An Implementation of Discrete Multi-Tone over Slowly Time-varying Multiple-Input/Multiple-Output Channels[1]

Ardavan Maleki Tehrani, Arash Hassibi, John Cioffi, Stephen Boyd

Information Systems Lab.
Department of Electrical Engineering
Stanford University, CA 94305, USA
ardavan, arash, cioffi, boyd@ISL.Stanford.EDU

Abstract

We have investigated a method for data transmission over slowly time-varying MIMO channels. A low complexity method is introduced that effectively diagonalizes the MIMO channel. This enables the use of Discrete Multi-Tone (DMT) modulation over the MIMO channel to achieve information transmission rates close to Shannon capacity.

DMT requires knowledge of the channel state information at the transmitter which is not always possible in practice. In this case the channel can be only made block diagonal and signal detection requires the solution to a least-squares problem with integer variables. This is a very challenging problem that is theoretically difficult (NP-hard). In this paper, a practically efficient method is proposed to solve this least-squares problem.

Introduction

Multichannel modulation methods such as Multi-Tone, OFDM (Orthogonal Frequency Division Multiplexing), and DMT (Discrete Multi-Tone) are in general one of the best methods for data transmission channels with severe inter-symbol interference (ISI). The concept of DMT [1] has been analyzed extensively for single-input, single-output (SISO) channels. It has been shown that DMT is able to achieve data transmission rates close to Shannon capacity, given the channel state information is provided at the transmitter. This motivates to investigate the application of DMT for multiple-input,multiple-output (MIMO) channels. Figure 1 shows a block diagram of such a system. The concept of DMT lies behind two factors. One is (fast) diagonalization of the channel matrix and the other is using water-filling on the input data to maximize the information rate sent over the diagonalized channel.

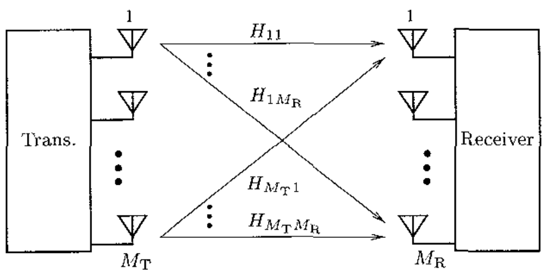

Figure 1: Block diagram of the MIMO system

Here we will consider the case of block-time-invariant MIMO channels, where the channel is assumed to be time-invariant during transmission of each block of data. One important aspect of such channels that has motivated us to study MIMO systems is the fact that the capacity of such channels depends linearly on the number of antennas utilized [2, 3, 4]. Though the formulations won't suggest how to achieve this capacity but it is a large gain compared to the SISO case. Few number of studies have been done on the problem of transmission over MIMO time invariant channels in the literature [5, 6, 7, 8, 9, 10]. However, less study has been done on time-varying MIMO channels. [11] has approached the problem by introducing a multi-layer transmission structure using multiple antennas, and has come up with capacity formulations for the proposed structure. In our opinion, the multi-carrier based approaches such as OFDM and DMT, are superior implementation-wise to approaches requiring multiple equalizers at the receiver. In this paper we will consider the approach taken by [4], where they have analyzed the capacity of a slowly time-varying wireless MIMO channel, and have proposed an OFDM communication structure for the

[1]Research supported in part by AFOSR (under F49620-95-1-0318), NSF (under ECS-9222391 and EEC-9420565), MURI (under F49620-95-1-0525), Lucent Technologies and Stanford Center for Telecommunciations.

0-7803-4984-9/98/$10.00 ©1998 IEEE.

Authorized licensed use limited to: Stanford University. Downloaded on April 06,2010 at 14:21:14 EDT from IEEE Xplore. Restrictions apply.

MIMO wireless channel. DMT requires knowledge of the channel state information at the transmitter. This is not always possible in practice. If the transmitter does not have the knowledge of the channel state information, the channel can be only made block diagonal, and in this case, (optimal) signal detection at the receiver becomes very challenging. More specifically, signal detection requires the solution to a least-squares problem with integer variables which is theoretically very difficult (NP-hard). The solution to such a least-squares problem is also required in GPS (Global Positioning System) signal processing and has been of significant research interest in the GPS field for some years now (cf. [12, 13] and references therein). We will show that although the solution to this problem is theoretically difficult, it can be solved rather efficiently in practice using an algorithm from the theory of geometry of numbers due to Lenstra, Lenstra, and Lovász [14, 15].

In §1 we describe the channel model and discuss channel block diagonalization, in §2 we will use the results obtained in §1 to completely diagonalize the channel, and will discuss the concept of MIMO DMT. Finally, in §3 we address the problem of signal detection.

1 MIMO System Model

The aim of this section is to present a matrix equation for the MIMO system and to show how this channel matrix can be block diagonalized efficiently using FFT and IFFT algorithms by adding redundancy to the input vector. The diagonalization of the channel matrix leads to N independent channels, over which information can be sent independently. Using a generalized water-filling solution the input vector can be optimized to achieve channel capacity.

We will assume a block time-invariant channel model, *i.e.*, the channel remains unchanged during one block period. A block consists of N data symbols and ν cyclic prefix symbols. We will also assume an additive white Gaussian noise (AWGN) channel with M_{T} transmitting and M_{R} receiving antennas.

Over one block of data transmission, the channel input/output relationship is given by

$$y_j(k) = h_{ij}(k) * x_i(k) + n_j(k), \quad i = 1, \ldots, M_{\mathrm{T}}, \quad j = 1, \ldots, M_{\mathrm{R}},$$

$y_j(k)$ is the channel output (at the jth receiving antenna), $x_i(k)$ is the channel input (at the ith transmitting antenna), and h_{ij} is the channel impulse response from the ith transmitting antenna to the jth receiving antenna. We assume all channel impulse responses to be of finite length ν, *i.e.*, $h_{ij}(k) = 0$ for $k < 0$ and $k > \nu$.

At the transmitter, each data block of length N is concatenated with its first ν data symbols (cyclic prefix). At the receiver, the first and last ν symbols will be discarded, and the middle N symbols from each receiving antenna will be retained. Let $H_{ij} \in \mathbf{C}^{N\times N}$ be the matrix representation of this block transmission so that

$$\mathbf{y}_j = H_{ij}\mathbf{x}_i + \mathbf{n}_j,$$

where $\mathbf{x}_i = [x_i(0) \ \cdots \ x_i(N-1)]^T$, $\mathbf{y}_j = [y_j(0) \ \cdots \ y_j(N-1)]^T$, and $\mathbf{n}_j = [n_j(0) \ \cdots \ n_j(N-1)]^T$ are respectively, the block of N data symbols at the ith transmitting antenna, block of N received symbols at the jth receiving antenna, and block of N additive white Gaussian noise samples at the jth receiving antenna.

Assuming the first ν symbols of $\mathbf{x}_i$ is the cyclic prefix that is added to the transmitted data block $\mathbf{x}_i$, then the $N \times N$ channel matrix is *circulant* and is given by

$$H_{ij} = \begin{bmatrix} h_{ij}(0) & 0 & \cdots & h_{ij}(\nu) & \cdots & h_{ij}(1) \\ \vdots & h_{ij}(0) & \ddots & 0 & \cdots & h_{ij}(2) \\ h_{ij}(\nu) & \vdots & & & & \vdots \\ \vdots & h_{ij}(\nu) & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & & & \\ 0 & 0 & \cdots & \cdots & h_{ij}(1) & h_{ij}(0) \end{bmatrix} \tag{1}$$

Now defining

$$\mathbf{x}^T = [\mathbf{x}_1^T \ \cdots \ \mathbf{x}_{M_{\mathrm{T}}}{}^T], \quad \mathbf{y}^T = [\mathbf{y}_1^T \ \cdots \ \mathbf{y}_{M_{\mathrm{R}}}{}^T],$$
$$\mathbf{n}^T = [\mathbf{n}_1^T \ \cdots \ \mathbf{n}_{M_{\mathrm{R}}}{}^T],$$

and

$$\mathbf{H} = \begin{bmatrix} H_{11} & H_{12} & \cdots & H_{1M_{\mathrm{T}}} \\ H_{21} & H_{22} & \cdots & H_{2M_{\mathrm{T}}} \\ \vdots & \ddots & \ddots & \vdots \\ H_{M_{\mathrm{R}}1} & \cdots & \cdots & H_{M_{\mathrm{T}}M_{\mathrm{T}}} \end{bmatrix}, \tag{2}$$

the matrix equation for the MIMO system becomes

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}. \tag{3}$$

If the (block) SVD of $\mathbf{H}$ can be easily computed, the (generalized) water-filling solution can be readily obtained, which would enable us to send data close to Shannon capacity. Since by adding the cyclic prefix we made the H_{ij} matrices circulant, the (block) SVD of $\mathbf{H}$ can be easily computed using FFT and IFFT algorithms.

The SVD of H_{ij} is given by $H_{ij} = Q^*\Lambda_{ij}Q$ where Q and Q^* are the FFT and IFFT matrices ($Q_{kl} = \frac{1}{\sqrt{N}}\exp(-j2\pi kl/N)$ for $k,l = 1,\ldots,N$), and Λ_{ij} is a diagonal matrix. Defining $\mathbf{Q}_{M_{\mathrm{T}}}$ and $\mathbf{Q}_{M_{\mathrm{R}}}$ respectively as block diagonal matrices with M_{T} and M_{R} blocks of Q on their diagonals, and $\Lambda_{ij} = \mathbf{diag}\,(\lambda_{ij}(0), \lambda_{ij}(1), \cdots, \lambda_{ij}(N-1))$ we have

$$\mathbf{Q}_{M_{\mathrm{R}}}\mathbf{H}\mathbf{Q}^*_{M_{\mathrm{T}}} = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} & \cdots & \Lambda_{1M_{\mathrm{T}}} \\ \Lambda_{21} & \Lambda_{22} & \cdots & \Lambda_{2M_{\mathrm{T}}} \\ \vdots & \ddots & \ddots & \vdots \\ \Lambda_{M_{\mathrm{R}}1} & \cdots & \cdots & \Lambda_{M_{\mathrm{R}}M_{\mathrm{T}}} \end{bmatrix}.$$

Let $\mathbf{\Lambda}$ be the matrix on the right, then matrix elements Λ_{ij} of the $\mathbf{\Lambda}$ are each diagonal. Therefore, by multiplying $\mathbf{\Lambda}$ on the left and right by permutation matrices $P_{M_{\mathrm{R}}}$ and $P_{M_{\mathrm{T}}}$ respectively, $\mathbf{\Lambda}$ can be transformed into a block diagonal matrix $\mathbf{\Lambda}_{\mathrm{D}} = \mathbf{diag}(\Delta_1, \cdots, \Delta_N)$ satisfying

$$\mathbf{\Lambda}_{\mathrm{D}} = P_{M_{\mathrm{R}}}\mathbf{\Lambda}P_{M_{\mathrm{T}}} = P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}}\mathbf{H}\mathbf{Q}^*_{M_{\mathrm{T}}}P_{M_{\mathrm{T}}} \tag{4}$$

Authorized licensed use limited to: Stanford University. Downloaded on April 06,2010 at 14:21:14 EDT from IEEE Xplore. Restrictions apply.

where for $i = 1, \ldots, N$

$$\Delta_i = \begin{bmatrix} \lambda_{11}(i) & \lambda_{12}(i) & \cdots & \lambda_{1M_{\mathrm{T}}}(i) \\ \lambda_{21}(i) & \lambda_{22}(i) & \cdots & \lambda_{2M_{\mathrm{T}}}(i) \\ & \ddots & \ddots & \vdots \\ \lambda_{M_{\mathrm{R}}1}(i) & \cdots & \cdots & \lambda_{M_{\mathrm{R}}M_{\mathrm{T}}}(i) \end{bmatrix}.$$

The permutation matrix $P_{M_{\mathrm{T}}}$ is an $NM_{\mathrm{T}} \times NM_{\mathrm{T}}$ matrix where all of the nonzero elements are unity and located at positions (i, j) satisfying

$$i = \left\lfloor \frac{j}{M_{\mathrm{T}}} \right\rfloor + Nk, \quad k = 0, \cdots, M_{\mathrm{T}} - 1, \;\; j = 1, \cdots, N \cdot M_{\mathrm{T}}.$$

Similarly, $P_{M_{\mathrm{R}}}$ is $NM_{\mathrm{R}} \times NM_{\mathrm{R}}$ with unity elements at positions (i, j) given by

$$j = \left\lfloor \frac{i}{M_{\mathrm{R}}} \right\rfloor + Nk \quad \text{where:} \quad k = 0, \cdots, M_{\mathrm{R}} - 1, \;\; i = 1, \cdots, M_{\mathrm{R}} \cdot N.$$

From the channel equation $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}$ we get

$$P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}}\mathbf{y} = P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}}\mathbf{H}\mathbf{x} + P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}}\mathbf{n},$$

and assuming

$$\mathbf{x} = \mathbf{Q}^*_{M_{\mathrm{T}}} P_{M_{\mathrm{T}}}\mathbf{X}, \quad \mathbf{Y} = P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}}\mathbf{y}, \quad \mathbf{N} = P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}}\mathbf{n}$$

we get

$$\mathbf{Y} = (P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}}\mathbf{H}\mathbf{Q}^*_{M_{\mathrm{T}}}P_{M_{\mathrm{T}}})\mathbf{X} + \mathbf{N} = \mathbf{\Lambda}_{\mathrm{D}}\mathbf{X} + \mathbf{N}$$

or

$$\mathbf{Y} = \mathbf{\Lambda}_{\mathrm{D}}\mathbf{X} + \mathbf{N} \tag{5}$$

Equation (5) represents the block diagonalized channel matrix equation. In practice, blocks of data are concatenated to form the vector $\mathbf{X}$ which is then permuted according to $P_{M_{\mathrm{T}}}$ and passed through the IFFT matrix $\mathbf{Q}^*_{M_{\mathrm{T}}}$. Then, consecutive length-N blocks of the resulting N-point IFFT are sent to the M_{T} transmitting antennas. Figure 2 shows a block diagram of the above implementation.

At the receiver, the output symbols from each of the M_{R} receiving antennas are grouped in blocks of length N and are then concatenated to form the vector $\mathbf{y}$ which is passed through the FFT matrix $\mathbf{Q}_{M_{\mathrm{R}}}$. The resulting N-point FFT vector is permuted according to $P_{M_{\mathrm{R}}}$ and would yield vector $\mathbf{Y}$.

Note that the effect of the permutation matrices, $P_{M_{\mathrm{T}}}$ and $P_{M_{\mathrm{R}}}$, on the vectors $\mathbf{X}$ and FFT of $\mathbf{y}$ is just simple reordering.

2 MIMO DMT

In this section we will discuss the issue of transmitter optimization and DMT for the MIMO channel (in terms of how much energy we should put in different frequency bins) to obtain the highest possible channel capacity.

In the SISO case this is achieved by a water-filling solution, and the whole multicarrier system is named DMT, which is basically OFDM with an optimized transmit data vector. A similar approach leads to the optimal transmit vector in the MIMO case, hence the name MIMO DMT. We would like to mention that Roy, Yang and Kumar [6] have solved a similar problem for the continuous-time case, however here we will focus on the discrete-time case.

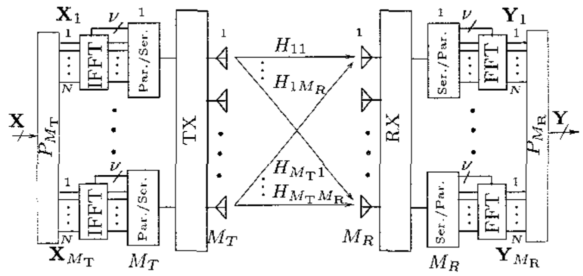

Figure 2: Detailed block diagram of the MIMO system

According to [4] the channel capacity for a slowly time-varying, ergodic, AWGN MIMO channel, with channel matrix $\mathbf{H}$ is obtained from

$$C_{M_{\mathrm{R}}M_{\mathrm{T}}} = \mathbf{E}\left\{ \sum_{i=1}^{n} \log\left[1 + \frac{\epsilon_i |\lambda_{H_i}|^2}{\sigma^2}\right] \right\}$$

where $\lambda_{H_i}s$ for $i = 1, \ldots, n$ are the singular values of the channel matrix $\mathbf{H}$ that are greater than a certain threshold that depends on the transmission energy, and ϵ_is are the energies assigned to each frequency bin of the transmitted vector $\mathbf{X}$ found from the water-filling solution (cf. [1] for details). Therefore, the key to finding the energy allocations ϵ_i that maximize the channel capacity is to compute the singular values of $\mathbf{H}$.

Since $\mathbf{H}$ is assumed to be known and can be block diagonalized efficiently to $\mathbf{\Lambda}_{\mathrm{D}} = \mathbf{diag}(\Delta_1, \ldots, \Delta_N)$ as in (4), the singular values of $\mathbf{H}$ can be computed easily by finding the singular values of the $M_{\mathrm{R}} \times M_{\mathrm{T}}$ matrices Δ_i.

Suppose that $F_i^* \Delta_i M_i = \left[\frac{\Sigma_i}{0}\right]$ (F_i, M_i are orthogonal and Σ_i is diagonal) follows from the SVD of the diagonal block Δ_i of $\mathbf{\Lambda}_{\mathrm{D}}$ (assuming $M_{\mathrm{R}} \geq M_{\mathrm{T}}$). Then

$$\mathbf{F}^*\mathbf{\Lambda}_{\mathrm{D}}\mathbf{M} = \mathbf{diag}\left(\left[\frac{\Sigma_1}{0}\right], \cdots, \left[\frac{\Sigma_N}{0}\right],\right) \tag{6}$$

where

$$\mathbf{F}^* = \mathbf{diag}\left(F_1^*, \cdots, F_N^*\right), \quad \mathbf{M} = (M_1, \cdots, M_N),$$
$$\mathbf{\Sigma} = \mathbf{diag}\left(\left[\tfrac{\Sigma_1}{0}\right], \cdots, \left[\tfrac{\Sigma_N}{0}\right]\right).$$

These relations lead to the following decomposition of H,

$$\mathbf{\Sigma} = \mathbf{F}^*\mathbf{\Lambda}_{\mathrm{D}}\mathbf{M} = (\mathbf{F}^* P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}})\mathbf{H}(\mathbf{Q}^*_{M_{\mathrm{T}}}P_{M_{\mathrm{T}}}\mathbf{M}) \tag{7}$$

which is in fact an SVD. (Since F_i, Q, M_i, $P_{M_{\mathrm{R}}}$ and $P_{M_{\mathrm{T}}}$ are unitary, it can be easily checked that so are $\mathbf{F}$, $\mathbf{Q}_{M_{\mathrm{R}}}$, $\mathbf{Q}_{M_{\mathrm{T}}}$, $\mathbf{M}$, and as a result the products $\mathbf{F}^* P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}}$ and $\mathbf{Q}^*_{M_{\mathrm{T}}}P_{M_{\mathrm{T}}}\mathbf{M}$.) Nonzero elements of $\mathbf{\Sigma}$ are the singular values of $\mathbf{H}$ and can now be used to compute the optimum energy allocations ϵ_i using the water-filling method.

The channel equation can now be written as

$$\mathbf{Y} \;\; = \;\; (\mathbf{F}^* P_{M_{\mathrm{R}}}\mathbf{Q}_{M_{\mathrm{R}}})\mathbf{H}(\mathbf{Q}^*_{M_{\mathrm{T}}}P_{M_{\mathrm{T}}}\mathbf{M})\mathbf{X} + \mathbf{N}$$

Authorized licensed use limited to: Stanford University. Downloaded on April 06,2010 at 14:21:14 EDT from IEEE Xplore. Restrictions apply.

$$= \mathbf{F}^*\Lambda_D\mathbf{M}+\mathbf{N}$$
$$= \Sigma\mathbf{X}+\mathbf{N}.$$

The above equation represents the complete diagonalized channel. Now in comparison to the block diagonalized case given in (5), the concatenated blocks of data forming vector $\mathbf{X}$, are first passed through the matrix M before being permuted according to P_{M_T} and passed through the IFFT matrix Q_{M_T}.

Similarly at the receiver, the received blocks of length N from the receiving antennas are concatenated to form vector $\mathbf{y}$ which is then passed through the FFT matrix Q_{M_R}. The resulting vector is permuted according to P_{M_R} and passed through matrix $\mathbf{F}^*$ which would finally yield vector $\mathbf{Y}$.

The input vector $\mathbf{X}$ is such that its energy in each frequency bin is optimal. The result would be the same as using a continuous time transmitting filter as proposed by [9].

The important fact to be noted is that the complexity of the above algorithm is much less than normal SVD algorithms. The complexity of performing a normal SVD on channel H is $O(N^3\alpha^3)$ while the complexity here is $O(N^2\alpha^4\log(N\alpha))$, where $\alpha=\min(M_T,M_R)$.

3 MIMO DMT Symbol Detection

The input/output relationship over one block of data for the MIMO AWGN channel is given by

$$\mathbf{Y}=\mathbf{H}\mathbf{X}+\mathbf{N}.$$

The goal of the symbol detection step is to estimate $\mathbf{X}$ given $\mathbf{Y}$. The optimum (maximum likelihood) estimate of $\mathbf{X}$ is given by the minimum least-squares formula

$$\hat{\mathbf{X}} = \underset{\mathbf{X}\in\mathcal{C}^{NM_T}}{\textbf{argmin}} \ \|\mathbf{Y}-\mathbf{H}\mathbf{X}\|^2 \quad (8)$$

$$= \underset{\mathbf{X}\in\mathcal{C}^{NM_T}}{\textbf{argmin}} \ \left(\mathbf{X}-(\mathbf{H}^*\mathbf{H})^{-1}\mathbf{H}^*\mathbf{Y}\right)^*\mathbf{H}^*\mathbf{H}\left(\mathbf{X}-(\mathbf{H}^*\mathbf{H})^{-1}\mathbf{H}^*\mathbf{Y}\right) \quad (9)$$

where $\mathcal{C}$, the symbol constellation, can be assumed to be a subset of the integer lattice $\mathbf{Z}$ (if the symbols are complex then $\mathcal{C}$ can be assumed to be a subset of $\mathbf{Z}^2$).

The fact is that solving (8) or (9) for general $\mathbf{H}$ is very difficult (NP-hard) due to the discrete nature of $\mathbf{X}$. A straightforward yet very inefficient method for finding $\hat{\mathbf{X}}$ is through an exhaustive search which requires trying all $|\mathcal{C}|^{NM_T}$ possibilities for $\mathbf{X}$.

The solution to (8) or (9) is very easy in one special case. This is when $\mathbf{H}$ has orthogonal columns so that $\mathbf{H}^*\mathbf{H}$ is diagonal and the minimum least-squares problem (9) decouples into NM_T one-dimensional least-squares problems

$$\hat{\mathbf{X}}_i = \underset{\mathbf{X}_i\in\mathcal{C}}{\textbf{argmin}} \ |\mathbf{X}_i-\left((\mathbf{H}^*\mathbf{H})^{-1}\mathbf{H}^*\mathbf{Y}\right)_{ii}|^2, \quad i=1,\ldots,NM_T,$$

and therefore $\mathbf{X}$ can be found by component-wise rounding of (the pseudo-inverse solution) $(\mathbf{H}^*\mathbf{H})^{-1}\mathbf{H}^*\mathbf{Y}$ to the nearest element in $\mathcal{C}$.

Under the MIMO DMT structure of §2, the channel is completely decoupled and $\mathbf{H}=\Sigma$ is orthogonal (or equivalently $\mathbf{H}^*\mathbf{H}$ is diagonal). Therefore, computing the maximum likelihood symbol estimates is no big issue (in terms of complexity) and it can be easily found by component-wise rounding of the pseudo-inverse solution, *i.e.*, $\hat{\mathbf{X}}_i=\textbf{round}\left([\Sigma_i^{-1}\ 0]\mathbf{Y}_i\right)$ (here $\hat{\mathbf{X}}_i$ and $\mathbf{Y}_i$ are the ith $M_T\times 1$ blocks of $\hat{\mathbf{X}}$ and $\mathbf{Y}$ respectively).

Under the structure of §1 (in which the transmitter does not incorporate the matrices $\mathbf{F}$ and $\mathbf{M}$ in its structure[1]), the channel matrix $\mathbf{H}$ is no longer orthogonal as in §2 so the minimum least-squares problem is not completely decoupled. However, $\mathbf{H}=\mathbf{\Lambda}_D$ is block diagonal and therefore the minimum least-squares problem *partially* decouples into N minimum least-squares problem of smaller size

$$\hat{\mathbf{X}}_i = \underset{\mathbf{X}_i\in\mathcal{C}^{M_T}}{\textbf{argmin}} \ \left(\mathbf{X}_i-(\Delta_i^*\Delta_i)^{-1}\Delta_i^*\mathbf{Y}_i\right)^*\Delta_i^*\Delta_i\left(\mathbf{X}_i-(\Delta_i^*\Delta_i)^{-1}\Delta_i^*\mathbf{Y}_i\right), \quad (10)$$

or equivalently,

$$\hat{\mathbf{X}}_i = \underset{\mathbf{X}_i\in\mathcal{C}^{M_T}}{\textbf{argmin}} \ \left\|\tilde{\mathbf{Y}}_i-G_i\mathbf{X}_i\right\|^2, \quad (11)$$

where $G_i=(\Delta_i^*\Delta_i)^{-1}$ and $\tilde{\mathbf{Y}}_i=G_i\Delta_i^*\mathbf{Y}_i$ for $i=1,\ldots,N$. In fact, we have reduced the problem of solving a least-squares problem of size NM_T to an easier problem of solving N minimum least-squares problems of size M_T. However, for an MIMO channel with $M_T>1$ it is still not straightforward to solve these N least-squares problems.

The set

$$\mathbf{L}(G_i)\triangleq\left\{\ G_i\mathbf{X}_i\ |\ \mathbf{X}_i\in\mathbf{Z}^{M_T}\right\}$$

is a lattice and therefore (11) can be interpreted as finding the closest lattice point to $\tilde{\mathbf{Y}}_i$ under the constraint $\mathbf{X}_i\in\mathcal{C}^{M_T}$. If G_i is orthogonal then $\hat{\mathbf{X}}_i$ can be simply found by rounding each component of $G_i^{-1}\tilde{\mathbf{Y}}_i$ to the nearest element in $\mathcal{C}$. Therefore, we would hope, that if G_i is in some sense "almost orthogonal", rounding the components of $G_i^{-1}\tilde{\mathbf{Y}}_i$ would yield a solution that is "close" if not exactly the same as the optimum solution $\hat{\mathbf{X}}_i$. This is basically the idea behind an algorithm to efficiently solve (10) or (11) which is the subject of the next subsections.

3.1 Suboptimal algorithm for the least-squares problem

In this subsection, we describe a suboptimal polynomial-time algorithm for solving (10) or (11). Suboptimal algorithms of this kind are important for a few reasons. First, suboptimal algorithms can be performed efficiently with a guaranteed low worst-case complexity. Second, they provide a relatively good initial guess for any global optimization algorithm, and finally, these suboptimal algorithms might

[1]This could be the case when the channel matrix $\mathbf{H}$ is unknown to the transmitter. Since the $\mathbf{H}$ is usually estimated at the receiver there should be some feedback from the receiver to the transmitter if we want to use the MIMO DMT structure of §2, but this is not always practical.

Authorized licensed use limited to: Stanford University. Downloaded on April 06,2010 at 14:21:14 EDT from IEEE Xplore. Restrictions apply.

find the global optimum as they often do in practice. If $d_{\min}$, the minimum length vector in $\mathbf{L}(G_i)$, or any lower bound $d \le d_{\min}$ on it is known, a sufficient condition for the suboptimal minimizer $\hat{\mathbf{X}}_{i,\text{sub}}$ to be the global minimizer $\hat{\mathbf{X}}_i$ is simply given by

$$\|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_{i,\text{sub}}\| \le \frac{d}{2} \Longrightarrow \hat{\mathbf{X}}_{i,\text{sub}} = \hat{\mathbf{X}}_i, \qquad (12)$$

as there is only one lattice point in a ball centered at $G_i\hat{\mathbf{X}}_{i,\text{sub}}$ and with radius $d_{\min}/2$.

Suppose that F_i is a unimodular matrix, *i.e.*, $F_i, F_i^{-1} \in \mathbf{Z}^{M_\text{T}\times M_\text{T}}$ so that it is an onto mapping from $\mathbf{Z}^{M_\text{T}}$ to $\mathbf{Z}^{M_\text{T}}$. Therefore, in (11) we can change variables to $\mathbf{W}_i = F_i^{-1}\mathbf{X}_i$ so the optimization in the new variable $\mathbf{W}_i \in \mathbf{Z}^{M_\text{T}}$ becomes

$$\hat{\mathbf{W}}_i = \underset{\mathbf{W}_i \in \bar{\mathcal{C}}}{\textbf{argmin}} \left\|\tilde{\mathbf{Y}}_i - G_iF_i\mathbf{W}_i\right\|^2,$$

where $\bar{\mathcal{C}}$ is the mapping of $\mathcal{C}^{M_\text{T}}$ under F_i^{-1}. If F_i can be chosen such that G_iF_i becomes orthogonal then $\hat{\mathbf{X}}_i$ can be found by, component-wise rounding of $(G_iF_i)^{-1}\tilde{\mathbf{Y}}_i$ to the closest integer, and multiplying the result by F_i to get a vector whose components should be mapped to the closest element in $\mathcal{C}$. However, such an F_i usually does not exist, and in practice, one can only hope to find an F_i that "almost orthogonalizes" G_iF_i. If such an F_i is found, it is reasonable to believe that the change of variables to $\mathbf{W}_i$ followed by rounding would give a "close" to optimal solution.

There is an algorithm due to Lenstra, Lenstra, and Lovász (LLL algorithm) that finds such an F_i. This algorithm is polynomial-time and practically efficient. For details of the LLL algorithm and its different variations refer to [14, 15, 16, 13, 17, 18] and references therein. The following suboptimal algorithm for solving the least-squares problem (which makes use of the LLL algorithm) is based on the heuristic that rounding would give a "close" to optimal solution if G_i is "almost" orthogonal.

Suboptimal algorithm for solving the least-squares problem. Suppose that G_i and $\tilde{\mathbf{Y}}_i$ are given. A suboptimal solution $\hat{\mathbf{X}}_{i,\text{sub}}$ to (11) in the sense that when $\mathcal{C} = \mathbf{Z}$

$$\|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_{i,\text{sub}}\| \le \left(1 + 2M_\text{T}\,(4.5)^{M_\text{T}/2}\right) \min_{\mathbf{X}_i \in \mathbf{Z}^{M_\text{T}}} \|\tilde{\mathbf{Y}}_i - G_i\mathbf{X}_i\|, \qquad (13)$$

exists that can be found as follows [16]:

- Perform the LLL algorithm on G_i. This results in a new matrix $\bar{G}_i$ which is almost orthogonal and a unimodular matrix F_i such that $\bar{G} = G_iF_i$.
- $\hat{\mathbf{X}}_{i,\text{tmp}} \leftarrow F_i\lceil \bar{G}_i^{-1}\tilde{\mathbf{Y}}_i\rfloor$ where $\lceil\cdot\rfloor$ is the component-wise rounding operation to the nearest integer.
- Components of $\hat{\mathbf{X}}_{i,\text{sub}}$ are components of $\hat{\mathbf{X}}_{i,\text{tmp}}$ mapped to the nearest element in $\mathcal{C}$.

Another heuristic to get a suboptimal solution is to do the component-wise rounding recursively, *i.e.*, round only one of the components of $\bar{G}_i^{-1}\tilde{\mathbf{Y}}_i$ (e.g., the one closest to an integer) at a time, then fix that component in the least-squares problem and repeat. Yet another suboptimal polynomial-time algorithm is due to Babai [15, 16]. In this method, $\hat{\mathbf{X}}_{i,\text{sub}}$ is found by recursively computing the closest point in sub-lattices of L to $\tilde{\mathbf{Y}}_i$. The provable worst-case bound we get is better than (13) with the price of some additional computation.

As reported in [19] and from our own experience, it should be noted that these suboptimal algorithms work *much* better in practice than the worst-case bounds suggest. In practice, optimality of $\hat{\mathbf{X}}_{i,\text{sub}}$ can be checked using condition (12). This is very easy since a (relatively sharp) lower bound d on $d_{\min}$ can be computed as the length of the shortest vector resulting from performing the Gram-Schmidt orthogonalization procedure on the columns of $\bar{G}_i$ (cf. [12, 13]).

Using the worst-case performance bounds of these suboptimal algorithms (for example the one in (13)) it is possible to find a lower bound on the probability that $\hat{\mathbf{X}}_{i,\text{sub}} = \mathbf{X}_i$ given $\hat{\mathbf{X}}_i = \mathbf{X}_i$, *i.e.*, the probability that the suboptimal estimate of the symbol is correct given the optimal estimate of the symbol is correct. Suppose that the known worst-case sub-optimality factor of the suboptimal algorithm is $\alpha_{M_\text{T}} > 1$ so

$$\|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_{i,\text{sub}}\| \le \alpha_{M_\text{T}}\|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_i\|.$$

We have

$$\begin{aligned}\|G_i(\hat{\mathbf{X}}_{i,\text{sub}} - \hat{\mathbf{X}}_i)\| &= \|(\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_i) - (\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_{i,\text{sub}})\| \\ &\le \|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_i\| + \|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_{i,\text{sub}}\| \\ &\le (1 + \alpha_{M_\text{T}})\|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_i\|.\end{aligned}$$

If $\hat{\mathbf{X}}_{i,\text{sub}} \ne \hat{\mathbf{X}}_i$ then $\|G_i(\hat{\mathbf{X}}_{i,\text{sub}} - \hat{\mathbf{X}}_i)\| \ge d_{\min}$ so that $\|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_i\| \ge d_{\min}/(1 + \alpha_{M_\text{T}})$. But $d_{\min}$ can be (lower) bounded by $P_\text{e} = \textbf{Prob}(\hat{\mathbf{X}}_i \ne \mathbf{X}_i)$ as $d_{\min} \ge 2Q^{-1}(P_\text{e}/2)$ where Q^{-1} is the inverse function of the Q function (the probability of the tail of the Gaussian PDF). Therefore if $\hat{\mathbf{X}}_{i,\text{sub}} \ne \hat{\mathbf{X}}_i$ we have $\|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_i\| \ge 2Q^{-1}(P_\text{e}/2)/(1 + \alpha_{M_\text{T}})$, or equivalently,

$$\|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_i\| \le \frac{2Q^{-1}(P_\text{e}/2)}{1 + \alpha_{M_\text{T}}} \Longrightarrow \hat{\mathbf{X}}_{i,\text{sub}} = \hat{\mathbf{X}}_i.$$

Now if $\hat{\mathbf{X}}_i = \mathbf{X}_i$ then $\|\tilde{\mathbf{Y}}_i - G_i\hat{\mathbf{X}}_i\|^2$ is χ^2 with M_T degrees of freedom and we finally get

$$\textbf{Prob}\left(\hat{\mathbf{X}}_{i,\text{sub}} = \mathbf{X}_i \mid \hat{\mathbf{X}}_i = \mathbf{X}_i\right) \ge F_{\chi^2}\left(\frac{4Q^{-1}(P_\text{e}/2)^2}{(1 + \alpha_{M_\text{T}})^2}; M_\text{T}\right) \qquad (14)$$

where $F_{\chi^2}(\cdot, M_\text{T})$ is the χ^2 CDF with M_T degrees of freedom. The interesting point about (14) is that

Authorized licensed use limited to: Stanford University. Downloaded on April 06,2010 at 14:21:14 EDT from IEEE Xplore. Restrictions apply.

as P_e gets smaller the bound on the probability gets larger, which means that the suboptimal algorithm is guaranteed to perform better. In communication systems, P_e is designed to be very small and therefore these suboptimal algorithms have a guarantee on their performance. Again, we must note that, in practice, the performance is much better than the worst-case bounds.

3.2 Global optimization algorithm for the least-squares problem

Once we have efficiently computed a suboptimal solution to the least-squares problem (for example using the method discussed in the previous subsection), we need to check whether any better solution exists or not. As noted in §3.1, it is easy to check the sufficient condition (12) for optimality. It turns out that, specially for "low" P_e, this condition is "most" of the times true and the optimality of the solution is guaranteed. However, if this condition is not true we cannot say anything about the optimality of the solution.

In this case, the problem of checking whether any better solution exists or not is equivalent to checking whether an ellipsoid contains any point with integer-valued coordinates. The global optimization algorithm basically consists of computing a "good" initial guess using the suboptimal algorithm of the previous subsection, and an exhaustive search for finding, if any, points with integral coordinates inside an ellipsoid. This exhaustive search can be performed relatively efficiently (cf. [12, 13] and references therein for details). In practice, our simulations show that for problem sizes of a few ten integer variables, the computation required for solving the least-squares problem is in the order of a matrix inversion of the same size. An implementation of the global optimization algorithm in Matlab can be obtained by contacting the authors.

4 Conclusions

In this paper we discussed in detail the MIMO channel model proposed by Raleigh and Cioffi [4]. It was shown that by using cyclic prefix in the transmitted block of data from each antenna, it is possible to effectively block diagonalize the channel matrix. However in order to optimize the input vector, the channel has to be decomposed and completely diagonalized. A low complexity method was introduced that effectively diagonalizes the MIMO channel. This enables the use of Discrete Multi-Tone (DMT) modulation over the MIMO channel that achieves information transmission rates close to Shannon capacity by using an optimized input vector.

However, DMT requires knowledge of the channel state information at the transmitter, which is not always possible in practice. In this case, the channel can be only made block diagonal and signal detection becomes very challenging. A practically efficient method was proposed to solve the signal detection problem which is basically a least-squares problem with integer variables.

Finally, we should note that we did not address the channel estimation problem that is crucial for any practical implementation of the DMT approach.

References

[1] A.R. Ruiz, J.M. Cioffi, and S. Kasturia. Discrete multiple tone modulation with coset coding for the spectrally spaced channel. *IEEE Transactions on Communications*, 40(6):1012–1029, June 1992.

[2] L.H. Brandenbur and A.D. Wyner. Capacity of Gaussian channel with memory: The multivariate case. *Bell Systems Technical Journal*, 53:745–78, May-June 1974.

[3] R.S. Cheng and S. Verdu. Guassian multiaccess channels with ISI: Capacity region and multiuser water-filling. *IEEE Transaction on Information Theory*, 39:773–85, May 1993.

[4] G.G. Raleigh and J.M. Cioffi. Spatio-temporal coding for wireless communications. *IEEE GLOBECOM 96*, 3:1809–14, 1996.

[5] B.C. Ng, J. Chen, and A. Paulraj. Space time processing for fast fading channels with co-channel interference. *IEEE 46th Vehicular Technology Conference*, 3:1491–5, 1996.

[6] S. Roy, J. Yang, and P.S. Kumar. Joint transmitter/receiver optimization for multiuser communication. In W. A. Gardner, editor, *Cyclostationarity in Communications and Signal Processing*. IEEE Press, 1994.

[7] J. Salz. Digital transmission over cross-coupled linear channels. *AT&T Technical Journal*, 64:1147–59, July-August 1985.

[8] J. Yang and S. Roy. On joint transmitter and receiver optimization for multiple input-multiple-output (MIMO) transmission systems. *IEEE Transactions on Communications*, 42:3221–31, December 1993.

[9] J. Yang and S. Roy. Joint transmitter and receiver optimization for multiple input-multiple-output (MIMO) systems with decision feedback. *IEEE Transactions on Information Theory*, 40:1334–47, September 1994.

[10] G.W. Wornell. Signal processing techniques for efficient use of transmit diversity in wireless communications. *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2:1057–60, 1996.

[11] G.J. Foschini. Layered space-time architecture for wireless communication in a fading environments using multi-element antennas. *Bell Labs Technical Journal*, 1(2):41–59, Autumn 1996.

[12] A. Hassibi and S. Boyd. Integer parameter estimation in linear models with GPS applications. In *Proc. IEEE Conf. on Decision and Control*, Kobe, Japan, 1996.

[13] A. Hassibi and S. Boyd. Integer parameter estimation in linear models with GPS applications. *Submitted to IEEE Trans. on Signal Proc.*

[14] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261:515–534, 1982.

[15] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization.* Springer-Verlag, 1988.

[16] L. Babai. On Lovász' lattice reduction and the nearest lattice point problem. *Combinatorica*, 6:1–13, 1986.

[17] C. P. Schnorr. A hierarchy of polynomial time basis reduction algorithms. In L. Lovász and E. Szemerédi, editors, *Theory of Algorithms*, pages 375–386, North-Holland, Amsterdam, 1985.

[18] R. Kannan. Improved algorithms for integer programming and related problems. In *23rd IEEE Symposium on Foundations of Computer Science*, 1983.

[19] A. M. Odlyzko and H. te Riele. Disproof of the Mertens' conjecture. *J. reine angew. Math.*, 357:138–160, 1985.

Authorized licensed use limited to: Stanford University. Downloaded on April 06,2010 at 14:21:14 EDT from IEEE Xplore. Restrictions apply.