{C_{t,j}} \mathbf{a}_t(\hat{\mathbf{k}}) \mathbf{a}_t^\dagger(\hat{\mathbf{k}}) \, d\hat{\mathbf{k}} \quad \text{and} \quad \mathbf{A}_{r,i} = \int_{C_{r,i}} \mathbf{a}_r(\hat{\mathbf{r}}) \mathbf{a}_r^\dagger(\hat{\mathbf{r}}) \, d\hat{\mathbf{r}} \]  

(17)

\[ G_{t,j} = \int_{C_{t,j}} G_t(\hat{\mathbf{k}}) \, d\hat{\mathbf{k}} \quad \text{and} \quad G_{r,i} = \int_{C_{r,i}} G_r(\hat{\mathbf{r}}) \, d\hat{\mathbf{r}} \]  

(18)

for \( i = 1, \ldots, N_r \) and \( j = 1, \ldots, N_t \). The discrete form of the angular channel spectra is thus defined as

\[ G_{h,ij} = \frac{1}{|C_{t,j}| |C_{r,i}|} \int_{C_{t,j}} \int_{C_{r,i}} G_h(\hat{\mathbf{k}}, \hat{\mathbf{r}}) \, d\hat{\mathbf{k}} \, d\hat{\mathbf{r}} \]  

(19)

As described above, the spectrum estimate at any angle in a specific cell is the average of the spectra within that cell. Thus, \( G_{t,j} \) is the total transmitted power over the cell \( C_{t,j} \), \( G_{r,i} \) is the total received power over the cell \( C_{r,i} \), and \( G_{h,ij} \) is the average channel power over the region \( C_{t,j} \times C_{r,i} \). As mentioned earlier, the receive and transmit array manifolds, and thus the transmit and receive array spectra, are estimated a priori; thus only \( G_{h,ij} \) must be determined.

The estimate of the angular channel spectra \( \tilde{G}_{h,ij} \) for the region \( C_{t,j} \times C_{r,i} \) is determined via spectral estimation by generalized prolate spheroidal sequences.

### 1.3 Feedback Schemes

Three feedback schemes will be examined, each providing a different level of detail about the angular channel matrix. In all three feedback schemes, the first real-time
processing step is angular channel estimation ($\hat{G}_{h,ij}$). After estimating the angular channel, the receiver will compute the value of

$$\hat{G}_{h,j}^t = \sum_l \hat{G}_{h,ij} G_{r,i}, \quad \forall j$$

for each transmit angular region $C_{t,j}$. This will determine whether there are physical paths connecting transmit directions in the partition $C_{t,j}$ to the receiver.

In the *transmit beamforming* feedback scheme, the receiver will send $\{\hat{G}_{h,j}^t\}$ back to the transmitter, where the short-term correlation matrix, $R_t$, can be approximated:

$$R_t \approx \sum_{ij} G_{h,ij} G_{r,i} A_{t,j}$$

The eigenvectors corresponding to the $M$ largest eigenvalues of the short-term transmit correlation matrix approximation, $\hat{R}_t$, will be the optimal transmit beamforming vectors. The sequence $\{\hat{G}_{h,j}^t\}$ has $N_t$ parameters, indicating a feedback rate of $N_t$ symbols per physical channel.

In the *signal space estimation* feedback scheme, $\{\hat{G}_{h,j}^t\}$ will be processed at the receiver and a binary $N_r$-length vector $\{J_{t,j}\}$ will be sent back to the transmitter whose $j^{th}$ element indicates whether there are physical paths connecting the transmit direction in the partition $C_{t,j}$ to the receiver:

$$J_{t,j} = \begin{cases} 1, & \text{when } \sum_l G_{h,ij} G_{r,i} > \theta_{th} \\ 0, & \text{otherwise.} \end{cases}$$

This is a very simple scheme, telling the transmitter whether the transmit partition $C_{t,j}$ has physical paths connecting to the receiver, but providing no information about the quality of the paths.

At the transmitter, the transmit signal space $A_t(\Omega_t)$ can be approximated

$$A_t(\Omega_t) \approx \sum_j J_{t,j} A_{t,j}$$
The eigenvectors of $\hat{A}_t(\Omega_c)$ will provide the optimal basis for transmission. Here again there are $N_t$ parameters, and the feedback rate is $N_t$ symbols per channel.

The *antenna selection* feedback scheme provides the least amount of information to the transmitter, sending back the number of angular partitions with physical paths connecting to the receiver, i.e. $|J_t|$, where $|J_t| = \sum_j J_{t,j}$.

\begin{align}
\text{rank}(R_t) &\approx |J_t| \\
\text{rank}(R_r) &\approx |J_r|
\end{align} \tag{24}

This feedback scheme provides the least information of the three schemes, only telling how many of the $N_t$ transmit directions should be used, not which ones nor the respective quality. The transmitter will use a look-up table for antenna selection based upon a priori blind antenna selection and exhaustive combinatorial search schemes.

### 1.4 Expected Performance

For the MIMO wireless flat-fading and underspread channel, with a finite number of scatterers, in the low-SNR regime, it is expected that the feedback scheme performance as measured by bit error rate (BER) will rank in the order transmit beamforming first, signal space estimation second, and antenna selection third. Transmit beamforming will require the highest feedback rate, but should generate the best performance with optimal beamforming and power allocation. Signal space estimation requires the same feedback rate, but should have slightly worse performance with equal power being allocated across the optimal transmission basis. Antenna selection requires the lowest feedback rate and should yield the worst performance, with the optimal transmit scheme of equal power allocation between the transmit antennas chosen based upon blind antenna selection and the exhaustive combinatorial search schemes.
2. PROCEDURE

2.1 Environment

For the purposes of this thesis, we assume an urban environment, modeled by a flat-fading, underspread Rayleigh channel. The transmit and receive antennas are critically spaced uniform linear arrays.

2.2 A Priori Processing

The MIMO angular domain a priori processing is based solely upon the physical antenna properties. Assuming uniform critically spaced linear transmit and receive antenna arrays, i.e. plugging $\Delta_t = \Delta_r = 1/2$ into Equation (5), the array manifolds will take the form:

$$a_t(\Omega_t) = \frac{1}{\sqrt{N_t}} \begin{bmatrix} 1 \\ e^{-j\pi \Omega_t} \\ e^{-j2\pi \Omega_t} \\ \vdots \\ e^{-j(N_t-1)\pi \Omega_t} \end{bmatrix} \quad a_r(\Omega_r) = \frac{1}{\sqrt{N_r}} \begin{bmatrix} 1 \\ e^{-j\pi \Omega_r} \\ e^{-j2\pi \Omega_r} \\ \vdots \\ e^{-j(N_r-1)\pi \Omega_r} \end{bmatrix} \quad (25)$$

The $(\ell, k)^{th}$ element of the discrete form of the transmit ($A_{t,j}$) and receive manifolds ($A_{r,i}$), using Equation (17), will then be:

$$A_{t,j}(\ell, k) = \frac{1}{N_t} \int_{\Omega_t,j} \exp(j\pi(\ell - k)\Omega) \, d\Omega \quad (26)$$

$$A_{r,i}(\ell, k) = \frac{1}{N_r} \int_{\Omega_r,i} \exp(j\pi(\ell - k)\Omega) \, d\Omega \quad (27)$$

where $A_{t,j}$ and $A_{r,i}$ are Hermetian matrices. The transmit and receive antenna array spectra will be:

$$G_{t,j} = \int_{\Omega_t,j} ||a_t(\Omega)||^2 \, d\Omega = \int_{\Omega_t,j} a_t^\dagger(\Omega)a_t(\Omega) \, d\Omega = \int_{\Omega_t,j} N_t \, d\Omega \quad (28)$$

$$G_{r,i} = \int_{\Omega_r,i} ||a_r(\Omega)||^2 \, d\Omega = \int_{\Omega_r,i} a_r^\dagger(\Omega)a_r(\Omega) \, d\Omega = \int_{\Omega_r,i} N_r \, d\Omega \quad (29)$$

For the uniformly spaced linear array, the non-overlapping angular cells evenly divide up the entire angular region, chosen such that $\{a_t(\Omega_t)\}$ and $\{a_r(\Omega_r)\}$ are orthogonal sets.
This will divide the angular spectra evenly between the angular cells at the transmit and receive antenna arrays.

The angular channel estimator as described in Poon’s “Angular-Domain Processing for MIMO Systems” [1] takes the following form:

\[ \hat{G}_{h,ij} = \frac{1}{N_t N_r L_t L_j} \sum_{l=1}^{N_t} \sum_{n=1}^{L_t} \sum_{m=1}^{L_j} |u_{in}^\dagger H_{jm}|^2 \]  

(30)

As shown by Poon, finding the spatial transmit \((v_{jm})\) and receive \((u_{in})\) filters to minimize the worst case spectral estimate bias requires solving the following generalized eigenvector equations

\[ A_{t,j} v_{jm} = \lambda_{jm} (\sum j' A_{t,j'}) v_{jm} \]
\[ A_{r,i} u_{in} = \sigma_{in} (\sum i' A_{r,i'}) u_{in} \]  

(31)

for the set of generalized receive and transmit eigenvectors:

\[ V = \{ v_{jm} : j = 1, \ldots, N_t \text{ and } m = 1, \ldots, L_t,j \} \]  

(32)

\[ U = \{ u_{in} : i = 1, \ldots, N_r \text{ and } n = 1, \ldots, L_r,i \} \]  

(33)

The optimal choice of resolvable angular cells for the uniformly spaced linear antenna array yields

\[ L_t,j = \ldots = L_t,N_t \]
\[ L_r,i = \ldots = L_r,N_r \]
\[ \lambda_{11} = \ldots = \lambda_{N_t,1} \]
\[ \sigma_{11} = \ldots = \sigma_{N_r,1} \]

Thus the generalized eigenvector equation simplifies to

\[ A_{t,j} v_j = \lambda_j (\sum j' A_{t,j'}) v_j \]
\[ A_{r,i} u_i = \sigma_{in} (\sum i' A_{r,i'}) u_i \]  

(34)

\[ V = \{ v_j : j = 1, \ldots, N_t \} \]
\[ U = \{ u_i : i = 1, \ldots, N_r \} \]

The generalized eigenvectors for the uniformly spaced linear antenna array are

\[ V(\kappa, \ell) = \frac{1}{\sqrt{N_t}} e^{-j 2 \pi \ell / N_t} \quad \kappa, \ell = 0, \ldots, N_t - 1 \]  

(35)

\[ U(\kappa, \ell) = \frac{1}{\sqrt{N_r}} e^{-j 2 \pi \ell / N_r} \quad \kappa, \ell = 0, \ldots, N_r - 1 \]  

(36)
The columns of unit matrices $V$ and $U$ form the spatial filters for the $N_t$ transmit and $N_r$ receive angular cells at the antenna arrays. The spatial filters $V$ and $U$ are the discrete-time Fourier transform matrices. This concludes the a priori processing for the antennas.

### 2.3 Real-Time Processing

The first step of real-time angular domain processing is estimation of the angular channel spectrum ($G_{h,ij}$). After computing the angular channel spectrum estimate ($\hat{G}_{h,ij}$) at the receiver, the user can then employ the discussed feedback schemes to optimize transmission for the current wireless channel.

#### 2.3.1 Spectral Estimation

The angular channel spectrum estimator ($\hat{G}_{h,ij}$) is obtained by first transmitting the transmit spatial filters, projecting the received signals against the receive spatial filters, and averaging the power over all eigenvectors for $(i,j)$ for all subcarriers.

For each $j \in \{1, ..., N_t\}$

\[
\begin{align*}
&\text{Let } x_l = v_{jm} \\
&y_l = H_l x_l + w_l \quad l = 1, ..., N_l \\
&r_{i,j,m,n,l} = u_{ln}^\dagger y_l, \quad \forall i, n, l. \\
&\hat{G}_{h,ij} = \frac{1}{N_t L_{r,i} L_{t,j}} \sum_{l=1}^{N_l} \sum_{n=1}^{L_{r,i}} \sum_{m=1}^{L_{t,j}} |r_{i,j,m,n,l}|^2, \quad \forall i. \\
\end{align*}
\]

For the uniform linear array with optimal choice of angular cells, $L_{r,i} = L_{t,j} = 1$. This simplifies the equation to averaging the spectrum over the subcarriers:

\[
\hat{G}_{h,ij} = \frac{1}{N_t} \sum_{l=1}^{N_l} |r_{i,j,m,n,l}|^2, \quad \forall i.
\]
### 2.3.2 Transmit Beamforming

With knowledge of the channel angular spectrum estimate \( \hat{G}_{h,ij} \) and the receive array angular spectra \( G_{r,i} \), the receiver can compute \( \{ \hat{G}_{h,j} \} \) for feedback to the transmitter.

\[
\hat{G}^t_{h,j} = \sum_i \hat{G}_{h,ij} G_{r,i}, \quad \forall j
\]

The transmit beamforming feedback scheme sends \( \{ \hat{G}^t_{h,j} \} \) back to the transmitter. At the transmitter, \( \{ \hat{G}^t_{h,j} \} \) will be used to approximate the short-term transmitter correlation matrix, \( \mathbf{R}_t \). The short-term transmit correlation matrix estimate \( \mathbf{R}_t \) is

\[
\mathbf{R}_t = \sum_j \hat{G}^t_{h,j} \mathbf{A}_{t,j}
\]

(40)

The transmitter will then perform the eigenvalue decomposition of the short-term transmit correlation matrix estimate \( \mathbf{R}_t \) to obtain the eigenvectors corresponding to the \( M \) largest eigenvalues. These eigenvectors will serve as the transmit beamforming vectors, and the eigenvalues will be used to help optimally allocate power.

### 2.3.3 Signal Space Estimation

With knowledge of the channel angular spectrum estimate \( \hat{G}_{h,ij} \) and the receive array angular spectra \( G_{r,i} \), the receiver can compute \( \{ \hat{G}^t_{h,j} \} \) as specified above. Define an \( N_r \)-length vector, \( \mathbf{j}_t \), whose \( j^{th} \) element indicates whether there are physical paths connecting the transmit direction in the partition \( C_{t,j} \) to the receiver:

\[
\mathbf{j}_{t,j} = \begin{cases} 1, & \text{when } \hat{G}^t_{h,j} > \theta_{th} \\ 0, & \text{otherwise.} \end{cases} \quad \text{where } \theta_{th} = \frac{1}{2} \max \{ \hat{G}^t_{h,j} \}
\]

(41)

Send \( \mathbf{j}_t \) back to the transmitter. At the transmitter, \( \mathbf{j}_t \) will be used to compute

\[
\mathbf{A}_t(\Omega_t) = \sum_j \hat{\mathbf{j}}_{t,j} \mathbf{A}_{t,j}
\]

(42)

whose eigenvalue decomposition will provide the optimal basis for transmission. Power will be equally allocated across the optimal basis for transmission.
2.3.4 Antenna Selection

With knowledge of the channel angular spectrum estimate \( (\hat{G}_{h,i}) \) and the receive array angular spectra \( (G_{r,i}) \), the receiver can compute \( \{\hat{G}^t_{h,j}\} \) and \( \hat{f}_r \) as specified above.

Similarly, compute \( \{\hat{G}^r_{h,j}\} \) and \( \hat{f}_r \) based upon knowledge of \( (\hat{G}_{h,i}) \) and the transmit array angular spectra \( (G_{t,i}) \) at the receiver:

\[
\hat{G}^r_{h,i} = \sum_j \hat{G}_{h,i}G_{t,j}, \quad \forall i
\]

(43)

Define an \( N_r \)-length vector, \( \hat{f}_r \), whose \( i^{th} \) element indicates whether there are physical paths connecting the receive direction in the partition \( C_{r,i} \) to the transmitter:

\[
\hat{f}_{r,i} = \begin{cases} 
1, & \text{when } \hat{G}^r_{h,i} > \theta_{th,r} \\
0, & \text{otherwise.}
\end{cases}
\]

where \( \theta_{th,r} = \frac{1}{2} \max \hat{G}^r_{h,i} \)

(44)

Now define \( \tilde{D} \) such that

\[
\tilde{D} = \min\{|\hat{f}_t|, |\hat{f}_r|\}
\]

(45)

At the receiver look up the optimal receiver antenna selection for \([\tilde{D}],|\hat{f}_r|\). Send \([\tilde{D}]\) and \(|\hat{f}_t|\) back to the transmitter. The transmitter will do the same look-up for the optimal antenna selection for \([\tilde{D}], |\hat{f}_t|\).

2.4 Simulation

The performance for the feedback schemes is examined for a MIMO wireless environment, with a flat-fading, underspread Rayleigh fading channel. There are four transmit and four receive antennas arranged in critically spaced uniform linear arrays. The transmit beamforming feedback scheme employs waterfilling for optimal power allocation. The signal space estimation feedback scheme and the antenna selection feedback scheme employ equal power allocation. The data being sent is gray-coded and modulated via 16-QAM. The receiver employs a decorrelator with successive interference cancellation (SIC) and performs maximum likelihood detection on the decoded 16-QAM data.
3. FINDINGS

The wireless MIMO simulations for each feedback scheme consisted of $10^4$ trials for each feedback scheme, with each trial having $N_l$ subcarriers and each subcarrier having an independent channel matrix $H_l$. To reduce the number of variables under examination, the same data symbols were transmitted during each trial and the total power at the transmit antenna array was kept constant. The bit error rate (BER) was measured over an SNR range of approximately 30 dB to demonstrate the angular channel feedback scheme's performance.

The first simulation is for the single subcarrier MIMO channel scenario ($N_l = 1$).

![Angular Domain Feedback Scheme Comparison](Figure 4: Angular Domain Feedback Scheme Performance ($N_l = 1$)

As can be seen in Figure 4, the angular domain feedback schemes largely worked out as expected, with the transmit beamforming feedback scheme providing the lowest BER, followed by the signal space estimation scheme, and lastly the antenna selection scheme.
The second simulation is for the multiple subcarrier MIMO channel scenario; here we use five subcarriers ($N_t = 5$).

Figure 5: Angular Domain Feedback Scheme Performance ($N_t = 5$)

We see in Figure 5 that once again the angular domain feedback schemes worked out as expected, with the transmit beamforming feedback scheme providing the best performance, then the signal space estimation scheme, and finally the antenna selection scheme.
4. ANALYSIS

The mathematical derivations and MATLAB simulations confirmed the expected results for the angular domain feedback schemes. The order of performance between the three angular domain feedback schemes was as predicted; however, a larger delta between the performances of the three feedback schemes was expected. The antenna selection feedback scheme only provides the number of transmit angular regions to use, so the antenna selection is based on a blind search algorithm done a priori. This is suboptimal compared to knowing exactly which transmit angular region to use, as is the case with the signal space estimation feedback scheme, which allows the transmitter to determine the optimal basis for transmission. Similarly, the transmit beamforming feedback scheme provides the transmitter more information than does signal space estimation, allowing for optimal beamforming and power allocation.

The feedback scheme performance is relatively close for the different number of subcarriers. For a single subcarrier, the transmission can be optimized for the individual angular channel, whereas for the multiple subcarriers, the transmission is optimized over all the independent angular channels. These tradeoffs offset providing comparable performance for the feedback schemes. As the number of subcarriers increases, the number of independent channels increases, and the transmit beamforming vectors approach the optimal angular domain beamforming vectors for a Rayleigh fading channel, i.e. the columns of the inverse discrete Fourier transform (IDFT) matrix.

The angular domain feedback schemes provide information on the state of the angular channel in a compact form, with the $N_r$-symbol vector transmit beamforming feedback scheme, the $N_r$-symbol vector signal space estimation feedback scheme, and the scalar antenna selection feedback scheme. This greatly decreases the feedback rate compared to the conventional statistical model where information would be relayed to the transmitter on the channel state for all $N_rN_t$ antenna pairs. The feedback rate to the transmitter is reduced by up to $N_t$ times for the transmit beamforming scheme and even more for the signal space estimation and antenna selection schemes. Thus the feedback reduction benefit increases with the number of antennas. In addition, the angular domain
feedback schemes off-load the processing from the receiver to the transmitter, a benefit for a broadcast system.
5. CONCLUSION

The angular domain provides a much more natural, compact representation of the MIMO wireless channel than the conventional statistical model. Harnessing the physical nature of the wireless channel, the angular channel model produces a sparse channel matrix allowing for compact feedback to the transmitter. The array processing techniques applied to the MIMO wireless channel modeled in the angular domain herein reduce the computation complexity and the feedback rate of angular domain processing. The transmit beamforming, signal space estimation, and antenna selection feedback schemes examined in this paper confirm those angular domain benefits of modeling the MIMO wireless channel.
REFERENCE

APPENDIX: MATLAB CODE

Angular Domain Transmit Beamforming Feedback Scheme MATLAB Code:

tic; %start clock

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% ENVIRONMENT AND SIMULATION SETUP -------------------------------
----

%Antenna Array Properties
Nr=4; %number of Rx antennas
Nt=4; %number of Tx antennas

%Simulation Setup
Nl=1; %number of subcarriers
Npower=-17; %AWG noise power
Pstepsize=0.5; %Noise power step size
Psteps=60; %Noise power steps
trials=10^4; %number of trials
BER=zeros(Psteps,4,Nt);

H=zeros(Nr,Nt,Nl); %wireless channel gain matrix
w=zeros(Nr,Nl); %AWGN at Rx antenna array

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% A PRIORI PROCESSING -----------------------------------------------
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%

% ANTENNA ARRAY MANIFOLDS -----------------------------------------
---
At2=zeros(Nt,Nt,Nt);
At=zeros(Nt,Nt,Nt);
Atfull=zeros(Nt,Nt);
deltat=1/2;
Ct=zeros(Nt+1,1);
Ct(1)=0;
for n=2:Nt+1,
    Ct(n)=Ct(n-1)+1/(deltat*Nt);
% 1/(deltat*Nt) = 1/2 for critically spaced 4x4
end

% sym x;
% Ct=[-0.25;0.25;0.75;1.25;1.75];
% Ct=[0;0.5;1.5;2];
% Ct=[-1;-0.5;0;0.5;1];

for k=1:Nt, %kth row of At
    for l=1:Nt, %lth column of At
        for n=1:Nt, %nth At (i.e. At for angular cell n)
            At(k,l,n)=1/Nt*exp(lj*2*pi*(l-k)*deltat*Ct(n));
        end
    end
end

20
Atfull(k,l)=Atfull(k,l)+At(k,l,n);
end
end
end
% At=At2;
Ar=At;

% INITIALIZE IDEAL TRANSFORMATION MATRICES -------------------------------
----
Ut=zeros(Nr,Nr); % angular transformation matrix for Tx antenna
Ur=zeros(Nt,Nt); % angular transformation matrix for Rx antenna

% generate angular transformation matrix for transmit antenna
for k=0:Nt-1
    for l=0:Nt-1,
        Ut(k+1,l+1)=1/sqrt(Nt)*exp(-1i*2*pi*k*l/Nt);
    end
end

% generate angular transformation matrix for receive antenna
if Nr==Nt
    Ur=Ut;
else
    % compute angular transfer matrix for Ur
    for k=0:Nr-1
        for l=0:Nr-1,
            Ur(k+1,l+1)=1/sqrt(Nr)*exp(-1i*2*pi*k*l/Nr);
        end
    end
end
Gr=ones(Nr,1);
Gt=ones(Nt,1);

%%%%
$\begin{array}{c}
\% SIMULATION \\
\% Create data signal for transmission --- \\
\% \text{numbits}=4;
\% \text{datanum}=[0,5,14,7];
\% \text{datanum}=[0,0,0,0];
\% \text{data}=\text{dec2bin}(...);
\% [\text{graydata}, \text{map}] = \text{bin2gray}(\text{...}, \text{\'qam\',16});
\% xin=-1*(2*(\%numbits/2)-1);
\% xa2=(xin+2*mod(graydata,\%numbits))+1i*(xin+2*floor(graydata/\%numbits));
\% P=xin.*conj(xa);
\% P=xa2.*conj(xa2);
\end{array}$

21
for Plvl=1:Psteps,
    disp(['On step: ' num2str(Plvl) ' of ' num2str(Psteps)]);
    % INITIALIZING CHANNEL AND ADDITIVE NOISE --------------------------
    No=Npower+(Plvl-1)*Pstepsize;
    
    for tr=1:trials,
        for l=1:Nl,
            H(:,:,l)=wgn(Nr,Nt,0,'complex'); %Rayleigh fading
            w(:,l)=wgn(Nr,1,No,'complex');   %AWGN at Rx
        end
    end
    
    % REAL-TIME PROCESSING
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % SPECTRUM ESTIMATION ---------------------------------------------
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % At the Rx
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    Ghthat=zeros(Nt,1);
    Ghrhat=zeros(Nr,1);
    %compute Ghthat to send back to the Tx
    for c=1:Nt
        for b=1:Nr
            Ghhat(b,c)=Ghhat(b,c)+(1/Nl)*abs(rbc)^2;
        end
    end
    % compute Ghrhat and Rrhat
    for b=1:Nr
        for c=1:Nt
            Ghrhat(b)=Ghrhat(b)+Ghhat(b,c)*Gt(c);
        end
    end
Ghrhat=Ghrhat/sqrt(sum(Ghrhat.*Ghrhat));

Rrhat=zeros(Nr,Nr);
for b=1:Nr
    Rrhat=Rrhat+Ghrhat(b)*Ar(:,:,b);
end

[Uhat,Duhat]=eig(Rrhat);

%At the Tx -----------------------------
Rthat=zeros(Nt,Nt);
for c=1:Nt
    Rthat=Rthat+Ghthat(c)*At(:,:,c);
end
[Vhat,Dvhat]=eig(Rthat);

Dvhat2=V'*Rthat*V;
EigValues=max(Dvhat2);

M=2;
EVMax=zeros(M,2);
for m=1:M,
    for n=1:Nt,
        if EigValues(n)==max(EigValues)
            EVMax(m,1)=n;
            EVMax(m,2)=EigValues(n);
        end
    end
    EigValues(EVMax(m,1))=0;
end

%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% SIMULATION USING TRANSMIT BEAMFORMING SCHEME
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%

Vhat2=V;
    Vhat2=Vhat;

    for c=1:Nt,
        useEV=0;
        for m=1:M,
            if EVMax(m,1)==c
                useEV=1;
            end
        end
        if useEV==0
            Vhat2(:,c)=0;
        end
    end

%Apply transmit spatial filter -------
xa=xa2;
for c=1:Nt
    useEV=0;
    for m=1:M,
        if EVMMax(m,1)==c
            useEV=1;
        end
    end
    if useEV==0
        xa(c)=0;
    end
end

scalev=sqrt(Nt/M);
%   scalev=1;
x=Vhat2*xa.*scalev;
%   x=Vhat*xa;
%   x=V*xa;

% Simulate MIMO channel -----------
y=zeros(Nr,Nl);
ya=zeros(Nr,Nl);
for l=1:Nl,
    y(:,l)=H(:,:,l)*x+w(:,l);
end

% Apply receive spatial filter-------
for l=1:Nl,
    ya(:,l)=Uhat'*y(:,l);
%   ya(:,l)=U'*y(:,l);
end

% Decode received signal -----------
for l=1:Nl,
    % CREATE THE BANK OF DECORRELATORS
    % the decorrelator for the kth stream is the kth row of the
    % pseudoinverse of the fading channel matrix H
    Ha=Uhat'*H(:,:,l)*Vhat2;
    %   Ha=U'*H(:,:,l)*V;
    Hpinv=pinv(Ha);
    % DECORRELATE K-th STREAM
    % projection operation followed by matched filter
    % channel & decorrelator for x(k) (for decoding)
    for k=1:Nt,
        if xa(k) ~= 0
            yprime_f=Hpinv(k,:)*ya(:,l);
            chf=Hpinv(k,:)*Ha(:,k);
            yprime=yprime_f;
            % ML DETECTION FOR GRAY CODED 16-QAM SYMBOL FOR K-th
            STREAM
code the real component
if real(yprime)>real(chf)*2*scalev
    xhat=3;
elseif real(yprime)>0
    xhat=1;
elseif real(yprime)>real(chf)*-2*scalev
    xhat=-1;
else
    xhat=-3;
end

%decoding the imaginary component
if imag(yprime)>real(chf)*2*scalev
    xhat=xhat+3j;
elseif imag(yprime)>0
    xhat=xhat+1j;
elseif imag(yprime)>real(chf)*-2*scalev
    xhat=xhat-1j;
else
    xhat=xhat-3j;
end

%SUCCESSION INTERFACE CANCELLATION (SIC)
xrec=zeros(Nt,1);
xrec(k)=xhat;

%DECODE RECEIVED SYMBOL
graydatahat=2*(-1*xmin+imag(xhat))+(-1*xmin+real(xhat))/2;
datanumhat=gray2bin(graydatahat,'qam',16);
datahat=dec2bin(datanumhat,numbits);

%DETERMINE IF DECODED DATA IS CORRECT
BER(Plvl,1,k)=10*log10(P(k))-No;
for a=1:numbits,
    if (strcmp(datahat(a),data(k,a)))
        BER(Plvl,2,k)=BER(Plvl,2,k)+1;
    end
    BER(Plvl,3,k)=BER(Plvl,3,k)+1;
end
end %count the number of bits correct for that symbol
end %Plvl
end %trials
end %Plvl

%%
% CALCULATE BER AND PLOT
for k=1:Nt,
    BER(:,4,k)=1-BER(:,2,k)./BER(:,3,k);
end

BER1=zeros(Psteps,4);
BER2=zeros(Psteps,4);
BER3=zeros(Psteps,4);
BER4=zeros(Psteps,4);

BER1(:,:,1)=BER(:,:,1);
BER2(:,:,2)=BER(:,:,2);
BER3(:,:,3)=BER(:,:,3);
BER4(:,:,4)=BER(:,:,4);

figure;
semilogy(BER1(:,1),BER1(:,4),BER2(:,1),BER2(:,4),BER3(:,1),BER3(:,4),BER4(:,1),BER4(:,4));
legend('BER1','BER2','BER3','BER4');

BER5=zeros(Psteps,4);
BER5(:,1)=BER(:,1,1);
BER5(:,2)=BER(:,2,1)+BER(:,2,2)+BER(:,2,3)+BER(:,2,4);
BER5(:,3)=BER(:,3,1)+BER(:,3,2)+BER(:,3,3)+BER(:,3,4);
BER5(:,4)=1-BER5(:,2)./BER5(:,3);

%%
for c=1:Nt,
   % for m=1:M,
      if EVMax(m,1)==c
         BER5(:,1)=BER(:,1,c);
         BER5(:,2)=BER5(:,2)+BER(:,2,c);
         BER5(:,3)=BER5(:,3)+BER(:,3,c);
      end
   % end
% end

figure;
semilogy(BER5(:,1),BER5(:,4));
xlim([0 35]);

stop clock

s=toc;
disp(['Time taken = ', num2str(s)]); 
disp('Tx Beamforming Complete');
Angular Domain Signal Space Estimation Feedback Scheme MATLAB Code:

tic; %start clock
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% ENVIRONMENT AND SIMULATION SETUP ------------------------------------
%%
$\text{Nr}=4$; %number of Rx antennas
$\text{Nt}=4$; %number of Tx antennas

%Simulation Setup
$\text{Nl}=5$; %number of subcarriers
$\text{Npower}=-17$; %AWG noise power
$\text{Pstepsize}=1$; %Noise power step size
$\text{Psteps}=30$; %Noise power steps
$\text{trials}=5*10^3$; %number of trials
$\text{BER} = \text{zeros}(\text{Psteps, 4, Nt})$;

$\text{H} = \text{zeros}(\text{Nr}, \text{Nt}, \text{Nl})$; %wireless channel gain matrix
$\text{w} = \text{zeros}(\text{Nr}, \text{Nl})$; %AWGN at Rx antenna array
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% A PRIORI PROCESSING
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%
% ANTENNA ARRAY MANIFOLDS ----------------------------------------------
---
$\text{At} = \text{zeros}(\text{Nt}, \text{Nt}, \text{Nt})$;
$\text{Atfull} = \text{zeros}(\text{Nt}, \text{Nt})$;
$\text{deltat}=1/2$;
$\text{Ct} = \text{zeros}(\text{Nt+1}, 1)$;
$\text{Ct}(1)=0$;
for $n=2: \text{Nt+1}$,
    $\text{Ct}(n)=\text{Ct}(n-1)+1/(\text{deltat*Nt})$; %1/(deltat*Nt) = 1/2 for critically spaced 4x4
end

for $k=1: \text{Nt}$, %kth row of At
    for $l=1: \text{Nt}$, %lth column of At
        for $n=1: \text{Nt}$, %nth At (i.e. At for angular cell n)
            $\text{At}(k,l,n)=\text{int}(1/\text{Nt}*[\text{exp}(\text{i}2\pi*(l-k)*\text{deltat}*x),x,\text{Ct}(n),\text{Ct}(n+1))];$
            $\text{At}(k,l,n)=1/\text{Nt}*[\text{exp}(\text{i}2\pi*(l-k)\text{deltat}*\text{Ct}(n))];$
            $\text{Atfull}(k,l)=\text{Atfull}(k,l)+\text{At}(k,l,n);$ %
        end
    end
end

$\text{Ar} = \text{At}$;
% INITIALIZE IDEAL TRANSFORMATION MATRICES -----------------------------

Ut=zeros(Nr,Nr); %angular transformation matrix for Tx antenna
Ur=zeros(Nt,Nt); %angular transformation matrix for Rx antenna

%generate angular transformation matrix for transmit antenna
for k=0:Nt-1
    for l=0:Nt-1,
        Ut(k+1,l+1)=1/sqrt(Nt)*exp(-1i*2*pi*k*l/Nt);
    end
end

%generate angular transformation matrix for receive antenna
if Nr==Nt
    Ur=Ut;
else
    %compute angular transfer matrix for Ur
    for k=0:Nr-1
        for l=0:Nr-1,
            Ur(k+1,l+1)=1/sqrt(Nr)*exp(-1i*2*pi*k*l/Nr);
        end
    end
end

Gr=ones(Nr,1);
Gt=ones(Nt,1);

%%% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% SIMULATION %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Create data signal for transmission ---
numbits=4;
% datanum=[0,5,14,7];
datanum=[0,0,0,0];
data=dec2bin(datanum,numbits);
[graydata,map] = bin2gray(datanum,'qam',16);
xmin=-1*(2*(numbits/2)-1);
xa2=(xmin+2*mod(graydata.',numbits))+1i*(xmin+2*floor(graydata.'./numbits));
P=xa2.*conj(xa2);

for Plvl=1:Psteps,
    disp(['On step: ' num2str(Plvl) ' of ' num2str(Psteps)]);
% INITIALIZING CHANNEL AND ADDITIVE NOISE ----------------------------

No=Npower+(Plvl-1)*Pstepsize;

for tr=1:trials,
    for l=1:Nl,
\( H(:,:,l) = \text{wgn}(N_r, N_t, 0, 'complex'); \)  

Rayleigh fading
\( w(:,l) = \text{wgn}(N_r, 1, N_o, 'complex'); \)  

AWGN at Rx antenna array
end

\%
% REAL-TIME PROCESSING
% SPECTRUM ESTIMATION
-----------------------------------------
-----------------
V=Ut;
U=Ur;
Ghhat=zeros(N_r,N_t);

for c=1:Nt,
  x=V(:,c);
  for b=1:Nr,
    for l=1:N_l,
      y=H(:,:,l)*x+w(:,l);
      rbc=U(:,b)'*y;
      Ghhat(b,c)=Ghhat(b,c)+(1/N_l)*abs(rbc)^2;
    end
  end
end

% TRANSMIT BEAMFORMING
-------------------------------
%At the Rx

Ghthat=zeros(N_t,1);
Ghrhat=zeros(N_r,1);
Jthat=zeros(N_t,1);
%compute Ghthat to send back to the Tx
for c=1:Nt
  for b=1:Nr
    Ghthat(c)=Ghthat(c)+Ghhat(b,c)*Gr(b);
  end
end
theta=0.5*max(Ghthat);
for c=1:Nt,
  if(Ghthat(c)>theta)
    Jthat(c)=1;
  end
end

%compute Ghrhat and Rrhat
for b=1:Nr
  for c=1:Nt
    Ghrhat(b)=Ghrhat(b)+Ghhat(b,c)*Gt(c);
  end
end
Rrhat=zeros(Nr,Nr);
for b=1:Nr
    Rrhat=Rrhat+Ghrhat(b)*Ar(:,:,b);
end

[Uhat,Duhat]=eig(Rrhat);

%At the Tx ------------------------
Rthat=zeros(Nt,Nt);
for c=1:Nt
    %             Rthat=Rthat+Ghthat(c)*At(:,:,c);
    Rthat=Rthat+Jthat(c)*At(:,:,c);
end
[Vhat,Dvhat]=eig(Rthat);

%%%$%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% SIMULATION USING TRANSMIT BEAMFORMING SCHEME
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%Dvhat2=V'*Rthat*V;

Vhat=V;
Uhat=U;

%  %
%  %for c=1:Nt,
%  %   if Jthat(c)==0
%  %      Vhat2(:,c)=0;
%  %end
%  %
%  %Vhat=Vhat2;

%Apply transmit spatial filter --------
% x=Vhat*xa.*sqrt(Nt/sum(Jthat));
% x=V*xa;
xa=xa2;
for c=1:Nt
    if Jthat(c)==0
        xa(c)=0;
    end
end

    scalev=sqrt(Nt/sum(Jthat));
% scalev=1;

x=Vhat2*xa.*scalev;
%Simulate MIMO channel
y=zeros(Nr,Nl);
ya=zeros(Nr,Nl);

for l=1:Nl,
    y(:,l)=H(:,;,:l)*x+w(:,l);
end

%Apply receive spatial filter
for l=1:Nl,
    ya(:,l)=Uhat'*y(:,l);
end

%Decode received signal
%CREATE THE BANK OF DECORRELATORS
%the decorrelator for the kth stream is the kth row of the
%pseudoinverse of the fading channel matrix H
Ha=Uhat'*H(:,;,:l)*Vhat;
% Ha=U'*H(:,;,:l)*V;
Hpinv=pinv(Ha);

%DECORRELATE K-th STREAM
%projection operation followed by matched filter
%channel & decorrelator for x(k) (for decoding)
for k=1:Nt,
    if xa(k) ~= 0
        yprime_f=Hpinv(k,:)*ya(:,l);
        chf=Hpinv(k,:)*Ha(:,k);
        yprime=yprime_f;
        %ML DETECTION FOR GRAY CODED 16-QAM SYMBOL FOR K-th
        %decoding the real component
        if real(yprime)>real(chf)*2*scalev
            xhat=3;
        elseif real(yprime)>0
            xhat=1;
        elseif real(yprime)>real(chf)*-2*scalev
            xhat=-1;
        else
            xhat=-3;
        end
        %decoding the imaginary component
        if imag(yprime)>real(chf)*2*scalev
            xhat=xhat+3j;
        elseif imag(yprime)>0
            xhat=xhat+1j;
        elseif imag(yprime)>real(chf)*-2*scalev
            xhat=xhat-1j;
        else
            xhat=xhat-3j;
        end
    end
end

%SUCCESSIVE INTERFERENCE CANCELLATION (SIC)
xrec=zeros(Nt,1);
\texttt{xrec(k)=xhat;}
\texttt{ya(:,l)=ya(:,l)-Ha*xrec.*scalev;}

\texttt{\%DECODE RECEIVED SYMBOL}
\texttt{graydatahat=2*(-1*xmin+imag(xhat))+(-1*xmin+real(xhat))/2;}
\texttt{datanumhat=gray2bin(graydatahat,'qam',16);}
\texttt{datat=dec2bin(datanumhat,numbits);} 
\texttt{\%DETERMINE IF DECODED DATA IS CORRECT}
\texttt{BER(Plvl,1,k)=10*log10(P(k))-No;}
\texttt{for a=1:numbits,}
\texttt{\quad if(strcmp(datahat(a),data(k,a)))}
\texttt{\quad \quad BER(Plvl,2,k)=BER(Plvl,2,k)+1;}
\texttt{\quad BER(Plvl,3,k)=BER(Plvl,3,k)+1;}
\texttt{end}
\texttt{\quad end %count the number of bits correct for that symbol}
\texttt{end %trials}
\texttt{end %Plvl}

\texttt{%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%}
\texttt{%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%}
\texttt{\% CALCULATE BER AND PLOT}
\texttt{%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%}
\texttt{%%%%}
\texttt{for k=1:Nt,}
\texttt{\quad BER(:,4,k)=1-BER(:,2,k)./BER(:,3,k);}
\texttt{end}

\texttt{BER1=zeros(Psteps,4);}
\texttt{BER2=zeros(Psteps,4);}
\texttt{BER3=zeros(Psteps,4);}
\texttt{BER4=zeros(Psteps,4);}
\texttt{BER1(:,::)=BER(:,::,1);}
\texttt{BER2(:,::)=BER(:,::,2);}
\texttt{BER3(:,::)=BER(:,::,3);}
\texttt{BER4(:,::)=BER(:,::,4);}

\texttt{figure;}
\texttt{semilogy(BER1(:,1),BER1(:,4),BER2(:,1),BER2(:,4),BER3(:,1),BER3(:,4),BE}
\texttt{R4(:,1),BER4(:,4));}
\texttt{legend('BER1','BER2','BER3','BER4');}

\texttt{BER5=zeros(Psteps,4);}
\texttt{BER5(:,1)=BER(:,1,1);}
\texttt{BER5(:,2)=BER(:,2,1)+BER(:,2,2)+BER(:,2,3)+BER(:,2,4);}
\texttt{BER5(:,3)=BER(:,3,1)+BER(:,3,2)+BER(:,3,3)+BER(:,3,4);}
\texttt{BER5(:,4)=1-BER5(:,2)./BER5(:,3);}
\texttt{figure;}
\texttt{semilogy(BER5(:,1),BER5(:,4));}
\texttt{xlim([0 35]);}
s = toc;                  % stop clock
disp(['Time taken = ', num2str(s)]);
disp('Signal Space Estimation Complete');
Angular Domain Signal Space Estimation Feedback Scheme MATLAB Code:

tic; % start clock

% ENVIRONMENT AND SIMULATION SETUP

---

% Antenna Array Properties
Nr=4; % number of Rx antennas
Nt=4; % number of Tx antennas

% Simulation Setup
Nl=5; % number of subcarriers
Npower=-17; % AWG noise power
Pstepsize=1; % Noise power step size
Psteps=30; % Noise power steps
trials=5*10^3; % number of trials
BER=zeros(Psteps,4,Nt); % wireless channel gain matrix
H=zeros(Nr,Nt,Nl); % AWGN at Rx antenna array
w=zeros(Nr,Nl); % noise at Rx antenna array

% A PRIORI PROCESSING

---

% ANTENNA ARRAY MANIFOLDS

At=zeros(Nt,Nt,Nt); %Antenna array manifolds
Atfull=zeros(Nt,Nt); %Antenna array full manifolds
deltat=1/2; % 1/(deltat*Nt) = 1/2 for critically spaced 4x4
Ct=zeros(Nt+1,1); % 1/(deltat*Nt) = 1/2 for critically spaced 4x4
Ct(1)=0;
for n=2:Nt+1,
    Ct(n)=Ct(n-1)+1/(deltat*Nt);
end
for k=1:Nt,
    for l=1:Nt,
        for n=1:Nt,
            At(k,l,n)=int(1/Nt*exp(1j*2*pi*(l-k)*deltat*x),x,Ct(n),Ct(n+1));
        end
    end
end
Ar=At;
% INITIALIZE IDEAL TRANSFORMATION MATRICES

Ut=zeros(Nr,Nr); % angular transformation matrix for Tx antenna
Ur=zeros(Nt,Nt); % angular transformation matrix for Rx antenna

% generate angular transformation matrix for transmit antenna
for k=0:Nt-1
    for l=0:Nt-1,
        Ut(k+1,l+1)=1/sqrt(Nt)*exp(-1i*2*pi*k*l/Nt);
    end
end

generate angular transformation matrix for receive antenna
if Nr==Nt
    Ur=Ut;
else
    % compute angular transfer matrix for Ur
    for k=0:Nr-1
        for l=0:Nr-1,
            Ur(k+1,l+1)=1/sqrt(Nr)*exp(-1i*2*pi*k*l/Nr);
        end
    end
end

Gr=ones(Nr,1);
Gt=ones(Nt,1);

% ANTENNA SELECTION LOOK UP TABLE

% generate binary strings for I_x
bin_num=zeros(2^Nt,Nt); % the binary numbers in a matrix
for n=1:2^Nt,
    bin_str=dec2bin(n-1,Nt); % binary string
    for k=1:Nt,
        bin_num(n,k)=str2double(bin_str(k));
    end
end

I_Nt=eye(Nt,Nt); % basic matrix for I_Nt
I_t=cell(Nt,Nt); % memory for Tx for I_t that minimizes w.c.

% minimizing the worst case condition number of the transmit signal space
for D=1:Nt,
    for n=1:Nt,
        cond_num_max=0;
        % for a given |x|=D
        for k=1:2^Nt,
            if sum(bin_num(k,:))==D
                I_x=I_Nt;
                for l=1:Nt,
                    I_x(l,l)=I_x(l,l)*bin_num(k,l);
                end
            end
        end
    end
end
%for a given |J|=n, maximize condition value
condition_num=0;
for b=1:2^Nt,
    TxSigSpace=zeros(Nt,Nt);
    if sum(bin_num(b,:))==n
        Jj=bin_num(b,:);
        for c=1:Nt,
            TxSigSpace=TxSigSpace+Jj(c)*I_x*At(:,:,c)*I_x;
        end
        d=eig(TxSigSpace);
        if min(d)==0
            temp_cond_num=inf;
        else
            temp_cond=max(d)/min(d);
        end
        if temp_cond_num > condition_num
            condition_num=temp_cond_num;
            Jj_max=Jj;
        end
    end
end
if condition_num < cond_num_max || cond_num_max==0
    I_x_minmax=I_x;
    Jj_minmax=Jj_max;
    cond_num_minmax=cond_num_max;
end
end
I_t{D,n}=I_x_minmax;
end
end
I_Nr=eye(Nr,Nr);
%basic matrix for I_Nt
I_r=cell(Nr,Nr);
%memory for Tx for I_t that minimizes w.c.
%minimizing the worst case condition number of the transmit signal space
for D=1:Nr,
    for n=1:Nr,
        cond_num_max=0;
        %for a given |x|=D
        for k=1:2^Nr,
            if sum(bin_num(k,:))==D
                I_x=I_Nr;
                for l=1:Nr,
                    I_x(l,l)=I_x(l,l)*bin_num(k,l);
                end
                %for a given |J|=n, maximize condition value
                condition_num=0;
                for b=1:2^Nr,
                    RxSigSpace=zeros(Nt,Nt);
                    if sum(bin_num(b,:))==n
                        Ji=bin_num(b,:);
                        for c=1:Nr,
RxSigSpace=RxSigSpace+Ji(c)*I_x*Ar(:,:,c)*I_x;
end

d=eig(RxSigSpace);
if min(d)==0
    temp_cond_num=inf;
else
    temp_cond=max(d)/min(d);
end
if temp_cond_num > condition_num
    condition_num=temp_cond_num;
    Ji_max=Ji;
end
end
end
if condition_num < cond_num_max || cond_num_max==0
    I_x_minmax=I_x;
    Ji_minmax=Ji_max;
    cond_num_minmax=cond_num_max;
end
end
I_r{D,n}=I_x_minmax;
end

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
% SIMULATION
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%
%Create data signal for transmission ---
numbits=4;
% datanum=[0,5,14,7];
datanum=[0,0,0,0];
data=dec2bin(datanum,numbits);
[graydata,map] = bin2gray(datanum,'qam',16);
xmin=-1*(2^(numbits/2)-1);
xa=(xmin+2*mod(graydata.',numbits))+1i*(xmin+2*floor(graydata.'./numbits));
P=xa.*conj(xa);

for Plvl=1:Psteps,
    disp(['On step: ' num2str(Plvl) ' of ' num2str(Psteps)]);
% INITIALIZING CHANNEL AND ADDITIVE NOISE -----------------------------
-------
No=Npower+(Plvl-1)*Pstepsize;

for tr=1:trials,
    for l=1:Nl,
        H(:,:,l)=wgn(Nr,Nt,0,'complex');
        %Rayleigh fading
\[ w(:,l)=\text{wgn}(Nr,1,No,'\text{complex}'); \] % AWGN at Rx

antenna array
end

%%% % REAL-TIME PROCESSING

% SPECTRUM ESTIMATION -----------------------------
V=Ut;
U=Ur;
Ghhat=zeros(Nr,Nt);

for c=1:Nt,
    x=V(:,c);
    for b=1:Nr,
        for l=1:Nl,
            y=H(:,:,l)*x+w(:,l);
            rbc=U(:,b)'*y;
            Ghhat(b,c)=Ghhat(b,c)+(1/Nl)*abs(rbc)^2;
        end
    end
end

% TRANSMIT BEAMFORMING -----------------------------
% At the Rx ----------------------------------------
Ghthat=zeros(Nt,1);
Ghrhat=zeros(Nr,1);
Jthat=zeros(Nt,1);
Jrhat=zeros(Nr,1);
% compute Ghthat to send back to the Tx
for c=1:Nt
    for b=1:Nr
        Ghthat(c)=Ghthat(c)+Ghhat(b,c)*Gr(b);
    end
end
theta=0.5*max(Ghthat);
for c=1:Nt,
    if(Ghthat(c)>theta)
        Jthat(c)=1;
    end
end

% compute Ghrhat and Rrhat
for b=1:Nr
    for c=1:Nt
        Ghrhat(b)=Ghrhat(b)+Ghhat(b,c)*Gt(c);
    end
end
\[ \theta = 0.5 \max(G_{\text{hrhat}}); \]
\[ \text{for } b = 1 : N_r, \]
\[ \quad \text{if } (G_{\text{hrhat}}(b) > \theta) \]
\[ \quad \quad J_{\text{rhat}}(b) = 1; \]
\[ \text{end} \]
\[ \text{end} \]

\[ R_{\text{rhat}} = \text{zeros}(N_r, N_r); \]
\[ \text{for } b = 1 : N_r \]
\[ \quad \text{R}_{\text{rhat}} = R_{\text{rhat}} + G_{\text{hrhat}}(b) \cdot A_r(\cdot, :, b); \]
\[ \quad R_{\text{rhat}} = R_{\text{rhat}} + J_{\text{rhat}}(b) \cdot A_r(\cdot, :, b); \]
\[ \text{end} \]

\[ [U_{\text{hat}}, D_{\text{uhat}}] = \text{eig}(R_{\text{rhat}}); \]

\[ % \text{At the Tx} \] %
\[ R_{\text{that}} = \text{zeros}(N_t, N_t); \]
\[ \text{for } c = 1 : N_t \]
\[ \quad \text{R}_{\text{that}} = R_{\text{that}} + G_{\text{hthat}}(c) \cdot A_t(\cdot, :, c); \]
\[ \quad R_{\text{that}} = R_{\text{that}} + J_{\text{that}}(c) \cdot A_t(\cdot, :, c); \]
\[ \text{end} \]

\[ [V_{\text{hat}}, D_{\text{vhat}}] = \text{eig}(R_{\text{that}}); \]

\[ J_{\text{rhatnum}} = \text{sum}(J_{\text{rhat}}); \]
\[ J_{\text{thatnum}} = \text{sum}(J_{\text{that}}); \]

\[ D_{\text{hat}} = \min(J_{\text{rhatnum}}, J_{\text{thatnum}}); \]
\[ I_{\text{r}}(D_{\text{hat}}, J_{\text{thatnum}}); \]
\[ I_{\text{t}}(D_{\text{hat}}, J_{\text{rhatnum}}); \]

\[ V_{\text{hat}} = V; \]
\[ U_{\text{hat}} = U; \]

\[ % \]
\[ % \text{SIMULATION USING TRANSMIT BEAMFORMING SCHEME} \]

\[ % \text{Apply transmit spatial filter} \]
\[ % \quad V_{\text{hat}} = V_{\text{hat}} \cdot I_{\text{t}}(D_{\text{hat}}, J_{\text{thatnum}}); \]
\[ % \quad U_{\text{hat}} = U_{\text{hat}} \cdot I_{\text{r}}(D_{\text{hat}}, J_{\text{rhatnum}}); \]

\[ x_a = x_a^2; \]
\[ \text{for } c = 1 : N_t \]
\[ \quad \text{if } I_{\text{t}}(D_{\text{hat}}, J_{\text{thatnum}})(c, c) == 0 \]
\[ \quad \quad x_a(c) = 0; \]
\[ \quad \text{end} \]
\[ \text{end} \]

\[ \text{scalev} = \sqrt{N_t / D_{\text{hat}}}; \]
\[ \text{scalev} = 1; \]
x=Vhat2*xa.*scalev;

% x=Vhat*xa*sqrt(Nt/Dhat);
% x=V*xa;

%Simulate MIMO channel ---------------
y=zeros(Nr,Nl);
ya=zeros(Nr,Nl);

for l=1:Nl,
    y(:,l)=H(:,:,l)*x+w(:,l);
end

%Apply receive spatial filter ---------
for l=1:Nl,
    ya(:,l)=Uhat'*y(:,l);
    ya(:,l)=U'*y(:,l);
end

%Decode received signal ---------------
for l=1:Nl,
    %CREATE THE BANK OF DECORRELATORS
    %the decorrelator for the kth stream is the kth row of the %pseudoinverse of the fading channel matrix H
    Ha=Uhat'*H(:,:,l)*Vhat;
    Ha=U'*H(:,:,l)*V;
    Hpinv=pinv(Ha);

    %DECORRELATE K-th STREAM
    %projection operation followed by matched filter
    %channel & decorrelator for x(k) (for decoding)
    for k=1:Nt,
        if I_r{Dhat,Jthatnum}(k,k)==1
            yprime_f=Hpinv(k,:)*ya(:,l);
            chf=Hpinv(k,:)*Ha(:,k);
            yprime=yprime_f;
            %ML DETECTION FOR GRAY CODED 16-QAM SYMBOL FOR K-th
            STREAM

            %decoding the real component
            if real(yprime)>real(chf)*2*scalev
                xhat=3;
            elseif real(yprime)>0
                xhat=1;
            elseif real(yprime)>real(chf)*-2*scalev
                xhat=-1;
            else
                xhat=-3;
            end

            %decoding the imaginary component
            if imag(yprime)>real(chf)*2*scalev
                xhat=xhat+3j;
            elseif imag(yprime)>0
                xhat=xhat+1j;
            elseif imag(yprime)>real(chf)*-2*scalev
                xhat=xhat-1j;
            else
                xhat=xhat-3j;
            end
xhat=xhat-3j;
end

%SUCCESSIONAL INTERFERENCE CANCELLATION (SIC)
xrec=zeros(Nt,1);
xrec(k)=xhat;
ya(:,l)=ya(:,l)-Ha*xrec.*scalev;

%DECODE RECEIVED SYMBOL
graydatahat=2*(-1*xmin+imag(xhat))+(-1*xmin+real(xhat))/2;
datanumhat=gray2bin(graydatahat,'qam',16);
dat handwritten by you
BER5(:,3) = BER(:,3,1) + BER(:,3,2) + BER(:,3,3) + BER(:,3,4);
BER5(:,4) = 1 - BER5(:,2) ./ BER5(:,3);

figure;
semilogy(BER5(:,1), BER5(:,4));
xlim([0 35]);

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% s = toc; % stop clock
disp(['Time taken = ', num2str(s)]);
disp('Antenna Selection Complete');