Blind Equalization and Identification for Differential Space-time Modulated Communication Systems

A Thesis

Presented in Partial Fulfillment of the Requirements for
the Degree Master of Science in the
Graduate School of The Ohio State University

By

Wei Hu, B.S.

* * * * *

The Ohio State University

2002

Master's Examination Committee:

Prof. Philip Schniter, Adviser

Prof. Hesham El-Gamal

Approved by

____

Adviser
Department of Electrical Engineering

ABSTRACT

The capacity of wireless communication systems over fading channels is enhanced by the use of multiple antennas at the transmitter and receiver. Differential space-time coding technique which does not require channel estimation is proposed for multiple input and multiple output (MIMO) system to achieve higher capacity. We consider the problem of blind identification and equalization for MIMO system with frequency-selective fading channels. We apply the differential unitary space-time (DUST) codes designed for flat fading channel to the frequency-selective channel and use the blind sub-space algorithm to reduce the frequency-selective fading channel to an unknown flat fading channel. We then apply the non-coherent decoder for the DUST codes and get an initial estimation of the transmitted symbols and channel responses. We also present two methods to derive better estimation of the channels and symbols with the aid of the initial estimation. One is the soft iterative least square projection algorithm and the other is the iterative per-survivor processing algorithm. Both are generalized to MIMO systems. The iterative per-survivor processing combined with the blind sub-space algorithm gives us a good estimation of our MIMO system when the channel memory is short. Constrained CR bound with parameters is derived and compared with the results of the algorithm proposed to evaluate its performance.

Blind Equalization and Identification for Differential Space-time Modulated Communication Systems

By

Wei Hu, M.S.

The Ohio State University, 2002

Prof. Philip Schniter, Adviser

The capacity of wireless communication systems over fading channels is enhanced by the use of multiple antennas at the transmitter and receiver. Differential space-time coding technique which does not require channel estimation is proposed for multiple input and multiple output (MIMO) system to achieve higher capacity. We consider the problem of blind identification and equalization for MIMO system with frequency-selective fading channels. We apply the differential unitary space-time (DUST) codes designed for flat fading channel to the frequency-selective channel and use the blind sub-space algorithm to reduce the frequency-selective fading channel to an unknown flat fading channel. We then apply the non-coherent decoder for the DUST codes and get an initial estimation of the transmitted symbols and channel responses. We also present two methods to derive better estimation of the channels and symbols with the aid of the initial estimation. One is the soft iterative least square projection algorithm and the other is the iterative per-survivor processing algorithm. Both are

generalized to MIMO systems. The iterative per-survivor processing combined with the blind sub-space algorithm gives us a good estimation of our MIMO system when the channel memory is short. Constrained CR bound with parameters is derived and compared with the results of the algorithm proposed to evaluate its performance.

ACKNOWLEDGMENTS

I would like to thank my supervisor Prof. Philip Schniter for his great help and many suggestions during this research. I am also thankful to Prof. Hesham El-Gamal for his early instructions of the advanced communication theory.

Thanks to Ashwin Iyer, Vidya Bhallamudi and Rudra Bandhu for sharing with me their knowledge of space time modulation. Thanks to Wei Lai for sharing with me her knowledge of algebraic methods for deterministic blind beamforming. Also thanks to my friends, Yu Luo and Sudha Dhoorjaty for the help in LaTex and the constant encouragement to me.

I am also very grateful to my family for their support and their love.

Wei Hu

July 24th, 2002

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION AND MIMO LINEAR SYSTEM MODEL

1.1 Introduction

The rapid growth in information technology demands higher data rate service and more reliable data transmission in modern communication systems. But due to multi-path propagation, the signal sent from a transmit antenna is usually reflected by various objects in its path. So the received signal is the sum of all these reflections in addition to the background noise and some other user interference. This fading phenomena can generate time varying attenuations and delays, which may cause great difficulties to recover the transmitted information signals.

In order to mitigate fading attenuation effect, different diversity techniques are proposed. Diversity means providing the receiver with more than one copy of the transmitted signals. There are several ways to do so. Transmitting the same information signals at different time is called time diversity. Transmitting the same signals over different frequency bands is called frequency diversity. However, they both have their disadvantages. Time diversity is inapplicable in slow-varying channel case since the delay required to achieve the diversity becomes large. Frequency diversity requires more bandwidth which may not be available. Foschini and Gans [16] show that systems using multiple input and multiple output antennas (MIMO) can increase data rate without loss of bandwidth efficiency. To fully exploit the spatial and temporal diversities in MIMO communication systems, lots of work on space-time coding has been done. Space-time trellis coding and space-time block coding are proposed for coherent detection, in which the channel responses are known to the receivers for detection. Differential space-time coding is proposed for non-coherent detection, in which the detection does not require channel responses to be known to the receivers.

According to the different fading types of the channel responses, the communication system can be divided as narrow-band systems and wide-band systems. Flat fading channel in narrow-band systems means that the the maximum delay spread of channel is smaller than the transmission interval, so the symbols transmit at different times do not interfere with each other. While frequency-selective fading channels in wide-band communication systems means the maximum delay spread of the channel is larger than the transmission interval, so the symbols transmitted at different times may interfere with each other and this is called inter-symbol interference (ISI). Knowledge of the channel coefficients is usually required to mitigate ISI. Sending pilot symbols may be one way of obtaining the channel coefficients. But this kind of training can be difficulty or costly, especially in fast fading environments. Estimation of the channel parameters or transmitted symbols using only the channel output is called blind identification or blind equalization. Our project is on the analysis of blind identification and equalization for wide-band wireless communication systems applying the differential unitary space-time (DUST) codes.

The wide-band differential space-time coded communication system we are studying is a MIMO linear system with frequency-selective channel fading. The input signals are specially structured in the spatial and temporal dimensions to increase the diversity and the band-width efficiency. The structure of the transmitted space-time codes are known to the receiver as a prior knowledge to blindly estimate the channel response and the transmitted signals. The idea of our scheme is that the DUST codes proposed by Hochwold [4] are used as the transmit symbols. Then the blind sub-space algorithm [5], which exploits the over sampled system output, is implemented to give the initial estimation of the the symbols subject to an unknown ambiguity matrix multiplication. Since the DUST codes are designed to tolerate this ambiguity, we can use non-coherent decoding to estimate the transmitted information. After we get the estimation of the transmit information and also the channel responses, we consider use of an iterative least square projection (ILSP) algorithm [9] to obtain improved estimates of the channel and transmit symbols. Since the performance of this projection algorithm is not as good as hoped, we also consider an iterative per-survivor processing (PSP) algorithm [11] which gives improved results. To evaluate the performance of the iterative PSP algorithm, we also derived the constrained CR bound of channel estimation error and compared with the estimation error resulted from our algorithm. The simulation results show that the iterative PSP algorithm is a good approach to solve the problem.

This thesis is organized as follows. In the next section in this chapter we give the system model. In Chapter 2, we introduce the blind sub-space algorithm generalized for MIMO system. In Chapter 3, we present the differential space-time coding technique and the non-coherent decoder. In Chapter 4, we describe the iterative least square projection and derive the soft ILSP algorithm. In Chapter 5, we derive the iterative PSP algorithm which is the final solution for our problem. In Chapter 6, we present the constrained CR bound and some simulation results to illustrate the performance of our algorithms.

1.2 MIMO Linear System Model

Consider a system with N_t transmit antennas and N_r receive antennas. The input N_t digital signals at time $t = nT$ are $s_1[n], s_2[n], \cdots, s_{N_t}[n]$. The symbol period is T. So the input signals at the nth symbol period are:

$$\mathbf{s}[n] = \begin{bmatrix} s_1[n] \\ s_2[n] \\ \vdots \\ s_{N_t}[n] \end{bmatrix} \in \mathbb{C}^{N_t \times 1}.$$

The output signals at time t are $x_1(t), x_2(t), \cdots, x_{N_r}(t)$. The signal received consists of multiple paths, with echoes arriving from different angles with different delays and attenuations. The impulse response of the channel from the jth transmit antenna to the ith receive antenna at delay t is denoted $h_{ij}(t)$. Assuming the delay spread of channel impulse is $N_h T$,

$$h_{ij}(t) = 0,\ t \notin [0, N_h T),\ \ i = 1, \cdots N_r;\ j = 1, \cdots N_t.$$

So at the nth transmit symbol period, only N_h consecutive symbols of transmit signals play a role in the received signal. Suppose

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_{N_r}(t) \end{bmatrix} \mathbf{H}(t) = \begin{bmatrix} h_{11}(t) & \cdots & h_{1N_t}(t) \\ \vdots & \ddots & \vdots \\ h_{N_r 1}(t) & \cdots & h_{N_r N_t}(t) \end{bmatrix} \mathbf{w}(t) = \begin{bmatrix} w_1(t) \\ \vdots \\ w_{N_r}(t) \end{bmatrix}.$$

$w_i(t)$ is the channel additive complex Gaussian noise to the ith receive antenna at time t. But we usually over-sample the received signal to improve the performance.

Variable	Description
T	symbol (baud) interval
T_c	coherence time for flat fading channel
N_t, N_r	number of transmit antennas, receive antennas
N	number of symbol intervals per frame interval
N_c	number of block codewords per frame interval
N_s	number of symbol intervals per block codeword
N_h	channel impulse response duration (in symbol intervals)
N_o	over-sampling rate of the received signal
N_m	maximum number of iterations in the iterative PSP algorithm
$h_{i,j}[l]$	channel gain from jth transmit antenna to ith receive antenna at lag $t = lT$
$\mathbf{H}[l]$	channel impulse response $N_rN_o \times N_t$ matrix during lag $t = lT$
$\mathbf{H}$	channel impulse response MIMO system model
$\tilde{\mathbf{H}}$	normalized channel impulse response
$\hat{\mathbf{H}}^{(k)}$	channel estimation in the kth iteration in iterative PSP and soft ILSP
$\mathcal{H}$	block-Toeplitz matrix of the channel response
$s_j[n], \mathbf{s}[n]$	transmitted symbols, $N_t \times 1$ vector across transmit antennas
$\mathbf{S}[n]$	transmitted $N_t \times N_s$ block codes
$\mathbf{S}$	all transmitted vectors $[\mathbf{s}[-N_h+1], \cdots, \mathbf{s}[N-1]]$
$\mathcal{S}, \mathcal{S}_{N_h}$	block-Toeplitz matrix of transmitted symbol
$\mathcal{V}$	group of DUST block codes transmitted in our system
$\mathbf{S}_\ell$	block code from group codes $\mathcal{V}$
$\mathcal{U}$	sets of all possible choices of $\mathbf{S}$
L	size of group codes $\mathcal{V}$
$\hat{\mathbf{S}}^{(k)}$	code sequence estimation in the kth iteration in iterative PSP and soft ILSP
$\hat{\mathcal{S}}^{(k)}$	estimation of block-Toeplitz matrix constructed using $\hat{\mathbf{S}}^{(k)}$
$s^{(k)}[n]$	transmitted signal from the kth transmit antenna at time $t = nT$
$w_i[n], \mathbf{w}[n], \mathbf{W}[n]$	noise sample, $N_rN_o \times 1$ vector across receive antennas, $N_rN_o \times N_s$ block
$\mathbf{W}$	all noise vectors $[\mathbf{w}[0], \cdots, \mathbf{w}[N-1]]$
$\mathcal{W}$	block-Toeplitz noise matrix
$x_i[n], \mathbf{x}[n], \mathbf{X}[n]$	received sample, $N_rN_o \times 1$ vector across receive antennas, $N_rN_o \times N_s$ block
$\mathbf{X}$	all received signal vectors $[\mathbf{x}[0], \cdots, \mathbf{x}[N-1]]$
$\mathcal{X}$	block-Toeplitz observation matrix

Table 1.1: Parameters and descriptions for the system model

Suppose we sample the channel impulse response, the received signal and the additive noise at intervals of $\frac{T}{N_0}$, where $N_0 \in \mathbb{N}$ is called the over-sampling rate. This means:

$$\begin{aligned} h_{ij}[m] &= h_{ij}(m\frac{T}{N_0}) \\ x_i[m] &= x_i(m\frac{T}{N_0}) \\ w_i[m] &= w_i(m\frac{T}{N_0}). \end{aligned}$$

So at the nth transmit signal period, we collect the receive samples:

$$\mathbf{x}[n] = \begin{bmatrix} \mathbf{x}(nN_0\frac{T}{N_0}) \\ \mathbf{x}((nN_0+1)\frac{T}{N_0}) \\ \vdots \\ \mathbf{x}((nN_0+N_0-1)\frac{T}{N_0}) \end{bmatrix} = \begin{bmatrix} x_1[nN_0] \\ \vdots \\ x_{N_r}[nN_0] \\ \vdots \\ \vdots \\ x_1[nN_0+N_0-1] \\ \vdots \\ x_{N_r}[nN_0+N_0-1] \end{bmatrix} \in \mathbb{C}^{N_oN_r\times 1}.$$

Note that $\mathbf{x}[n]$ contains the N_oN_r spatial and temporal samples during the nth transmit symbol interval. The over-sampled channel impulse response at delay lT is:

$$\begin{aligned} \mathbf{H}[l] &= \begin{bmatrix} \mathbf{H}(lN_0\frac{T}{N_0}) \\ \mathbf{H}((lN_0+1)\frac{T}{N_0}) \\ \vdots \\ \mathbf{H}((lN_0+N_0-1)\frac{T}{N_0}) \end{bmatrix} \\ &= \begin{bmatrix} h_{11}[lN_0] & \cdots & h_{1N_t}[lN_0] \\ \vdots & \ddots & \vdots \\ h_{N_r1}[lN_0] & \cdots & h_{N_rN_t}[lN_0] \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ h_{11}[lN_0+N_0+1] & \cdots & h_{1N_t}[lN_0+N_0-1] \\ \vdots & \ddots & \vdots \\ h_{N_r1}[lN_0+N_0-1] & \cdots & h_{N_rN_t}[lN_0+N_0-1] \end{bmatrix} \in \mathbb{C}^{N_oN_r\times N_t}. \end{aligned}$$

Similarly we can define the over-sampled additive noise at the nth transmit symbol period as:

$$\mathbf{w}[n] = \begin{bmatrix} \mathbf{w}(nN_0\frac{T}{N_0}) \\ \mathbf{w}((nN_0+1)\frac{T}{N_0}) \\ \vdots \\ \mathbf{w}((nN_0+N_0-1)\frac{T}{N_0}) \end{bmatrix} = \begin{bmatrix} w_1[nN_0] \\ \vdots \\ w_{N_r}[nN_0] \\ \vdots \\ \vdots \\ w_1[nN_0+N_0-1] \\ \vdots \\ w_{N_r}[nN_0+N_0-1] \end{bmatrix} \in \mathbb{C}^{N_oN_r\times 1}.$$

So the system model can be described by the following equation:

$$\mathbf{x}[n] = \sum_{l=0}^{N_h-1} \mathbf{H}[l]\mathbf{s}[n-l] + \mathbf{w}[n]. \tag{1.1}$$

In a frame, we collect samples during N symbol periods. Note, in this thesis "frame" means a whole observation interval for our estimation while "block" means the DUST block codeword. A frame usually contains a certain number of block codes. The received signals for a frame can be written as:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}[0] & \cdots & \mathbf{x}[N-1] \end{bmatrix} \in \mathbb{C}^{N_oN_r\times N}.$$

Since the length of the channel response is N_h, we define the over-sampled channel response matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}[0] & \cdots & \mathbf{H}[N_h-1] \end{bmatrix} \in \mathbb{C}^{N_oN_r\times N_tN_h}.$$

The over-sampled additive noise matrix in a frame of N symbol periods is:

$$\mathbf{W} = \begin{bmatrix} \mathbf{w}[0] & \cdots & \mathbf{w}[N-1] \end{bmatrix}.$$

Given the input signal $\mathbf{s}[n] \in \mathbb{C}^{N_t\times 1}$, we define a block-Toeplitz transmit signal matrix for a frame with N symbol periods as

$$\mathcal{S}_{N_h} = \begin{bmatrix} \mathbf{s}[0] & \mathbf{s}[1] & \ddots & \mathbf{s}[N-1] \\ \ddots & \ddots & \ddots & \mathbf{s}[N-2] \\ \mathbf{s}[-N_h+2] & \mathbf{s}[-N_h+3] & \ddots & \ddots \\ \mathbf{s}[-N_h+1] & \mathbf{s}[-N_h+2] & \ddots & \mathbf{s}[N-N_h] \end{bmatrix} \in \mathbb{C}^{N_t N_h \times N}.$$

The subscript index N_h in $\mathcal{S}_{N_h}$ represents how many input $N_t \times 1$ signal vectors are stacked.

Based on the MIMO linear system model (1.1), we get

$$\mathbf{X} = \mathbf{H}\mathcal{S}_{N_h} + \mathbf{W}. \tag{1.2}$$

The above equation is our frequency selective MIMO linear system model. In blind identification, we estimate the channel coefficients $\mathbf{H}$ observing only $\mathbf{X}$. In blind equalization, we estimate the block vector symbols $\mathbf{S} = [\mathbf{s}[-N_h+1], \cdots, \mathbf{s}[N-1]]$ observing only $\mathbf{X}$. Given $\mathbf{X}$, the blind subspace method in the next section will try to find $\mathcal{S}_{N_h}$ such that $\mathcal{S}_{N_h}$ is a block-Toeplitz matrix and the transmitted symbols in $\mathcal{S}_{N_h}$ satisfy the differential unitary space-time code properties which we will discuss later. Table (1.1) lists most of the important notations used in this thesis.

CHAPTER 2

DETERMINISTIC SUBSPACE METHOD

The deterministic subspace method developed by Liu and Xu [5] and van der Veen et al. [2] forms the first part of our algorithm.

We typically desire a blind equalization method that performs perfectly in the absence of noise. So we first consider the noiseless case of system model (1.2):

$$\mathbf{X} = \mathbf{H}\mathcal{S}_{N_h}. \tag{2.1}$$

Thus the goal is to recover $\mathcal{S}_{N_h}$ knowing $\mathbf{X}$ but not $\mathbf{H}$. Clearly, this requires $\mathbf{H}$ to be left invertible, which means there must exist a "filtering matrix" $\mathbf{F}$ such that $\mathbf{FX} = \mathcal{S}_{N_h}$. This is equivalent to having an $\mathbf{H} \in \mathbb{C}^{N_o N_r \times N_t N_h}$ that is of full column rank, which requires $N_o N_r \geq N_t N_h$. But this may put undue requirements on the number of antennas or over-sampling rate. We can ease this condition by making use of the structure of $\mathcal{S}_{N_h}$ and rearranging the structure of (2.1).

We first extend $\mathbf{X}$ to a block-Toeplitz matrix by left shifting and stacking $k \in \mathbb{N}$ times. The parameter k can be viewed as an equalizer length (in symbol periods). So we get:

$$\mathcal{X}_k = \begin{bmatrix} \mathbf{x}[k-1] & \mathbf{x}[k] & \ddots & \mathbf{x}[N-1] \\ \mathbf{x}[k-2] & \mathbf{x}[k-1] & \ddots & \mathbf{x}[N-2] \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{x}[0] & \mathbf{x}[1] & \cdots & \mathbf{x}[N-k] \end{bmatrix} \in \mathbb{C}^{kN_r N_o \times (N-k+1)}.$$

Extending the data matrix leads to the following system model:

$$\begin{aligned}\mathcal{X}_k &= \mathcal{H}_k \mathcal{S}_{N_h+k-1} \qquad (2.2)\\ &= \underbrace{\begin{bmatrix} \mathbf{H}[0] & \cdots & \mathbf{H}[N_h-1] & & \mathbf{0} \\ & \ddots & & \ddots & \\ \mathbf{0} & & \mathbf{H}[0] & \cdots & \mathbf{H}[N_h-1] \end{bmatrix}}_{\mathcal{H}_k} \underbrace{\begin{bmatrix} \mathbf{s}[k-1] & \cdots & \mathbf{s}[N-1] \\ \vdots & \ddots & \vdots \\ \mathbf{s}[-N_h+1] & \cdots & \mathbf{s}[N-k-N_h+1] \end{bmatrix}}_{\mathcal{S}_{N_h+k-1}},\end{aligned}$$

where $\mathcal{H}_k \in \mathbb{C}^{kN_rN_o \times N_t(N_h+k-1)}$ and $\mathcal{S}_{N_h+k-1} \in \mathbb{C}^{N_t(N_h+k-1)\times(N-k+1)}$ are both block-Toeplitz. Note that, for any $k \in \mathbb{N}$, the system model (2.2) has the same block-Toeplitz form. As k increases, the matrices in (2.2) get taller. For simplicity, we adopt the notation $\mathcal{X} = \mathcal{X}_k$, $\mathcal{H} = \mathcal{H}_k$, $\mathcal{S} = \mathcal{S}_{N_h+k-1}$. Given $\mathcal{X}$, we would like to determine $\mathcal{H}$ and $\mathcal{S}$ with the block-Toeplitz structures.

A necessary condition for $\mathcal{X}$ to have a unique factorization $\mathcal{X} = \mathcal{H}\mathcal{S}$ is that $\mathcal{H}$ is a "tall" matrix and $\mathcal{S}$ is a "wide" matrix. Note also that a tall $\mathcal{H}$ requires tall $\mathbf{H}[l]$. Thus the following conditions are necessary for unique factorization,

$$\begin{aligned} \text{Tall } \mathbf{H}[l] \in \mathbb{C}^{N_oN_r \times N_t} &\Rightarrow N_oN_r > N_t \\ \text{Tall } \mathcal{H} \in \mathbb{C}^{kN_rN_o \times N_t(N_h+k-1)} &\Rightarrow k \geq \frac{N_t(N_h-1)}{N_oN_r - N_t} \qquad (2.3) \\ \text{Wide } \mathcal{S} \in \mathbb{C}^{N_t(N_h+k-1)\times(N-k+1)} &\Rightarrow N \geq N_tN_h + (N_t+1)(k-1). \end{aligned}$$

In the above conditions, "tall" $\mathcal{H}$ requires that k should be sufficiently large and "wide" $\mathcal{S}$ requires that N is sufficiently large. Assuming k and N can be made sufficiently large, then the first condition $N_oN_r > N_t$ is a fundamental identification restriction. Our two assumptions for the subspace algorithms to work are:

1. $\mathcal{H}_k$ has full column rank for some chosen value of k;

2. $\mathcal{S}_{N_h+k-1}$ has full row rank for k specified above and some chosen value of N.

Given the model $\mathcal{X} = \mathcal{H}\mathcal{S}$ and the above two assumptions, we have the following property:

$$\mathcal{H} \text{ full column rank} \Rightarrow \ \text{row}(\mathcal{X}) = \ \text{row}(\mathcal{S}). \tag{2.4}$$

This indicates that without knowing the input sequences, the row span of the input matrix $\mathcal{S}$ can be obtained from the row span of the observed matrix $\mathcal{X}$.

To factor $\mathcal{X}$ into $\mathcal{X} = \mathcal{H}\mathcal{S}$, we must find $\mathcal{S}$ such that:

1. Row span of $\mathcal{S}$ is equivalent to row span of $\mathcal{X}$;

2. $\mathcal{S}$ has a block-Toeplitz structure.

Accordingly, the deterministic blind subspace method is described by the following two steps, each making use of one property above.

Step 1: Obtain the row span of $\mathcal{S}$ Suppose as stated above, there is no noise and $\mathcal{H}$ has full column rank. Based on property (2.4), the row span of $\mathcal{S}$ can be obtained from $\mathcal{X}$. We can compute the SVD of $\mathcal{X}$, $\mathcal{X} = U\Sigma V$, where U, V are unitary matrices, and Σ is a diagonal matrix containing the singular values in non-increasing order. The rank of $\mathcal{X}$ is $r_{\mathcal{X}}$ which equals to the number of the non-zero singular values. Suppose $\hat{V}$ is the first $r_{\mathcal{X}}$ rows of V, so that the rows of $\hat{V}$ form an orthonormal basis for the row span of $\mathcal{X}$. For well-conditioned problems, since $\mathcal{S} \in \mathbb{C}^{N_t(N_h+k-1)\times(N-k+1)}$ is a "wide" matrix, we expect $r_{\mathcal{X}} = N_t(N_h + k - 1)$. And thus $\hat{V}$ is of dimension $N_t(N_h + k - 1) \times (N - k + 1)$. Let the column of G form an orthonormal basis for the orthogonal complement of row($\hat{V}$). Then G has the dimension $(N - k + 1) \times (N - k + 1 - N_t(N_h + k - 1))$. Since $\hat{V}G = 0$, $\mathcal{X}G = 0$ and so $\mathcal{S}G = 0$. If there is noise in the system, then the effective rank $\hat{r}_{\mathcal{X}}$ of $\mathcal{X}$ would be estimated by deciding how many singular values of $\mathcal{X}$ are above the noise level. The estimated row span $\hat{V}$ would then be given by the first $\hat{r}_{\mathcal{X}}$ rows of V.

Step 2. Forcing the Toeplitz structure of $\mathcal{S}$ The next step for computing the structured factorization is to find all possible matrices $\mathcal{S}$ which have a block-Toeplitz structure with $k+N_h-1$ block rows and which obey $\text{row}(\mathcal{S}) = \text{row}(\mathcal{X})$. This requires that each block row of $\mathcal{S}$ is in the row span of $\mathcal{X}$:

$$\begin{cases} \begin{bmatrix} \mathbf{s}[k-1] & \cdots & \mathbf{s}[N-1] \end{bmatrix} & \in \text{row}(\mathcal{X}) \\ & \vdots \\ \begin{bmatrix} \mathbf{s}[-N_h+1] & \cdots & \mathbf{s}[N-k-N_h+1] \end{bmatrix} & \in \text{row}(\mathcal{X}) \end{cases}$$

Given that columns of G form an orthonormal basis for the orthogonal complement of $\text{row}(\mathcal{X})$, we have $\mathcal{X}G = 0$ and so $\mathcal{S}G = 0$,

$$\begin{cases} \begin{bmatrix} \mathbf{s}[k-1] & \cdots & \mathbf{s}[N-1] \end{bmatrix} G = 0 \\ \qquad\vdots \\ \begin{bmatrix} \mathbf{s}[-N_h+1] & \cdots & \mathbf{s}[N-k-N_h+1] \end{bmatrix} G = 0. \end{cases}$$

If we define the generator of the Toeplitz matrix $\mathcal{S}_{N_h+k-1}$ as the block vector:

$$\mathbf{S} = [\mathbf{s}[-N_h+1], \cdots, \mathbf{s}[N-1]] \in \mathbb{C}^{N_t \times (N+N_h-1)},$$

then,

$$\begin{cases} \begin{bmatrix} \mathbf{s}[k-1] & \cdots & \mathbf{s}[N-1] \end{bmatrix} G = 0 & \Rightarrow \mathbf{S} \underbrace{\begin{bmatrix} \mathbf{0}_{(N_h+k-2)\times(N-k+1-N_t(N_h+k-1))} \\ G \end{bmatrix}}_{G_1} = 0 \\ \begin{bmatrix} \mathbf{s}[k-2] & \cdots & \mathbf{s}[N-2] \end{bmatrix} G = 0 & \Rightarrow \mathbf{S} \underbrace{\begin{bmatrix} \mathbf{0}_{(N_h+k-3)\times(N-k+1-N_t(N_h+k-1))} \\ G \\ \mathbf{0}_{1\times(N-k+1-N_t(N_h+k-1))} \end{bmatrix}}_{G_2} = 0 \\ & \vdots \\ \begin{bmatrix} \mathbf{s}[-N_h+1] & \cdots & \mathbf{s}[N-k-N_h+1] \end{bmatrix} G = 0 & \Rightarrow \mathbf{S} \underbrace{\begin{bmatrix} G \\ \mathbf{0}_{(N_h+k-2)\times(N-k+1-N_t(N_h+k-1))} \end{bmatrix}}_{G_{N_h+k-1}} = 0. \end{cases}$$

To meet the above $k + N_h - 1$ conditions, the generator block vector $\mathbf{S}$ must be orthogonal to the union of the column spans of $G_1, G_2, \cdots, G_{N_h+k-1}$. Defining $\mathbf{G}$ as

$$\mathbf{G} = \left[\begin{array}{ccc} G_1 & \cdots & G_{N_h+k-1} \end{array} \right],$$

the above condition becomes:

$$\mathbf{SG} = \mathbf{0} \tag{2.5}$$

If Y is a matrix whose rows form a basis for the orthogonal complement of col($\mathbf{G}$), then

$$Y = A\mathbf{S}, \tag{2.6}$$

where A is an arbitrary $N_t \times N_t$ invertible "ambiguity matrix". In other words, the solution of (2.5) is not unique, and so $\mathbf{S}$ can only be determined up to a matrix ambiguity. Later we make use of DUST codes to tolerate this ambiguity. This is the result for the noiseless model. If noise is added, the output Y contains also noise from the sub-space method, the output can be written as:

$$Y = A\mathbf{S} + Z.$$

CHAPTER 3

DIFFERENTIAL SPACE-TIME MODULATION

3.1 Space-time coding for Rayleigh flat fading channel

Recently multi-antenna wireless communication has been a research focus because it can support high data rate with low error probability. Space-time coding has been proposed for multi-antenna systems, especially with channels that are characterized as Rayleigh flat fading. The difference between the frequency-selective channel we discussed earlier and the flat fading channel here is that the flat fading channel is memoryless while the frequency selective channel has delay spread $N_h > 1$ symbol intervals. So in flat fading channel, for the received signal at the nth symbol interval, only the symbols transmitted at the same time can influence it. Assume that N_sT is small compared with the channel coherence time T_c, so that the channel coefficients can be considered constant over N_s symbols. Then we use the abbreviation $\tilde{h}_{ij}$ to denote the normalized channel gain from the jth transmit antenna to the ith receive antenna during the current block. For Rayleigh flat fading channel, the normalized path gains $\tilde{h}_{ij}$ are unit variance independent and identically distributed complex Gaussian random variables,

$$p(\tilde{h}_{ij}) = (1/\pi)e^{(-|\tilde{h}_{ij}|^2)} \text{ for } \tilde{h}_{ij} \in \mathbb{C}.$$

Consider the nth block of symbols, i.e. symbols transmitted from nN_sT to $(n+1)N_sT-T$:

$$\mathbf{S}[n] = \begin{bmatrix} s_1[nN_s] & s_1[nN_s+1] & \cdots & s_1[nN_s+N_s-1] \\ s_2[nN_s] & s_2[nN_s+1] & \cdots & s_2[nN_s+N_s-1] \\ \vdots & \vdots & \ddots & \vdots \\ s_{N_t}[nN_s] & s_{N_t}[nN_s+1] & \cdots & s_{N_t}[nN_s+N_s-1] \end{bmatrix} \in \mathbb{C}^{N_t\times N_s}.$$

Consider the channel matrix for the same block:

$$\tilde{\mathbf{H}} = \begin{bmatrix} \tilde{h}_{11} & \tilde{h}_{12} & \cdots & \tilde{h}_{1N_t} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{h}_{N_r1} & \tilde{h}_{N_r2} & \cdots & \tilde{h}_{N_rN_t} \end{bmatrix} \in \mathbb{C}^{N_r\times N_t}.$$

The nth block of received signals is:

$$\mathbf{X}[n] = \begin{bmatrix} x_1[nN_s] & x_1[nN_s+1] & \cdots & x_1[nN_s+N_s-1] \\ x_2[nN_s] & x_2[nN_s+1] & \cdots & x_2[nN_s+N_s-1] \\ \vdots & \vdots & \ddots & \vdots \\ x_{N_r}[nN_s] & x_{N_r}[nN_s+1] & \cdots & x_{N_r}[nN_s+N_s-1] \end{bmatrix} \in \mathbb{C}^{N_r\times N_s}.$$

The nth block of noise matrix is:

$$\mathbf{W}[n] = \begin{bmatrix} w_1[nN_s] & w_1[nN_s+1] & \cdots & w_1[nN_s+N_s-1] \\ w_2[nN_s] & w_2[nN_s+1] & \cdots & w_2[nN_s+N_s-1] \\ \vdots & \vdots & \ddots & \vdots \\ w_{N_r}[nN_s] & w_{N_r}[nN_s+1] & \cdots & w_{N_r}[nN_s+N_s-1] \end{bmatrix} \in \mathbb{C}^{N_r\times N_s}.$$

Assume that the elements in the code matrix are normalized such that the average power per transmitted antenna equals one: $\frac{1}{N_t}\sum_{j=0}^{N_t-1} E|s_j[n]|^2 = 1$. Then the signal model for Rayleigh flat fading channel is:

$$\mathbf{X}[n] = \sqrt{\frac{\rho}{N_t}}\tilde{\mathbf{H}}\mathbf{S}[n] + \mathbf{W}[n]. \tag{3.1}$$

For simplicity, we assume that $\mathbf{W}[n]$ contain zero mean unit variance i.i.d. complex Gaussian noise, so that ρ is the SNR at each receive antenna.

For space-time coding, the transmitter passes the information bit stream into words of N_b bits and maps each word to a $N_t \times N_s$ matrix $\mathbf{S}_\ell$, where $\ell \in \{0, \cdots, L-1\}$ $(L = 2^{N_b})$. The result is a sequence of code matrices $\mathbf{S}[n] \in \{\mathbf{S}_0, \mathbf{S}_1, \cdots, \mathbf{S}_{L-1}\}$.

3.2 Decoding with perfect CSI at the receiver

Most work on space-time coding has assumed perfect channel state information (CSI) is available, i.e. the block channel matrix $\tilde{\mathbf{H}}$ is known at the receiver. The likelihood of $\mathbf{X}[n]$ conditioned on $\mathbf{S}[n]$ and $\tilde{\mathbf{H}}$ is:

$$p(\mathbf{X}[n]|\tilde{\mathbf{H}},\mathbf{S}[n]) = \frac{1}{\pi^{N_s N_r}} \exp(-\mathrm{tr}(\mathbf{X}[n] - \sqrt{\frac{\rho}{N_t}}\tilde{\mathbf{H}}\mathbf{S}[n])(\mathbf{X}[n] - \sqrt{\frac{\rho}{N_t}}\tilde{\mathbf{H}}\mathbf{S}[n])^H),$$

where $\mathrm{tr}(\cdot)$ means trace and $(\cdot)^H$ means complex conjugate transpose. So the ML detector becomes:

$$\hat{\ell} = \arg \min_{\ell \in \{0,1,\cdots L-1\}} \mathrm{tr}(\mathbf{X}[n] - \sqrt{\frac{\rho}{N_t}}\tilde{\mathbf{H}}\mathbf{S}_\ell)(\mathbf{X}[n] - \sqrt{\frac{\rho}{N_t}}\tilde{\mathbf{H}}\mathbf{S}_\ell)^H. \tag{3.2}$$

If we assume that each transmitted codeword is of equal probability, then the probability of incorrectly ML decoding $\mathbf{S}[n] = \mathbf{S}_\ell$ as $\mathbf{S}[n] = \mathbf{S}_{\ell'}$ in a code consisting of only these two matrices is defined as:

$$\begin{aligned} p\left\{\mathbf{S}\ell \to \mathbf{S}\ell'\right\} &:= p\left\{\mathbf{S}_{\ell'} \text{ detected}|\mathbf{S}_\ell\right\} \\ &= p\left\{p(\mathbf{X}[n]|\tilde{\mathbf{H}},\mathbf{S}_{\ell'}) \geq p(\mathbf{X}[n]|\tilde{\mathbf{H}},\mathbf{S}_\ell)|\mathbf{S}[n] = \mathbf{S}_\ell\right\}. \end{aligned}$$

$p\{\mathbf{S}_\ell \to \mathbf{S}_{\ell'}\}$ is called the "pairwise error probability". Let us define the matrix difference outer product:

$$\Delta\mathbf{S}[n] = (\mathbf{S}[n] - \mathbf{S}_{\ell'})(\mathbf{S}[n] - \mathbf{S}_{\ell'})^H \in \mathbb{C}^{N_t \times N_s}.$$

An upper bound of the pairwise error probability was derived in [6] that depends on $\Delta\mathbf{S}[n]$:

$$\begin{aligned} p\{\mathbf{S}_\ell \to \mathbf{S}_{\ell'}\} &\leq (\prod_{j=1}^{N_t}(1 + \frac{\lambda_j(\ell,\ell')\rho}{4}))^{-N_r} \\ &\leq (\prod_{j=1}^{r(\ell,\ell')} \lambda_j(\ell,\ell'))^{-N_r}(\frac{\rho}{4})^{-r(\ell,\ell')N_r}. \end{aligned}$$

Here, $r(\ell, \ell')$ is the rank of $\Delta\mathbf{S}[n]$ and $\prod_{j=1}^{r(\ell,\ell')} \lambda_j(\ell, \ell')$ is the product of its non-zero eigenvalues. The second expression above approaches the first as ρ increases. The parameter $r(\ell, \ell')$ can be interpreted as the "diversity advantage" of the code pair of $\mathbf{S}_\ell$ and $\mathbf{S}_{\ell'}$, and equals the slope of the log BER vs. log SNR plot at high SNR. The maximum attainable diversity advantage is therefore N_t, since $\Delta\mathbf{S} \in \mathbb{C}^{N_t \times N_t}$ when $N_s \geq N_t$. The quantity $\Pi_{j=1}^{N_t} \lambda_j(\ell, \ell')$ is called the "coding advantage" or "product distance", and affects the left/right shift of the BER vs. SNR plot. Error probability is minimized by maximizing both the diversity advantage and the coding advantage over all possible symbol difference matrices. Suppose:

$$r = \min_{\ell \neq \ell'} r(\ell, \ell') \quad \ell, \ell' \in \{0, 1, \cdots, L-1\}.$$

So r is the minimum diversity advantage over all possible code pairs. Similarly define:

$$\Lambda = \min_{\ell \neq \ell'} \left(\prod_{j=1}^{r(\ell,\ell')} \lambda_j(\ell, \ell') \right) \quad \ell, \ell' \in \{0, 1, \cdots, L-1\},$$

and Λ is the minimum coding advantage over all possible code pairs. So for lower error probability, we want codes with maximum value of r and Λ. At high SNR, the performance is determined primarily by the minimum diversity r, which attains a maximum value of N_t when all the difference matrices of the space-time code pairs are of full rank.

3.3 Unitary space-time modulation without CSI at the receiver

The above ML detector and performance analysis is based on the case in which the channel state information is known to the receiver. In that case, training symbols must be sent to obtain the channel state information. However, the use of training symbols may result in a significant loss of throughput. So we need to derive systems that work well without the knowledge of channels. Such schemes are referred to as non-coherent schemes. Hochwald and Marzetta [7] have proved that the capacity of multiple-antenna communication systems can be approached for large ρ or for $T_c \gg N_t T$ using so-called "unitary space-time codes", which have the property that all code matrices $\mathbf{S}_\ell$ contain orthogonal rows and equal energy:

$$\mathbf{S}_\ell \mathbf{S}_\ell^H = N_t I, \text{ for all } \ell \in \{0, 1, \cdots, L-1\}.$$

For comparison with the previous known channel case, we give the probability of error and ML detector form for unknown channel case from [3]. With the model equation:

$$\mathbf{X}[n] = \sqrt{\frac{\rho}{N_t}} \tilde{\mathbf{H}} \mathbf{S}[n] + \mathbf{W}[n],$$

when $\mathbf{S}[n] = \mathbf{S}_\ell$ is transmitted and $\tilde{\mathbf{H}}$ is unknown, the received matrix $\mathbf{X}[n]$ is Gaussian with conditional pdf [7]:

$$p(\mathbf{X}[n]|\mathbf{S}_\ell) = \frac{\exp(-\mathrm{tr}(\mathbf{X}[n]\Sigma_\ell^{-1}\mathbf{X}^H[n]))}{|\pi\Sigma_\ell|^r},$$

where $\Sigma_\ell = I + \sqrt{\frac{\rho}{N_t}} \mathbf{S}_\ell^H \mathbf{S}_\ell$. Note that due to the unitary code matrix property, $|\Sigma_\ell|$ does not depend on ℓ. Furthermore,

$$\Sigma_\ell^{-1} = I - \frac{\sqrt{\frac{\rho}{N_t}}}{N_s \sqrt{\frac{\rho}{N_t}} + 1} \mathbf{S}_\ell^H \mathbf{S}_\ell.$$

So the ML detector for a unitary code has the form:

$$\begin{aligned} \hat{\ell} &= \arg \max_{\ell \in \{0,1,\cdots,L-1\}} p(\mathbf{X}[n]|\mathbf{S}_\ell) \\ &= \arg \max_{\ell \in \{0,1,\cdots,L-1\}} \mathrm{tr}(\mathbf{X}[n]\mathbf{S}_\ell^H \mathbf{S}_\ell \mathbf{X}^H[n]). \end{aligned} \quad (3.3)$$

3.4 Differential unitary space-time modulation

Based on the unitary space-time modulation, the differential unitary space-time modulation (DUST) is proposed by Hughes [3] and Hochwald [4] separately for non-coherent detection. Consideration of continuous (rather than block) channel variation motivated differential schemes in which the channel is assumed to be constant only over the short duration of $T_c = 2N_tT$. DUST can be considered an extension of the differential phase-shift keying (DPSK) to multiple antennas.

We first review DPSK. Here we send symbol sequence $s[n]$ where $s[n] = s[n-1]\phi[n]$. Note $s[n]$ is the transmitted symbol while $\phi[n]$ is the information symbol and is in the constellation of PSK. For example, if the rate is R bits/channel use, we need $L = 2^R$ constellation size, giving $\phi[n]$ the L-PSK constellation $\{\phi_0, \phi_1, \cdots, \phi_{L-1}\}$. The channel coefficient h is assumed to be the same for each pair of two consecutive symbols, allowing the receiver to detect the information symbol via comparing the phase difference between successive received symbols. This yields on ML receiver which has a very simple form:

$$\hat{\ell}[n] = \arg \min_{\ell \in \{0,1,\cdots,L-1\}} |\phi_\ell - s[n]s^*[n-1]|.$$

In DUST modulation, it is assumed that the channels are constant over each pair of consecutive block symbols $\mathbf{S}[n], \mathbf{S}[n-1]$. This scheme uses data at the current and previous block for encoding and decoding. The block symbol matrices satisfy the following rule:

$$\mathbf{S}[n] = \mathbf{S}[n-1]V_{\ell[n]}, \quad \mathbf{S}[n] \in \mathbb{C}^{N_t \times N_t},$$

where $V_{\ell[n]} \in \mathbb{C}^{N_t \times N_t}$ is a unitary matrix and $\ell[n] \in \{0, 1, \cdots, L-1\}$ is the index of the unitary constellation matrix at time n. Here the block codeword length N_s of the

DUST code we use in our system equals N_t. The transmitter sends block symbols $\mathbf{S}[n]$, while $V_{\ell[n]}$ represents the actual data contained in the block sequence. For example, if the transmission rate is R bits/channel use for a N_t transmit antenna scheme, the constellation size will be $L = 2^{RN_t}$ and we need L unitary matrix choices for $V_{\ell[n]}$. Similar to DPSK above, the receiver estimates $V_{\ell[n]}$ using the last two received blocks $\mathbf{X}[n]$ and $\mathbf{X}[n-1]$. Since:

$$\mathbf{X}[n-1] = \sqrt{\frac{\rho}{N_t}}\tilde{\mathbf{H}}\mathbf{S}[n-1] + \mathbf{W}[n-1] \tag{3.4}$$

$$\mathbf{X}[n] = \sqrt{\frac{\rho}{N_t}}\tilde{\mathbf{H}}\mathbf{S}[n] + \mathbf{W}[n]. \tag{3.5}$$

Define:

$$\overline{\mathbf{X}}[n] = (\mathbf{X}[n-1], \mathbf{X}[n])$$

$$\overline{\mathbf{S}}[n] = (\mathbf{S}[n-1], \mathbf{S}[n-1]V_{\ell[n]})$$

$$\overline{\mathbf{W}}[n] = (\mathbf{W}[n-1], \mathbf{W}[n]).$$

So we get:

$$\overline{\mathbf{X}}[n] = \sqrt{\frac{\rho}{N_t}}\tilde{\mathbf{H}}\overline{\mathbf{S}}[n] + \overline{\mathbf{W}}[n].$$

With the property of the unitary codes, $V_{\ell[n]}V_{\ell[n]}^H = N_t I$,

$$\begin{aligned}
\overline{\mathbf{S}}[n]^H\overline{\mathbf{S}}[n] &= \begin{pmatrix} \mathbf{S}[n-1]^H\mathbf{S}[n-1] & \mathbf{S}[n-1]^H\mathbf{S}[n-1]V_{\ell[n]} \\ V_{\ell[n]}^H\mathbf{S}[n-1]^H\mathbf{S}[n-1] & V_{\ell[n]}^H\mathbf{S}[n-1]^H\mathbf{S}[n-1]V_{\ell[n]} \end{pmatrix} \\
&= \begin{pmatrix} N_t I & N_t V_{\ell[n]} \\ N_t V_{\ell[n]}^H & N_t I \end{pmatrix},
\end{aligned}$$

so the ML detector for the above model from 3.3 is:

$$\begin{aligned}
\hat{\ell}[n] &= \arg \max_{\ell\in\{0,1,\cdots,L-1\}} \operatorname{tr}\left\{\overline{\mathbf{X}}\overline{\mathbf{S}}^H\overline{\mathbf{S}}\overline{\mathbf{X}}^H\right\} \\
&= \arg \max_{\ell\in\{0,1,L-1\}} \operatorname{tr}\left\{(\mathbf{X}[n-1], \mathbf{X}[n]) \begin{pmatrix} N_t I & N_t V_{\ell[n]} \\ N_t V_{\ell[n]}^H & N_t I \end{pmatrix} \begin{pmatrix} \mathbf{X}^H[n-1] \\ \mathbf{X}^H[n] \end{pmatrix}\right\} \\
&= \arg \max_{\ell\in\{0,1,\cdots,L-1\}} \operatorname{Re}\left(\operatorname{tr}\left\{\mathbf{X}[n-1]V_{\ell[n]}\mathbf{X}[n]^H\right\}\right),
\end{aligned}$$

where Re(·) means taking the real part.

From (3.4) and (3.5), we get the following expression:

$$\begin{aligned}\mathbf{X}[n] &= \sqrt{\frac{\rho}{N_t}}\mathbf{HS}[n-1]V_{\ell[n]} + \mathbf{W}[n] \\ &= \mathbf{X}[n-1]V_{\ell[n]} + \mathbf{W}[n-1]V_{\ell[n]} + \mathbf{W}[n] \\ &= \mathbf{X}[n-1]V_{\ell[n]} + \sqrt{2}\mathbf{W}'[n]. \end{aligned} \tag{3.6}$$

Equation (3.6) is called the "fundamental difference equation" in [4], where $\mathbf{W}'$ has the same statistics as $\mathbf{W}$. Thus the information block $V_{\ell[n]}$ goes through an effective known channel with response $\mathbf{X}[n-1]$ and is corrupted by effective noise $\mathbf{W}'$ with twice the variance of the channel noise $\mathbf{W}$. This results in a 3dB loss in performance relative to coherent detection. Note that the restriction to unitary alphabets further reduces the performance of DUST relative to coherent space-time modulation.

We will describe the property of the DUST code now. As we have stated that $V_{\ell[n]}$ is a unitary matrices from L-ary alphabets. Because group constellations can simplify the differential scheme, both Hughes [3] and Hochwald [4] suggest the group design method, i.e., let $\mathcal{V}$ be an algebraic group of L $N_t \times N_t$ unitary matrices. Using group structure, the transmitters don't need to explicitly multiply any matrices, since a group is closed under multiplication.

In this thesis, we use the DUST code construction proposed by Hughes in [3] which is a general approach to differential modulation and can be applied to any number of transmit antennas and any target constellation. These unitary group codes have the property:

$$\mathbf{S}[n] = \mathbf{S}[n-1]V_{\ell[n]},\ \mathbf{S}[0] = V_k \quad k \in \{0, 1, \cdots, L-1\},$$

with $\mathbf{S}[0]$ being any matrix in the group. $\mathbf{S}[0]$ doesn't need to be known to the receiver, because the difference codeword $V_{\ell[n]}$ contains the real information to be transmitted. $V_{\ell[n]}$ is the nth information block and $\mathbf{S}[n]$ is the nth transmitted block, which are all elements of a group of unitary matrices. As we mentioned before, the DUST code we use has the property $N_s = N_t$. For example, for $N_t = 2$, the construction might be:

$$\mathcal{V} = \left\{ \pm \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \pm \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \pm \begin{bmatrix} j & 0 \\ 0 & -j \end{bmatrix}, \pm \begin{bmatrix} 0 & j \\ j & 0 \end{bmatrix} \right\} \qquad \mathbf{S}[0] \in \mathcal{V}.$$

As suggested by (3.6) and (3.3), the ML decoder has a very simple form:

$$\hat{\ell} = \arg \max_{\ell \in \{0,1,\cdots,L-1\}} \mathrm{Re}(\mathrm{tr}(\mathbf{X}[n-1] V_\ell \mathbf{X}^H [n-1])). \qquad (3.7)$$

In this thesis, we assume that the DUST codes, designed for flat fading, are used in frequency-selective fading as described in Section 1.2. Recall that deterministic MIMO blind identification and equalization techniques introduced in Chapter 2 can estimate the symbols up to a $N_t \times N_t$ matrix ambiguity, meaning they can effectively reduce a frequency-selective fading channel to an unknown flat fading channel. Then, the DUST code property and the soft ILSP or iterative PSP method (which we will describe later) can yield fully-blind estimation of the symbols in our MIMO frequency-selective fading model.

CHAPTER 4

ITERATIVE LEAST SQUARE WITH PROJECTION ALGORITHM

4.1 Initial blind estimation of the code sequence

After application of the deterministic sub-space method in Chapter 2 to our MIMO linear system model (1.1) introduced in Section 1.2, we get:

$$Y = A\mathbf{S} + Z. \tag{4.1}$$

Y is the estimated signal sequence of size $N_t \times N$. A is the "ambiguity matrix" of size $N_t \times N_t$. Z is the residual noise and estimation error introduced by the deterministic sub-space algorithm. We need to recover the input sequence $\mathbf{S} = (\mathbf{s}[-N_h+1], \cdots \mathbf{s}[N-1]) \in \mathbb{C}^{N_t \times N+N_h-1}$ from Y. This can be viewed as an equivalent flat fading model with unknown channel response A. The transmitted DUST block codewords are of size $N_t \times N_t$. For simplicity, we assume the transmitted signal vectors with minus index are all $\mathbf{0}$, i.e., $[\mathbf{s}[-N_h+1], \cdots, \mathbf{s}[-1]] = \mathbf{0}$ and they are the "guard" bits between frames. So we group $\mathbf{s}[n]$ in block codewords of length N_t, obtaining:

$$\mathbf{S}[m] = \begin{pmatrix} \mathbf{s}[mN_t] & \mathbf{s}[mN_t+1] & \cdots & \mathbf{s}[(m+1)N_t-1] \end{pmatrix} \in \mathbb{C}^{N_t \times N_t}.$$

Assuming $N_c = \lfloor \frac{N}{N_t} \rfloor$, we can get N_c complete DUST block codewords in each frame, i.e., $\mathbf{S} = (\mathbf{S}[0], \cdots, \mathbf{S}[N_c - 1])$. We group the estimated sequence Y in the same way, so $Y = (Y[0], \cdots, Y[N_c - 1])$. Since the transmitted block symbols are differentially encoded, we can use the decoding scheme (3.7) introduced in the DUST modulation part to get the initial estimation $\hat{\mathbf{S}}^{(0)}$ of the transmitted information block codewords.

Recall that the transmitted block codeword $\mathbf{S}[m]$ has the property that $\mathbf{S}[m] = \mathbf{S}[m-1]V_{\ell[m]}$. Then for $m = 1, \cdots, N_c - 1$,

$$\hat{\ell}[m] = \arg \max_{\ell[m] \in \{0, \cdots, L-1\}} \mathrm{Re}(\mathrm{tr}\left\{Y[m-1]V_{\ell[m]}Y^H[m]\right\}).$$

Given the estimate $\hat{\ell}[m]$ and supposing the first block codeword is any arbitrary codeword in the group, i.e., $\hat{\mathbf{S}}^{(0)}[0] = \mathbf{S}[0] \in \mathcal{V}$ as introduced in Section 3.4, set $\hat{\mathbf{S}}^{(0)}[m] = \hat{\mathbf{S}}^{(0)}[m-1]V_{\hat{\ell}[m]}$. For $m = 1, \cdots, N_c - 1$,

$$\hat{\mathbf{S}}^{(0)} = \left(\hat{\mathbf{S}}^{(0)}[0], \cdots, \hat{\mathbf{S}}^{(0)}[N_c - 1]\right).$$

This initial estimation $\hat{\mathbf{S}}^{(0)}$ is perfect if the system model (1.1) doesn't contain the noise part $\mathbf{w}[n]$, because the blind sub-space method introduced in Chapter 2 is perfect in noiseless case, i.e., the output error Z from the blind sub-space algorithm is $\mathbf{0}$. But if noise is added to the system model (1.1), the blind sub-space algorithm introduces great noise in Z part. So errors are introduced in the initial estimates $\hat{\mathbf{S}}^{(0)}$. To improve the performance of our blind algorithm, we apply the Iterative Least Square Projection (ILSP) method and soft ILSP further.

4.2 ILSP

ILSP is proposed by Talwar, et al. in [9] for separating and estimating the input digital signals in MIMO systems when the channel coefficients $\mathbf{H}$ are unknown and the digital signals $\mathbf{S}$ are of finite alphabet.

Recall our MIMO linear system model (1.1) is:

$$\mathbf{x}[n] = \sum_{l=0}^{N_h-1} \mathbf{H}[l]\mathbf{s}[n-l] + \mathbf{w}[n] \quad \text{for } n = 0, \cdots, N-1,$$

N is the number of transmit symbol periods in a frame, $\mathbf{w}[n]$ is the white noise. Then

$$\underbrace{\begin{bmatrix} \mathbf{x}[0] & \cdots & \mathbf{x}[N-1] \end{bmatrix}}_{\mathbf{X}} = \underbrace{\begin{bmatrix} \mathbf{H}[0] & \cdots & \mathbf{H}[N_h-1] \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} \mathbf{s}[0] & \cdots & \mathbf{s}[N-1] \\ \vdots & \ddots & \vdots \\ \underbrace{\mathbf{s}[-N_h+1]}_{\underline{\mathbf{s}}[0]} & \cdots & \underbrace{\mathbf{s}[N-N_h]}_{\underline{\mathbf{s}}[N-1]} \end{bmatrix}}_{\mathcal{S}_{N_h}} + \mathbf{W} \tag{4.2}$$

Equation (4.2) can be simplified as:

$$\mathbf{x}[n] = \mathbf{H}\underline{\mathbf{s}}[n] + \mathbf{w}[n], \tag{4.3}$$

since the noise $\mathbf{w}[n]$ is spatially white and complex Gaussian, the probability of $\mathbf{x}[n]$ given $\underline{\mathbf{s}}[n]$ as a function of $\mathbf{H}$ is:

$$p(\mathbf{x}[n]|\underline{\mathbf{s}}[n]; \mathbf{H}) = \mathrm{C}_1 \exp(-\frac{\|\mathbf{x}[n] - \mathbf{H}\underline{\mathbf{s}}[n]\|^2}{\sigma_w^2}),$$

where C_1 is some constant and σ_w^2 is the variance of the entries in $\mathbf{w}[n]$. Assuming the noise is temporally white, then the log likelihood of the observed data over N symbol periods is:

$$\log p(\mathbf{X}|\mathcal{S}_{N_h}; \mathbf{H}) = C_2 - \frac{1}{\sigma_w^2} \sum_{n=0}^{N-1} \|\mathbf{x}[n] - \mathbf{H}\underline{\mathbf{s}}[n]\|^2,$$

where C_2 is some constant. So the ML estimator maximizes $\log p(\mathbf{X}|\mathcal{S}_{N_h};\mathbf{H})$ with respect to the unknown parameter $\mathbf{H}$ and finite-alphabet $\mathcal{S}_{N_h}$. If DUST codes are used for $\mathbf{S}$, then each block codeword $\mathbf{S}[n]$ in $\mathbf{S}$ is in the group codes $\mathcal{V}$ which is of finite alphabet. So the transmit signals $\mathbf{S}$ is also constrained to a finite alphabet $\mathcal{U}$. Since $\mathcal{S}_{N_h}$ is generated from $\mathbf{S}$, the ML criteria can be written as:

$$\hat{\mathbf{S}} = \arg\min_{\mathbf{H},\mathbf{S}\in\mathcal{U}} \|\mathbf{X} - \mathbf{H}\mathcal{S}_{N_h}\|_F^2, \tag{4.4}$$

Equation (4.4) is a non-linear separable optimization problem with mixed discrete and non-discrete variables. We can solve this optimization problem in the following steps [10].

First, since $\mathbf{H}$ is unconstrained, we can minimize (4.4) with respect to $\mathbf{H}$, so that for any $\mathbf{S}$,

$$\hat{\mathbf{H}} = \mathbf{X}\mathcal{S}_{N_h}^{\dagger},$$

where $\mathcal{S}_{N_h}^{\dagger}$ means the pseudo-inverse of $\mathcal{S}_{N_h}$, and $\mathcal{S}_{N_h}^{\dagger} = \mathcal{S}_{N_h}^{H}(\mathcal{S}_{N_h}\mathcal{S}_{N_h}^{H})^{-1}$. Then plugging $\hat{\mathbf{H}}$ to (4.4), we get:

$$\hat{\mathbf{S}} = \arg\min_{\mathbf{S}\in\mathcal{U}} \|\mathbf{X}(I - \mathcal{S}_{N_h}^{H}(\mathcal{S}_{N_h}\mathcal{S}_{N_h}^{H})^{-1}\mathcal{S}_{N_h})\|_F^2.$$

The global minimum of the above can be found by enumeration of all possible $\mathbf{S}\in\mathcal{U}$, but the complexity grows exponentially with frame duration N. The ILSP algorithm below is proposed to save complexity and retain reasonably good estimation of joint $\mathbf{S}$ and $\mathbf{H}$.

Assume the cost function:

$$d(\mathbf{H},\mathbf{S}) = \|\mathbf{X} - \mathbf{H}\mathcal{S}_{N_h}\|_F^2.$$

Given an initial estimate $\hat{\mathbf{S}}^{(0)}$ in Section 4.1, the initial estimate of the block-Toeplitz matrix $\hat{\mathcal{S}}_{N_h}^{(0)}$ can be constructed from $\hat{\mathbf{S}}^{(0)}$, then the minimization of $d(\mathbf{H},\hat{\mathbf{S}}^{(0)})$ with respect to $\mathbf{H} \in \mathbb{C}^{N_r N_o \times N_t N_h}$ is a least square problem, which can be solved via $\hat{\mathbf{H}}^{(0)} = \mathbf{X}\hat{\mathcal{S}}_{N_h}^{(0)\dagger}$.

Given the initial estimate $\hat{\mathbf{H}}^{(0)}$, the minimization of $d(\hat{\mathbf{H}}^{(0)}, \mathbf{S})$ with respect to $\mathbf{S} \in \mathbb{C}^{N_t \times N}$ is also a least-square problem, but since $\mathbf{H}$ is not of full column rank, the least square estimation of $\mathbf{S}$ can not be derived from $\hat{\mathcal{S}}^{(1)} = \hat{\mathbf{H}}^{(0)\dagger}\mathbf{X}$, instead we need to transform the MIMO system model (1.1) to the following equivalent form:

$$\underbrace{\begin{bmatrix} \mathbf{x}[N-1] \\ \vdots \\ \mathbf{x}[0] \end{bmatrix}}_{\underline{\mathbf{x}}} = \underbrace{\begin{bmatrix} \mathbf{H}[0] & \cdots & \mathbf{H}[N_h-1] & & \mathbf{0} \\ & \ddots & & \ddots & \\ \mathbf{0} & & \mathbf{H}[0] & \cdots & \mathbf{H}[N_h-1] \end{bmatrix}}_{\mathcal{H}} \underbrace{\begin{bmatrix} \mathbf{s}[N-1] \\ \vdots \\ \mathbf{s}[-N_h+1] \end{bmatrix}}_{\underline{\mathbf{s}}} + \underbrace{\begin{bmatrix} \mathbf{w}[N-1] \\ \vdots \\ \mathbf{w}[0] \end{bmatrix}}_{\underline{\mathbf{w}}}, \tag{4.5}$$

where $\underline{\mathbf{w}}$ is the stacked white noise. Given the initial channel estimation $\hat{\mathbf{H}}^{(0)}$, we can construct the block-Toeplitz matrix $\hat{\mathcal{H}}^{(0)}$. So we get the model equation $\underline{\mathbf{x}} = \hat{\mathcal{H}}^{(0)}\underline{\mathbf{s}} + \underline{\mathbf{w}}^{(0)}$, where now $\underline{\mathbf{w}}^{(0)}$ captures estimation errors in $\hat{\mathcal{H}}^{(0)}$. Assuming $\underline{\mathbf{w}}^{(0)}$ is white and Gaussian, we can get the maximum likelihood estimation of $\mathbf{S}$:

$$\hat{\mathbf{S}}_{ML} = \arg \min_{\substack{\mathbf{S}[m]\in\mathcal{V} \\ m=0,\cdots,N_c-1}} \|\underline{\mathbf{x}} - \hat{\mathcal{H}}^{(0)}\underline{\mathbf{s}}\|^2. \tag{4.6}$$

Note the complexity of the above maximum likelihood decoding is exponential in the number of blocks N_c. To reduce the complexity, we can simplify (4.6) and find the updated estimated code sequence $\hat{\mathbf{S}}^{(1)} = (\hat{\mathbf{S}}^{(1)}[0], \cdots, \hat{\mathbf{S}}^{(1)}[N_c-1])$ by the following steps: first, find the maximum likelihood estimation of $\underline{\mathbf{s}}$ in the *complex field*, denoted by $\tilde{\underline{\mathbf{s}}}^{(1)}$; second, arrange the elements of $\tilde{\underline{\mathbf{s}}}^{(1)}$ in blocks of size $N_t \times N_t$ and form a sequence $(\tilde{\mathbf{S}}^{(1)}[0], \cdots, \tilde{\mathbf{S}}^{(1)}[N_c-1])$; third, project the block codeword in $(\tilde{\mathbf{S}}^{(1)}[0], \cdots, \tilde{\mathbf{S}}^{(1)}[N_c-1])$ onto the discrete alphabet $\mathcal{V}$ to get $(\hat{\mathbf{S}}^{(1)}[0], \cdots, \hat{\mathbf{S}}^{(1)}[N_c-1])$. The codeword projection process can be expressed as the following:

1. $\tilde{\underline{\mathbf{s}}}^{(1)} = \arg \min_{\underline{\mathbf{s}}\in\mathbb{C}} \|\underline{\mathbf{x}} - \hat{\mathcal{H}}^{(0)}\underline{\mathbf{s}}\| = \hat{\mathcal{H}}^{(0)\dagger}\underline{\mathbf{x}}$,

2. $\underline{\tilde{\mathbf{s}}}^{(1)} \rightarrow (\tilde{\mathbf{S}}^{(1)}[0], \cdots, \tilde{\mathbf{S}}^{(1)}[N_c - 1])$,

3. $\hat{\mathbf{S}}^{(1)}[m] = \text{Project}(\tilde{\mathbf{S}}^{(1)}[m])$ onto $\mathcal{V}$ for $m = 0, \cdots, N_c - 1$.

When doing the projection, we use the following similarity criteria between the codeword $\mathbf{S}[m]$ and the choice V_ℓ from the group codes $\mathcal{V}$:

$$d_{m,\ell} = \frac{\exp(-\|V_\ell - \tilde{\mathbf{S}}^{(k)}[m]\|_F^2)}{\max_q \exp(-\|V_q - \tilde{\mathbf{S}}^{(k)}[m]\|_F^2)}. \tag{4.7}$$

Note that the mth block codeword is most likely corresponding to the codeword with index:

$$\hat{\ell}[m] = \arg\min_{\ell} d_{m,\ell}.$$

Then the updated estimate of the code sequence becomes,

$$\hat{\mathbf{S}}^{(1)} = \left[\hat{\mathbf{S}}^{(1)}[0], \cdots, \hat{\mathbf{S}}^{(1)}[N_c - 1]\right], \quad \text{where} \quad \hat{\mathbf{S}}^{(1)}[m] = V_{\hat{\ell}[m]}.$$

After we get $\hat{\mathbf{S}}^{(1)}$, $\mathbf{H}$ is re-estimated by minimizing $d(\mathbf{H}, \hat{\mathbf{S}}^{(0)})$ with respect to $\mathbf{H}$, yielding $\hat{\mathbf{H}}^{(1)} = \mathbf{X}\hat{\mathcal{S}}_{N_h}^{(1)\dagger}$. Then we can get updated estimation $\hat{\mathbf{S}}^{(2)}$ from projection method using $\hat{\mathbf{H}}^{(1)}$. This iteration is repeated until $\hat{\mathbf{S}}^{(k)}$ converges. ILSP can be summarized below:

ILSP

1. Given $\hat{\mathbf{S}}^{(0)}$ for $k = 0$.

2. Initial channel estimation: $\hat{\mathbf{H}}^{(0)} = \mathbf{X}\hat{\mathcal{S}}_{N_h}^{(0)\dagger}$.

3. $k = k + 1$

   (a) Update estimation $\hat{\mathbf{S}}^{(k)}$ from projection method using $\hat{\mathbf{H}}^{(k-1)}$:

      i. $\underline{\tilde{\mathbf{s}}}^{(k)} = \hat{\mathcal{H}}^{(k-1)\dagger}\underline{\mathbf{x}}$,

ii. $\underline{\tilde{\mathbf{s}}}^{(k)} \to (\tilde{\mathbf{S}}^{(k)}[0], \cdots, \tilde{\mathbf{S}}^{(k)}[N_c - 1])$,

iii. Project $\tilde{\mathbf{S}}^{(k)}$ to closest discrete values and get $\hat{\mathbf{S}}^{(k)}$.

(b) Update estimation $\hat{\mathbf{H}}^{(k)}$ from least square method using $\hat{\mathbf{S}}^{(k)}$:

$\hat{\mathbf{H}}^{(k)} = \mathbf{X}\hat{\mathcal{S}}_{N_h}^{(k)\dagger}$.

(c) If $\hat{\mathbf{S}}^{(k)} \neq \hat{\mathbf{S}}^{(k-1)}$, goto 3.

ILSP can be used to separate an instantaneous linear mixture of finite alphabet signals. It reduces computational complexity because it avoids enumeration all possibilities of $\mathbf{S}$. However, since it can not guarantee that the cost is minimized at each iteration due to the projection step, it is suboptimal. It is important to have a reasonably accurate initial estimate $\hat{\mathbf{S}}^{(0)}$ so that ILSP has a good chance to converge to the global minimum of $d(\mathbf{H}, \mathbf{S})$. For "typical" matrix dimension and noise level, ILSP usually converge to a fixed point in less than $5 - 10$ iterations [9]. The cost $\|\mathbf{X} - \hat{\mathbf{H}}^{(k)}\hat{\mathcal{S}}_{N_h}^{(k)}\|_F^2$ indicates how close the estimated values are to the true optima.

4.3 Soft ILSP

To improve the performance further, we apply a modified version of ILSP called "soft ILSP". The process of soft ILSP can be summarized below starting from an initial estimate $\hat{\mathbf{S}}^{(0)}$ from Section 4.1.

Soft ILSP

1. Given $\hat{\mathbf{S}}^{(0)}$, $k = 0$.

2. $\hat{\mathbf{H}}^{(0)} = \mathbf{X}\hat{\mathcal{S}}_{N_h}^{(0)\dagger}$.

3. for $k = 1$ to N_m(Maximum number of iterations)

(a) Update estimation of pseudo-probability $p_{n,m}^{(k-1)}$ with projection method using $\hat{\mathbf{H}}^{(k-1)}$:

i. $\tilde{\underline{\mathbf{s}}}^{(k)} = \hat{\mathcal{H}}^{(k-1)\dagger}\underline{\mathbf{x}}$,

ii. Estimation of codeword pseudo-probabilities $p_{n,m}^{(k-1)}$ using $\tilde{\underline{\mathbf{s}}}^{(k)}$.

(b) Update estimation $\hat{\mathbf{H}}^{(k)}$ with EM algorithm using codeword pseudo-probabilities $p_{n,m}^{(k-1)}$.

Soft ILSP is similar to ILSP, they both are iterative process and use the same initializations. The difference between them is that ILSP use projection to get the single most possible choice for each block codeword $\mathbf{S}[n]$ while soft ILSP use projection to get several possible choices for each column vector in $\mathcal{S}_{N_h}$. The other difference is that ILSP use least square method to re-estimate the channel response while soft ILSP use EM-based algorithm to re-estimate the channel response. We will give the details of the different updating process in soft ILSP below.

Step $3(a)$. Update estimation of soft codeword pseudo-probabilities $p_{n,m}^{(k-1)}$ using $\mathbf{H}^{(k-1)}$.

Consider the MIMO system model (4.2), each column vector $\underline{\mathbf{s}}[n]$ is decided by block codewords $\left[\mathbf{S}\left[\lfloor\frac{n}{N_t}\rfloor\right],\cdots,\mathbf{S}\left[\lfloor\frac{n-N_h+1}{N_t}\rfloor\right]\right]$. Since each codeword $\mathbf{S}[n]\in\mathcal{V}$ is of finite alphabet, each column vector $\underline{\mathbf{s}}[n]$ in $\mathcal{S}_{N_h}$ is also of finite alphabet. Suppose the set of all choices of column vector $\underline{\mathbf{s}}[n]$ is $\underline{V}_n = \left\{\underline{\mathbf{s}}_{n,i}\right\}_{i=1}^{L_n}$, so the size of $\underline{V}_n$ is L_n. Given the current estimated codewords $\tilde{\underline{\mathbf{s}}}^{(k)}$ in complex field from $\tilde{\underline{\mathbf{s}}}^{(k)} = \hat{\mathcal{H}}^{(k-1)\dagger}\underline{\mathbf{x}}$, we can construct the estimated block-Toeplitz matrix $\tilde{\mathcal{S}}_{N_h}^{(k)}$. Based on this estimation, we can define the following criteria of distance similar to (4.7). For each choice $\underline{\mathbf{s}}_{n,i}$ in the set

$\underline{V}_n$, the distance between the column vector $\underline{\mathbf{s}}[n]$ and the choice $\underline{\mathbf{s}}_{n,i}$ is $d_{n,i}$,

$$d_{n,i} = \frac{\exp\left(-\|\underline{\mathbf{s}}_{n,i} - \tilde{\mathbf{s}}^{(k)}[n]\|^2\right)}{\max_j \exp\left(\|\underline{\mathbf{s}}_{n,j} - \tilde{\mathbf{s}}^{(k)}[n]\|^2\right)}. \tag{4.8}$$

For each $\underline{\mathbf{s}}[n]$, there are L_n choices, each with similarity coefficient $d_{n,i}$. To simplify the algorithm, we only consider the most possible choices for $\underline{\mathbf{s}}[n]$. Specifically, we set a threshold D_n. If $d_{n,m} \geq D_n$, we consider $\mathbf{s}[n]$ as a valid possibility for $\underline{\mathbf{s}}[n]$. If $d_{n,m} < D_n$, we do not consider the possibility $\underline{\mathbf{s}}_{n,m}$ as valid. Suppose for $\underline{\mathbf{s}}[n]$ there are $l_n \leq L_n$ valid choices. Furthermore assume that the set $\underline{V}_n$ was constructed so that the first l_n elements are these valid choices, i.e., $\{\underline{\mathbf{s}}_{n,m}\}_{m=1}^{l_n}$. Now define $\underline{v}_n = \{\underline{\mathbf{s}}_{n,m}\}_{m=1}^{l_n} \subseteq \underline{V}_n$. The valid element $\underline{\mathbf{s}}_{n,m}$ is assigned "pseudo-probability" $p_{n,m}^{(k-1)}$, defined as:

$$\begin{aligned} p_{n,m}^{(k-1)} &:= \frac{d_{n,m}}{\sum_{m=1}^{l_n} d_{n,m}} \\ &\approx p(\underline{\mathbf{s}}[n] = \underline{\mathbf{s}}_{n,m} | \mathbf{X}, \hat{\mathbf{H}}^{(k-1)}). \end{aligned} \tag{4.9}$$

The threshold of D_n depends on how many choices we can afford to keep for each n. For example, if we have $D_n = \min_i(d_{n,i})$, there are L_n choices for each $\underline{\mathbf{s}}[n]$. This is the case of enumeration all choices of $\underline{V}_n$ and is of the highest complexity. When the threshold $D_n = \max_i(d_{n,i})$, we are doing a "hard" projection similar to ILSP: each $\underline{\mathbf{s}}[n]$ has just one choice and this case has the lowest complexity. By setting the threshold D_n, we can adjust the complexity of the algorithm. We call this "soft" projection, because for each column vector $\underline{\mathbf{s}}[n]$, there might be multiple choices. And these multiple choices together with their pseudo-probability $p_{n,m}^{(k-1)}$ will be used in the re-estimation of $\mathbf{H}$ as described below.

Step $3(b)$. Using expectation estimation (EM) algorithm to update estimation $\hat{\mathbf{H}}^{(k)}$ with the pseudo-probabilities.

The EM algorithm can produce the maximum-likelihood estimates of parameters when there is a many-to-one mapping from an underlying distribution to the distribution governing the observation [8]. With the system model (4.2), given the observation data sequence $\mathbf{X}$ and the estimated soft codewords with corresponding pseudo-probabilities, we would like to estimate the parameter $\mathbf{H}$.

Since $\mathbf{W}$ in (4.2) is white Gaussian noise, the likelihood of $\mathbf{X}$ conditioned on the transmitted symbols $\mathcal{S}_{N_h}$ and the channel response $\mathbf{H}$ is:

$$p(\mathbf{X}|\mathcal{S}_{N_h}, \mathbf{H}) = C_3 \exp(-\frac{\|\mathbf{X} - \mathbf{H}\mathcal{S}_{N_h}\|_F^2}{\sigma_w^2}).$$

Then the joint probability of $\mathbf{X}$ and $\mathcal{S}_{N_h}$ conditioned on $\mathbf{H}$ is:

$$\begin{aligned} p(\mathbf{X}, \mathcal{S}_{N_h}|\mathbf{H}) &= p(\mathbf{X}|\mathcal{S}_{N_h}, \mathbf{H}) p(\mathcal{S}_{N_h}; \mathbf{H}) \\ &= p(\mathbf{X}|\mathcal{S}_{N_h}, \mathbf{H}) p(\mathcal{S}_{N_h}) \\ &= C_3 \exp(-\frac{\|\mathbf{X} - \mathbf{H}\mathcal{S}_{N_h}\|_F^2}{\sigma_w^2}) p(\mathcal{S}_{N_h}). \end{aligned}$$

Taking log of the above probability,

$$\log p(\mathbf{X}, \mathcal{S}_{N_h}|\mathbf{H}) = C_4 - \frac{1}{\sigma_w^2}\|\mathbf{X} - \mathbf{H}\mathcal{S}_{N_h}\|_F^2 + \log p(\mathcal{S}_{N_h}).$$

The basic idea of EM is that we want to minimize the above log-likelihood, but we don't have the data $\mathbf{H}$ to compute it. So instead, we maximize the expectation of the log-likelihood given the observed data and our previous estimation $\hat{\mathbf{H}}^{(k-1)}$. This can be expressed in two steps [8].

Let $\hat{\mathbf{H}}^{(k-1)}$ be our previous estimate of parameter $\mathbf{H}$ from the $(k-1)$th iteration. For the E-step, we compute:

$$Q(\hat{\mathbf{H}}, \hat{\mathbf{H}}^{(k-1)}) \quad := \quad \mathrm{E}(\log p(\mathbf{X}, \mathcal{S}_{N_h}|\mathbf{H} = \hat{\mathbf{H}})|\mathbf{X}, \mathbf{H} = \hat{\mathbf{H}}^{(k-1)})$$

$$
\begin{aligned}
&= \int_{\mathcal{S}_{N_h}} \log p(\mathbf{X}, \mathcal{S}_{N_h} | \mathbf{H} = \hat{\mathbf{H}}) p(\mathcal{S}_{N_h} | \mathbf{X}, \mathbf{H} = \hat{\mathbf{H}}^{(k-1)}) d\mathcal{S}_{N_h} \\
&= \int_{\mathcal{S}_{N_h}} \left[C_4 - \frac{1}{\sigma_w^2} \|\mathbf{X} - \mathbf{H}\mathcal{S}_{N_h}\|_F^2 + \log p(\mathcal{S}_{N_h}) \right] p(\mathcal{S}_{N_h} | \mathbf{X}, \mathbf{H} = \hat{\mathbf{H}}^{(k-1)}) d\mathcal{S}_{N_h} \\
&= C_5 - \frac{1}{\sigma_w^2} \int_{\mathcal{S}_{N_h}} \|\mathbf{X} - \hat{\mathbf{H}}\mathcal{S}_{N_h}\|_F^2 p(\mathcal{S}_{N_h} | \mathbf{X}, \mathbf{H} = \hat{\mathbf{H}}^{(k-1)}) d\mathcal{S}_{N_h}.
\end{aligned}
$$

Since,

$$
\|\mathbf{X} - \hat{\mathbf{H}}\mathcal{S}_{N_h}\|_F^2 = \sum_{n=0}^{N-1} \|\mathbf{x}[n] - \hat{\mathbf{H}}\underline{\mathbf{s}}[n]\|^2,
$$

where $\underline{\mathbf{s}}[n] \in \underline{v}_n$, the above Q function can be expressed as:

$$
\begin{aligned}
&Q(\hat{\mathbf{H}}, \hat{\mathbf{H}}^{(k-1)}) \\
&= C_5 - \frac{1}{\sigma_w^2} \sum_{n=0}^{N-1} \int_{\underline{v}_0} \int_{\underline{v}_1} \cdots \int_{\underline{v}_{N-1}} \|\mathbf{x}[n] - \hat{\mathbf{H}}\underline{\mathbf{s}}[n]\|^2 p(\underline{\mathbf{s}}[0], \cdots, \underline{\mathbf{s}}[N-1] | \mathbf{X}, \hat{\mathbf{H}}^{(k-1)}) d\underline{\mathbf{s}}[0] \cdots d\underline{\mathbf{s}}[N-1] \\
&= C_5 - \frac{1}{\sigma_w^2} \sum_{n=0}^{N-1} \int_{\underline{v}_n} \|\mathbf{x}[n] - \hat{\mathbf{H}}\underline{\mathbf{s}}[n]\|^2 \int \cdots \int_{\underline{v}_{j \neq n}} p(\underline{\mathbf{s}}[0], \cdots, \underline{\mathbf{s}}[N-1] | \mathbf{X}, \hat{\mathbf{H}}^{(k-1)}) d\underline{\mathbf{s}}[0] \cdots d\underline{\mathbf{s}}[N-1],
\end{aligned}
$$

where,

$$
\int \cdots \int_{\underline{v}_{j \neq n}} p(\underline{\mathbf{s}}[0], \cdots, \underline{\mathbf{s}}[N-1] | \mathbf{X}, \hat{\mathbf{H}}^{(k-1)}) d\underline{\mathbf{s}}[0] \cdots d\underline{\mathbf{s}}[N-1] = p(\underline{\mathbf{s}}[n] | \mathbf{X}, \hat{\mathbf{H}}^{(k-1)}) d\underline{\mathbf{s}}[n].
$$

The above Q function can be further simplified as:

$$
\begin{aligned}
Q(\hat{\mathbf{H}}, \hat{\mathbf{H}}^{(k-1)}) &= C_5 - \frac{1}{\sigma_w^2} \sum_{n=0}^{N-1} \int_{\underline{v}_n} \|\mathbf{x}[n] - \hat{\mathbf{H}}\underline{\mathbf{s}}[n]\|^2 p(\underline{\mathbf{s}}[n] | \mathbf{X}, \hat{\mathbf{H}}^{(k-1)}) d\underline{\mathbf{s}}[n] \\
&= C_5 - \frac{1}{\sigma_w^2} \sum_{n=0}^{N-1} \sum_{m=1}^{L_n} \|\mathbf{x}[n] - \hat{\mathbf{H}}\underline{\mathbf{s}}_{n,m}\|^2 p(\underline{\mathbf{s}}[n] = \underline{\mathbf{s}}_{n,m} | \mathbf{X}, \hat{\mathbf{H}}^{(k-1)}).
\end{aligned}
$$

From (4.9), we make the approximation:

$$
p_{n,m}^{(k-1)} \approx p(\underline{\mathbf{s}}[n] = \underline{\mathbf{s}}_{n,m} | \mathbf{X}, \hat{\mathbf{H}}^{(k-1)}).
$$

Then the Q function can be approximated by the following expression:

$$
Q(\hat{\mathbf{H}}, \hat{\mathbf{H}}^{(k-1)}) \approx C_5 - \frac{1}{\sigma_w^2} \sum_{n=0}^{N-1} \sum_{m=1}^{l_n} p_{n,m}^{(k-1)} \|\mathbf{x}[n] - \hat{\mathbf{H}}\underline{\mathbf{s}}_{n,m}\|^2,
$$

since $p_{n,m}^{(k-1)} = 0$ for $m > l_n$. The new estimation $\hat{\mathbf{H}}^{(k)}$ is $\hat{\mathbf{H}}$ which maximizes the Q function above.

$$\begin{aligned}\hat{\mathbf{H}}^{(k)} &= \arg\max_{\hat{\mathbf{H}}} Q(\hat{\mathbf{H}}, \hat{\mathbf{H}}^{(k-1)}) \\ &= \arg\min_{\hat{\mathbf{H}}} \sum_{n,m} p_{n,m}^{k-1} \|\mathbf{x}[n] - \hat{\mathbf{H}}\underline{\mathbf{s}}_{n,m}\|^2.\end{aligned}$$

Since a necessary condition for the minimizer is:

$$\frac{\partial}{\partial \hat{\mathbf{H}}} \sum_{n,m} p_{n,m}^{(k-1)} \|\mathbf{x}[n] - \hat{\mathbf{H}}\underline{\mathbf{s}}_{n,m}\|^2 = 2\sum_{n,m} p_{n,m}^{(k-1)} \mathbf{x}[n]\underline{\mathbf{s}}_{n,m}^H - 2\sum_{n,m} p_{n,m}^{(k-1)} \hat{\mathbf{H}}\underline{\mathbf{s}}_{n,m}\underline{\mathbf{s}}_{n,m}^H = \mathbf{0},$$

we get:

$$\hat{\mathbf{H}}^{(k)} = (\sum_{n,m} p_{n,m}^{(k-1)} \mathbf{x}[n]\underline{\mathbf{s}}_{n,m}^H)(\sum_{n,m} p_{n,m}^{(k-1)} \underline{\mathbf{s}}_{n,m}\underline{\mathbf{s}}_{n,m}^H)^{-1}.$$

After we get the new channel estimation $\hat{\mathbf{H}}^{(k)}$, goto step 3(a) and continue the iteration.

CHAPTER 5

ITERATIVE PER-SURVIVOR PROCESSING ALGORITHM

As is well known, Viterbi decoding can be used to implement maximum likely sequence detection in ISI channels when the channel information is known perfectly by the receiver [11]. In our system, the channel information is unknown though we have the initial estimated channel information from the blind sub-space algorithm. So the Viterbi algorithm is not directly applicable here. An alternative is to use the generalized per-survivor processing (PSP) receiver [11]. Using PSP, we can update our estimated channel information at every stage when we search for the most likely sequence.

5.1 MLSE with perfect CSI

Recall model (1.2) in Section 1.2, repeated below for convenience,

$$\begin{bmatrix} \mathbf{x}[0] & \cdots & \mathbf{x}[N-1] \end{bmatrix} = \mathbf{H} \underbrace{\begin{bmatrix} \mathbf{s}[0] & \cdots & \mathbf{s}[N-1] \\ \vdots & \ddots & \vdots \\ \underbrace{\mathbf{s}[-N_h+1]}_{\underline{\mathbf{s}}[0]} & \cdots & \underbrace{\mathbf{s}[N-N_h]}_{\underline{\mathbf{s}}[N-1]} \end{bmatrix}}_{\mathcal{S}_{N_h}} + \mathbf{W}.$$

Note that we can also write our model as:

$$\mathbf{x}[k] = \mathbf{H}\underline{\mathbf{s}}[k] + \mathbf{w}[k] \qquad k = 0, \cdots, N-1.$$

Given the perfect channel information $\mathbf{H}$, the probability density function of the received data conditioned on the transmitted block code sequence $\mathbf{S}$ is:

$$p(\mathbf{X}|\mathbf{S}) = \frac{1}{(\pi\delta_w^2)^N}\Pi_{k=0}^{N-1} e^{-\frac{||\mathbf{x}[k]-\mathbf{H}\underline{\mathbf{s}}[k]||^2}{\delta_w^2}}.$$

Taking the logarithm of the probability above, we obtain:

$$\log(p(\mathbf{X}|\mathbf{S})) = C_6 - \sum_{k=0}^{N-1} \frac{||\mathbf{x}[k] - \mathbf{H}\underline{\mathbf{s}}[k]||^2}{\delta_w^2},$$

where C_6 is a constant. The maximum likelihood detection of the transmitted sequence is:

$$\hat{\mathbf{S}} = \arg\min_{\mathbf{S}\in\mathcal{U}} \sum_{k=0}^{N-1} ||\mathbf{x}[k] - \mathbf{H}\underline{\mathbf{s}}[k]||^2. \tag{5.1}$$

Since the channel is of length N_h and the block codes are of length N_t, each column vector $\underline{\mathbf{s}}[k]$ spans up to $M = \lfloor \frac{N_h}{N_t} \rfloor + 1$ codewords. If the channel information $\mathbf{H}$ is perfectly known, the optimum receiver is a Viterbi decoder that searches for the path with minimum metric in the trellis diagram of a finite state machine.

Assume for simplicity as stated before, the transmitted signal vectors with minus index can be viewed as guard signals which are $[\mathbf{s}[-N_h+1], \cdots, \mathbf{s}[-1]] = \mathbf{0}$, so we can group the signal vectors in a frame $\mathbf{S} = [\mathbf{s}[0], \cdots, \mathbf{s}[N-1]]$ into $N_c = \lfloor \frac{N}{N_t} \rfloor$ DUST codewords, i.e., $\mathbf{S} = (\mathbf{S}[0], \cdots, \mathbf{S}[N_c - 1])$. Then divide the block-Toeplitz matrix $\mathcal{S}_{N_h}$ to N_c block columns, $\mathcal{S}_{N_h} = (\underline{\mathbf{S}}[0], \cdots, \underline{\mathbf{S}}[N_c - 1])$, each block column having N_t column vectors. In other words, the n-th block column $\underline{\mathbf{S}}[n]$ contains the column vectors $(\underline{\mathbf{s}}[nN_t], \cdots, \underline{\mathbf{s}}[(n+1)N_t - 1])$. Divide the observed data matrix $\mathbf{X}$ in the same way into N_c blocks, $\mathbf{X} = (\underline{\mathbf{X}}[0], \cdots, \underline{\mathbf{X}}[N_c - 1])$ with the n-th block represented as $\underline{\mathbf{X}}[n]$. Then the maximum likelihood criteria from (5.1) can be restated as:

$$\hat{\mathbf{S}} = \arg\min_{\mathbf{S}\in\mathcal{U}} \sum_{n=0}^{N_c-1} ||\underline{\mathbf{X}}[n] - \mathbf{H}\underline{\mathbf{S}}[n]||_F^2. \tag{5.2}$$

Define the state of trellis diagram as:

$$\mu_n = \left[\hat{\mathbf{S}}[n], \cdots, \hat{\mathbf{S}}[n-M+1]\right], \tag{5.3}$$

where M is the channel response duration in terms of code blocks. So there are L^M possibilities for μ_n. The transition of states can be represented as $\mu_n \rightarrow \mu_{n+1}$. The transition metric at step n is defined as:

$$\lambda_v(\mu_n \rightarrow \mu_{n+1}) = ||\underline{\mathbf{X}}[n+1] - \mathbf{H}\underline{\hat{\mathbf{S}}}[n+1]||_F^2, \tag{5.4}$$

where state $\mu_{n+1} = \left[\hat{\mathbf{S}}[n+1], \cdots, \hat{\mathbf{S}}[n+2-M]\right]$ shares $\left[\hat{\mathbf{S}}[n], \cdots, \hat{\mathbf{S}}[n+2-M]\right]$ in common with μ_n. Let $M_v(\mu_n)$ denote the survivor metric as in the standard Viterbi algorithm. The accumulated metric $M_v(\mu_{n+1})$ is determined by performing a minimization over the set of states transitioning to μ_{n+1}:

$$M_v(\mu_{n+1}) = \min_{\mu_n}[M_v(\mu_n) + \lambda_v(\mu_n \rightarrow \mu_{n+1})]. \tag{5.5}$$

By choosing the trellis path with the minimized metric, we can achieve the maximum likelihood sequence detection of (5.2).

5.2 PSP for imperfect CSI

When $\mathbf{H}$ is unknown, a per-survivor estimation of $\mathbf{H}$ can be implemented. Recall the state μ_n at step n from (5.3). Since $\mathbf{H}$ is unknown, the branch metric in (5.4) is modified as:

$$\lambda_p(\mu_n \rightarrow \mu_{n+1}) = ||\underline{\mathbf{X}}[n+1] - \hat{\mathbf{H}}\underline{\hat{\mathbf{S}}}[n+1]||_F^2, \tag{5.6}$$

which means λ_p is also a function of estimate $\hat{\mathbf{H}}$. Note that if $\mathbf{H}$ is known, (5.6) reduces to the metric (5.4). The codeword sequence associated with each surviving path is used as a training sequence for the per-survivor estimation of $\mathbf{H}$. Define the codeword sequence associated with the surviving path terminating in state μ_n as $\{\hat{\mathbf{S}}[k](\mu_n)\}_{k=0}^{n}{}^{SV}$. Define the data-aided channel estimator as $G[\cdot]$ and the per-survivor estimation of $\mathbf{H}$ as:

$$\hat{\mathbf{H}}(\mu_n)^{SV} = G[\{\underline{\mathbf{X}}[k]\}_{k=0}^{n}, \{\hat{\mathbf{S}}[k](\mu_n)\}_{k=0}^{n}{}^{SV}].$$

The per-survivor estimate $\hat{\mathbf{H}}(\mu_n)^{SV}$ is then inserted in the computation of the branch metric (5.6):

$$\lambda_p(\mu_n \to \mu_{n+1}) = ||\underline{\mathbf{X}}[n+1] - \hat{\mathbf{H}}(\mu_n)^{SV}\hat{\underline{\mathbf{S}}}[n+1]||_F^2.$$

We then find the survivor metric $M_p(\mu_{n+1})$ similar to (5.5) which is:

$$M_p(\mu_{n+1}) = \min_{\mu_n}[M_p(\mu_n) + \lambda_p(\mu_n \to \mu_{n+1})], \tag{5.7}$$

and continue the process until $n = N_c - 1$.

Note that when a survivor is correct, the corresponding estimate $\hat{\mathbf{H}}$ is computed using the correct data sequence. Assuming the data-aided estimator $G[\cdot]$ has the property that it can perfectly estimate $\mathbf{H}$ given the correct codeword sequence in the absence of noise, then PSP will detect $\mathbf{S}$ in the absence of noise. For this reason, PSP is asymptotically optimal as SNR increases [11]. Adaptive algorithms such as Least Mean Square (LMS) and Recursive Least Square (RLS) can be used to implement $G[\cdot]$. We will discuss LMS and RLS based PSP in detail in the next two subsections. Table 5.1 lists the notation used for the PSP algorithm.

5.2.1 PSP using LMS

LMS is proposed in [11] to accomplish the channel identification component of PSP sequence decoding. LMS is a linear adaptive filtering algorithm based on two

Variable	Description
μ_{n+1}	one of L^M states at step $n+1$
$\mu_n \to \mu_{n+1}$	path transition from μ_n to μ_{n+1}
μ_{n+1}^{SV}	surviving path connected to μ_{n+1}
$\lambda_p(\mu_n \to \mu_{n+1})$	branch metric corresponding to transition $\mu_n \to \mu_{n+1}$
$M(\mu_{n+1})$	surviving path metric connected to state μ_{n+1}
$\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=0}^{n+1}$	tentative decisions of the DUST codes connected to the state μ_{n+1}
$\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=0}^{n+1^{SV}}$	surviving path connected to the state μ_{n+1}
$\underline{\hat{\mathbf{S}}}(\mu_{n+1})$	block columns constructed from the tentative decisions $\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=n-M+2}^{n+1}$
$\underline{\hat{\mathbf{S}}}(\mu_{n+1})^{SV}$	block column constructed from the surviving path $\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=0}^{n+1^{SV}}$ connected to the state μ_{n+1}
$\underline{\mathbf{E}}(\mu_{n+1})$	error between the received signal and its estimation along transition $\mu_n \to \mu_{n+1}$
$\underline{\mathbf{E}}(\mu_{n+1})^{SV}$	error between the received signal and its estimation along transition of the surviving path connected to μ_{n+1}
$\mathbf{K}(\mu_{n+1})^{SV}$	gain of the surviving path connected to the state μ_{n+1}
$\mathbf{P}(\mu_{n+1})^{SV}$	inverse of the correlation matrix of the surviving path connected to the state μ_{n+1}
$\hat{\mathbf{H}}(\mu_{n+1})^{SV}$	channel estimation for the surviving path transition connected to the state μ_{n+1}

Table 5.1: Parameter and description for PSP algorithm

steps: first, compute a filtered output and generate the error between the output and the desired response; second, adjust the filter according to the output error [12]. We use a single-input single-output (SISO) model to further describe LMS. Let $\mathbf{f}$ denote a vector of FIR channel response coefficients, $t[n]$ as the input, $\hat{\mathbf{f}}[n]$ as the estimate of $\mathbf{f}$, $\hat{r}[n]$ as the filtered output, $r[n]$ as the desired output and $e[n]$ as the error. Then briefly, LMS can be written as:

1. Generate output $\hat{r}[n] = \hat{\mathbf{f}}^H[n]\mathbf{t}[n]$ and estimation error $e[n] = r[n] - \hat{r}[n]$,
2. Update the channel estimate $\hat{\mathbf{f}}[n+1] = \hat{\mathbf{f}}[n] + \beta\mathbf{t}[n]e^H[n]$.

β, a positive constant, is the step-size parameter. The iterative procedure starts with an initial estimate $\hat{\mathbf{f}}[0]$.

In our system, the unknown channel coefficients are contained in $\mathbf{H}$. Suppose the tentative decision for the code sequence associated with the transition $\mu_n \rightarrow \mu_{n+1}$ is the codeword sequence $\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=0}^{n+1}$. Arrange this data sequence into the form of block column $\underline{\hat{\mathbf{S}}}(\mu_{n+1})$ having the same structure as $\underline{\mathbf{S}}[n+1]$. Then the PSP based on LMS channel identification proceeds in similar way as in step 1 of LMS: for all the transitions $\mu_n \rightarrow \mu_{n+1}$, calculate the errors,

$$\underline{\mathbf{E}}(\mu_{n+1}) = \underline{\mathbf{X}}[n+1] - \hat{\mathbf{H}}(\mu_n)^{SV}\underline{\hat{\mathbf{S}}}(\mu_{n+1}). \tag{5.8}$$

The transition metric is:

$$\lambda_p(\mu_n \rightarrow \mu_{n+1}) = ||\underline{\mathbf{E}}(\mu_{n+1})||_F^2. \tag{5.9}$$

The surviving metric $M_p(\mu_{n+1})$ is calculated as in (5.7). The surviving path $\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=0}^{n+1}{}^{SV}$ connected to the state μ_{n+1} is the tentative decision of code sequence which has the surviving metric $M_p(\mu_{n+1})$. Next the channel estimation for state μ_{n+1} is updated in similar way as in step 2 of LMS,

$$\hat{\mathbf{H}}(\mu_{n+1})^{SV} = \hat{\mathbf{H}}(\mu_n)^{SV} + \beta\underline{\mathbf{E}}(\mu_{n+1})^{SV}\hat{\underline{\mathbf{S}}}^H(\mu_{n+1})^{SV}. \tag{5.10}$$

The updated estimation $\hat{\mathbf{H}}(\mu_{n+1})^{SV}$ is computed for each surviving path $\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=0}^{n+1}{}^{SV}$.

The PSP sequence decoder based on LMS channel identification is summarized below.

PSP using LMS

1. Start with an initial estimation $\hat{\mathbf{H}}^{(0)}$.

2. $n = n + 1$, $0 \leq n \leq N_c - 1$,

   (a) For each state μ_{n+1}, find the groups $\{\mu_n\}$ that can be connected to state μ_{n+1}.

   (b) Find the tentative decisions of the DUST codes $\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=0}^{n+1}$ along the transition $\mu_n \rightarrow \mu_{n+1}$.

   (c) Use the codes $\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=n-M+2}^{n+1}$ from the tentative decisions above to construct block column $\hat{\underline{\mathbf{S}}}(\mu_{n+1})$.

   (d) Find the block column error between the actual received signal and the desired response approximated on $\hat{\mathbf{H}}(\mu_n)^{SV}$,

   $$\underline{\mathbf{E}}(\mu_{n+1}) = \underline{\mathbf{X}}[n+1] - \hat{\mathbf{H}}(\mu_n)^{SV}\hat{\underline{\mathbf{S}}}(\mu_{n+1}).$$

   (e) Find the branch metric from the error $\underline{\mathbf{E}}(\mu_{n+1})$,

   $$\lambda_p(\mu_n \rightarrow \mu_{n+1}) = ||\underline{\mathbf{E}}(\mu_{n+1})||_F^2.$$

(f) Find the surviving path metric connected to state μ_{n+1} using the criteria,

$$M_p(\mu_{n+1}) = \min_{\mu_n}[M_p(\mu_n) + \lambda_p(\mu_n \rightarrow \mu_{n+1})],$$

and keep the surviving path connected to μ_{n+1} as $\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=0}^{n+1}{}^{SV}$.

(g) Update the channel estimation using the errors and the block column constructed from the surviving path $\{\hat{\mathbf{S}}[k](\mu_{n+1})\}_{k=0}^{n+1}{}^{SV}$ connected to the state μ_{n+1},

$$\hat{\mathbf{H}}(\mu_{n+1})^{SV} = \hat{\mathbf{H}}(\mu_n)^{SV} + \beta\underline{\mathbf{E}}(\mu_{n+1})^{SV}\underline{\hat{\mathbf{S}}}(\mu_{n+1})^{SV}.$$

3. Find the minimum path metric $\min_{\mu_{N_c-1}} M_p(\mu_{N_c-1})$ and the surviving path $\{\hat{\mathbf{S}}[k](\mu_{N_c-1})\}_{k=0}^{N_c-1}{}^{SV}$ which generate this minimum path metric. This is the output of PSP sequence decoder.

5.2.2 PSP using RLS

RLS is also proposed in [11] to accomplish the channel identification in PSP sequence decoding. RLS algorithm can be viewed as a special kind of Kalman filter [12]. Assume the same SISO model as in the description of LMS. In addition, define γ as a "forgetting factor". In the method of exponential weighed least squares, we want to minimize the cost function $\sum_{i=1}^{n} \gamma^{n-i}|e(i)|^2$. Defining $\mathbf{\Phi}(n)$ as the correlation matrix of the input signal $\mathbf{t}(n)$ and $\mathbf{p}(n) = \mathbf{\Phi}^{-1}(n)$ and using the Matrix Inversion Lemma [12], we obtain the RLS algorithm:

1. Initialize correlation matrix inverse $\mathbf{p}[0] = \mathbf{\Phi}[0] = (\mathrm{E}(\mathbf{t}[0]\mathbf{t}^H[0]))^{-1}$.

2. At $n = 1, 2, \cdots$, find:

   gain vector: $\mathbf{k}[n] = \frac{\gamma^{-1}\mathbf{p}[n-1]\mathbf{t}[n]}{1+\gamma^{-1}\mathbf{t}^H[n]\mathbf{p}[n-1]\mathbf{t}[n]}$, estimation error: $e[n] = r[n] - \hat{\mathbf{f}}^H[n-1]\mathbf{t}[n]$,

channel estimate: $\hat{\mathbf{f}}[n] = \hat{\mathbf{f}}[n-1] + \mathbf{k}[n]e^H[n]$,

correlation matrix inverse: $\mathbf{p}[n] = \gamma^{-1}\mathbf{p}[n-1] - \gamma^{-1}\mathbf{k}[n]\mathbf{t}^H[n]\mathbf{p}[n-1]$.

If we combine RLS channel estimation with PSP sequence decoder, $\hat{\mathbf{H}}(\mu_{n-1})^{SV}$ is estimated by recursively minimizing the exponentially weighted cost:

$$\sum_{k=0}^{N_c-1} \gamma^{N_c-1-k} ||\underline{\mathbf{X}}[k] - \hat{\mathbf{H}}(\mu_{N_c-1})^{SV} \hat{\underline{\mathbf{S}}}(\mu_k)^{SV}||_F^2, \tag{5.11}$$

where γ is the forgetting factor used to track possibly time-varying channels ($0 < \gamma \leq 1$). We outline PSP based on RLS below:

PSP using RLS

1. Start with the initial estimate $\hat{\mathbf{H}}^{(0)}$, $\hat{\mathbf{S}}^{(0)}$ and the inverse of the correlation matrix $\hat{\mathbf{P}}^{(0)} = (\hat{\mathcal{S}}_{N_h}^{(0)} \hat{\mathcal{S}}_{N_h}^{(0)H})^{-1}$.

2. $n = n+1$, $0 \leq n \leq N_c - 1$,

   (a) to (f) are the same as in Section 5.2.1.

   (g) Update the gain of the surviving path connected to state μ_{n+1},

$$\mathbf{K}(\mu_{n+1})^{SV} = \frac{\mathbf{P}(\mu_n)^{SV} \hat{\underline{\mathbf{S}}}(\mu_{n+1})^{SV}}{\hat{\underline{\mathbf{S}}}^H(\mu_{n+1})^{SV} \mathbf{P}^n(\mu_n)^{SV} \hat{\underline{\mathbf{S}}}(\mu_{n+1})^{SV} + \gamma \mathbf{I}},$$

   Update the inverse of the correlation matrix of the surviving path connected to state μ_{n+1},

$$\mathbf{P}(\mu_{n+1})^{SV} = \gamma^{-1} \left[\mathbf{I} - \mathbf{K}(\mu_{n+1})^{SV} (\hat{\underline{\mathbf{S}}}^H(\mu_{n+1})^{SV}) \right] \mathbf{P}(\mu_n)^{SV},$$

   Update the channel estimation using the errors and gain of the surviving path connected to μ_{n+1},

$$\hat{\mathbf{H}}(\mu_{n+1})^{SV} = \hat{\mathbf{H}}(\mu_n)^{SV} + \underline{\mathbf{E}}(\mu_{n+1})^{SV} \mathbf{K}(\mu_{n+1})^{SV}.$$

3. Find the minimum path metric $M_p(\mu_{N_c-1})$ and the surviving path $\{\hat{\mathbf{S}}[k](\mu_{N_c-1})\}_{k=0}^{N_c-1}{}^{SV}$ which generate this minimum path metric. This is the output of PSP sequence decoder.

5.3 Iterative PSP Sequence Estimation

According to the ML criteria (4.4) derived in Chapter 4, the optimal estimation of the codewords is obtained from:

$$\hat{\mathbf{S}} = \arg\min_{\mathbf{H},\mathbf{S}\in\mathcal{U}} \|\mathbf{X} - \mathbf{H}\mathcal{S}_{N_h}\|_F^2,$$

which is a minimization over $\mathbf{H}$ and $\mathbf{S}$. If we rewrite the above equation as:

$$\hat{\mathbf{S}} = \arg\min_{\mathbf{S}\in\mathcal{U}} \min_{\mathbf{H}} \left(\sum_{n=0}^{N_c-1} ||\underline{\mathbf{X}}[n] - \mathbf{H}\underline{\mathbf{S}}[n]||_F^2\right). \tag{5.12}$$

We can do the optimization iteratively if given an initial estimation $\hat{\mathbf{S}}^{(0)}$. In our system, the initial estimate $\hat{\mathbf{S}}^{(0)}$ is obtained using blind sub-space algorithm and the non-coherent decoder for DUST codes. Using the inner minimization in (5.12), the initial estimate $\hat{\mathbf{H}}^{(0)}$ is obtained from least square method: $\hat{\mathbf{H}}^{(0)} = \mathbf{X}\hat{\mathcal{S}}_{N_h}^{(0)\dagger}$, which gives ML estimate $\hat{\mathbf{H}}^{(0)}$ given $\hat{\mathbf{S}}^{(0)}$. $\hat{\mathbf{H}}^{(0)}$ in turn suggests an updated estimation $\hat{\mathbf{S}}^{(1)}$ and we can use PSP based on LMS or RLS to get $\hat{\mathbf{S}}^{(1)}$. With $\hat{\mathbf{S}}^{(1)}$, the inner minimization gives an updated estimation $\hat{\mathbf{H}}^{(1)}$, and PSP works much better with the updated channel estimation $\hat{\mathbf{H}}^{(1)}$. So we can use PSP in an iterative way as (5.12) suggested: after we get the output code sequence estimation $\hat{\mathbf{S}}^{(k)}$ from the kth time using PSP, least square estimation $\hat{\mathbf{H}}^{(k)} = \mathbf{X}\hat{\mathcal{S}}_{N_h}^{(k)\dagger}$ is obtained. We then send $\hat{\mathbf{H}}^{(k)}$ to the PSP sequence decoder again and get $\hat{\mathbf{S}}^{(k+1)}$. The iteration is stopped when the channel estimation $\hat{\mathbf{H}}^{(k)} = \hat{\mathbf{H}}^{(k+1)}$. Usually after two to three iterations, the algorithm stops. In some special cases, the estimation $\hat{\mathbf{H}}^{(k)}$ converges very slowly. To save complexity and avoid too many times of iteration, we can set the maximum number of iteration N_m as before. So we always have fewer than N_m iterations.

Our final blind equalization and identification algorithm for our MIMO differential space-time modulated systems can be summarized below:

1. Obtain the initial block code sequence estimation $\hat{\mathbf{S}}^{(0)}$ from blind sub-space method and non-coherent decor for DUST codes.

2. Get the initial channel estimation $\hat{\mathbf{H}}^{(0)} = \mathbf{X}\hat{\mathcal{S}}_{N_h}^{(0)\dagger}$ using least square method.

3. $k = k + 1$, $1 \le k \le N_m$,

   (a) Use $\hat{\mathbf{H}}^{(k-1)}$ in PSP algorithm and get $\hat{\mathbf{S}}^{(k)}$.

   (b) Least square estimation of channels: $\hat{\mathbf{H}}^{(k)} = \mathbf{X}\hat{\mathcal{S}}_{N_h}^{(k)\dagger}$.

   (c) If $\hat{\mathbf{H}}^{(k)} \neq \hat{\mathbf{H}}^{(k-1)}$, goto (a).

CHAPTER 6

CR BOUND ANALYSIS AND SIMULATION RESULTS

6.1 Constrained Cramér-Rao Bound

To evaluate the effect of the iterative PSP algorithm we proposed, we want to find the bound on MIMO channel estimation error with side information. Here we implement the method of computing the constrained CR bound introduced by Sadler, et al. [13]. The side information for our blind channel estimation is the structure of the DUST codewords. To simplify the derivation process for the constrained CR bound, we will use most of the conclusions in [13]. For proof details, please refer to [13], [14].

First, we transform our MIMO linear system model introduced in Chapter 1 to an equivalent model described in [13] and then we use the results derived in [13] directly. With the model equation (1.1), the channel response $\mathbf{H}[k] \in \mathbb{C}^{N_r N_o \times N_t}$, $k = 0, \cdots, N_h - 1$, can be written as:

$$\mathbf{H}[k] = \begin{bmatrix} c_{1,1}[k] & \cdots & c_{1,N_t}[k] \\ \vdots & \ddots & \vdots \\ c_{N_r N_o,1}[k] & \cdots & c_{N_r N_o,N_t}[k] \end{bmatrix}.$$

Assume that $s_k[n]$ denotes the kth element of the transmitted signal vector $\mathbf{s}[n]$, $x_i[n]$ denotes the ith element of the received signal vector $\mathbf{x}[n]$, $w_i[n]$ denotes the ith element of the noise vector $\mathbf{w}[n]$, $i = 1, \cdots, N_r N_o$. Rearranging the MIMO model (1.1) we get,

$$x_i[n] = \sum_{k=1}^{N_t} \sum_{l=0}^{N_h-1} c_{i,k}[l] s_k[n-l]. \tag{6.1}$$

If we take the ith element of all the received vectors $\mathbf{x}[0], \cdots, \mathbf{x}[N-1]$, and stacking them into a vector: $\mathbf{x}_i = [x_i[0], \cdots, x_i[N-1]]^T$, take the ith element of all the noise vectors $\mathbf{w}[0], \cdots, \mathbf{w}[N-1]$, and stacking them into a vector: $\mathbf{w}_i = [w_i[0], \cdots, w_i[N-1]]^T$, then from (6.1) we get,

$$\begin{aligned}
\mathbf{x}_i &= \sum_{k=1}^{N_t} \begin{bmatrix} c_{i,k}[N_h-1] & & \cdots & c_{i,k}[0] & \\ & & \ddots & & \ddots & \\ & & c_{i,k}[N_h-1] & \cdots & c_{i,k}[0] \end{bmatrix}_{N \times (N+N_h-1)} \begin{bmatrix} s_k[-N_h+1] \\ \vdots \\ s_k[N] \end{bmatrix} + \mathbf{w}_i \\
&= \sum_{k=1}^{N_t} \mathbf{C}_{i,k} \mathbf{s}_k + \mathbf{w}_i.
\end{aligned}$$

If we define $\mathbf{x} = [\mathbf{x}_1^T, \cdots, \mathbf{x}_{N_r N_o}^T]^T$, $\mathbf{s}_k = [s_k[-N_h+1], \cdots, s_k[N-1]]^T$, $\mathbf{w} = [\mathbf{w}_1^T, \cdots, \mathbf{w}_{N_r N_o}^T]^T$, the system model can be written as:

$$\begin{aligned}
\mathbf{x} &= \sum_{k=1}^{N_t} \begin{bmatrix} \mathbf{C}_{1,k} \\ \vdots \\ \mathbf{C}_{N_r N_o,k} \end{bmatrix} \mathbf{s}_k + \mathbf{w} \\
&= \sum_{k=1}^{N_t} \mathbf{C}_k \mathbf{s}_k + \mathbf{w} \qquad (6.2)
\end{aligned}$$

This is an equivalent model as (5) in [13], which is a MIMO model with N_t users and $N_r N_o$ channels.

We may use the conclusions in [13] now. Define the complex vector of unknown parameter (channel response and symbols) as (15) in [13]:

$$\Theta = \left[\mathbf{c}_1^T, \mathbf{s}_1^T, \cdots, \cdots, \mathbf{c}_{N_t}^T, \mathbf{s}_{N_t}^T\right]^T, \tag{6.3}$$

where

$$\mathbf{c}_k = [\mathbf{c}_{1,k}^T, \cdots, \mathbf{c}_{N_r N_o,k}^T], \quad \mathbf{c}_{i,k} = [c_{i,k}[0], \cdots, c_{i,k}[N_h-1]]^T.$$

The mean of $\mathbf{x}$ conditioned on $\mathbf{C}_k$ and $\mathbf{s}_k$ from (6.2) is:

$$\mu(\Theta) = \sum_{k=1}^{N_t} \mathbf{C}_k \mathbf{s}_k. \tag{6.4}$$

The covariance matrix of $\mathbf{x}$ conditioned on $\mathbf{C}_k$ and $\mathbf{s}_k$ is $\sigma_w^2 \mathbf{I}$. From (17) in [13], we get complex-valued Fisher Information matrix:

$$\mathbf{J}_c = \frac{2}{\sigma_w^2} (\frac{\partial \mu(\Theta)}{\partial \Theta})^H \frac{\partial \mu(\Theta)}{\partial \Theta}. \tag{6.5}$$

Define:

$$\left[\frac{\partial \mu(\Theta)}{\partial \Theta}\right]_{ij} = \frac{\partial [\mu(\Theta)]_i}{\partial [\Theta]_j},$$

where $[\mu(\Theta)]_i$ means the ith element of $\mu(\Theta)$ and $[\Theta]_j$ means the $j$the element of Θ. From (11) and (12) in [13], we get,

$$\frac{\partial \mu(\Theta)}{\partial \Theta} = [\mathbf{Q}_{\mathbf{1}}, \cdots, \mathbf{Q}_{N_t}] \tag{6.6}$$

$$\mathbf{Q}_k = [\mathbf{I}_{N_r N_o} \otimes \mathcal{S}^{(k)}, \mathbf{C}_k] \qquad k = 1, \cdots, N_t, \tag{6.7}$$

where $\mathbf{I}_{N_r N_o}$ is the $N_r N_o \times N_r N_o$ identity matrix, $\otimes$ denotes the Kronecker product, and

$$\mathcal{S}^{(k)} = \begin{bmatrix} s_k[0] & \cdots & s_k[-N_h+1] \\ \vdots & \ddots & \vdots \\ s_k[N-1] & \cdots & s_k[N-N_h+1] \end{bmatrix} \qquad k = 1, \cdots, N_t. \tag{6.8}$$

So the complex Fisher information matrix in (6.5) can be rewritten as:

$$\mathbf{J}_c = \frac{2}{\sigma_w^2} \begin{bmatrix} \mathbf{Q}_1^H \mathbf{Q}_1 & \cdots & \mathbf{Q}_1^H \mathbf{Q}_{N_t} \\ \vdots & \ddots & \vdots \\ \mathbf{Q}_{N_t}^H \mathbf{Q}_1 & \cdots & \mathbf{Q}_{N_t}^H \mathbf{Q}_{N_t} \end{bmatrix}. \tag{6.9}$$

Define the real parameter vector as:

$$\xi = [\mathrm{Re}(\Theta)^T, \mathrm{Im}(\Theta)^T]^T. \tag{6.10}$$

The real-valued FIM corresponding to real valued unknown parameter ξ in (6.10) is:

$$\mathbf{J}_r = 2 \begin{bmatrix} \mathrm{Re}(\mathbf{J}_c) & -\mathrm{Im}(\mathbf{J}_c) \\ \mathrm{Im}(\mathbf{J}_c) & \mathrm{Re}(\mathbf{J}_c) \end{bmatrix}. \tag{6.11}$$

Now consider our side information from the diagonal structure of the DUST codewords. For any codeword:

$$\mathbf{S}[n] = \begin{bmatrix} s_{1,1}[n] & \cdots & s_{1,N_t}[n] \\ \vdots & \ddots & \vdots \\ s_{N_t,1}[n] & \cdots & s_{N_t,N_t}[n] \end{bmatrix},$$

all the diagonal elements are unit modulus, $|s_{k,k}[n]| = 1$, and all the off-diagonal elements equals 0. Using this, we can get $R = N_c N_t^2$ equality constraints with the form:

$$s_{k,j\neq k}[n] = 0 \quad \text{and} \quad |s_{k,k}[n]| - 1 = 0 \;\; \text{for } j,k = 1,\cdots,N_t, n = 0,\cdots,N_c - 1.$$

Suppose the dimension of ξ is D, then define a $R \times D$ gradient matrix

$$F(\xi) = \frac{\partial \mathbf{f}(\xi)}{\partial \xi}. \tag{6.12}$$

where $\mathbf{f}(\xi)$ collects the R equality constraints. Now define F equals to $F(\xi_o)$ where ξ_o is the true value of the parameter vector. Let U be a $D \times (D - R)$ matrix whose columns are an orthonormal basis for the null space of F, so that $FU = 0$, $U^T U = \mathbf{I}$, then the constrained CR bound is:

$$E[(\hat{\xi} - \xi_o)(\hat{\xi} - \xi_o)^T] \geq U(U^T \mathbf{J}_r U)^{-1} U^T. \tag{6.13}$$

From (6.13), we can compute the channel estimation error $||\mathbf{H} - \hat{\mathbf{H}}||_F^2$ and compare it with the estimation error from the iterative PSP algorithm. We have done simulations for some specific cases in the next section to evaluate the performance of our algorithms.

6.2 Simulation results

The basic problem for this project is that in the MIMO system with frequency-selective channel response, if we use DUST codewords, how to blindly estimate the codeword sequence and the channel response. The blind equalization and identification algorithm we present mainly contains two steps: first, find the initialization estimation of the code sequence using blind sub-space algorithm and the non-coherent decoder for DUST codewords; second, use the initialization estimation to aid further estimation of the code sequence and channel response. As to the second step, we consider two methods, one is the ILSP and soft ILSP introduced in chapter 4, the other is the iterative PSP algorithm introduced in chapter 5. For the iterative PSP algorithm, there are two types: the iterative PSP using LMS and the iterative PSP using RLS.

For the first group of simulation, we compare the effect of the bit error rate (BER) and the frame error rate (FER) of all our blind algorithms. We also give the curve for the known channel response case (non-blind). For the non-blind case, the optimal decoder is the maximum likelihood sequence decoder. We set the parameters for the simulation as: $N_t = 2$ transmit antennas, $N_r = 2$ receiving antennas, up-sampling rate for the received signal $N_o = 2$, number of frequency selective channel taps is $N_h = 3$. The channels are generated as multi-ray channels with pulse shaping. Every frame contains $N_c = 51$ codewords. The step size β for the iterative PSP on LMS is 0.2. The forgetting factor γ for the iterative PSP on RLS is 0.8. The size of group codewords is $L = 4$. They are diagonal and unitary matrices from [4]:

$$\mathbf{S}[n] \in \left[\begin{bmatrix} j & 0 \\ 0 & -j \end{bmatrix}, \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \begin{bmatrix} -j & 0 \\ 0 & j \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right].$$

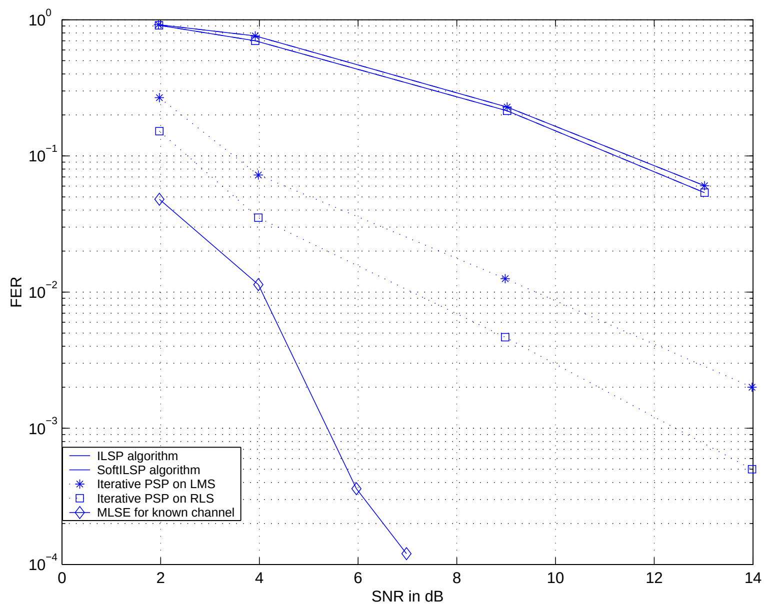

Figure 6.1: FER comparison of different algorithms

Figure 6.1 gives the simulation results for the FER versus SNR of all the algorithms proposed. Frame error rate is computed as the number of frames in which all the codewords are recovered correctly over the total number of frames for experiments.

Figure 6.2 gives the simulation results for the BER versus SNR. From these two figures, we can see that the iterative PSP algorithm is better than soft ILSP and ILSP algorithm. The iterative PSP on RLS is better than iterative PSP on LMS. Since PSP on LMS is much simpler, the complexity of PSP on RLS is the expense for its increase of performance. But there is still difference between the performance in the non-blind case and our blind case. Theoretically the BER and FER of blind case should be higher than the non-blind case. To evaluate how good our iterative PSP algorithms performs in the blind case, we give the constrained CR bound simulation as a comparison.

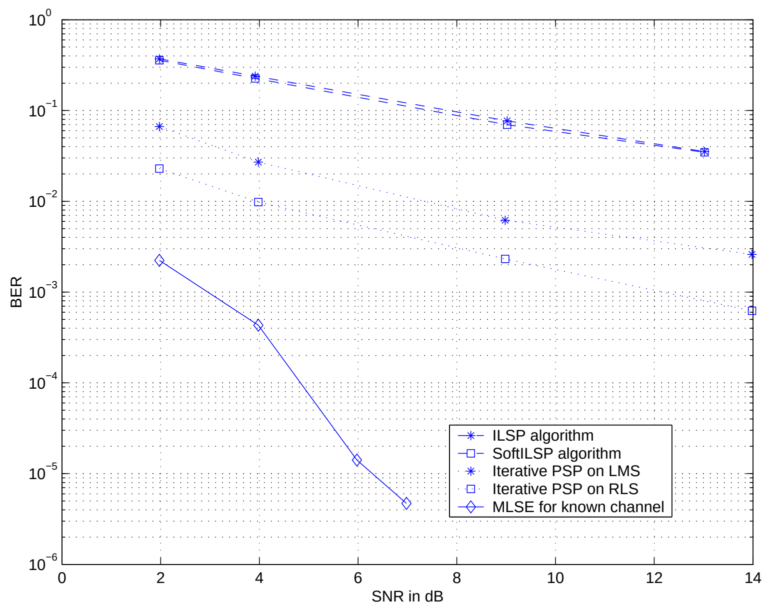

Figure 6.2: BER comparison of different algorithms

Figure 6.3 shows the CR bound for channel estimation error $||\mathbf{H}-\hat{\mathbf{H}}||_F^2$ and channel estimation error from iterative PSP on RLS algorithm and channel estimation error from the initialization estimation of the blind sup-space algorithm. From this plot, we can see that the iterative PSP on RLS algorithm based on initialization from sub-space method is a good way of blind equalization and identification for our MIMO system. Although it can not achieve the constrained CR bound, it is approaching the CR bound especially in high SNR case. We can also see that the initialization channel estimation from the blind sub-space algorithm does not perform very well in the noisy case.

We also investigate the effect of the number of the receiving antennas, the number of the over-sampling rate and the frame length to our iterative PSP on RLS algorithms. Figure 6.4 shows the effect of the number of receiving antenna to the

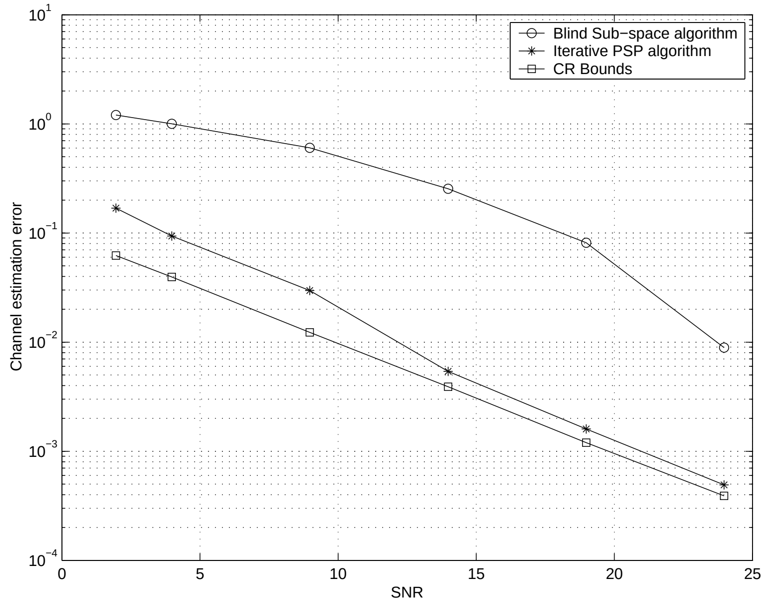

Figure 6.3: Channel Estimation Error Comparison

iterative PSP algorithm. We keep all the parameters the same as those of the first group of simulation except changing N_r from 2 to 4. When we increase the number of antennas, the performance becomes much better.

Figure 6.5 shows the effect of the up-sampling rate. We keep all the parameters the same as those of the first group simulation except changing the up-sampling rate N_o. If there is no up-sampling, then $N_o = 1$. We use $N_o = 2$ as the default up-sampling rate in our algorithm. The plot shows that when the up-sampling rate is 2, it's much better than no up-sampling case.

Figure 6.6 shows the effect of the frame length to the iterative PSP on RLS algorithm. We keep all the parameters the same as those of the first group of simulation except changing the frame length from $N_c = 51$ to 25. And the plot shows the longer the frame length, the better the performance. This is in accordance with our intuition,

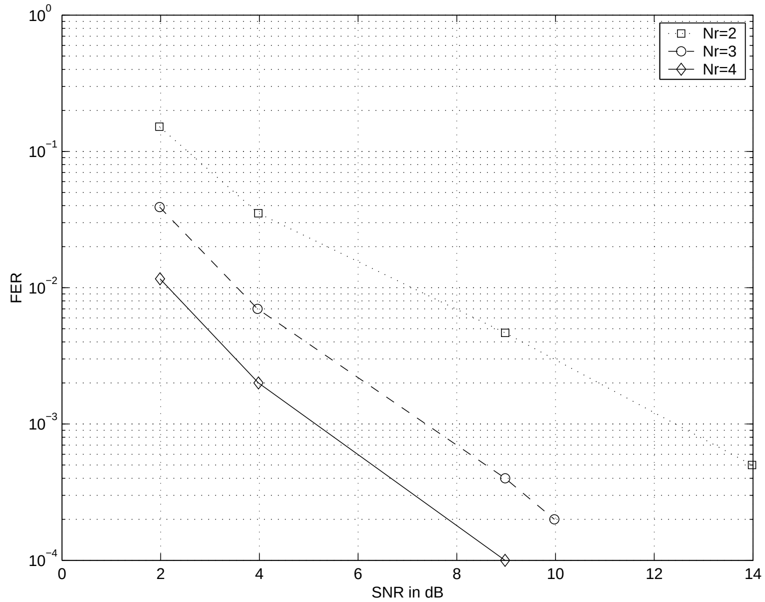

Figure 6.4: Effect of different number of receiver to the algorithm

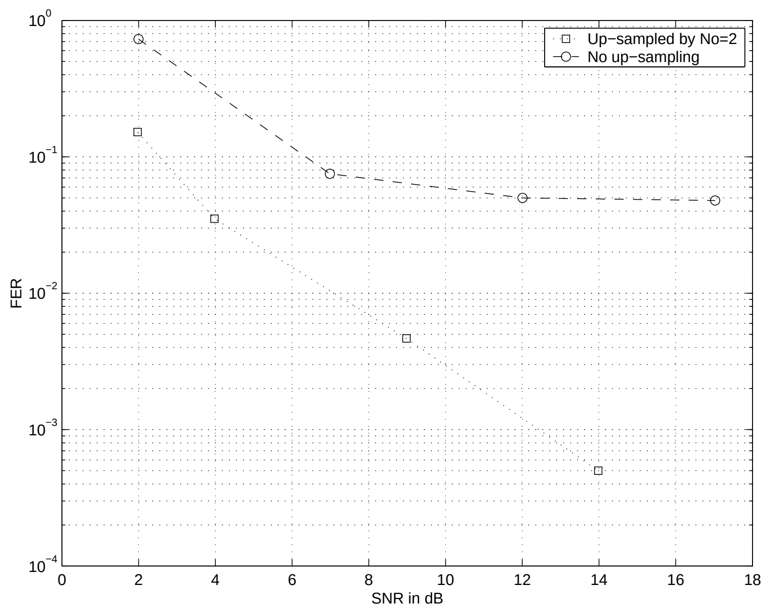

Figure 6.5: Effect of up-sampling to the algorithm

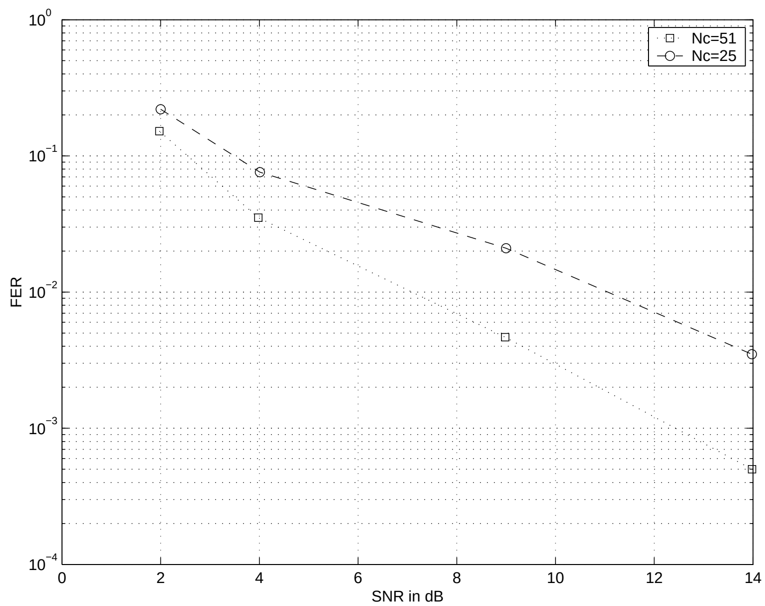

Figure 6.6: Effect of different frame length to the algorithm

since the longer the frame, the algorithm has more chances to *learn* the channels. For small length of 25 codewords in a frame, we can still blindly identify the channels and estimate the transmit codewords using this algorithm.

6.3 Conclusion

This thesis presents an approach of blind equalization and identification for MIMO communication system with frequency-selective fading channels. The blind sub-space algorithm plus the non-coherent decoder for the DUST codewords gives a blind equalization as initialization. This scheme works perfect in the absence of noise because the deterministic subspace method gives perfect results for the ideal case. But when noise is added, the deterministic subspace method gives an estimate with great noise, so the initialization estimation of both the channels and codeword sequence contains great noise.

To improve the accuracy of our blind algorithm, ILSP and soft ILSP are considered for further estimation of the channels and symbols. These approaches are based on projection, since the DUST codewords are block codewords in a group with finite alphabet, we can project every codeword in a frame to the group codewords. But ILSP and soft ILSP does not improve the performance as we hoped. The reason might be that the initialization estimation from the sub-space method is not accurate enough.

Iterative PSP on LMS or RLS based on sequence detection generalized for MIMO system is considered also. Although the PSP algorithm is sub-optimal, this approach gives great improvement in performance. Constrained CR bound are theoretically and computationally derived to evaluate the performance of the iterative PSP on RLS algorithm. Simulations show that it works well since it is approaching the constrained CR bound especially in high SNR case.

Generally speaking, we present an approach of blind identification and equalization for the differential space-time coded wide-band MIMO communication system. We also investigated some properties of the algorithm, such as the effect of the number of receive antennas and the number of block codewords in a frame. The simulation results are in consistent with what we derived theoretically. We showed the importance of over-sampling for the system. The blind sub-space algorithm is making use of over-sampled output and the initialization estimation from the sub-space algorithm is crucial to the iterative PSP algorithm.

There are still some limits for our algorithm. For example, this scheme is only designed for small number of taps of channel response because the complexity for the iterative PSP grows exponentially with the number of taps. How to solve the problem of longer taps of channel response can be further research topics. Another problem is that, after the sub-space method, we get an estimation of the symbols with an ambiguity matrix plus some additional noise. The property of the noise influences the non-coherent decoder we are using for the DUST code. How to analyze the property of the noise from the sub-space method may be further studied. Since the iterative PSP works better with better initialization, how to improve the accuracy of initial estimation from the blind sub-space method may need further investigation. Besides, if some other space-time codewords other than DUST code is employed, how to accomplish the blind equalization and identification for wide-band MIMO systems are broad topics for further research.

BIBLIOGRAPHY

[1] A. J. van der Veen, "An analytical constant modulus algorithm", *IEEE Trans. on Signal Processing*, vol. 44, no. 5, pp. 1136-1155, May 1996.

[2] A. J. van der Veen, S. Talwar, and A. Paulraj, "A subspace approach to blind space-time signal processing for wireless communication systems", *IEEE Trans. on Signal Processing*, vol. 47, no. 3, pp. 856-859, Mar. 1999.

[3] B. L. Hughes, "Differential space-time modulation", *IEEE Trans on infomation theory*, vol. 46, no. 7, Nov. 2000.

[4] B. M. Hochwald and W. Sweldens, "Diferential unitary space-time modulation", *IEEE Trans. on communications*, vol. 48, no. 12, pp. 2041-2052, Dec. 2000.

[5] H. Liu and G. Xu, "Closed-form blind symbol estimation in digital communication ", *IEEE Trans. on signal processing*, vol. 43, no. 11, pp. 2714-2723, Nov. 1995.

[6] V. Tarokh and N. Seshadri, "Space-time codes for high data rate wireless communication: performance criterion and code construction", *IEEE Trans. on information theory*, vol. 44, no. 2, Mar. 1998.

[7] B. M. Hochwald and T. Marzetta, "Unitary space-time modulation for multiple-antenna communication in Rayleigh flat-fading", *IEEE Trans. on information theory*, vol. 46, pp. 543-564, Mar. 2000.

[8] T. K. Moon, "The expectation-maximizaation algorithm," *IEEE Signal Processing Magazine,* pp. 47-60, Nov. 1996.

[9] S. Talwar, M. Viberg and A. Paulraj, "Blind estimation of multiple co-channel digital signals arriving at Antenna array", *IEEE Signal Processing Letters*, vol. 1, no. 2, Feb. 1994.

[10] G. Golub and V. Pereyra "The differentialtion of pseudo-interses and nonlinear least squares problems whose variables sepearate", *SIAM J. Num Anal.*, 10: 413-432, 1973.

[11] R. Raheli and A. Polydoros and C. Tzou "Per-survivor Processing: a general approach to MLSE in uncertain environments", *IEEE Trans. on communications*, vol. 43, no. 2, Feb. 1995.

[12] S. Haykin "Adaptive filter theory, Third Edition", *Prentice-Hall, Inc.*, 1996.

[13] B. M. Sadler, R. Kozick and T. Moore "Bounds on MIMO channel estimation and equalization with side information", *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, pp. 2145-2148, 2001.

[14] Y. Hua, "Fast maximum likelihood for blind identification of multiple FIR channels", *IEEE Trans. Signal Processing*, vol. 44, no. 3, pp. 661-672, Mar. 1996.

[15] K. Chugg, A. Anastasopoulos and X. Chen, "Iterative detection", *Kluwer Academic Publishers*, Dec. 2000

[16] G. J. Foschini, Jr and M. J. Gans, "On limits of wireless communication in a fading enviroment when using multiple antennas", *Wireless Personal Communnications*, vol. 6, pp. 311-335, Mar. 1998.

[17] B. M. Sadler, R. J. Kozick and T. Moore, "Bounds on bearing and symbol estimation with side information", *IEEE Trans. Signal Processng*, vol. 49, no. 4, Apr. 2001.

[18] P. Stoica and B. C. Ng, "On the Cramér-Rao bound under parametric constraints", *IEEE Trans. Signal Processing Letters*, vol. 5, no. 7, Jul. 1998.

[19] W. Choi and J. M. Cioffi, "Multiple input/multiple output (MIMO) equalization for space-time coding", *IEEE Pacific Rim Conference on Communication, Comupters and Signal Processing*, pp. 341-344, 1999.

[20] E. L. Pinto and C. J. Silva, "Performance evaluation of blind channel identification methods based on oversampling", *IEEE Proceedings on Military Communications Conference*, vol. 1, pp. 165-169, 1998.

[21] A. J. van der Veen, S. Talwar and A. Paulraj, "Blind estimation of multiple digital signals transmitted over FIR channels", *IEEE Trans. Signal Processing Letters*, vol. 2, no. 5, May 1995.

[22] S. Talwar, M. Viberg and A. Paulraj, "Blind estimation of multiple co-channel digital signals arriving at an antenna array", *Record of the Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 349-353, 1993.

[23] L. Tong, G. Xu and T. Kailath, "Blind identification and equlization based on second-order statistics: a time domain approach", *IEEE Trans. Information Theory*, vol. 40, no. 2, Mar. 1994.

[24] H. Chen, K. Buckley and R. Perry, "Time-recursive maximum likelihood based sequence estimation for unknown ISI channels", *Record of the Thirty-Fourth Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1005-1009, 2000.

[25] C. N. Georghiades and J. C. Han, "Sequence estimation if the persence of random parameters via the EM algorithm", *IEEE Trans. Communications*, vol. 45, pp. 300-308, Mar. 1997.

[26] J. F. Galdino and M. S. Alencar,"Blind equlization for fast frequency selective fading channels", *IEEE International Conference on Communcitions*, vol. 10, pp. 3082-3086, 2001.

[27] J. W. Brewer, "Kronecker products and matrix calculus in system theory", *IEEE Trans. Circuits And Systems*, vol. CAS-25, no. 9, Sep. 1976.

[28] H. Kubo, K. Murakami and T. Fujino, "An adaptive maximum-likelihood sequence estimator for fast time-varying intersymbol interference channels", *IEEE Trans. Communications*, vol. 42, no. 2, Feb. 1994.

[29] N. Seshadri, "Joint data and channel estimation using blind trellis search techniques", *IEEE Trans. Communications*, vol. 2, no. 2, Feb. 1994.

[30] E. Moulines, P. Duhamel, J. F. Cardoso and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters", *IEEE Trans. Signal Processing*, vol. 43, no. 2, Feb. 1995.