Sparse Equalizer Filter Design for Multi-path Channels

by

Xue Feng

B. S., Electrical Engineering
Xi'an Jiaotong University, China, 2010

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2012

Author .....
Department of Electrical Engineering and Computer Science
May 23, 2012

Certified by .....
Alan V. Oppenheim
Ford Professor of Engineering
Thesis Supervisor

Accepted by .....
Leslie A. Kolodziejski
Chairman, Department Committee on Graduate Students

Sparse Equalizer Filter Design for Multi-path Channels

by

Xue Feng

Submitted to the Department of Electrical Engineering and Computer Science
on May 23, 2012, in partial fulfillment of the
requirements for the degree of
Master of Science

Abstract

In this thesis, sparse Finite Impulse Response (FIR) equalizers are designed for sparse multi-path channels under a pre-defined Mean Squared Error (MSE) constraint. We start by examining the intrinsic sparsity of the Zero Forcing equalizers and the FIR Minimum MSE (MMSE) equalizers. Next the equalization MSE is formulated as a quadratic function of the equalizer coefficients. Both the Linear Equalizer (LE) and the Decision Feedback Equalizer (DFE) are analyzed. Utilizing the quadratic form, designing a sparse equalizer under a single MSE constraint becomes an l_0-norm minimization problem under a quadratic constraint, as described in [2]. Three previously developed methods for solving this problem are applied, namely the successive thinning algorithm, the branch-and-bound algorithm, and the simple linear programming algorithm. Simulations under various channel specifications, equalizer specifications and algorithm specifications are conducted to show the dependency of the sparsity on these factors. The channels include the ideal discrete multi-path channels and the Vehicular A multi-path channels in both the Single-Input-Single-Output (SISO) and the Multiple-Input-Multiple-Output scenarios. Additionally, the sparse FIR equalizer is designed for MIMO channels under two MSE constraints. This is formulated as an l_0-norm minimization problem under two quadratic constraints. A sub-optimal solution by decoupling the two constraints is proposed.

Thesis Supervisor: Alan V. Oppenheim
Title: Ford Professor of Engineering

Acknowledgments

First, I am most grateful for my research advisor, Professor Alan Oppenheim. I am fortunate to have had the privilege of being mentored by Al. Al's guidance and support are crucial for the development of this thesis. At the beginning of this path, Al's vision helped me orient myself on the right path. Encouraging me to unconventional thinking and to creativity, stimulating me, and providing me with unlimited freedom, he has made this journey enjoyable and rewarding. I have benefited tremendously from his dedication not only to my intellectual growth, but also my personality development. I deeply appreciate Al's help in shaping me a more organized person.

I would like to express my sincere thanks to Dennis Wei for his helpful contribution to this thesis. Dennis has been a close research collaborator and friend over the last two years. Part of this thesis is based on the algorithm that Dennis developed in his PhD thesis. I wish him all the best in his future endeavors.

It is a privilege to have been part of the Digital Signal Processing Group (DSPG) at MIT. I would like to thank past and present members of DSPG including Tom Baran, Ballard Blair, Petros Boufounos, Sefa Demirtas, Dan Dudgeon, Zahi Karam, Tarek Aziz Lahlou, Shay Maymon, Joseph McMichael, Martin McCormick, Milutin Pajovic, Charles Rohrs, Guolong Su, and Dennis Wei. The intellectual atmosphere of the group as well as the willingness to share ideas and to collaborate on research problems has made it a very exciting and enriching experience.

I also want to thank my fellow students Xun Cai, Atulya Yellepeddi and Feng Gao for their intellectual help in developing this thesis. I gained a lot from the stimulating discussions and I feel very lucky to have such inspiring and encouraging friends.

Finally, I am deeply grateful to my parents for their ever present love, their support throughout my education, and their encouraging me to strive for the best. I sincerely thank my whole family.

Contents

1. Introduction **13**

1.1. Background 13

1.2. Outline of the Thesis 15

2. Zero-Forcing Equalizer, MMSE Equalizer and Their Sparsity **17**

2.1. Zero-Forcing Equalizer 17

2.1.1. The existence of a ZF equalizer 17

2.1.2. Sparsity of ZF Equalizer for multi-path channels 19

2.2. FIR MMSE Equalizer 20

3. Mean Squared Error of Channel Equalization **25**

3.1. Linear Equalization 25

3.1.1. Signal Models and Problem Reductions 26

The SISO case 26

The SIMO case 29

The MIMO case 34

3.1.2. The Effect of Equalizer Length and Decision Delay on the MMSE . 37

3.2. Decision Feedback Equalization 41

3.2.1. DFE structure and assumption 41

3.2.2. Problem Reductions 43

3.2.3. The effect of Decision Delay, FFF lengh, and FBF length on the MMSE 45

4. L_0 norm minimization under a quadratic contraint **47**

4.1. Problem Formulation 47

4.1.1. Norm 47

4.1.2. Problem Formulation and Visualization 49
4.1.3. Comparison with Compressive Sensing 50
4.2. Problem Solutions 52
4.2.1. Sub-optimal Solution 52
4.2.2. The Branch-and-Bound Solution 54
4.2.3. Simple Linear Programming Algorithm 56

5. Sparse Equalizer Design and Simulation Results **59**
5.1. The effect of Channel Specifications 59
5.1.1. Simple Multi-path Channel 59
5.1.2. Practical Wireless Communication Channel 60
5.2. The effect of Equalizer Specifications 63
5.3. The effect of Algorithm Specifications 65

6. Zero-norm Minimization under Two Quadratic Constraints **69**
6.1. MIMO MSE 69
6.2. Problem Formulation and Visualization 70
6.3. Sub-optimal Solution 71
Part I 72
Part II 75

7. Conclusion and Future Work **77**
7.1. Future Work 78

List of Figures

2-1. (a) The ZF equalizer coefficients for a multi-path channel with impulse response $h[n] = 1 + 0.6\delta[n-7] + 0.4\delta[n-23]$. (b) The ZF equalizer coefficients for a multi-path channel with impulse response $h[n] = 1 + 0.23\delta[n-7] + 0.4\delta[n-13] - 0.7\delta[n-29]$. (c) The ZF equalizer coefficients for a multi-path channel with impulse response $h[n] = 1 + 0.17\delta[n-5] + 0.54\delta[n-11] - 0.12\delta[n-23] + 0.3\delta[n-31]$. 23

2-2. (a) The ZF equalizer coefficients for a multi-path channel with impulse response $h[n] = 1 + 0.6\delta[n-7] + 0.4\delta[n-23]$. (b) The FIR MMSE equalizer coefficients for the same multi-path channel with no noise. (c) The FIR MMSE equalizer coefficients for the same multi-path channel with Gaussian white noise and the equalizer input SNR = 10 dB. 24

3-1. Equalization System using linear equalizer 25

3-2. Equalization of a SISO Channel 26

3-3. Equalization of a discrete SISO channel 27

3-4. Equalization of a single channel with oversampling 30

3-5. Equalization of a discrete SIMO channel 31

3-6. Equalization of a MIMO channel 34

3-7. MMSE normalized by δ_x^2 as a function of the equalizer length N for (a) N_1=7, N_2=23, and (b) N_1=3, N_2=23 39

3-8. MMSE normalized by δ_x^2 as a function of the equalization decision delay Δ for channel length $N_c = 24$ and equalizer length $N = 40$. 40

3-9. MMSE normalized by δ_x^2 as a function of the channel SNR for N_1=7, N_2=23, and $N = 30$. 41

3-10. Equalization using DFE 42
3-11. Histogram of the equalization MSE caused by past estimation errors 43

4-1. Visualization of zero-norm minimization under one quadratic constraint . . 50
4-2. Visualization of compressive sensing 52
4-3. The subproblem tree of the branch-and-bound method 55
4-4. Visualization of the simple linear programming technique for the sparse equalizer design problem. 57

5-1. Number of nonzero taps vs. Channel SNR 60
5-2. Vehicular A Channel Setup 61
5-3. Effective discrete-time channel response for the Vehicular A multi-path channel example 62
5-4. (a) Coefficient values for the length 50 MMSE equalizer with SNR = 10 dB. (b) A corresponding sparse equalizer with excess error ratio = 0.2. 63
5-5. Number of nonzero taps vs. equalizer length 64
5-6. Number of nonzero taps vs. equalizer length using LE and DFE. 65
5-7. Number of nonzero taps vs. excess error 66
5-8. The number of non-zero coefficients and the computing time vs. the number of iterations in the Branch-and-Bound algorithm. 67
5-9. Histogram of the difference between the successive thinning result and the optimal result 68

6-1. Visualization of the l_0-norm minimization under two quadratic constraints 72
6-2. Visualization of the constraints separation in the sub-optimal method . . . 73
6-3. The tree of feasible separation searching 74
6-4. Iterative approach for finding the valid separation in problem 6.9 75

List of Tables

3.1. MMSE with respect to N_f and Δ 46

5.1. An example of Vehicular A multi-path Channel 61

Chapter 1
Introduction

1.1 Background

Filter design can be viewed as a tradeoff between improving level of performance and reducing complexity. Various complexity measures, such as total filter length, or number of nonzero coefficients, have been defined for different implementation situations. The total number of coefficients is traditionally used as an indication of the complexity of a filter design. However, for cases where the cost of implementation is dominated by arithmetic operations, the number of non-zero coefficients may be a more appropriate metric given that operations associated with zero-valued coefficients may be ignored. This leads to a demand for designs with fewer non-zero coefficients, i.e., sparse designs, which can be exploited to reduce computation, hardware, or power consumption, depending on the form of implementation. In the context of this thesis, a sparse equalizer refers to an equalizer that has a large number of zero coefficients.

In this thesis, attention is restricted to FIR filters. Finite impulse response filters are generally preferred over an infinite impulse response (IIR) implementation when linear phase is important, but they typically require more computation, memory and delay. Although memory is relatively inexpensive, increased computation not only makes the filter more expensive, but also increases the effects of quantization noise. In these cases, sparse FIR filters in which several of the multiplying coefficients are zero, plays a significant role. Additionally, it is often possible to cascade sparse filter sections with a non-sparse section to efficiently realize a high-quality narrowband filter.

In this thesis, the sparse FIR filter design application is focused on multi-path channel equalization. The necessity and feasibility are given below. For communication channels characterized by long channel impulse responses that could span tens of symbol periods, very long equalizers have to be employed to mitigate the resulting inter-symbol interference (ISI). This increases the complexity of computing and implementing finite impulse response

equalizers, which grows in proportion to the number of taps. It is reasonable to use the number of nonzero coefficients as a measure of filter complexity, and thus sparse equalization becomes critical in order to reduce the complexity at the expense of a tolerable performance loss. In practice, many multi-path channels have a few dominant paths over the entire channel impulse response duration. These examples include the power-delay profiles of the Pedestrian B and Vehicular A wireless channels defined by the ITU [3]. For multi-path communication channels, its even more important to evaluate the decrease of complexity introduced by a sparse equalizer. It is also intriguing to see whether we can obtain sparse equalizers with satisfactory performance if the channel itself is sparse. Moreover, as we will see in Chapter 2, the equalizer of a multi-path channel tends to be sparse and contains a significant percentage of small coefficients. This phenomenon allows us to design a sparse equalizer with a small sacrifice of the equalization performance.

The concept of sparse filter design has been proposed in the literatures. In [1], the number of nonzero coefficients is reduced by using only the largest taps of the minimum mean square error (MMSE) solution. An improved subset selection method [7] has also been developed under the weighted least-squares criterion. Wei [2] proposed two algorithms to determine the locations of nonzero taps given the maximum error under a weighted least square criterion. One is a low-complexity greedy algorithm that gives a sparse solution which is not necessarily optimal. The second focuses on the optimal solution based on the branch-and-bound (BNB) procedure. There are also algorithms that relax the number of nonzero coefficients constraint to the l_2-norm or the l_1-norm of the coefficients. It has been seen that the l_2-norm does not reflect the sparsity of the vector very well, and the l_1-norm recovers the sparse signals with very high probability (Baranuik [20]). In [21], a modified l_1-norm minimization method is proposed in order to obtain a sparse equalizer. The straightforward exhaustive search method is numerically unstable and NP-hard, which requires enormous iterations on all the non-zero elements. This is the major factor that many algorithms avoids using the l_0-norm to solve this optimization problem, and instead introducing relaxations or bounds to either find the sub-optimal solution or fasten the optimality searching speed.

Additionally, both the Linear Equalizer (LE) and the Decision Feedback Equalizer (DFE) are considered for the sparse equalizer design. The DFE is well known for its capability in combating intersymbol interference in communication channels. In [4] and [5], investigations were conducted on different types of equalizers. Instead of the LE, the DFE is considered in [4] as a way to combat ISI and is further suggested in [5] as a way to improve sparsity in conjunction with linear programming.

Both Single-Input-Single-Output (SISO) multi-path channels and Multiple-Input-Multiple-Output (MIMO) multi-path channels are investigated in this thesis. The analysis of the MIMO case is motived by the popularity of MIMO channels. MIMO technology has attracted attention in wireless communications, because it offers significant increases in data throughput and link range without additional bandwidth or increased transmit power. It is motivated by the desire to increase the capacity of digital wireless networks by allowing multiple transmissions sharing the same time slot and the frequency band tand separating them spatio-tempolrally at the receiver. In multi-user communication over linear, dispersive, and noisy channels, the received signal is composed of the sum of several transmitted signals corrupted by intersymbol interference (ISI), interuser interference (IUI), and noise. Typical examples of MIMO systems include TDMA digital cellular systems with multiple transmit/receive antennas, wideband asychronous CDMA system, and high-density digital magnetic recording.

■ 1.2 Outline of the Thesis

In this thesis, a framework is explored to design sparse FIR equalizers for sparse multi-path channels.

In Chapter 2, the natural sparsity of equalizers for sparse multi-path channels is explored. This understanding of the intrinsic equalizer sparsity is necessary for the sparse equalizer design in Chapter 4 and 6. The discussion first focuses on Zero-Forcing (ZF) equalizers. We define the ZF-equalizability and analyze the sparsity of ZF equalizer. Then we relate the FIR finite length Minimum Mean Squared Error (MMSE) equalizers to ZF equalizers and extended the discussion on sparsity to the FIR MMSE equalizers. Simulations results are presented to show the sparsity of the equalizers.

In Chapter 3, the equalization Mean Squared Error (MSE) is analyzed for linear, time-invariant, and noisy channels. To design a sparse FIR equalizer filter under the MSE criterion, it is necessary to obtain the mathematical expression of the equalization MSE as a function of the FIR equalizer filter coefficients. We first formulate the MSE expression for linear equalizers and evaluates the minimum MSE value under different channel and equalizer specifications. Then we extend the formulation of the MSE expression using advanced equalizers with feedbacks. The minimum MSE value of the DFE system is also evaluated under various channel and equalizer specifications.

In Chapter 4, the sparse equalizer design problem under a single MSE constraint is formulated. We first define the l_0-norm as the number of non-zero elements in a vector and show that this is a l_0-norm minimization problem under a quadratic constraint. Second, three algorithms to solve this problem are presented. These algorithms are implemented and evaluated in Chapter 5 for the sparse equalizer design. Additionally, a comparison between this equalizer design problem and the compressive sensing problem is made in this chapter in order to explain the difference.

In Chapter 5, simulation results of the sparse equalizer design problem defined in Chapter 4 are shown. The sparse equalizer filter design result can be affected by many factors including the channel specifications, the equalizer specifications and the algorithm specifications. In this chapter, we examine the effect of the channel SNR, the channel response, the equalizer type, the equalizer length, the MSE allowance, as well as the algorithm that is applied.

In Chapter 6, sparse equalizer design is considered for MIMO channels with two MSE constraints. This problem arises when one or more user subsets have their individual equalization MSE constraints other than the overall equalization MSE constraint. We start by defining the MIMO channel equalization MSE. Then the sparse equalizer design problem is formulated as a zero-norm minimization problem under two quadratic constraints. Due to the complexity of solving this problem, we propose a low-cost method, which decouples the two constraints in the original problem and computes a non-optimal solution. The simulation of this method is planned in future work (Section 7.1).

Chapter 2
Zero-Forcing Equalizer, MMSE Equalizer and Their Sparsity

In this chapter, the intrinsic sparsity of equalizers for sparse multi-path channels is explored. Before designing the sparse equalizer, understanding the natural sparsity of the equalizers is necessary. In Section 2.1, the discussion focuses on Zero-Forcing (ZF) equalizers. We show that if a channel is sparse, its ZF equalizer filter also exhibits a sparse pattern. In Section 2.2, the discussion is extended to finite length Minimum Mean Squared Error (MMSE) equalizers, which are shown to be non-sparse but often have a significant percentage of small coefficients.

■ 2.1 Zero-Forcing Equalizer

In this section, the equalization of a noise-free communication channel is considered. When the channel noise v is negligible, removing the inter-symbol interference at the receiver to recover the input signals results in the zero-forcing equalization problem. Zero-forcing equalizers are essentially channel inverses used to recover the unknown input signal of a linear system from its measurable output signal.

■ 2.1.1 The existence of a ZF equalizer

The goal of the equalizer is to recover the input signal from the output measurements. Therefore, zero-forcing requires that the input signal be recovered exactly with a possible finite delay in the absence of noise. One problem is that a zero forcing (ZF) equalizer may not exist. In this section, conditions for the existence of a ZF equalizer are defined in various sparsity multi-channel systems. In the next section, the sparsity of the ZF equalizer is discussed if it exists.

The following terminology is introduced in order to formulate the existence of a ZF equalizer. As defined in [8], a matrix $Q(z)$ is said to be unimodular if $\det Q(z)$ is a non-zero constant independent of z^{-1}. Moreover, every matrix has a Smith form [8], which is given

as follows. For any $\mathbf{H}(z) \in C^{n\times m}[z]$, there exist some unimodular matrices $Q(z) \in C^{n\times n}[z]$ and $P(z) \in C^{m\times m}$ such that (a)

$$Q(z)\mathbf{H}(z)P(z) = \begin{bmatrix} \alpha_1(z) & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \alpha_2(z) & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \alpha_p(z) & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix} \tag{2.1}$$

for some monic polynomials $\alpha_i(z), i = 1, 2, \dots, p$, where $p \leq \min n, m$. (b) $\alpha_1(z)$ is the greatest common divisor of all entries of $\mathbf{H}(\mathbf{z})$.

A transfer function has an IIR ZF-equalizer if there exists a finite order, stable and causal rational function $W(z)$ such that $W(z)H(z) = z^{-d}\mathbf{I}$ for some finite integer delay $d \geq 0$. A transfer function has an FIR ZF-equalizer if there exists a finite order FIR filter $W(z)$ such that $W(z)H(z) = z^{-d}\mathbf{I}$ for some finite integer delay $d \geq 0$.

The necessary and sufficient conditions for the two cases are respectively

(1) $H(z)$ has an IIR ZF-equalizer if and only if $H(z)$ is minimum phase, i.e., all of $\alpha_i(z)$ are minimum phase.

(2) $H(z)$ has an FIR ZF-equalizer if and only if $\alpha_i(z) = z^{-d_i}$ for some $d_i \geq 0$.

The proof is given below. We first show the necessity of (1). Suppose there is a $|z_0| \geq 1$ and $\alpha(z_0) = 0$ for some i, rank $H(z_0) < p$. Then rank $W(z_0)H(z_0) < p$. But rank $z_0^{-d}I = p$.

This completes the necessity part. For sufficiency, let

$$W(z) = z^{-d}P(z)\begin{bmatrix} \frac{1}{\alpha_1(z)} & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \frac{1}{\alpha_2(z)} & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \dots & 0 \\ 0 & 0 & \dots & \frac{1}{\alpha_p(z)} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix} Q(z)$$

for some $d \geq 0$ so that $W(z)$ is causal. Then it straightforward that $W(z)H(z) = z^{-d}I$. This completes the sufficiency part of (1). The proof of (2) is similar.

■ 2.1.2 Sparsity of ZF Equalizer for multi-path channels

In this section, sparsity and the locations of non-zero coefficients in the ZF equalizer are explored. We show that if a channel is sparse, i.e., its impulse response contains a large amount of zero coefficients, the ZF equalizer is sparse as well.

The zero-forcing equalizer for a causal multi-path channel is generally infinitely long, i.e., is an IIR filter. For cases where FIR ZF equalizers are not achievable, long division is adopted to compute the IIR zero-forcing equalizer.

To begin with, consider a multi-path channel $H(z) = 1 + \alpha_1 z^{-N_1} + \alpha_2 z^{-N_2}$ with two delay taps at $n = N_1$, and $n = N_2$. The zero forcing equalizer's transfer function is $W(z) = \frac{1}{1+\alpha_1 z^{-N_1}+\alpha_2 z^{-N_2}}$. By long division, it becomes

$$W(z) = 1 - \alpha_1 z^{-N_1} - \alpha_2 z^{-N_2} + \alpha_1^2 z^{-2N_1} + 2\alpha_1\alpha_2 z^{-(N_1+N_2)} + \cdots$$

or more compactly,

$$W(z) = \sum_{\lambda_1=0}^{\infty}\sum_{\lambda_2=0}^{\infty} c\ \alpha_1^{\lambda_1}\alpha_2^{\lambda_2} z^{-(\lambda_1 N_1+\lambda_2 N_2)}, \tag{2.2}$$

where c and λ_i $(i = 1, 2)$ are independent integers.

Note that each time a new term $\beta z^{pN_1+qN_2}$ is introduced in the long division result, another two terms $\alpha_1\beta z^{(p+1)N_1+qN_2}$ and $\alpha_2\beta z^{pN_1+(q+1)N_2}$ will be added to the residual, and thus these two term will appear in the long division. The merging of similar items is taken into account by the integer c. By induction, the pattern shown in Equation (2.2) holds.

This idea can be applied to a more general multi-path channel $H(z) = 1+\sum_{i=1}^{s-1}\alpha_i z^{-N_i}$, whose IIR ZF equalizer transfer function can be written as

$$W(z) = \sum_{\lambda_1=0}^{\infty}\sum_{\lambda_2=0}^{\infty}\cdots\sum_{\lambda_{s-1}=0}^{\infty} c\prod \alpha_i^{\lambda_i} z^{-\sum\lambda_i N_i},$$

where c and λ_i $(i = 1, 2, \ldots, s-1)$ are independent integers. Therefore, as the number of N_i grows, i.e., as the number of taps in the multi-path channel increases, the ZF equalizer $W(z)$ contains more taps and thus is less sparse. This is illustrated in Figure 2-1.

■ 2.2 FIR MMSE Equalizer

In the context of this thesis, the error criterion to evaluate the FIR equalization performance is chosen to be MSE. For the scope of this thesis, understanding the sparsity of the FIR MMSE equalizer is helpful. This section discusses the FIR MMSE equalizer filter for a finite length channel. Specifically, we relate the FIR equalizer coefficients with the ZF equalizer coefficients and observe the sparsity both in the absence of noise and with additive noise.

For simplicity, we start with a two path channel whose system function is

$$H(z) = 1 + \alpha z^{-d}.$$

Denote the IIR zero-forcing equalizer filter system function as

$$W_a(z) = \sum_{i=0}^{\infty} p_i z^{-i}$$

and the FIR MMSE equalizer filter system function as

$$W_b(z) = \sum_{i=0}^{N-1} q_i z^{-i}$$

The combined system function of the channel and FIR MMSE equalizer is characterized by

$$\begin{aligned} H(z)W_b(z) =& H(z)W_a(z) - H(z)(W_a(z) - W_b(z)) \\ =& 1 - H(z)(W_a(z) - W_b(z)) \\ =& 1 - (1 + \alpha z^{-d})(\sum_{i=0}^{N-1}(p_i - q_i)z^{-i} + \sum_{i=N}^{\infty} p_i z^{-i}) \\ =& 1 - \left[\sum_{i=0}^{d-1}(p_i - q_i)z^{-i} + \sum_{i=d}^{N-1}(\alpha(p_{i-d} - q_{i-d}) + (p_i - q_i))z^{-i} \right. \\ & \left. + \sum_{N}^{N+d-1}(\alpha(p_{i-d} - q_{i-d}) + p_i)z^{-i} + \sum_{N+d}^{\infty}(\alpha p_{i-d} + p_i)z^{-i}\right] \end{aligned} \tag{2.3}$$

Therefore, if $x[k]$ and $\hat{x}[k]$ denote the channel input and equalization output at any time k respectively, they are related by

$$\begin{aligned} \hat{x}[k] = x[k] - \left\{ \sum_{i=0}^{d-1}(p_i - q_i)x[k-i] + \sum_{i=d}^{N-1}(\alpha(p_{i-d} - q_{i-d}) + (p_i - q_i))x[k-i] \right. \\ \left. + \sum_{N}^{N+d-1}(\alpha(p_{i-d} - q_{i-d}) + p_i)x[k-i] + \sum_{N+d}^{\infty}(\alpha p_{i-d} + p_i)x[k-i]\right\}, \end{aligned}$$

$$\begin{aligned} e[k] =& x[k] - \hat{x}[k] \\ =& \left\{ \sum_{i=0}^{d-1}(p_i - q_i)x[k-i] + \sum_{i=d}^{N-1}(\alpha(p_{i-d} - q_{i-d}) + (p_i - q_i))x[k-i] \right. \\ & \left. + \sum_{N}^{N+d-1}(\alpha(p_{i-d} - q_{i-d}) + p_i)x[k-i] + \sum_{N+d}^{\infty}(\alpha p_{i-d} + p_i)x[k-i]\right\}. \end{aligned}$$

Assuming that the input signal is white, its auto-correlation function is given by

$$E_k\left[x[k]x[k-n]\right] = \begin{cases} \sigma_x^2 & k = n \\ 0 & k \neq n \end{cases}$$

Therefore, the MSE is given by

$$E[e[k]^2] = \delta_x^2 \left[\sum_{i=0}^{d-1} (p_i - q_i)^2 + \sum_{i=d}^{N-1} (\alpha(p_{i-d} - q_{i-d}) + (p_i - q_i))^2 + \sum_{N}^{N+d-1} (\alpha(p_{i-d} - q_{i-d}) + p_i)^2 + \sum_{N+d}^{\infty} (\alpha p_{i-d} + p_i)^2 \right] . \tag{2.4}$$

Since $\{p_i\}$ are the known ZF equalizer coefficients for a given channel, minimizing Equation (2.4) is equal to minimize the following function of $\{q_i\}$

$$f(q_0, q_1, \ldots, q_{N-1}) = \sum_{i=0}^{d-1} (p_i - q_i)^2 + \sum_{i=d}^{N-1} (\alpha(p_{i-d} - q_{i-d}) + (p_i - q_i))^2 + \sum_{N}^{N+d-1} (\alpha(p_{i-d} - q_{i-d}) + p_i)^2, \tag{2.5}$$

By taking partial derivative with respect to q_i and setting it to be zero, each q_i can be determined by the $N + d + 1$ p_i values. In general, the solution $q_i \neq p_i$, which means the FIR MMSE equalizer filter is not a truncated IIR ZF equalizer filter even in the noise-free case. The truncation affect all values of p_is. Therefore, the sparsity pattern shown in ZF equalizer coefficients does not hold for the FIR MMSE equalizer coefficients.

The sparsities of the ZF equalizer and FIR MMSE equalizer are compared in Figure 2-2. The difference from (a) to (b) indicates that the sparse pattern is destroyed due to the finite length constraint. (c) shows that with noise added, the values of the FIR MMSE equalizer coefficients are further slightly changed. But although (b) and (c) are not sparse, they contain a lot of small coefficients. These features are important for designing a sparse FIR equalizer under the MSE criterion with low cost, as will be shown in Chapter 5.

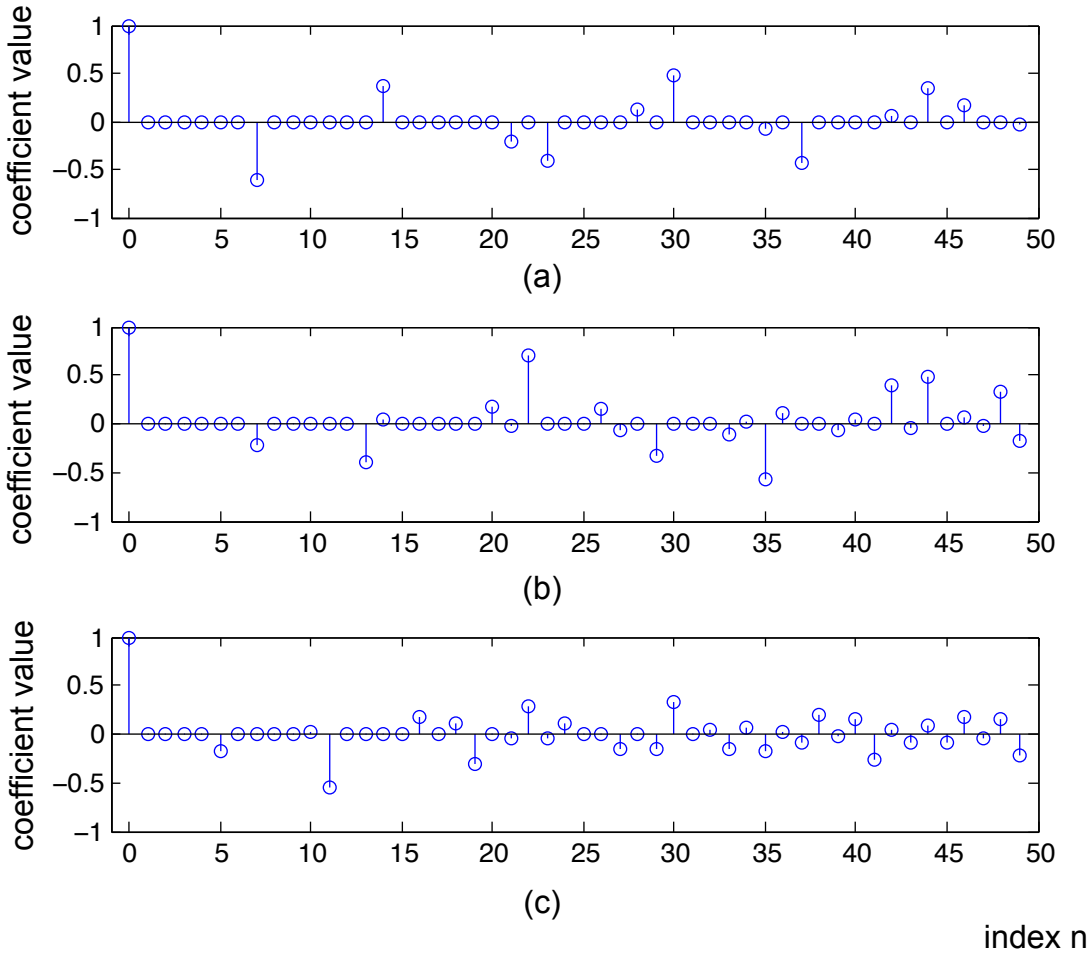

Figure 2-1: (a) The ZF equalizer coefficients for a multi-path channel with impulse response $h[n] = 1 + 0.6\delta[n-7] + 0.4\delta[n-23]$. (b) The ZF equalizer coefficients for a multi-path channel with impulse response $h[n] = 1 + 0.23\delta[n-7] + 0.4\delta[n-13] - 0.7\delta[n-29]$. (c) The ZF equalizer coefficients for a multi-path channel with impulse response $h[n] = 1 + 0.17\delta[n-5] + 0.54\delta[n-11] - 0.12\delta[n-23] + 0.3\delta[n-31]$.

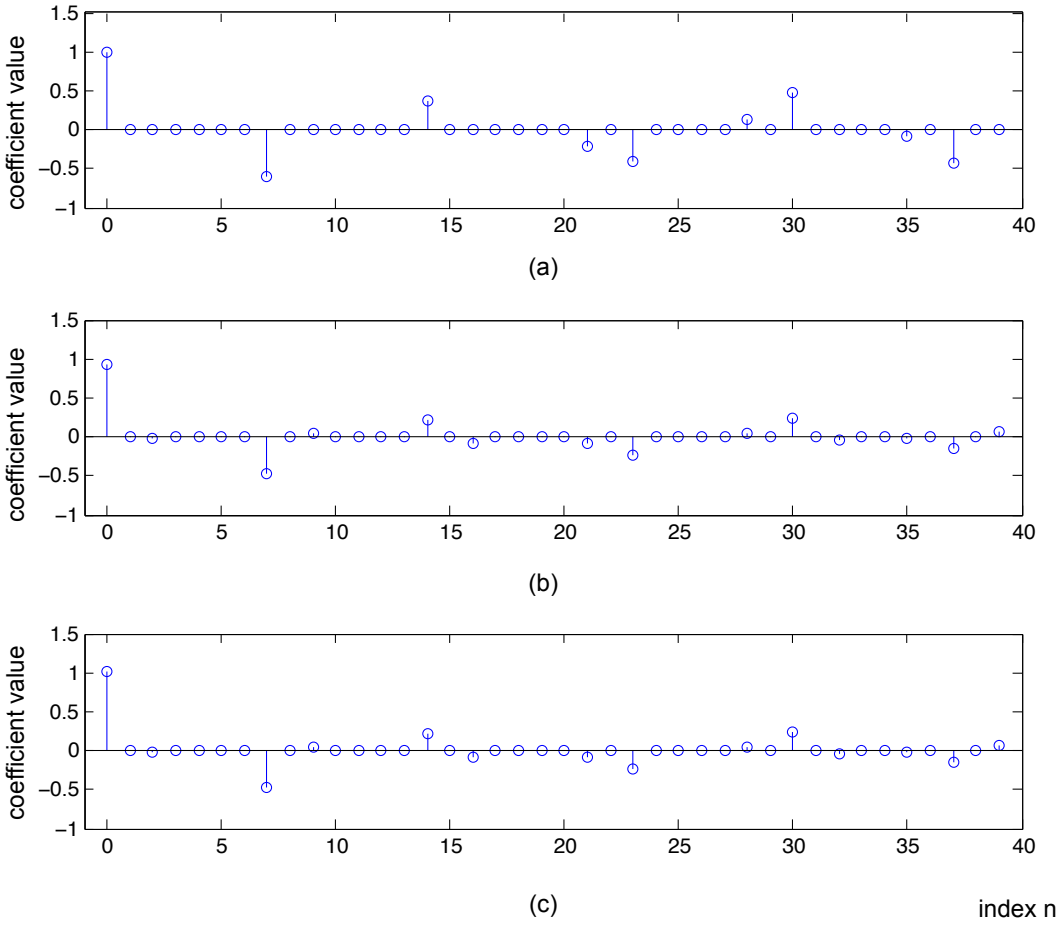

Figure 2-2: (a) The ZF equalizer coefficients for a multi-path channel with impulse response $h[n] = 1 + 0.6\delta[n-7] + 0.4\delta[n-23]$. (b) The FIR MMSE equalizer coefficients for the same multi-path channel with no noise. (c) The FIR MMSE equalizer coefficients for the same multi-path channel with Gaussian white noise and the equalizer input SNR = 10 dB.

Chapter 3
Mean Squared Error of Channel Equalization

To design a sparse FIR equalizer filter under the Mean Squared Error (MSE) criteria, it is necessary to first obtain the mathematical expression of the equalization MSE as a function of the FIR equalizer filter coefficients. In this chapter, the MSE is analyzed for linear, time-invariant, and noisy channels with two different equalization structures. Section 3.1 formulates the MSE expression for linear equalizers and evaluates the MSE value under different equalizer specifications. Section 3.2 turns considerations to advanced equalizers with feedbacks and derives the corresponding MSE expression.

As shown in Figure 3-1, the channels that are discussed in this chapter are modeled by

$$Y(z) = H(z)X(z) + V(z). \tag{3.1}$$

For a channel with n_i inputs and n_o outputs, $X(z)$ is an $n_i \times 1$ vector, and $Y(z)$ is an $n_o \times 1$ vector. $H(z)$ the $n_i \times n_o$ dimensional transfer function matrix. This model can be used to represent a large class of communication channels with multiple inputs and multiple outputs.

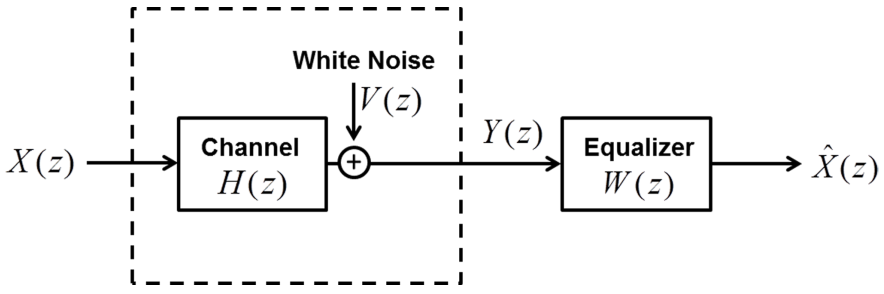

Figure 3-1: Equalization System using linear equalizer

■ 3.1 Linear Equalization

This section evaluate the MSE of linear equalizers for linear, and time-invariant channels with additive noises.

■ 3.1.1 Signal Models and Problem Reductions

Channels can be categorized into four types, namely, the Single-Input-Single-Output (SISO) channel, the Single-Input-Multiple-Output (SIMO) channel and the Multiple-Input-Multiple-Output (MIMO) channel. The equalization MSE for all channel types are discussed in the following subsections respectively. For a MISO channel, since the equalizer estimates every input signal and the MSE is calculated only between a specific input and its estimation, the MISO channel equalization system can be reduced to multiple SISO channel equalization systems.

The SISO case

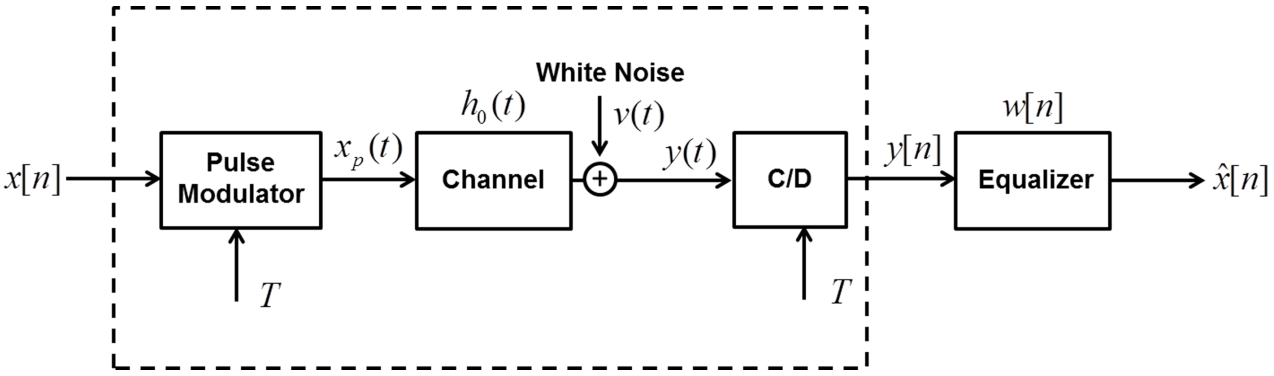

Figure 3-2: Equalization of a SISO Channel

Let us assume a general transmission scheme given in Figure(3-2). In the discrete equalization context, $x[n]$ represents the sequence of transmitted samples, and $y[n]$ represents the sequence of received samples. The channel impulse response is denoted by $h_0(t)$. Note that the pulse modulator and the C/D converter share the same sampling period T. Every input sample $x[k]$ produces exactly one output sample $y[k]$. Therefore, it is equivalent to a discrete SISO channel.

Denote the combined response of the pulse modulator $p(t)$ and the channel $h_0(t)$ as $h(t)$, i.e., $h(t) = p(t) \star h_0(t)$, where $\star$ denotes convolution. We assume that there is no ISI in the combined response. The output from the continuous-time transmission channel is

given by

$$y(t) = \sum_{l} h(t - lT)x[l] + v(t). \tag{3.2}$$

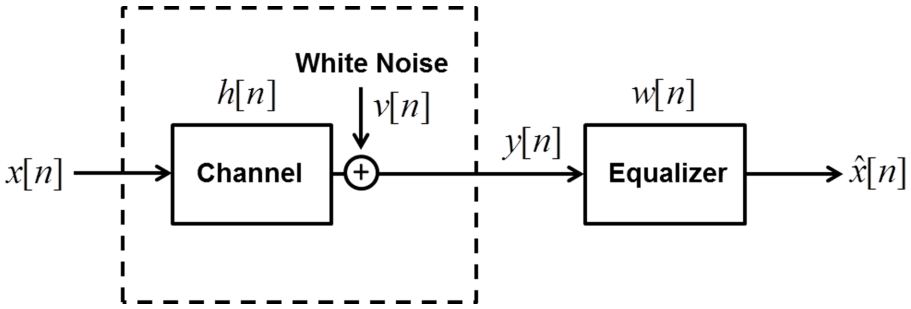

Figure 3-3: Equalization of a discrete SISO channel

After C/D conversion, the input to the equalizer is sampled at a rate of 1/T so that

$$\begin{aligned} y[n] &= y(nT) \\ &= \sum_{l} h(nT - lT)x[l] + v(nT) \\ &= \sum_{l} h[n - l]x[l] + v[n], \end{aligned} \tag{3.3}$$

where $h[n] = h(nT)$, and $v[n] = v(nT)$. Hence, the overall system is equivalent to the system given in Figure(3-3). Suppose the sampled channel impulse response $h[n]$ is of length N_c. By denoting h[n] as h_n, Equation (3.3) can be rewritten as

$$y_k = \sum_{l=0}^{N_c-1} h[l]x[n - l] + v[n] \tag{3.4}$$

where $h[n]$ is the channel impulse response, and $v[n]$ represents the additive noise.

An FIR equalizer of length N_f is applied to the received samples $\{y_k\}$in order to recover $x[n]$. Note that for every recovered sample $\hat{x}_k$, only the previous N_f samples of $y[n]$ are involved in the equalization, which means only samples $\{y_k, \ldots, y_{k-N_c+1}\}$ are used. For

these N_f-long samples of interest, it follows from Equation (3.4) that

$$\begin{bmatrix} y_{k+N_f-1} \\ y_{k+N_f-2} \\ \vdots \\ y_k \end{bmatrix} = \begin{bmatrix} h_0 & h_1 & \dots & h_{N_c-1} & 0 & \dots & \dots & 0 \\ 0 & h_0 & h_1 & \dots & h_{N_c-1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots & \vdots \\ 0 & \dots & \dots & 0 & h_0 & h_1 & \dots & h_{N_c-1} \end{bmatrix} \begin{bmatrix} x_{k+N_f-1} \\ x_{k+N_f-2} \\ \vdots \\ x_{k-N_c} \end{bmatrix} + \begin{bmatrix} v_{k+N_f-1} \\ v_{k+N_f-2} \\ \vdots \\ v_k \end{bmatrix} \tag{3.5}$$

or more compactly,

$$y_{k:k-N_f+1} = H x_{k:k-N_f-N_c+1} + v_{k:k-N_f+1} \tag{3.6}$$

where $y_{k:k-N_f+1}$, $x_{k:k-N_f-N_c+1}$, $n_{k:k-N_f+1}$ are column vectors grouping the received, transmitted, and noise samples over that block, and H is a $N_f \times (N_c + N_f)$ Toeplitz matrix as follows.

$$H = \begin{bmatrix} h_0 & h_1 & \dots & h_{N_c-1} & 0 & \dots & \dots & 0 \\ 0 & h_0 & h_1 & \dots & h_{N_c-1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots & \vdots \\ 0 & \dots & \dots & 0 & h_0 & h_1 & \dots & h_{N_c-1} \end{bmatrix} \tag{3.7}$$

Taking the decision delay into consideration, the k^{th} equalization error sample is given by

$$e_k = x_{k-\Delta} - \hat{x}_k \quad = x_{k-\Delta} - \mathbf{w}^H y_{k:k-N_f+1} \tag{3.8}$$

where $\mathbf{w}$ is the $N_f \times 1$ vector of equalizer coefficients and Δ is an integer representing the decision delay, with $0 \leq \Delta \leq N_f + N_c - 1$. Therefore, the MSE can be written as

$$\begin{aligned} \text{MSE} &= E[\,|e_k|^2\,] \\ &= E[\,e_k^H e_k\,] \\ &= E[\,(x_{k-\Delta} - \mathbf{w}^H y_{k:k-N_f+1})^H (x_{k-\Delta} - \mathbf{w}^H y_{k:k-N_f+1})\,] \\ &= E[\,x_{k-\Delta}^2\,] - \mathbf{w}^H E[\,y_{k:k-N_f+1} x_{k-\Delta}\,] - E[\,y_{k:k-N_f+1} x_{k-\Delta}\,]^H \mathbf{w} + \mathbf{w}^H E[\,y_{k:k-N_f+1} y_{k:k-N_f+1}^H \mathbf{w}\,] \end{aligned} \tag{3.9}$$

Therefore, as we will see soon, it is useful to define the auto-correlation and the cross-correlation based on the block of length N_f.

The $(N_f + N_c) \times (N_f + N_c)$ input correlation matrix is given by

$$R_{xx} \equiv E[\, x_{k:k-N_f-N_c+1}\, x^H_{k:k-N_f-N_c+1}\,]. \tag{3.10}$$

The noise correlation matrix is given by

$$R_{vv} \equiv E[\, v_{k:k-N_f+1}\, v^H_{k:k-N_f+1}\,]. \tag{3.11}$$

The input-output cross-correlation matrix is defined as

$$R_{yx} \equiv E[\, y_{k:k-N_f+1}\, x^H_{k:k-N_f-N_c+1}\,] = HR_{xx}. \tag{3.12}$$

The output auto-correlation matrix is defined as

$$R_{yy} \equiv E[\, y_{k:k-N_f+1}\, y^H_{k:k-N_f+1}\,] = HR_{xx}H^H + R_{vv}. \tag{3.13}$$

Denote $\mathbf{1}_\Delta \equiv [\underbrace{0\,0\,\cdots\,0}_{\Delta}\,1\,\underbrace{0\,0\,\cdots\,0}_{N_f+N_c-\Delta-1}\,]^H$, then $x_{k-\Delta}$ can be rewritten as $x_{k-\Delta} = \mathbf{1}^H_\Delta x_{k:k-N_f-v+1}$. Furthermore, denote $\mathbf{r}_\Delta = R_{yx}\mathbf{1}_\Delta$. Assuming that the source signal $x[n]$ is stationary, we have $E[\, x^2_{k-\Delta}\,] = \delta^2_x$, where δ^2_x is the source signal power. Then (3.9) becomes

$$\text{MSE} = \delta^2_x - \mathbf{w}^H \cdot \mathbf{r}_\Delta - \mathbf{r}^H_\Delta \mathbf{w} + \mathbf{w}^H R_{yy}\mathbf{w}, \tag{3.14}$$

which is a quadratic function of the coefficients $\{\mathbf{w}\}$

The SIMO case

In this section, the MSE is analyzed for channels with oversampling. We show that channels with oversampling are equivalent to discrete SIMO channels.

As a very common approach in communication systems to improve transmission accuracy, oversampling is widely adopted to combat the error introduced by noise. If multiple samples are taken of the same quantity with uncorrelated noise added to each sample, then averaging N samples reduces the noise power by a factor of 1/N. This requires a higher sampling rate at the receiver than that at the transmitter. The typical framework is given

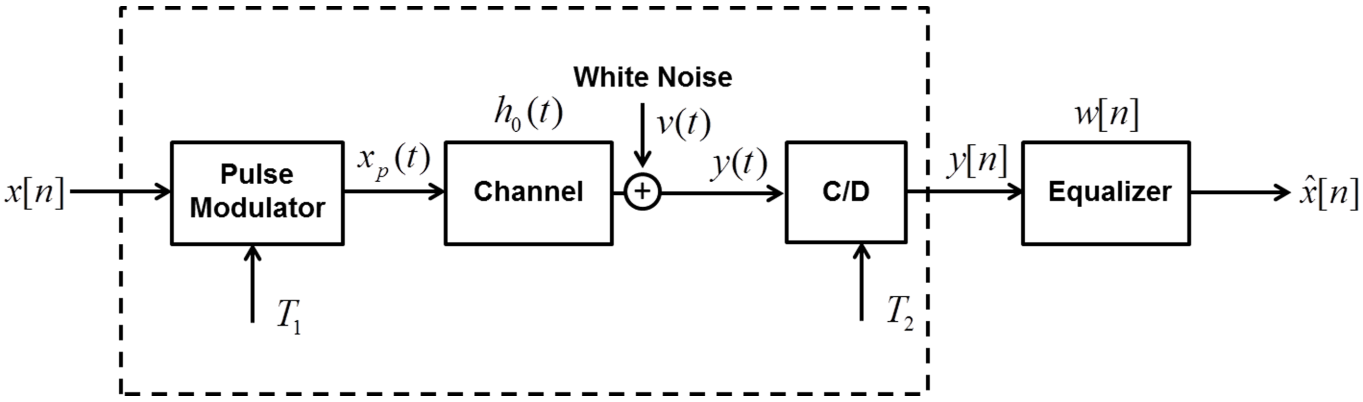

Figure 3-4: Equalization of a single channel with oversampling

in Figure 3-4, with oversampling rate $p = \frac{T_1}{T_2}$. Again, we assume that there is no ISI in the combined response of the pulse modulator and the channel.

Similar to the SISO case described in the previous section, the received signal of the equalizer is given by

$$y(t) = \sum_l h(t - lT_1)x[l] + v(t). \tag{3.15}$$

After C/D conversion, the input to the equalizer is sampled at a rate of $1/(T_2)$ so that

$$\begin{aligned} y[m] &= y(mT_2) \\ &= \sum_l h(mT_2 - lT_1)x[l] + v(mT_2) \end{aligned} \tag{3.16}$$

Since $y[m]$ is operated p times faster than the source x[n], samples $\{y[np], y[np+1], \ldots, y[np+p-1]\}$ are utilized together at the same time, i.e., input sample time n, to generate an estimate of $x[n]$. We rewrite y[m] as a sum of p parallel signals, i.e.,

$$y[m] = \sum_{i=1}^{p} y^{(i)}[n], \tag{3.17}$$

where $n = m \bmod p$, and each $y^{(i)}[n]$ is operated at the rate of $1/T_1$ with $y^{(i)}[n] = y[pn+i]$. More precisely, $y^{(i)}[n]$ is the sub-sampled polyphase components of $y[n]$. Hence, the overall system is equivalent to a discrete SIMO as is drawn in FIgure(3-5). Plugging the notation

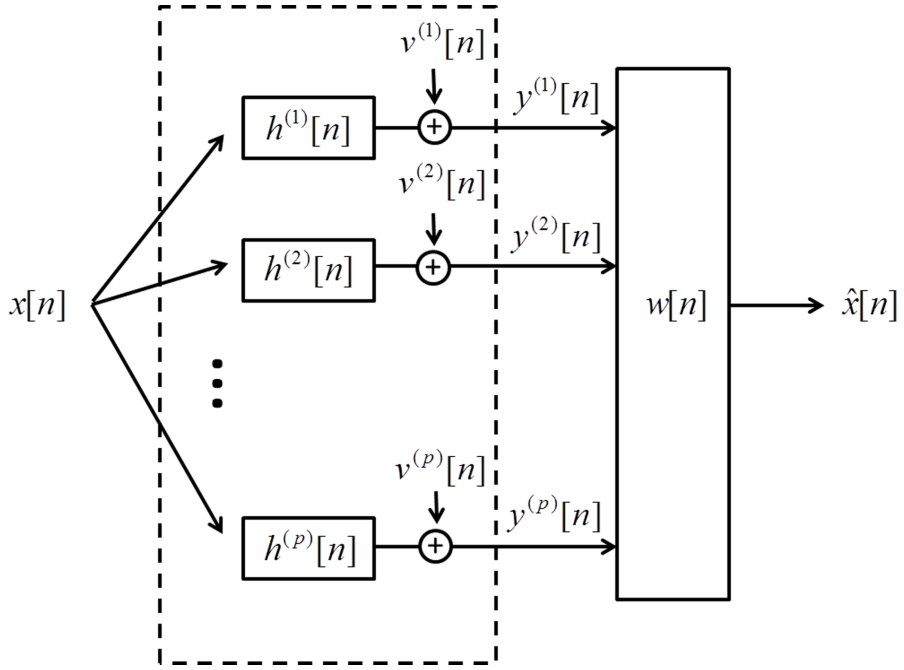

Figure 3-5: Equalization of a discrete SIMO channel

of $y^{(i)}[n]$ into Equation (3.16) we obtain the input-output relationship between each path

$$\begin{aligned} y^{(i)}[n] &= \sum_{l} h((pn+i)T_2 - lT_1)x[l] + v((pn+i)T_2) \\ &= \sum_{l} h((n-l)T_1 + iT_2)x[l] + v(nT_1 + iT_2) \\ &= \sum_{l=1} N_c^{(i)} h^{(i)}[n-l]x[l] + v^{(i)}[n] \end{aligned} \tag{3.18}$$

where $h^{(i)}[n] = h(nT_1 + iT_2)$, $v^{(i)}[n] = v(nT_1 + iT_2)$ and $N_c^{(i)}$ is the length of $h^{(i)}[n]$. Denoting $y^{(i)}[k]$ as $y_k^{(i)}$, the matrix form of quation (3.18) becomes

$$y_k^{(i)} = \begin{bmatrix} h_1^{(i)} & h_2^{(i)} & \dots & h_{N_c^{(i)}}^{(i)} \end{bmatrix} \begin{bmatrix} x_k \\ x_{k-1} \\ \vdots \\ x_{k-N_c+1} \end{bmatrix} + v_k^{(i)} \tag{3.19}$$

To simplify the expression, it is beneficial to group the received samples at the equalizer from all p channel outputs at sample time k into a $p \times 1$ column vector $\mathbf{y_k}$. Denote the

largest length of sampled channel impulse responses $h^{(i)}[n]$ as N_c, i.e., $N_c = \max_i N_c^{(i)}$. In this notation, Equation (3.18) becomes

$$\begin{bmatrix} y_k^{(1)} \\ y_k^{(2)} \\ \vdots \\ y_k^{(p)} \end{bmatrix} = \begin{bmatrix} h_1^{(1)} & h_2^{(1)} & \dots & h_{N_c}^{(1)} \\ h_1^{(2)} & h_2^{(2)} & \dots & h_{N_c}^{(2)} \\ \vdots & \ddots & \ddots & \vdots \\ h_1^{(p)} & h_2^{(p)} & \dots & h_{N_c}^{(p)} \end{bmatrix} \begin{bmatrix} x_k \\ x_{k-1} \\ \vdots \\ x_{k-N_c+1} \end{bmatrix} + \begin{bmatrix} v_k^{(1)} \\ v_k^{(2)} \\ \vdots \\ v_k^{(p)} \end{bmatrix} \tag{3.20}$$

An equalizer of length N_f is applied to the received samples in order to recover $x[n]$. Similar to the SISO case, for every recovered sample $\hat{x}[k]$, only the previous N_f samples of $y^{(i)}[n]$ are involved in the equalization. Again, grouping these samples of interest, Equation 3.21 can be extended to

$$\begin{bmatrix} y_k^{(1)} \\ y_k^{(2)} \\ \vdots \\ y_k^{(p)} \\ \hdashline y_{k-1}^{(1)} \\ y_{k-1}^{(2)} \\ \vdots \\ y_{k-1}^{(p)} \\ \hdashline \vdots \\ \hdashline y_{k-N_f+1}^{(1)} \\ y_{k-N_f+1}^{(2)} \\ \vdots \\ y_{k-N_f+1}^{(p)} \end{bmatrix} = \begin{bmatrix} h_1^{(1)} & h_2^{(1)} & \dots & h_{N_c}^{(1)} & & & \\ h_1^{(2)} & h_2^{(2)} & \dots & h_{N_c}^{(2)} & & 0 & \\ \vdots & \ddots & \ddots & \vdots & & & \\ h_1^{(p)} & h_2^{(p)} & \dots & h_{N_c}^{(p)} & & & \\ \hdashline 0 & h_1^{(1)} & h_2^{(1)} & \dots & h_{N_c}^{(1)} & & \\ 0 & h_1^{(2)} & h_2^{(2)} & \dots & h_{N_c}^{(2)} & 0 & \\ 0 & \vdots & \ddots & \ddots & \vdots & & \\ 0 & h_1^{(p)} & h_2^{(p)} & \dots & h_{N_c}^{(p)} & & \\ \hdashline & & & \ddots & & & \\ \hdashline & & & & h_1^{(1)} & h_2^{(1)} & \dots & h_{N_c}^{(1)} \\ & 0 & & & h_1^{(2)} & h_2^{(2)} & \dots & h_{N_c}^{(2)} \\ & & & & \vdots & \ddots & \ddots & \vdots \\ & & & & h_1^{(p)} & h_2^{(p)} & \dots & h_{N_c}^{(p)} \end{bmatrix} \begin{bmatrix} x_k \\ x_{k-1} \\ \vdots \\ x_{k-N_c-N_f+1} \end{bmatrix} + \begin{bmatrix} v_k^{(1)} \\ v_k^{(2)} \\ \vdots \\ v_k^{(p)} \\ \hdashline v_{k-1}^{(1)} \\ v_{k-1}^{(2)} \\ \vdots \\ v_{k-1}^{(p)} \\ \hdashline \vdots \\ \hdashline v_{k-N_c+1}^{(1)} \\ v_{k-N_c+1}^{(2)} \\ \vdots \\ v_{k-N_c+1}^{(p)} \end{bmatrix} \tag{3.21}$$

or in a more compact form,

$$\mathbf{y}_{k:k-N_f+1} = \mathbf{H}x_{k:k-N_f-N_c+1} + \mathbf{v}_{k:k-N_f+1} \tag{3.22}$$

where $y_{k:k-N_f+1}$ and $n_{k:k-N_f+1}$ are column vectors grouping the received and noise samples over that block respectively. Moreover, H is a $pN_f \times (N_c + N_f)$ matrix composed by $h_k^{(i)}$.

Similar to the SISO case, the k^{th} equalization error sample is given by

$$e_k = x_{k-\Delta} - \hat{x}_k \quad = x_{k-\Delta} - \mathbf{w}^H \mathbf{y}_{k:k-N_f+1} \tag{3.23}$$

where $\mathbf{w}$ is the $N_f \times 1$ vector of equalizer coefficients and Δ is an integer representing the decision delay, with $0 \leq \Delta \leq N_f + N_c - 1$. And the MSE can also be written as

$$\text{MSE} = E[\, x_{k-\Delta}^2 \,] - \mathbf{w}^H E[\, y_{k:k-N_f+1} x_{k-\Delta} \,] - E[\, y_{k:k-N_f+1} x_{k-\Delta} \,]^H \mathbf{w} + \mathbf{w}^H E[\, y_{k:k-N_f+1} y_{k:k-N_f+1}^H \,] \mathbf{w} \tag{3.24}$$

By defining the $(N_f + N_c) \times (N_f + N_c)$ input auto-correlation matrix

$$R_{xx} \equiv E[\, x_{k:k-N_f-N_c+1}\, x_{k:k-N_f-N_c+1}^H \,], \tag{3.25}$$

the $(pN_f) \times (pN_f)$ noise auto-correlation matrix

$$R_{vv} \equiv E[\, v_{k:k-N_f+1}\, v_{k:k-N_f+1}^H \,], \tag{3.26}$$

the $(pN_f) \times (N_f + N_c)$ output-input cross-correlation matrix

$$R_{yx} \equiv E[\, y_{k:k-N_f+1}\, x_{k:k-N_f-N_c+1}^H \,] = \mathbf{H} R_{xx} \tag{3.27}$$

and the $(pN_f) \times (pN_f)$ output auto-correlation matrix

$$R_{yy} \equiv E[\, y_{k:k-N_f+1}\, y_{k:k-N_f+1}^H \,] = \mathbf{H} R_{xx} \mathbf{H}^H + R_{vv}, \tag{3.28}$$

the equalization MSE for a SIMO channel in Equation (3.24) can be rewritten as

$$\text{MSE} = \delta_x^2 - \mathbf{w}^H \cdot \mathbf{r}_\Delta - \mathbf{r}_\Delta^H \mathbf{w} + \mathbf{w}^H R_{yy} \mathbf{w}, \tag{3.29}$$

which is reduced to the same quadratic form as in the case of SISO channel equalizations given in Equation (3.14).

The MIMO case

In this section, the equalization MSE for MIMO channels is investigated. We show that the MSE for MIMO channels can also be reduced to a quadratic form when the channel matrix is arranged in a certain way.

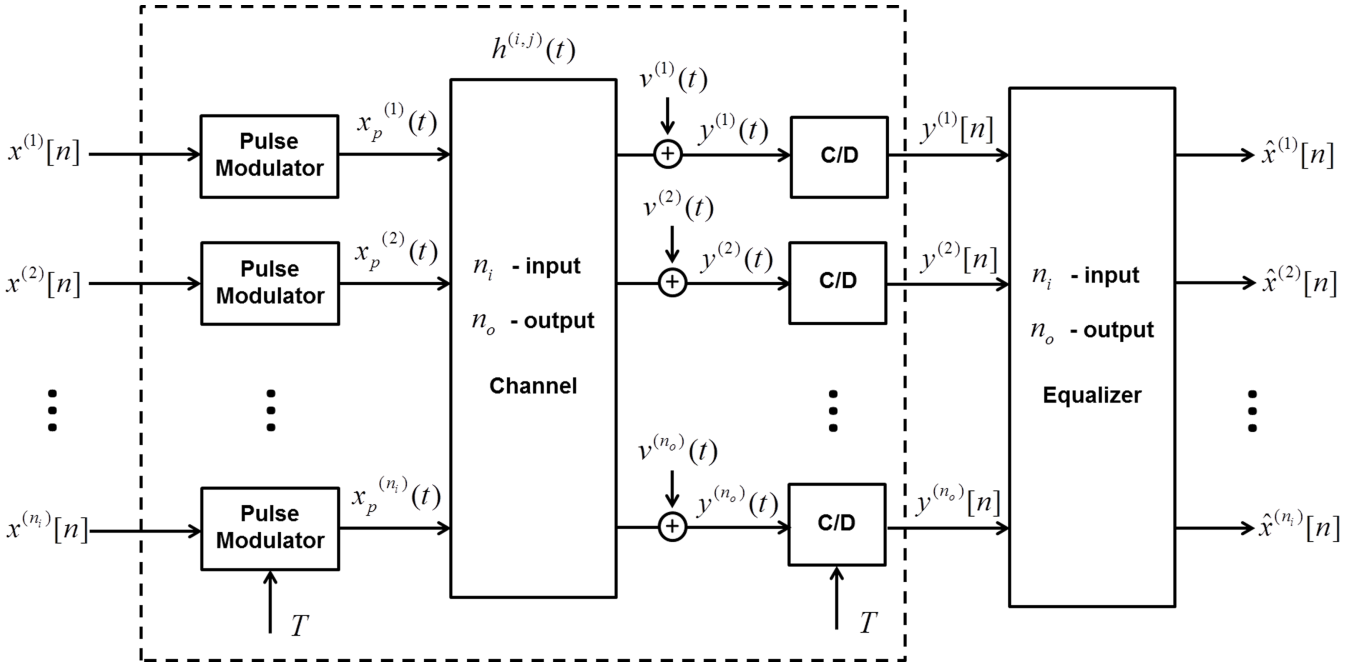

Figure 3-6: Equalization of a MIMO channel

A general case of a digital communication system with n_i inputs and n_o outputs is considered. Additive noise for each channel are assumed to be i.i.d. The block diagram of the multi-input multi-output channel model is given in Figure (3-6). The samples at the j^{th} output $(1 \leq j \leq n_o)$ have the standard form

$$y_k^{(j)} = \sum_{i=1}^{n_i} \sum_{m=0}^{N_c^{(i,j)}} h_m^{(i,j)} x_{k-m}^{(i)} + v_k^{(j)} \tag{3.30}$$

where

$y_k^{(j)}$ is the j^{th} channel output;

$h_m^{(i,j)}$ is the channel impulse response between the i^{th} input and the j^{th} output, whose length is denoted by $N_c^{(i,j)}$;

$v_k^{(j)}$ is the noise vector at the j^{th} output.

Similar to the SIMO case, to simplify the notation, the received samples from all n_o channel outputs at sample time k are grouped into a $n_o \times 1$ column vector $\mathbf{y_k}$ as follows

$$\mathbf{y_k} = \sum_{m=0}^{N_c} \mathbf{H_m} \mathbf{x}_{k-m} + \mathbf{v}_k \tag{3.31}$$

, where H_m is the $n_o \times n_i$ m^{th} $(1 < m < N_c)$MIMO channel matrix coefficient described as follows

$$\mathbf{H}_m = \begin{bmatrix} h_m^{(1,1)} & h_m^{(2,1)} & ... & h_m^{(n_i,1)} \\ h_m^{(1,2)} & h_m^{(2,2)} & ... & h_m^{(n_i,2)} \\ \vdots & \ddots & \ddots & \vdots \\ h_m^{(1,n_o)} & h_m^{(2,n_o)} & ... & h_m^{(n_i,n_o)}. \end{bmatrix}$$

$$\mathbf{x}_{k-m} = \begin{bmatrix} x_{k-m}^{(n_1)} \\ x_{k-m}^{(n_2)} \\ \vdots \\ x_{k-m}^{(n_i)} \end{bmatrix} \quad \mathbf{v}_k = \begin{bmatrix} v_k^{n_1} \\ v_k^{n_2} \\ \vdots \\ v_k^{n_o} \end{bmatrix}$$

are the input vector at time $k-m$, and noise vector at time k respectively. The parameter N_c is the maximum length of all the $n_o \times n_i$ channel impulse responses, i.e., $N_c = \max_{i,j} N_c^{(i,j)}$

An equalizer of length N_f is applied to the received samples in order to recover $x[n]$. Similar to the SISO and SIMO cases, for every recovered sample $\hat{x}[k]$, only the previous N_f samples of $y^{(i)}[n]$ are involved in the equalization. Hence, grouping these samples of

interest,

$$\begin{bmatrix} \mathbf{y}_{k+N_f-1} \\ \mathbf{y}_{k+N_f-2} \\ \vdots \\ \mathbf{y}_k \end{bmatrix} = \begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_{N_c-1} & 0 & \dots & \dots & 0 \\ 0 & \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_{N_c-1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots & \vdots \\ 0 & \dots & \dots & 0 & \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_{N_c-1} \end{bmatrix} \begin{bmatrix} x_{k+N_f-1} \\ x_{k+N_f-2} \\ \vdots \\ x_{k-N_c} \end{bmatrix} + \begin{bmatrix} v_{k+N_f-1} \\ v_{k+N_f-2} \\ \vdots \\ v_k \end{bmatrix}$$

or more compactly

$$\mathbf{y}_{k+N_f-1:k} = \mathbf{H}\mathbf{x}_{k-N_f-1:k-N_c} + \mathbf{v}_{k-N_f-1:k} \tag{3.32}$$

where

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_{N_c-1} & 0 & \dots & \dots & 0 \\ 0 & \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_{N_c-1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots & \vdots \\ 0 & \dots & \dots & 0 & \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_{N_c-1} \end{bmatrix}$$

In equalization systems for MIMO channels, the k^{th} equalization error sample becomes a vector given by

$$\mathbf{e}_k = \mathbf{x}_{k-\Delta} - \hat{\mathbf{x}}_k = \begin{bmatrix} e_k^{(1)} \\ e_k^{(2)} \\ \vdots \\ e_k^{(n_i)}, \end{bmatrix} \tag{3.33}$$

where $e_k^{(i)} = x_{k-\Delta} - \hat{x}_k$ is the equalization error of user i.

The equalization MSE for the whole system is defined as

$$\begin{aligned} \text{MSE} &= E[\, \mathbf{e}_k^H \mathbf{e}_k \,] \\ &= E[x_{k-\Delta}^H x_{k-\Delta}] - \mathbf{w}^H E[\mathbf{y}_{k:k-N_f+1} x_{k-\Delta}] - E[\mathbf{y}_{k:k-N_f+1} x_{k-\Delta}]^H \mathbf{w} \\ &\quad + \mathbf{w}^H E[\mathbf{y}_{k:k-N_f+1} \mathbf{y}_{k:k-N_f+1}^H \mathbf{w}] \end{aligned} \tag{3.34}$$

As we will see soon, it is useful to define the auto-correlation and the cross-correlation in the following way.

The $n_i(N_f + N_c) \times n_i(N_f + N_c)$ input auto-correlation matrix is defined as

$$\mathbf{R}_{xx} = E[\mathbf{x}_{k+N_f-1:k-N_c}\, \mathbf{x}^H_{k+N_f-1:k-N_c}] \tag{3.35}$$

The $n_o N_f \times n_o N_f$ noise auto-correlation matrix is defined as

$$\mathbf{R}_{vv} = E[\mathbf{n}_{k+N_f-1:k}\, \mathbf{v}^H_{k+N_f-1:k}] \tag{3.36}$$

The channel's input-output cross-correlation matrix is defined as

$$\mathbf{R}_{xy} = E[\mathbf{x}_{k+N_f-1:k-N_c}\, \mathbf{y}^H_{k+N_f-1:k}] = R_{xx}\mathbf{H}^H \tag{3.37}$$

The channel's output auto-correlation matrix is defined as

$$\mathbf{R}_{yy} = E[\mathbf{y}_{k+N_f-1:k}\, \mathbf{y}^H_{k+N_f-1:k}] = \mathbf{H}R_{xx}\mathbf{H}^H + R_{vv} \tag{3.38}$$

Furthermore, denote $\mathbf{1}_\Delta \equiv [\underbrace{0\,0\,\cdots\,0}_{\Delta}\,1\,\underbrace{0\,0\,\cdots\,0}_{N_f+N_c-\Delta-1}]^H$, then $x_{k-\Delta}$ can be rewritten as $x_{k-\Delta} = \mathbf{1}^H_\Delta x_{k:k-N_f-v+1}$. Denote $\mathbf{r}_\Delta = R_{yx}\mathbf{1}_\Delta$. Denote the i^{th} input signal power $E[\, {x_k^{(i)}}^H x_k^{(i)}\,]$ as ${\delta_x^{(i)}}^2$, and the total input signal power $\sum_{i=1}^{n_i} {\delta_x^{(i)}}^2$ as δ_x^2. Together with the assumption of stationary input signals, we have $E[\, \mathbf{x}^H_{k-\Delta}\mathbf{x}_{k-\Delta}\,] = \sum_{i=1}^{n_i} {\delta_x^{(i)}}^2 = \delta_x^2$.

Using all the above notations, the MSE can be rewritten as follows

$$\text{MSE} = \delta_x^2 - \mathbf{w}^H \cdot \mathbf{r}_\Delta - \mathbf{r}^H_\Delta \mathbf{w} + \mathbf{w}^H R_{yy}\mathbf{w}, \tag{3.39}$$

which results in the same quadratic form as in the SISO and SIMO cases.

■ 3.1.2 The Effect of Equalizer Length and Decision Delay on the MMSE

The equalizer that produces the Minimum Mean Squared Error (MMSE) is called the MMSE equalizer, which is also referred to as the Wiener filter. This section focuses the behavior of MMSE equalizers. The MMSE can be affected by various channel and equalizer

specifications. In the context of this chapter, the equalizer length, the decision delay and the Signal-to-Noise Ratio (SNR) are considered.

We first formulate the expression of the MMSE. From Section 3.1.1, the equalization MSE of all types of channels has the same quadratic form in Equations (3.14), (3.29) and (3.39). They can be rewritten as

$$\text{MSE} = \delta_x^2 - \mathbf{r}_\Delta^H \mathbf{R}_{yy}^{-1} \mathbf{r}_\Delta + (\mathbf{w} - \mathbf{R}_{yy}^{-1}\mathbf{r}_\Delta)^H R_{yy} (\mathbf{w} - \mathbf{R}_{yy}^{-1}\mathbf{r}_\Delta). \tag{3.40}$$

Since R_{yy} is positive-definite, from Equation 3.40, the MMSE is achieved by $\mathbf{w}_{\text{MMSE}} = \mathbf{R}_{yy}^{-1}\mathbf{r}_\Delta$. This $\mathbf{w}_{\text{MMSE}}$ solution also follows the Orthogonality Principle. Substituting $\mathbf{w}_{\text{MMSE}}$ for w in Equation 3.40, we obtain

$$\text{MMSE} = \delta_x^2 - \mathbf{r}_\Delta^H \mathbf{R}_{yy}^{-1} \mathbf{r}_\Delta. \tag{3.41}$$

In the following subsection, experimental results are given to show the effect of the equalizer length on MMSE, the effect of the decision delay on MMSE, and the effect of SNR on MMSE respectively. The channel impulse response $h[n]$ is chosen to represent an ideal multipath channel with one direct path and two delayed paths that are aligned with the sampling grid. More precisely,

$$h[n] = \delta[n] + a_1\delta[n - N_1] + a_2\delta[n - N_2],$$

where the delays N_1 and N_2 are positive integers and the amplitudes a_1 and a_2 are sampled randomly from the interval [-1,1].

First, we examine in figure 3-7 the effect of the equalizer length N on the equalization MMSE. For this experiment, SNR is fixed at 10 dB, and each data point represents the average of 1600 (a_1, a_2) pairs. The MMSE is non-decreasing as N increases. It is straightforward the set of equalizer coefficients with a smaller N is a subset of that with a greater N, and therefore the former cannot perform better than the latter. This can also be explained by reference to Equation 3.42. As N increases, the positive-definite matrix R_{yy}^{-1} grows, and thus $\mathbf{r}_\Delta^H \mathbf{R}_{yy}^{-1} \mathbf{r}_\Delta$ is non-decreasing, which will result in a smaller MMSE.

The staircase patterns can be explained by reference to the relationship between the FIR MMSE equalizer and the IIR ZF equalizer in Equation 2.5. As N increases, significant non-zero values in the IIR ZF equalizer are incorporated in the FIR MMSE equalizer only at certain values of N. As a result , the MMSE decreases the most at these points.

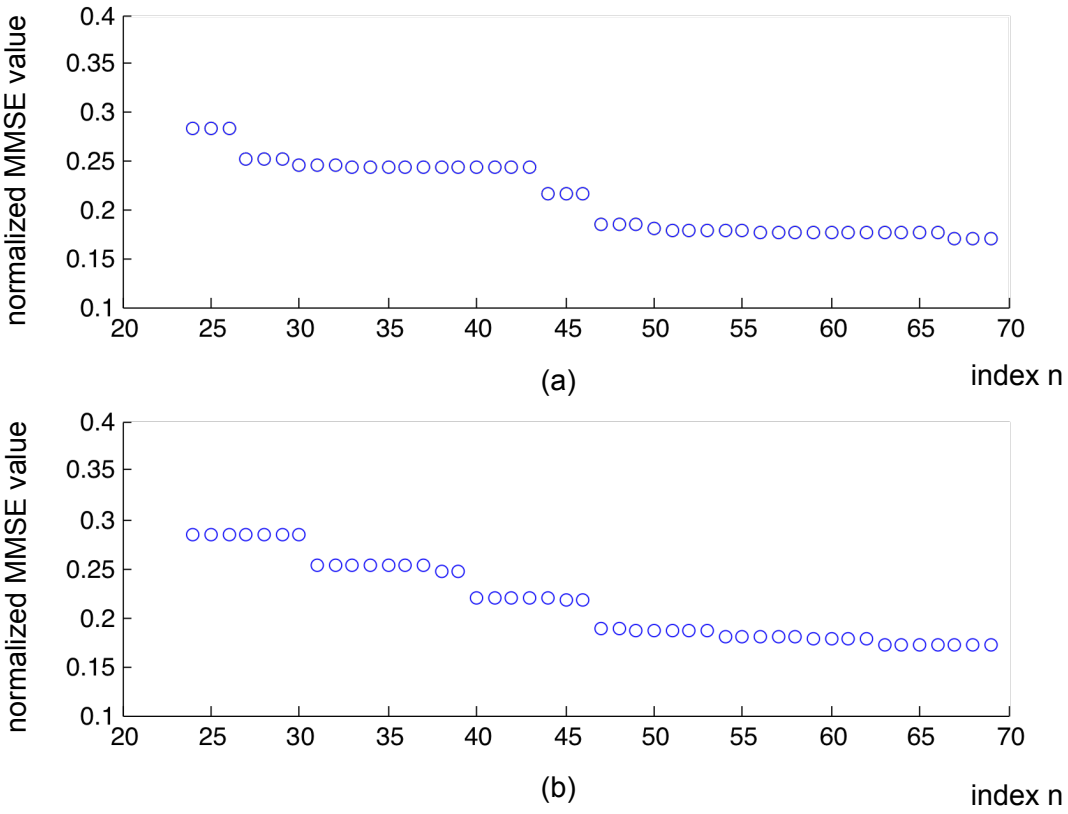

Figure 3-7: MMSE normalized by δ_x^2 as a function of the equalizer length N for (a) N_1=7, N_2=23, and (b) N_1=3, N_2=23

Next, for causality, a channel-equalizer system delay Δ has been introduced during the analysis of the equalization MSE. The delay is important in finite-length design because a non-causal filter cannot be implemented, and the delay allows time for the transmit data symbol to reach the receiver. Figure 3-8 plots the MMSE against the decision delay for $N_1 = 7, N_2 = 23$, and $N = 40$. Again each data point represents the average of 1600 (a_1, a_2) pairs. Simulations demonstrate that the optimal decision delay that produces the least MMSE is given by $\Delta = \frac{N_c+N}{2}$. This empirical result is also noted in [13]. Therefore,

in the sparse linear equalizer design examples given in Chapter 6, the decision delay Δ is chosen to be $\frac{N_c+N}{2}$.

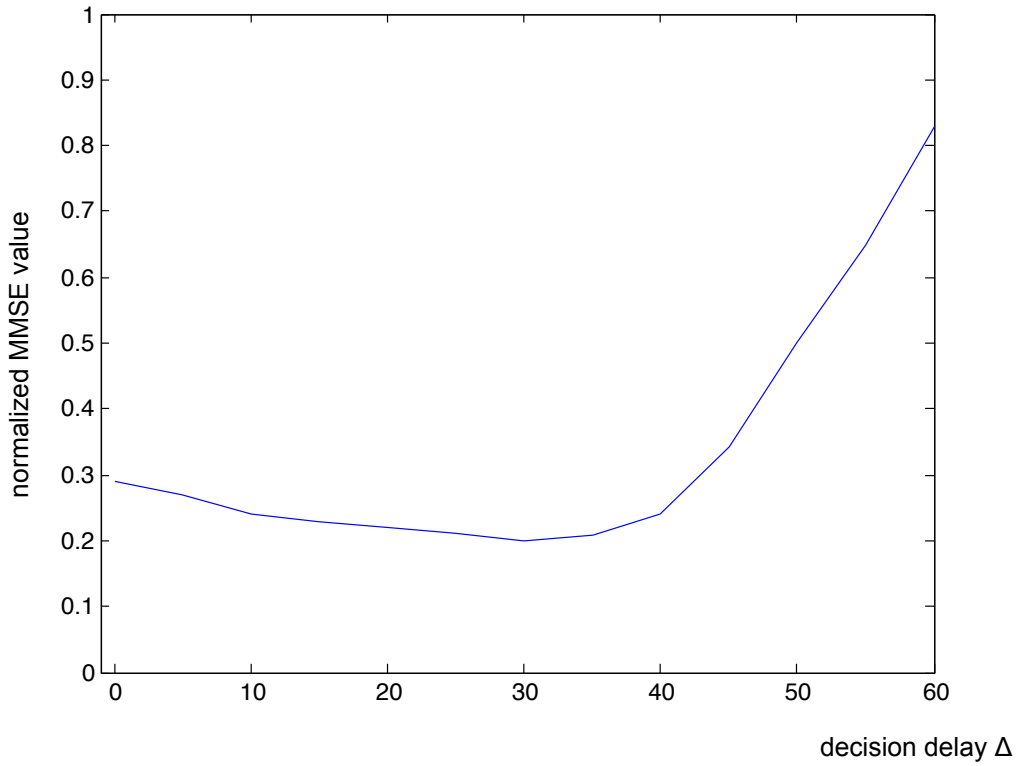

Figure 3-8: MMSE normalized by δ_x^2 as a function of the equalization decision delay Δ for channel length $N_c = 24$ and equalizer length $N = 40$.

Last, we show in Figure 3-9 the effect of the channel SNR on the MMSE for $N_1 = 7, N_2 = 23$, and $N = 30$. The MMSE decreases monotonically with SNR. This can be understood by rewriting Equation 3.42 as

$$\frac{\text{MMSE}}{\delta_x^2} = 1 - \mathbf{r}_\Delta^H [HH^H + \frac{1}{SNR} I]^{-1} \mathbf{r}_\Delta. \tag{3.42}$$

As SNR increases, the values of the diagonal elements in the positive-definite matrix $[HH^H + \frac{1}{SNR} I]^{-1}$ increase. As a result, $\mathbf{r}_\Delta^H [HH^H + \frac{1}{SNR} I]^{-1} \mathbf{r}_\Delta$ increases and the normalized MMSE decreases.

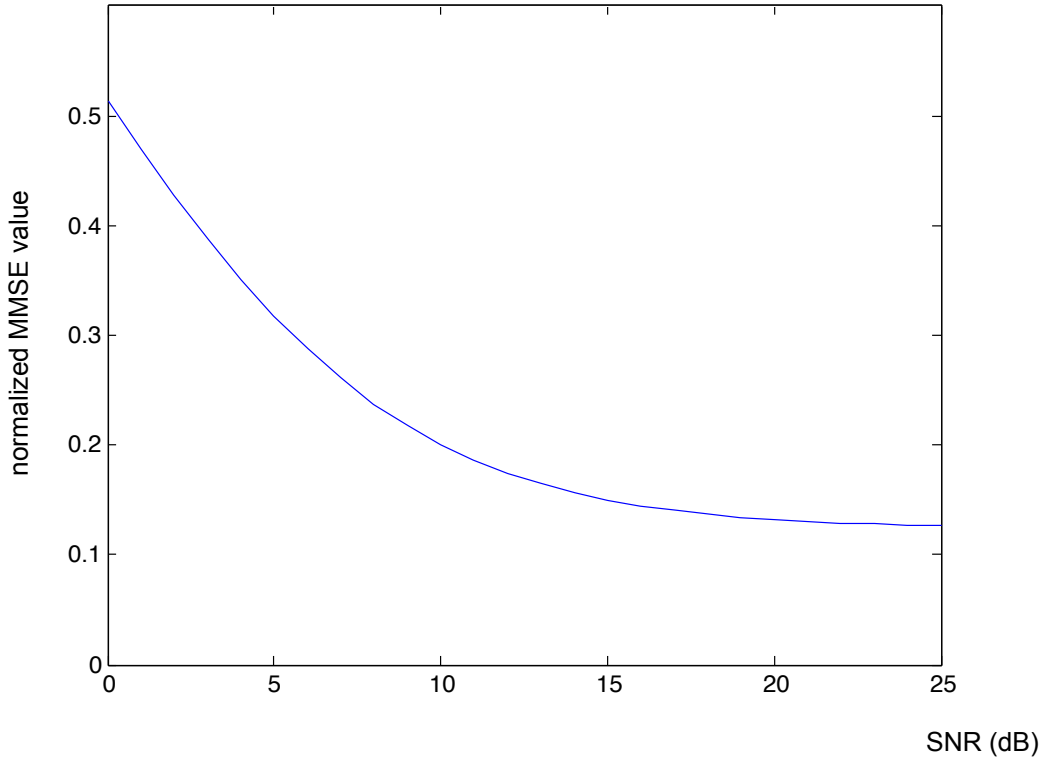

Figure 3-9: MMSE normalized by δ_x^2 as a function of the channel SNR for N_1=7, N_2=23, and $N = 30$.

■ 3.2 Decision Feedback Equalization

In this section, the equalization performance using Decision Feedback Equalizers (DFEs) is discussed. First, the DFE structure and its assumption are stated in Section 3.2.1. Next, the equalization MSE is derived in Section 3.2.2. Finally, Section 3.2.3 evaluates the MMSE value under various equalizer specifications. These results are useful for developing the sparse equalizers as will be shown in Chapter 5.

■ 3.2.1 DFE structure and assumption

The basic structure of DFE consists of a feedforward filter (FFF), a feedback filter (FBF) and a memoryless decision device. The typical framework of a Decision Feedback Equalization system is given in Figure (3-10).

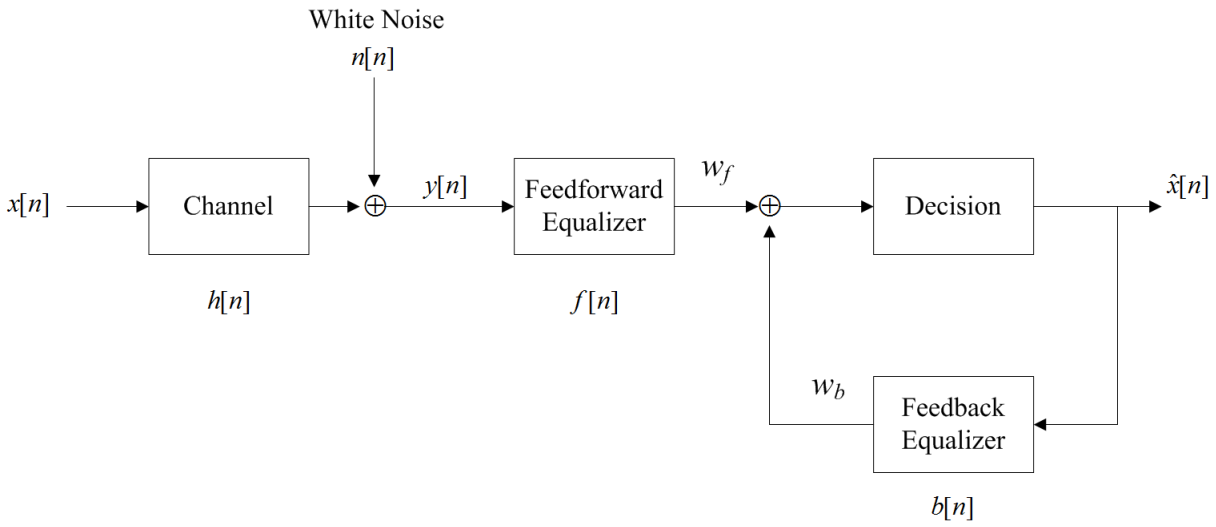

Figure 3-10: Equalization using DFE

The design of decision feedback equalizers has been traditionally carried out assuming that the past estimations are error-free, thus simplifying the mathematical derivations involved. The analysis of the performance becomes very complicated when past errors are taken into consideration. When an error is made by the receiver, the output of the FBF is no longer the desired value and the probability of subsequent errors is increased. Simulations also show that errors tend to occur in bursts. This can be understood from the structure of the feedback loop, where if one error occurs, it will be fed back and will bias the input to the decision block and thus most likely produce another error. This error propagation phenomenon is more severe when the tap weights and/or the number of the feedback taps are large. Duttweiler [12] derived an upper bound on the probability of error. The result is a relation between the true performance of the system and the performance it would have if the past errors were ignored in the FBF. In real applications, if the MMSE caused by the estimation errors is less than -30 dB, the correct past estimation assumption will not bring much loss in the DFE performance; If it is greater than -30 dB, the error is believed to severely affect the equalization results and becomes non-negligible. Figure 3-11 shows the histogram of the MSE caused by past esimation errors based on 1270 different channel setups. The MSE is below -30 dB in over 95% of the cases.

To mitigate the error propagation, a couple of techniques can be applied as proposed by [9], [10] and [11]. Typical approaches include inserting an additional delay in the decision-making device and estimating based on a block of symbols to gather more information. Other methods turn to state machines to model the error propagation and cancel the error according to the probabilistic model. Given that the error propagation in the problem of this thesis produces a small MSE, as is shown in Figure 3-11, for all the following analysis, correct past estimations are assumed.

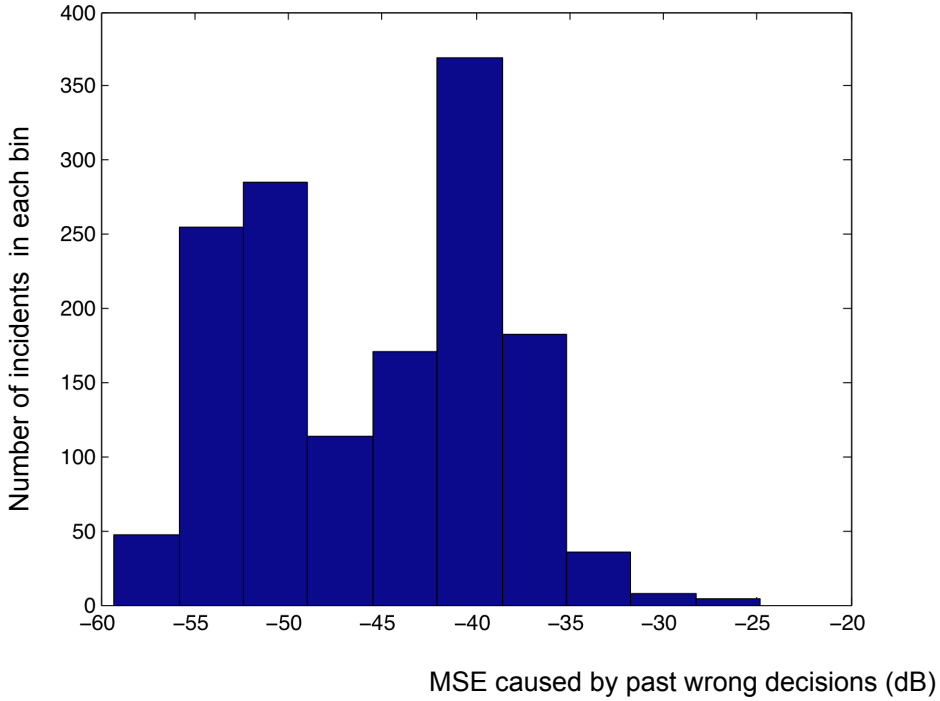

Figure 3-11: Histogram of the equalization MSE caused by past estimation errors

3.2.2 Problem Reductions

This section aims at obtaining the MSE as a function of the FFF and FBF coefficients. Section 3.1.1 has demonstrated that for all types of channels, by grouping the samples and constructing channel transfer matrix $\mathbf{H}$ in a specific way, the transmission system resembles a discrete SISO channel. Therefore, this section ignores the discussion on channels and directly analyzes discrete SISO channel equalizations.

Denote $\mathbf{w_f}$ as the feedforward filter coefficient vector, and $\mathbf{w_b}$ as the feedback filter coefficient vector respectively. In the context of this thesis, we do not investigate the effect of the decision box and thus assume that the decision box is an identity system. The k^{th} error sample is defined as

$$e_k = x_{k-\Delta} - (\mathbf{w}_f^H \, y_{k:k-N_f+1} - \mathbf{w}_b^H \, \hat{x}_{k-\Delta-1:k-\Delta-N_b}) \tag{3.43}$$

where $\hat{x}_{k-\Delta-1:k-\Delta-N_b}$ is the output vector segment that represents the estimations for $x_{k-\Delta-1:k-\Delta-N_b}$. Assuming correct past estimations, we can be substitute $x_{k-\Delta-1:k-\Delta-N_b}$ for $\hat{x}_{k-\Delta-1:k-\Delta-N_b}$, and therefore

$$e_k = x_{k-\Delta} - \underbrace{\left[\mathbf{w_f}^H \;\; \mathbf{w_b}^H \right]}_{\equiv \mathbf{w}^H} \underbrace{\begin{bmatrix} y_{k:k-N_f+1} \\ x_{k-\Delta-1:k-\Delta-N_b} \end{bmatrix}}_{\equiv \tilde{y}} \tag{3.44}$$

where $\mathbf{w}$ is the coefficient vector in combination of the feedforward filter coefficient vector and the feedback filter coefficient vector, with length $(N_f + N_b)$.

As we will see soon, it is useful to define the auto-correlation and the cross-correlation in the following way.

$$\begin{aligned} R_{\tilde{y}\tilde{y}} &\equiv E[\tilde{y}\tilde{y}^H] \\ R_{\tilde{y}x} &\equiv E[\tilde{y}\, x_{k:k-N_f-v+1}^H]. \end{aligned} \tag{3.45}$$

Furthermore, denote $\tilde{\mathbf{r}}_\Delta = R_{\tilde{y}x}\mathbf{1}_\Delta$, and $T_\Delta = \begin{bmatrix} 0_{N_b\times(\Delta+1)} & I_{N_b} & 0_{N_b\times(N_f+N_c-N_b-\Delta-1)} \end{bmatrix}^H$. Plugging in $\tilde{y}$, Equation 3.45 becomes

$$R_{\tilde{y}\tilde{y}} = \begin{bmatrix} R_{yy} & E[\,|x_{k-\Delta}|^2\,]HT_\Delta \\ E[\,|x_{k-\Delta}|^2\,]T_\Delta^H H^H & E[\,|x_{k-\Delta}|^2\,]I_{N_b} \end{bmatrix} \tag{3.46}$$

and

$$R_{\tilde{y}x} = \begin{bmatrix} E[\,|x_{k-\Delta}|^2 H \\ E[\,|x_{k-\Delta}|^2 T_\Delta^H \end{bmatrix}. \tag{3.47}$$

With the above notations, the MSE can be written as

$$\text{MSE} = E[\,|x_{k-\Delta}|^2\,] - \mathbf{w}^H \tilde{\mathbf{r}}_\Delta - \tilde{\mathbf{r}}_\Delta^H \mathbf{w} + \mathbf{w}^H R_{\tilde{y}\tilde{y}} \mathbf{w} \tag{3.48}$$

which is also a quadratic function of the equalizer coefficients $\mathbf{w} = \mathbf{w_f}\,;\mathbf{w_b}$.

■ 3.2.3 The effect of Decision Delay, FFF lengh, and FBF length on the MMSE

In most practical situations, computational and implementational complexity considerations often place a constraint on the maximum number of total (the sum of feedforward and feedback) filter coefficients that can be used. In addition, the decision delay setting could affect performance significantly as well, as is in the case of LE. The effect of decision delay becomes more severe when the filter is short. Therefore, it becomes important to understand the dependence of the finite length MMSE-DFE performance on the number of its feedforward filter taps N_f, the number of its feedback filter taps N_b, and on the decision delay Δ in order to set them properly. These results are useful in the sparse DFE design as we will show in Chapter 5.

Currently, all techniques for setting these three parameters, (N_f, N_b, Δ) are either ad-hoc or computationally intense, which requires an exhaustive search over these three parameters and inversion of a matrix whose size is equal to the total number of feedforward and feedback taps for each step in the search. This optimal 2-dimensional search algorithm is presented as follows.

Using the Orthogonality Principle of linear least square estimation, to minimize the MSE in Equation 3.48, w_{MMSE} is chosen to be $R_{\tilde{y}\tilde{y}}^{-1}\tilde{r}_\Delta$. Plugging in the w_{MMSE} into Equation 3.48, we get

$$\text{MMSE}(N, N_b, \Delta) = \text{ Constant} - \mathbf{r}_\Delta^H R_{\tilde{y}\tilde{y}}^{-1} \mathbf{r}_\Delta^* \tag{3.49}$$

Hence, for any choice of the triplet (N, N_b, Δ) Equation 3.49 is used to evaluate its performance. Unfortunately, there is no closed-form solution for the optimum triplet (N, N_b, Δ)

Table 3.1: MMSE with respect to N_f and Δ

	$\Delta=4$	$\Delta=5$	$\Delta=6$	$\Delta=7$	$\Delta=8$	$\Delta=9$	$\Delta=10$	$\Delta=11$	$\Delta=12$	$\Delta=13$	$\Delta=14$
N_f=7	0.3000	0.2800	0.2670	0.2200	**0.2011**	0.8437					
N_f=8	0.3000	0.2800	0.2670	0.2176	**0.1977**	0.1979	0.8437				
N_f=9	0.3000	0.2800	0.2670	0.2176	0.1963	**0.1796**	0.1805	0.8437			
N_f=10	0.3000	0.2800	0.2670	0.2176	0.1963	0.1744	**0.1712**	0.1805	0.8437		
N_f=11	0.3000	0.2800	0.2670	0.2176	0.1963	0.1744	**0.1699**	0.1711	0.1805	0.8437	
N_f=12	0.3000	0.2800	0.2670	0.2176	0.1963	0.1744	0.1699	**0.1682**	0.1711	0.1805	0.8437

that best balances the performance and complexity, and it can only be found by numerical search. The optimal triplet (N, N_b, Δ) that gives the least MMSE(N, N_b, Δ) is used to implement the DFE. Under the implementation constraint $N_f + N_b = N_{total}$ (where N_{total} is the given constraint), determining the optimum triplet is computationally intense since it requires an exhaustive 2-dimensional search and computation of the inverse $R_{\tilde{y}\tilde{y}}$.

Several theorems are given in [14] on the relationship between the three parameters under certain conditions. Suppose the channel length is N_c. The two theorems that are related to this thesis are described as follows:

Theorem I: If $N_b \geq N_c$, MSE$(\Delta = 0) \geq$ MSE$(\Delta = 1) \geq ... \geq$ MSE$(\Delta = N_f - 1)$, where MSE$(\Delta = k)$ is the MMSE when $\Delta = k$ with N_f and N_b being fixed.

Theorem II: If $N_b \geq N_c$, there must exist a positive N_0 so that when $N_f > N_0$, MSE$(\Delta = N_f - 1) = \min\{\text{MSE}(\Delta = N_f - 1), ..., \text{MSE}(\Delta = N_f + N_c - 1)\}$.

From the two theorems, if $N_b \geq N_c$ and N_f is large enough, the optimum decision delay is $N_b - 1$. There is no closed-form solution for N_0 defined in theorem II, but it can be verified via extensive simulations. We show in Table 3.2.3 the MMSE values with respect to N_f and Δ. The channel response $h[n]$ is given by $h = 1 + a_1 h[n-3] + a_2 h[n-7]$. The feedback filter length $N_b = N_c = 8$. For illustration, the numbers in bold font in Table 3.2.3 correspond to the optimum Δ for the specified N_f. It is shown that, in this example, the N_0 defined in Theorem II is 11, i.e., when $N_f > 11$, the optimum Δ equals $N_f - 1$; otherwise it does not. It is also shown that $N_b \geq N_c$, MSE$(\Delta = 0) \geq$ MSE$(\Delta = 1) \geq ... \geq$ MSE$(\Delta = N_f - 1)$ holds for every N_f, as expected by Theorem I.

Chapter 4
L_0 norm minimization under a quadratic contraint

In this chapter, the sparse equalizer design problem under a single MSE constraint is formulated. In Section 4.1 we show that it can be reduced to an l_0-norm minimization problem under a quadratic constraint. The viewing of the sparse equalizer design as an l_0-norm minimization problem under a quadratic constraint is first established by Wei in [2]. Next, in Section 4.2, three previously developed classes of algorithms are applied to solve this problem. Two of the algorithms discussed were originally investigated by Wei in [2].

■ 4.1 Problem Formulation

■ 4.1.1 Norm

Before defining the l_0-norm minimization problem, the definition of Norm is given in this subsection.

p-norm

For a real number $p \geq 1$, the p-norm or l_p-norm of x is defined by

$$\|x\|_p = (|x_1|^p + |x_2|^p + \ldots + |x_n|^p)^{\frac{1}{p}}. \tag{4.1}$$

Note that for p = 1, $\|x\|_1 = (|x_1| + |x_2| + \ldots + |x_n|)$ is the taxicab norm or Manhattan norm; For p = 2, $\|x\|_2 = (|x_1|^2 + |x_2|^2 + \ldots + |x_n|^2)^{\frac{1}{2}}$ is the Euclidean norm; As p approaches $\infty, \lim_{p\to\infty}(|x_1|^p + |x_2|^p + \ldots + |x_n|^p)^{\frac{1}{p}} = \max_n\{x_1, x_2, \ldots, x_n\}$the p-norm approaches maximum norm.

For all $p \geq 1$, the p-norms and maximum norm as defined above indeed satisfy the properties of a "length function" (or norm), which are that: (1) only the zero vector has zero length; (2) the length of the vector is positive homogeneous with respect to multiplication

by a scalar; (3) the length of the sum of two vectors is no larger than the sum of lengths of the vectors (triangle inequality). Abstractly speaking, this means that R^n together with the p-norm is a Banach space [18].

This definition is still of some interest for $0 < p < 1$. Accordingly, Equation 4.1 defines an absolutely homogeneous function. However, it does not define a norm [1], because it violates the triangle inequality, i.e, is not subadditive. However, the resulting function does not define an F-norm, because it is not subadditive. In R^n for $n > 1$, the formula for $0 < p < 1$

$$\|x\|_p = (|x_1|^p + |x_2|^p + \ldots + |x_n|^p)$$

defines a subadditive function, which defines an F-norm. This F-norm is not homogeneous.

zero norm

When p goes to zero, it is referred to as zero norm. There are two definitions of l_0 norms.

The mathematical definition of the l_0 norm was established by Banach's Theory of Linear Operations. The space of sequences has a complete metric topology provided by the Fnorm $\sum_n 2^{-n} \frac{x_n}{1+x_n}$, which is discussed by Rolewicz [15] in Metric Linear Spaces. This l_0-normed space is studied in functional analysis, probability theory, and harmonic analysis.

Another function was called the l_0 "norm" by Donoho [16], whose quotation marks warn that this function is not a proper norm. Donoho suggested the terminology p-"norm" locally, on bounded sets. When this "norm" is localized to a bounded set, it is the limit of p-norms as p approaches 0, i.e.,

$$\lim_{p \to 0} (|x_1|^p + |x_2|^p + \ldots + |x_n|^p).$$

This is not a norm, because it is not continuous with respect to scalar-vector multiplication (as the scalar approaches zero); it is not a proper norm because it is not homogeneous. Defining $0^0 = 0$, Donoho's zero "norm" of x is equal to the number of non-zero coordinates of vector x.

[1] except in R^1, where it coincides with the Euclidean norm, and in R^0, where it is trivial.

In the context of this thesis, the second definition of l_0 "norm" in order to denote the number of non-zero elements in a vector. From this point on, the quotation marks are omitted for simplicity.

■ 4.1.2 Problem Formulation and Visualization

This section focuses on the sparse equalizer design under a pre-defined MSE constraint ξ. From Chapter 2 we know that typically an FIR MMSE equalizer is not sparse. Thus it is necessary to bring in a looser MSE constraint than the MMSE constraint. The problem can be formulated as

$$\begin{aligned} &\min_{w} \|\mathbf{w}\|_{\mathbf{0}} \\ s.t. \quad &\text{MSE} \leq \xi = \text{MMSE } \xi_m + \text{Excess Error } \xi_e \end{aligned} \tag{4.2}$$

From Chapter 3, we obtain the MSE as a function of the equalizer coefficients as follows

$$\begin{aligned} \text{MSE} &= \delta_x^2 - \mathbf{w}^H \cdot \mathbf{r}_\Delta - \mathbf{r}_\Delta^H \mathbf{w} + \mathbf{w}^H R_{yy} \mathbf{w} \\ &= \text{MMSE} + (\mathbf{w} - R_{yy}^{-1}\mathbf{r}_\Delta)^H R_{yy} (\mathbf{w} - R_{yy}^{-1}\mathbf{r}_\Delta) \end{aligned} \tag{4.3}$$

where $\text{MMSE} = \delta_x^2 - \mathbf{r}_\Delta^H \mathbf{R}_{yy}^{-1} \mathbf{r}_\Delta$. Therefore, problem 4.2 can be rewritten as

$$\boxed{\begin{aligned} &\min_{w} \|\mathbf{w}\|_{\mathbf{0}} \\ s.t. \quad &(\mathbf{w} - \mathbf{w}_M)^H R (\mathbf{w} - \mathbf{w}_M) \leq \xi_e \end{aligned}} \tag{4.4}$$

where $\mathbf{w}_M = R_{yy}^{-1}\mathbf{r}_\Delta$ and $R = R_{yy}$.

This problem is visualized in Figure (4-1). The feasible set that satisfies the quadratic constraint is an ellipse centered at $\mathbf{w}_M$ in the N-dimensional space. The size of the ellipse is determined by the excess error tolerance ξ_e. The problem is to find the most sparse $\mathbf{w}$ in this region.

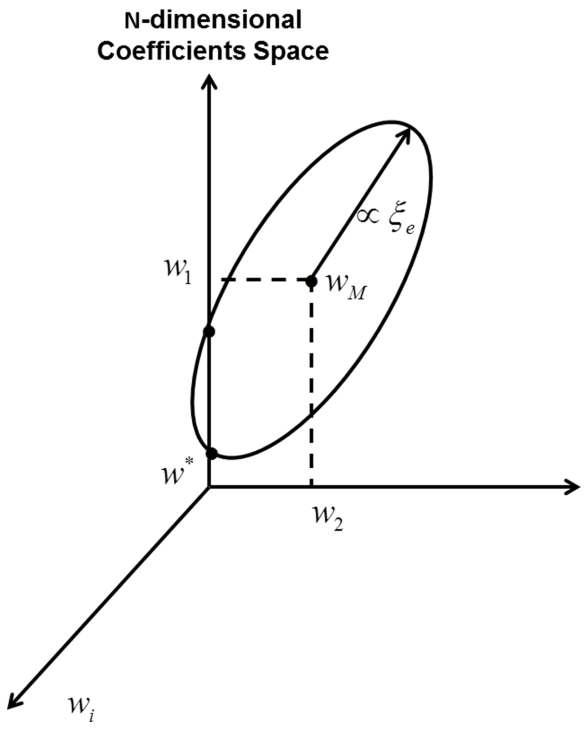

Figure 4-1: Visualization of zero-norm minimization under one quadratic constraint

■ 4.1.3 Comparison with Compressive Sensing

Zero-norm in compressive sensing helps in compressing image signals by eliminating coefficients that are zero or very small, while considering only the large and needed signal structures. Compressive sensing asserts that a sparse signal can be recovered from much fewer samples or measurements than the number of samples suggested by Shannon- Nyquist's traditional method. Compressive sensing is based mainly on the principles of:

(1) sparsity, relevant to the signal to be compressed;

Sparsity of the signal helps expressing the idea that the information rate of the signal is much smaller than suggested by its bandwidth. Compressive sensing helps in exploiting the fact that most of the natural signals are compressible and sparse, in the sense that they have a specific representation when they are expressed in the proper basis, say ψ.

(2) incoherence, relevant to the sensing models;

Incoherence expresses the idea that the signals having a sparse representation must be spread out in the domain in which they are acquired. Incoherence suggests that unlike the signals of interest, the sampling waveform has an extremely dense representation in ψ.

Compressive sensing is analyzed in three basic stages.

1. Matrix transforming, where the orthogonal matrix ψ changes.
2. Encoding, where measurement matrix is encoded by ψ, to form $\phi_i =< \psi_i, A >$.
3. Decoding, where $K+1$ values recover A under ψ.

Therefore, the problem is to find a stable measurement matrix A such that there is minimum or no damage to the compressed signal during the dimensional reduction from R^N to R^{K+1}. Also, a reconstruction algorithm is needed to recover the original signal from the $K+1$ measurements.

That defines the compressive sensing problem

$$\min_{w} \|\mathbf{w}\|_0 \quad s.t. \;\; A\mathbf{w} = \mathbf{b},$$

where the original N-dimensional signal, say w_M is recoverable from the (K+1)-dimensional b. In general, the exact solution to the above problem can be solved by a subset problem of the form

$$\begin{aligned} &\min_{w} \|w\|_0 \\ s.t. \;\; &\|b - Aw\|_2^2 \leq \epsilon. \end{aligned} \tag{4.5}$$

for a sufficiently small ϵ, and is a NP-hard problem. The region of feasible sets in the (K+1)-dimensional space is depicted in Figure 4-2.

When formulated in this way, the compressive sensing problem shares some similarities with our sparse equalizer filter design problem. Rewriting 4.5 as

$$\begin{aligned} &\min_{w} \|\mathbf{w}\|_\mathbf{0} \\ s.t. \;\; &(w - A^{-1}b)^T A^T A (w - A^{-1}b) \leq \epsilon, \end{aligned} \tag{4.6}$$

By comparing the sparse filter design problem 4.4 and the compressive sensing problem 4.6, we see that if R can be decomposed into $A^T A$ and w_M can be written as $w_M = A^{-1}b$, where b is a lower dimensional vector, the two problems are very similar. However, this requires

that w_M is compressible to a lower dimensional space without any information loss, which is often not true in the filter design problem. Therefore, our problem is distinct from the compressive sensing problem.

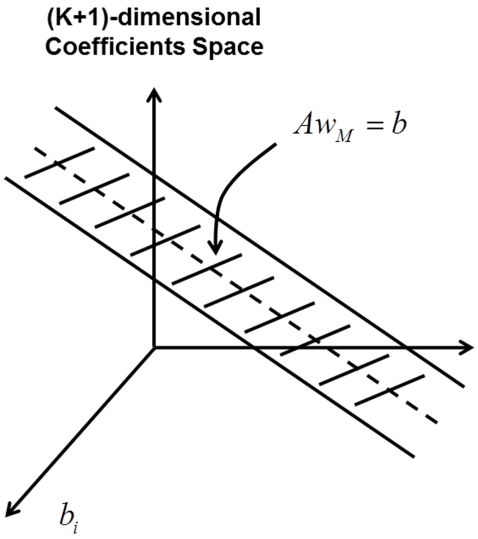

Figure 4-2: Visualization of compressive sensing

■ 4.2 Problem Solutions

In this section we discuss the solutions to problem 4.4. It has been seen that the l_2-norm does not reflect the sparsity of the vector very well, and the l_1-norm recovers the sparse signals with very high probability (Baranuik [20]). This can further be challenged by the actual l_0-norm optimization that counts the number of zero elements in w. The straightforward exhaustive search method is numerically unstable and NP-hard, which requires enormous iterations on all the non-zero elements. This is the major factor that many algorithms avoids using the l_0-norm to solve this optimization problem, and instead introducing relaxations or bounds to either find the sub-optimal solution or fasten the optimality searching speed.

■ 4.2.1 Sub-optimal Solution

In this section, a heuristic algorithm for solving 4.4 that we refer to as successive thinning is presented. This method is discussed in Wei [2] as a low-complexity algorithm to compute the sparse solution. Similar approaches were proposed in Sui *et al* [19] for channel

equalization. The separability can be with respect to a generalized notion of summation, e.g., a product of non-negative functions is also regarded as separable.

Here the algorithm is adapted to problem 4.4. The algorithm was applied to sparse filter design by [2]. The basic idea is to re-optimize the non-zero coefficients by setting more and more coefficients to zero. It terminates when the re-optimized result no longer satisfies the error constraint. Problem 4.4 may be solved by determining for each $K = 1, 2, \ldots, N$ whether a feasible solution with K zero coefficients exists. As K increases from 1 (or decreases from N), since the number of subsets is proportional to $\binom{N}{K}$, the complexity of evaluating the existence grows as least as fast as $\binom{N}{K}$. For K = 1, the successive thinning algorithm carries out the minimization of the excess error. We denote $Z^{(1)}$ as the resulting minimizing subset (in this case a single index). For K = 2, we restrict the minimization to only those sets of indices that include $Z^{(1)}$. By induction, let $Z^{(K)}$ represent the minimizer over this restricted collection of subsets of size K. For larger values of K, the subsets considered in the minimization are constrained to contain $Z^{(K-1)}$, the minimizer for the previous step $K-1$, thus limiting the search to just adding one new index to $Z^{(K-1)}$. As stated at the beginning of this paragraph, the algorithm terminates when the minimum value corresponding to $Z^{(K+1)}$ exceeds the MSE constraint for some K, at which point the last feasible subset $Z^{(K)}$ becomes the final subset of zero-valued coefficients. Given $Z^{(K)}$, i.e., the indices of the zero coefficients, we may then solve for the values of the non-zero coefficients to produce a feasible solution with zero-norm equal to $N-K$. The successive thinning algorithm greatly reduces the number of subsets that are explored in the optimal case. The number of subsets evaluated in the K^{th} step is at most $N-K+1$, corresponding to the $N-(K-1)$ choices for the index to be added. Since the number of steps can be at most N, the total number of subsets grows only quadratically with N. As we will see in the simulation results given in section 5.3, the successive thinning algorithm discussed in this section produces solutions that are in many instances either optimally sparse or close to optimal. Optimality or near-optimality is certified by running the branch-and-bound algorithm [2] described in the next section, which does guarantee an optimal solution. Thus the successive thinning algorithm is useful as a method for obtaining sparse solutions with relatively low complexity.

4.2.2 The Branch-and-Bound Solution

In Section 4.2.1, low-complexity algorithms were presented for solving problem 4.4 but were not aiming at obtaining the optimal solution. In this section, attention is turned to the optimal algorithms. Specifically, an algorithm based on a standard approach to combinatorial optimization known as branch-and-bound is considered due to its ability to avoid an exhaustive enumeration of all 2^N potential subproblems. The branch-and-bound procedure is applied to problem 4.4 in [2]. In the context of this thesis, only an overview of our branch-and-bound algorithm is presented. More details are described by [2]. Additional background on branch-and-bound can be found in [17].

First we introduce a binary indicator variable u_n with the u_n=0 if the corresponding $w_n = 0$ and $u_n = 1$ otherwise. Then the sum of the indicator variables u_n equals to the zero-norm of w. Using this fact, problem 4.4 can be re-written as follows:

$$\begin{aligned} &\min_{w,u} \sum_{n=1}^{N} u_n \\ &s.t.\ (w - w_M)^H R(w - w_M) \leq \xi_e, \\ &\quad |w_n| \leq B_n u_n \ \ \forall n, \\ &\quad u_n \in \{0, 1\} \ \ \forall n, \end{aligned} \tag{4.7}$$

where B_n is a positive constant for each n. The second constraint ensures that u_n behaves as an indicator variable, specifically by requiring that $w_n = 0$ if $u_n = 0$ and also forcing u_n to zero if $w_n = 0$ because the sum of the u_n is being minimized. When $u_n = 1$, the second constraint becomes a bound on the absolute value of w_n . The constants B_n are chosen to be large enough so that these bounds on $|w_n|$ do not affect feasible sets of w. The B_n values are determined in the relaxation part and is discussed in [2].

In the branch-and-bound procedure, problem 4.4 is successively divided into subproblems in a tree manner with fewer variables. The first two subproblems are formed by selecting an indicator variable u_n and making it to 0 in the left child subproblem and to 1 in the other one. Each of the two subproblems, if not solved directly, is divided into another two subproblems by fixing another indicator variable to 0 or 1. This process therefore

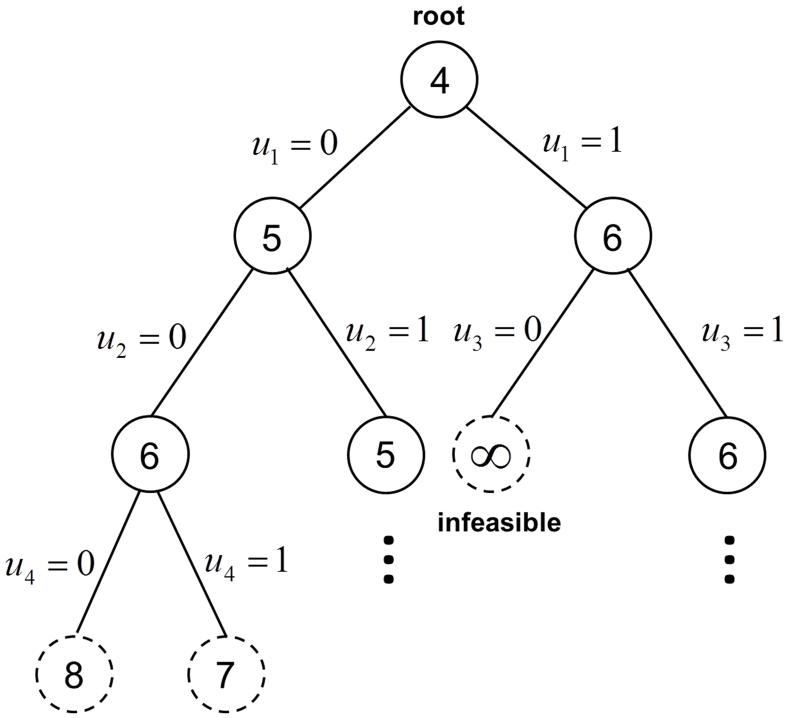

Figure 4-3: The subproblem tree of the branch-and-bound method

produces a binary tree of subproblems as depicted in Figure 4-3. A bound is described in [2] to help restrict the number of enumerations. The bounding part includes the computation of a lower bound on the optimal value of each subproblemInfeasible subproblems are denoted by a lower bound of $+\infty$. Since a child subproblem has one more constraint added than its parent problem, Now we explain the relationship of the lower bound between two generations and the passing of the lower bound.

the lower bound for the child cannot be less than that of the parent. Note that the lower bounds illustrated in Figure(4-3) is non-decreasing from the root to the leaves. In addition, feasible solutions may be found for certain subproblems. The algorithm keeps a dynamic record of the best feasible solution, i.e., the solution that produces the lowest cost so far, referred to as the incumbent solution in [2]. If the lower bound for a subproblem is equal to or higher than the cost of the incumbent solution, then the subproblem cannot lead to better solutions and can thus be eliminated from the tree along with all of its descendants. The elimination operation helps avoiding an exhaustive search over all potential subproblems,

although in worst-case examples the complexity of branch-and-bound remains exponential in N [17], for typical instances the situation can be much improved.

■ 4.2.3 Simple Linear Programming Algorithm

An even simpler approach to obtain a non-optimal solution to problem 4.4 is presented in this section. This greedy algorithm based approach has been developed in previous research including [1], [23] and [24]. The idea is to find the smallest element in the current equalizer coefficient vector and set it to zero. If the resulting coefficient vector still satisfies the MSE constraint, the above procedure is repeated. This method is visualized in Figure 4-4. At each iteration step, the feasible set is projected to a lower dimensional space by setting the current smallest coefficient to be zero. The algorithm terminates at step $N - K$ where the projection of the feasible set onto the K-dimensional space is empty.

To explain the non-optimality of this approach, note that in Figure 4-1, when a projection of the w_M on a lower dimensional space does not exist in the elliptic region, it is possible that a sparse solution does exist. In other words, it is possible that the projection of w_M on any axis is not in the elliptic region while the region does have intersections with some axises.

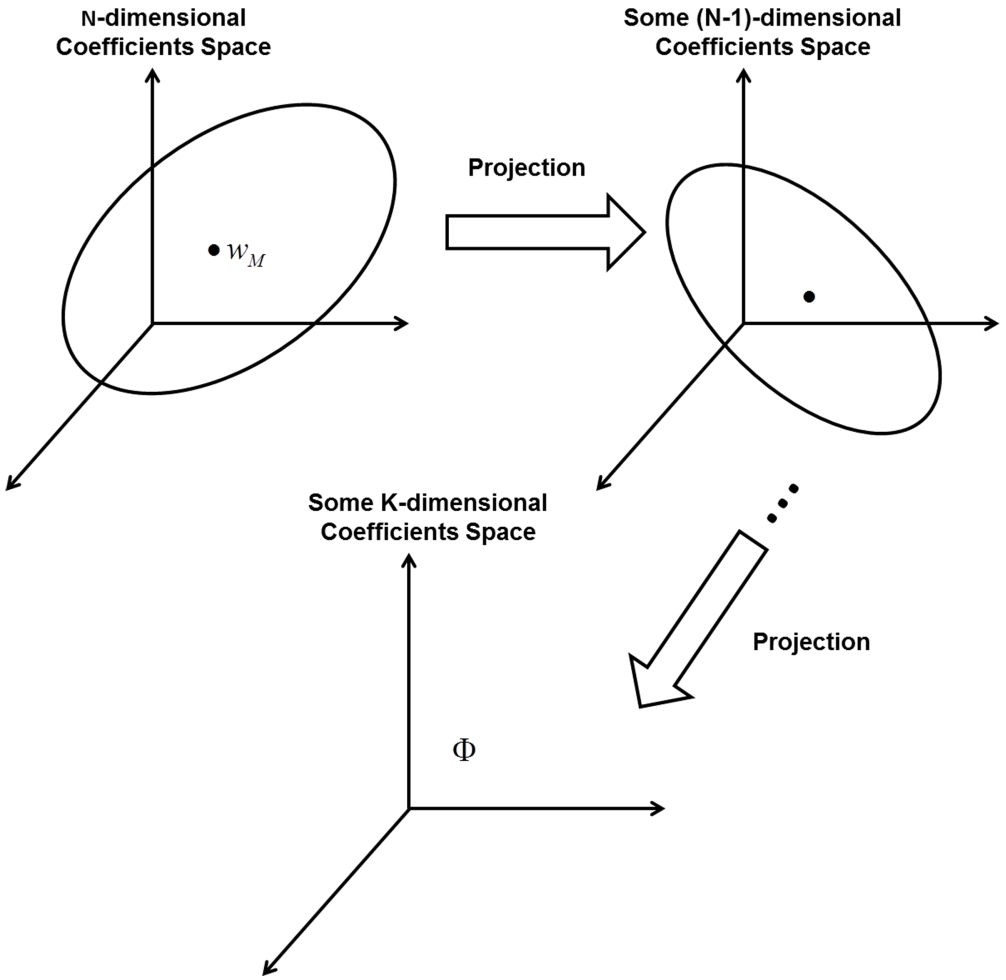

Figure 4-4: Visualization of the simple linear programming technique for the sparse equalizer design problem.

Chapter 5
Sparse Equalizer Design and Simulation Results

This chapter presents the simulation results for the sparse equalizer filter design problem defined in Chapter 4. The sparse equalizer filter design result can be affected by the channel specifications including SNR and the channel response, by the equalizer specifications including equalizer type and length, and by the algorithm specifications including the MSE tolerance. We examine the effect of the above factors in the following sections.

■ 5.1 The effect of Channel Specifications

In this section, sparse equalizer design is simulated for two types of channels, the simple multi-path channel and the Vehicular A channel respectively. Results are given to illustrate the dependency of the number of non-zero coefficients in the sparse equalizer filter on various channel setups. The optimal algorithm described in Section 4.2.2 is applied to eliminate the effect of non-optimal solutions.

■ 5.1.1 Simple Multi-path Channel

In the initial simulation, the channel response h[n] is chosen to represent an ideal multi-path channel with a direct path and two delayed paths. More precisely,

$$h[n] = \delta[n] + a_1\delta[n - N_1] + a_2\delta[n - N_2] \tag{5.1}$$

Linear equalizers are adopted in this subsection. Based on the discussion in Section 3.1.2, a good choice for Δ is $(N_f + N_c)/2$, and is used for all the analysis related to linear equalizer in this chapter.

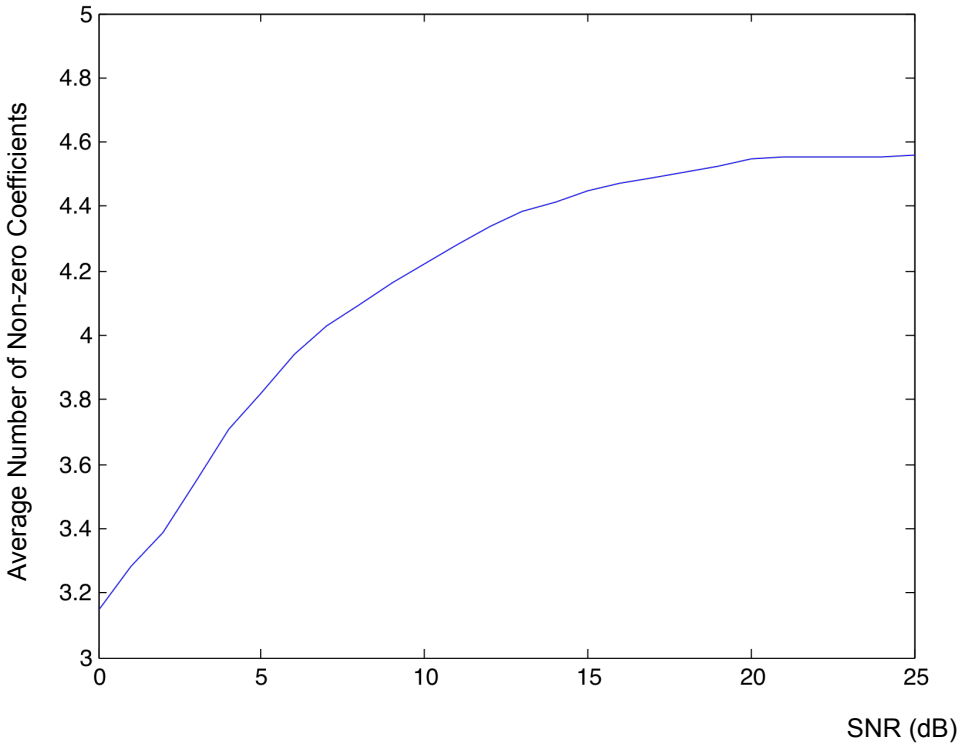

Figure 5-1: Number of nonzero taps vs. Channel SNR

■ 5.1.2 Practical Wireless Communication Channel

In this section we demonstrate the sparse equalizer design feasibility for a realistic communication channel. In practice, besides the physical propagation channel, transmitter and receiver filters have their system transfer function as well, which will also shape the transmitted signal. As is shown in Figure (5-2), we assume that the transmit and receive filters are square-root raised-cosine filters. The raised-cosine filter is a filter frequently used for pulse-shaping in digital modulation due to its ability to minimize the ISI. Its frequency-domain description is a piecewise function given by

$$H(\Omega) = \begin{cases} T, & |\Omega| \leq \frac{1-\beta}{2T} \\ \frac{T}{2}\left[1+\cos\left(\frac{\pi T}{\beta}\left(|\Omega|-\frac{1-\beta}{2T}\right)\right)\right], & \frac{1-\beta}{2T} \leq |\Omega| \leq \frac{1+\beta}{2T} \\ 0, & otherwise \end{cases} \tag{5.2}$$

Table 5.1: An example of Vehicular A multi-path Channel

a_k distribution	a_k (dB)	τ_k (ns)
Rayleigh	-5	21
Rayleigh	-6	76
Rayleigh	-8	127
Rayleigh	-9	213
Rayleigh	-11	350

where β is the roll-off factor which takes value between 0 and 1. The name raised-cosine stems from the fact that the non-zero portion of the frequency spectrum of its simplest form ($\beta = 1$) is a cosine function raised up to sit above the frequency axis. The excess bandwidth parameter β is set to 0.8.

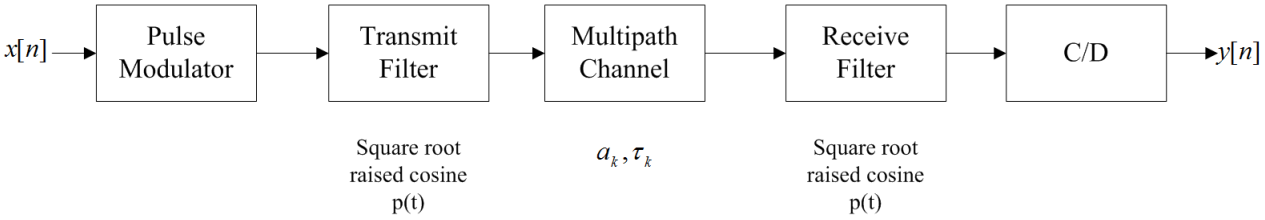

Figure 5-2: Vehicular A Channel Setup

An example of a five-delay-path channel propagation response can be expressed as

$$h(t) = \sum_{i=0}^{5} a_i \delta(t - \tau_i) \tag{5.3}$$

According to the Vehicular A multi-path Channel specifications described in the ITU-2000 standard [3], the parameter a_i and τ_i are given in Table 5.1.2 In particular, to obtain a representative evaluation on the equalization performance, we averaged 1000 cases. Each case we generate a_k and τ_i with their mean values determined by Table 5.1.2 and with variances to be 5% of the corresponding absolute mean values.

To fit into the context of the discrete equalization sparsity formulation, the effective discrete-time channel response is obtained by

$$h[n] = \sum_{i=0}^{5} a_i p(n - \tau_i) \tag{5.4}$$

where the pulse $p(t)$ is the convolution of the transmit and receive filter responses and the sampling period has been normalized to unity.

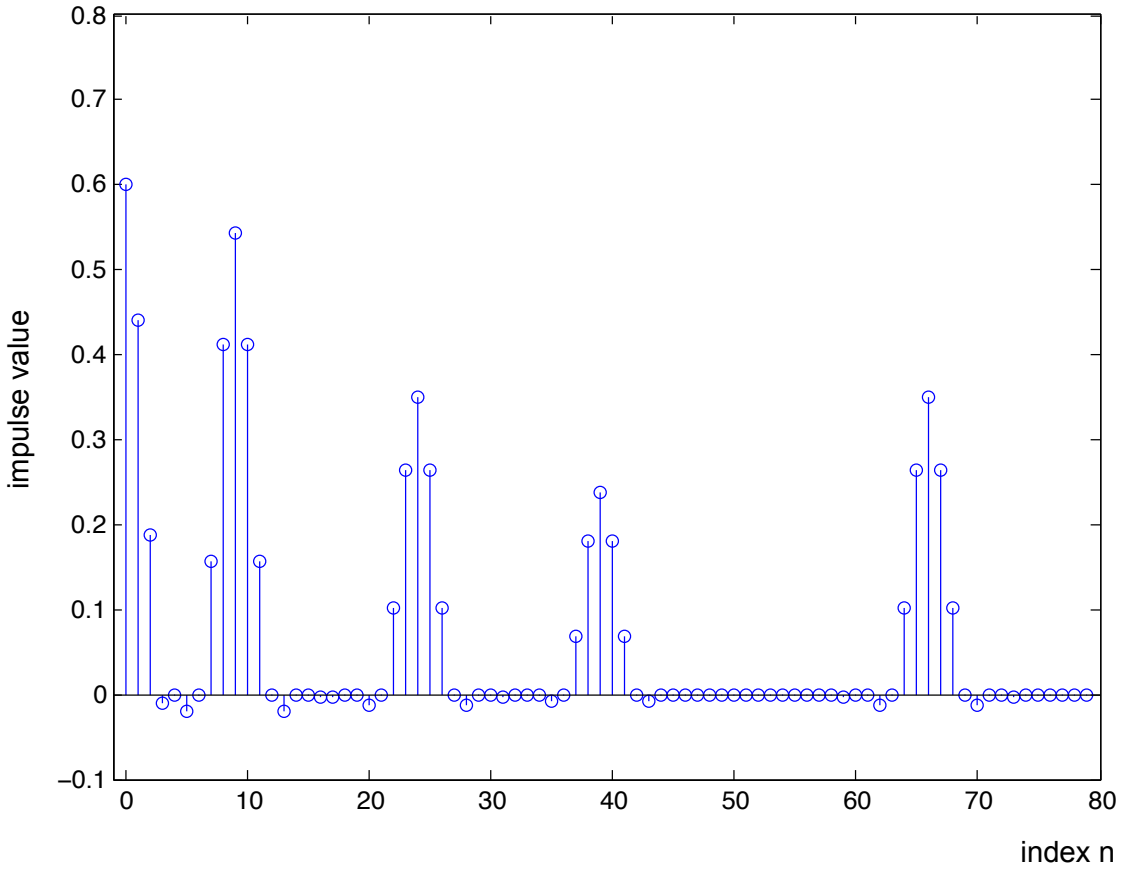

Figure 5-3: Effective discrete-time channel response for the Vehicular A multi-path channel example

The resulting discrete-time channel response is plotted in Figure 5-3. The remainder of the experimental setup is the same as in Section 5.1.1.

In Figure 5-4, we show the coefficient values for the length 50 equalizer for SNR=10 dB and a sparse equalizer with an MSE that is 20% higher. The sparse equalizer has about one

third as many non-zero coefficients as the MMSE equalizer. The larger coefficients in the MMSE equalizer tend to be retained in the sparse equalizer, including a cluster surrounding the largest coefficient that corresponds to the strongest path in the channel.

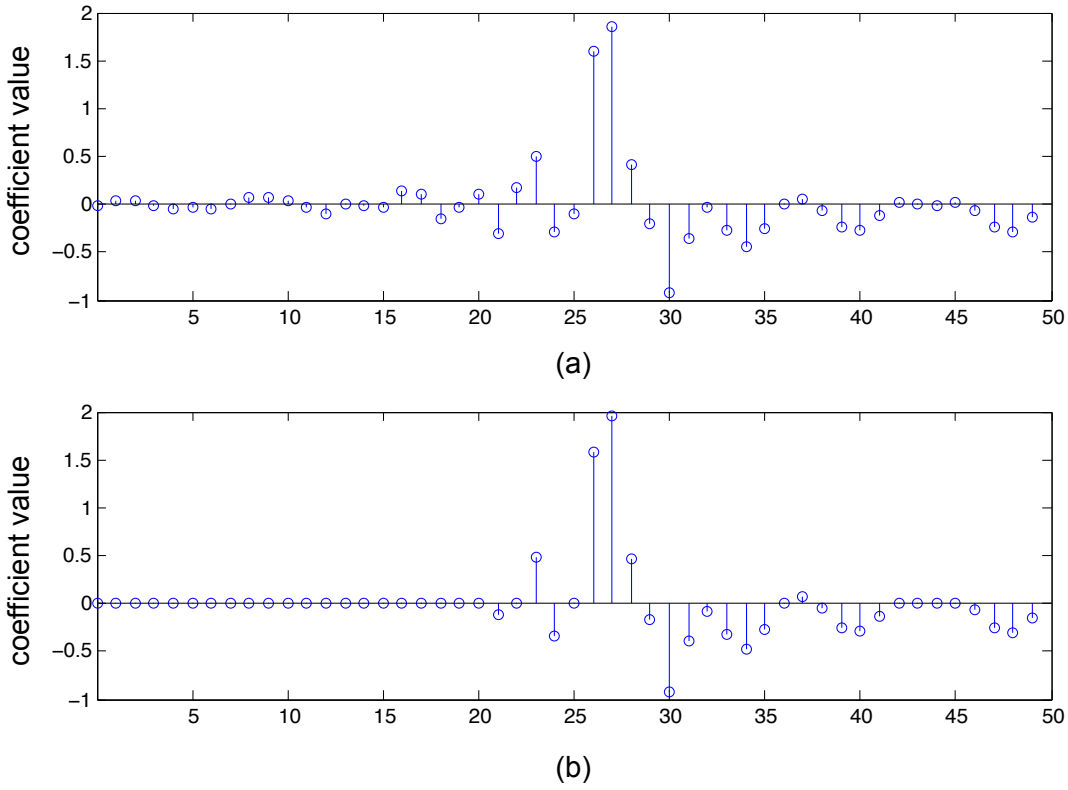

Figure 5-4: (a) Coefficient values for the length 50 MMSE equalizer with SNR = 10 dB. (b) A corresponding sparse equalizer with excess error ratio = 0.2.

■ 5.2 The effect of Equalizer Specifications

In this section, the simple multi-path channel given in Section 5.1.1 is implemented to analyze the effect of the equalizer specifications on the number of non-zero coefficients in the sparse equalizer. Again, the Branch-and-Bound algorithm is applied to eliminate the effect of non-optimal solutions.

Figure (5-5) shows the effect of the length N on both the MMSE and the number of non-zero coefficients in a sparse equalizer. For this experiment, the excess error ratio is fixed to 20%. SNR is 10 dB, and each data point again represents the average of 1600 (a1,a2) pairs. The staircase patterns can be explained by reference to Section 3.1.2. The MMSE

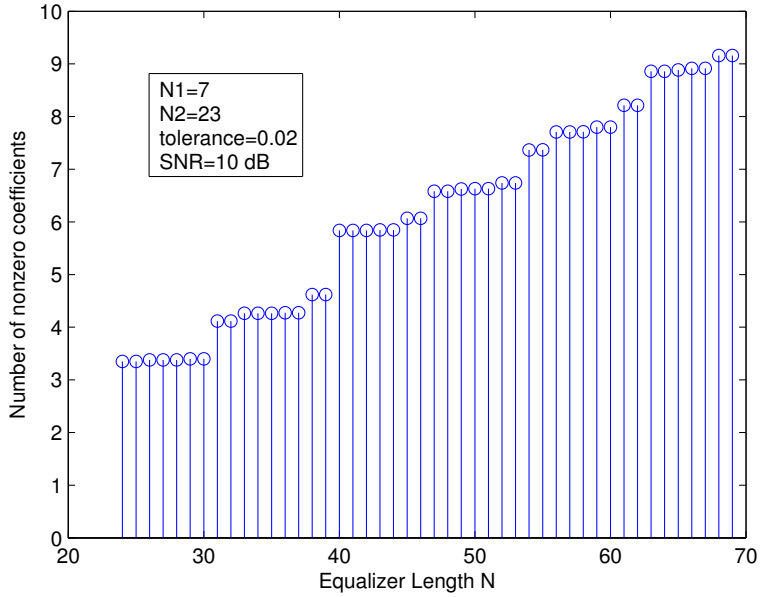

Figure 5-5: Number of nonzero taps vs. equalizer length

decreases the most at some special values of N, which means the excess error tolerance decreases at these special values of N. This error constraint change leads to an increase in the number of non-zero coefficients that are required. For example, for $N_1 = 7$ and $N_2 = 23$, the largest decreases occur at $30 = N_2 + N_1$, $39 = 2N_2 - N_1$, and $46 = 2N_2$, followed by smaller decreases at other integer combinations of N_1 and N_2.

Figure (5-5) compares the the number of non-zero coefficients using the linear equalizer and the decision feedback equalizer. For this experiment, the excess error ratio is fixed to 20%. SNR is 10 dB, and each data point again represents the average of 1600 (a1,a2) pairs. Based on the discussion in Section 3.1.2, the decision delay Δ is set to $(N_f + N_c)/2$ for linear equalizers. Based on the discussion in Section 3.2.3, the decision delay Δ is set to $N_f - 1$. N_b varies from 1 to N_c and the corresponding N_f varies from $N - 1$ to $N - N_c$. In (b), each data point represents the average of the N_c FFF/FBF length pairs. The error bar in (b) reflect the change of the number of non-zero taps as the FFF/FBF length ratio varies. The comparison between (a) and (b) demonstrate that using the DFE structure can achieve a more sparse equalizer than using the LE under the same equalizer length constraint and the MSE constraint for the same channel.

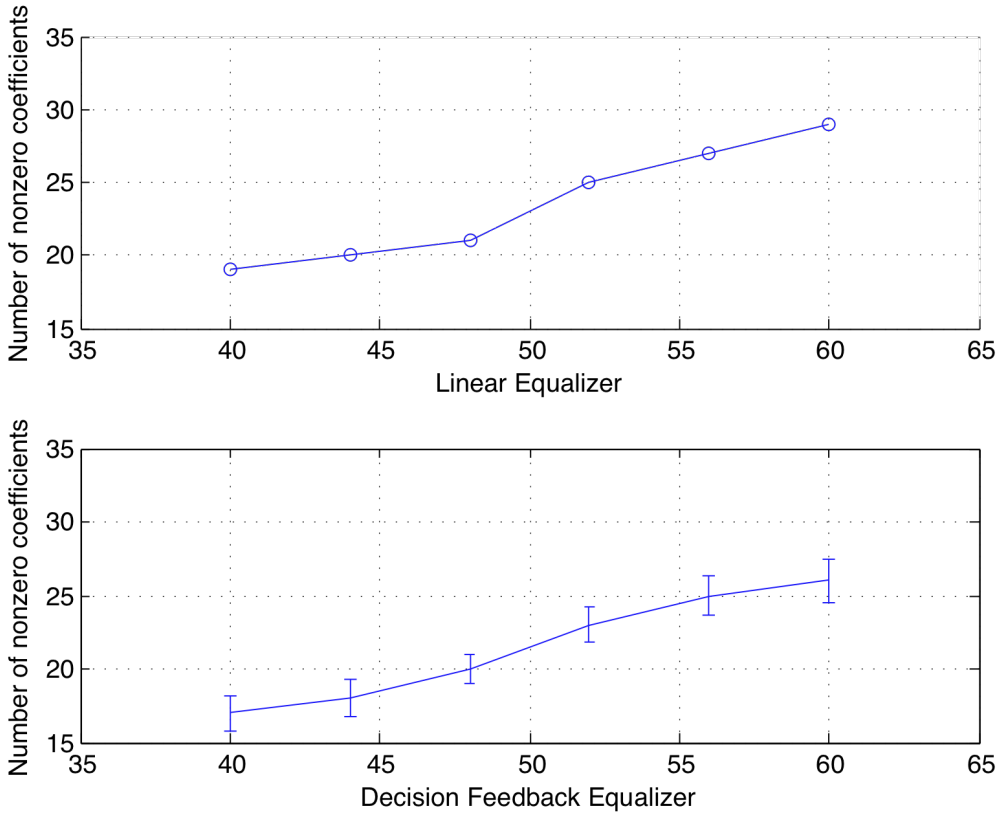

Figure 5-6: Number of nonzero taps vs. equalizer length using LE and DFE.

■ 5.3 The effect of Algorithm Specifications

Figure 5-7 shows the number of nonzero coefficients versus excess error ratio by applying the branch-and-bound algorithm. As a parameter of the algorithm input, the excess error ratio is defined as $(\xi - \xi_{\text{MMSE}})/\xi_{\text{MMSE}}$, where ξ represents the estimation error of the sparse equalizer filter, and ξ_{MMSE} denotes the estimation error of the MMSE equalizer filter. The number of non-zero coefficients, averaged over 1600 uniformly distributed (a1,a2) pairs, is viewed as a function of the excess error ratio for $N_1 = 7, N_2 = 23$, $N = 2N_2$, and SNR = 10 dB. The left-most point corresponds to the MMSE equalizer, which in general does not have any zero values. However, for sparse multi-path channels, the MMSE equalizer has a lot of coefficients with small values, which introduces small excess errors when forced to zero. Hence there is an abrupt decrease in the number of non-zero coefficients as soon as ξ/ξ_{MMSE} exceeds 1, followed by a rapid approach toward an asymptote. The gain in sparsity

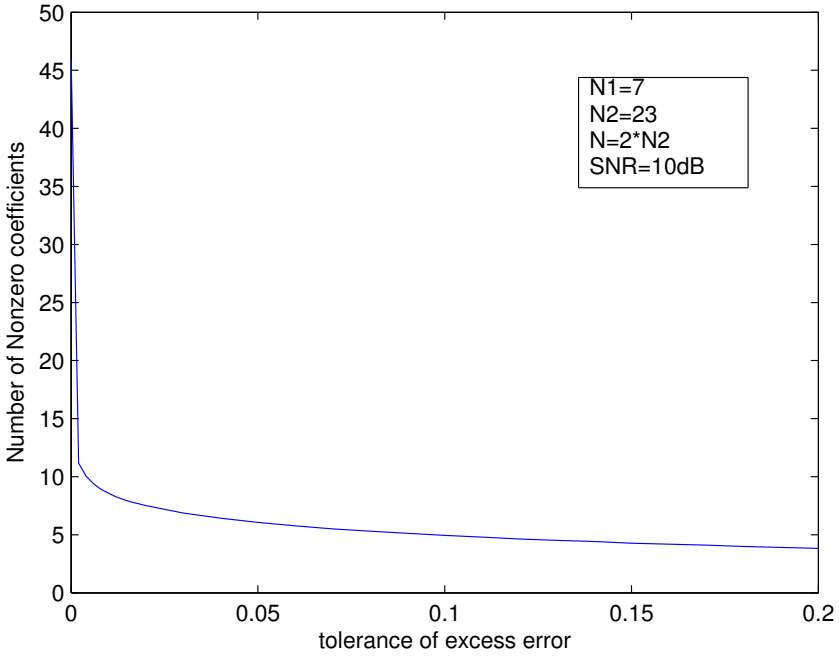

Figure 5-7: Number of nonzero taps vs. excess error

is slightly smaller for the higher SNR value and the behavior is similar for all values of N. It has been observed that the tradeoff between sparsity and MSE is quite favorable in the sense that the number of non-zero coefficients can be reduced substantially with only a small increase in the MSE.

Figure 5-8 plots the time and the corresponding number of non-zero coefficients at each iteration. The computing time tends to grow exponentially as the number of iteration increases. The number of non-zero coefficients in the sparse equalizer decreases at each iteration. Therefore, there is a tradeoff when we place a limit on the number of iterations. Observations indicate that the number of non-zero coefficients converges mostly in less than 20 iterations.

It is also helpful to examine the optimality of the lower bound given by the successive thinning algorithm. We have observed in the experiments that it is rare for the successive thinning solution to not be optimal. Figure (5-9) indicates that for over 85% of the 1600 experiments, the successive thinning algorithm gives an exact estimation of the actual cost. The tightness of the this non-optimal solution may be due to a number of factors present

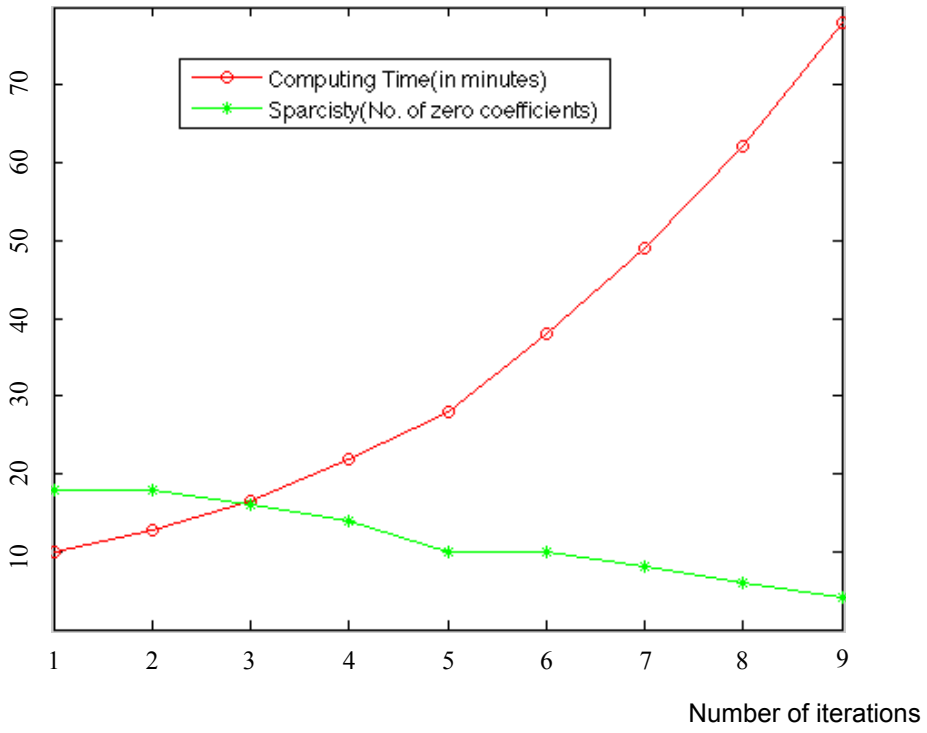

Figure 5-8: The number of non-zero coefficients and the computing time vs. the number of iterations in the Branch-and-Bound algorithm.

in this idealized example, including the high level of sparsity, the relative unambiguity regarding which coefficients should be non-zero, and the diagonally dominant structure of the matrix R. Moreover, considering the computational cost of the BNB algorithm shown in Figure 5-8, we can turn to the successive thinning method as a low-cost substitute.

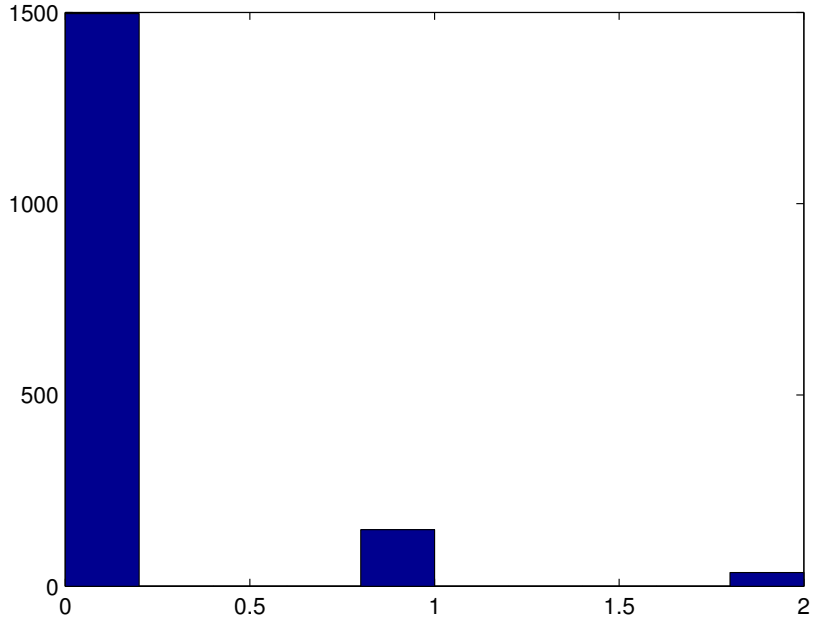

Figure 5-9: Histogram of the difference between the successive thinning result and the optimal result

Chapter 6
Zero-norm Minimization under Two Quadratic Constraints

In this chapter, sparse equalizer design is considered for MIMO channels with two MSE constraints. This problem arises when one or more user subsets have their individual MSE constraints other than the overall equalization MSE constraint. First, the MSE of the MIMO channel equalization is defined in Section 6.1. Next, Section 6.2 shows that this design problem can be formulated as a zero-norm minimization under two quadratic constraints. Due to the complexity to solve this problem, in Section 6.3 we propose a method to decouple the two constraints in the original problem, which will generate a sub-optimal solution. The simulation of this method is planned in Future Work (Section 7.1)).

■ 6.1 MIMO MSE

Suppose the MIMO channel has n_i inputs and n_o outputs, and the corresponding linear equalizer has n_o inputs and n_i outputs. The equalization MSE of a MIMO channel can be defined between each channel-input-equalization-output pair, and thus there are n_i of them. For example, the MSE between the l^{th} channel input and the equalization output is defined as

$$\text{MSE}_l = E[(x_k^{(l)} - \hat{x}_{k-\Delta}^{(l)})^2] \tag{6.1}$$

As we will see soon, it is useful to define the following auto-correlation and cross-correlation matrices in addition to Equations 3.35, 3.36, 3.37 and 3.38. Suppose the maximum channel length is N_c and the equalizer length is N_f.
The l^{th} $(N_f + N_c) \times (N_f + N_c)$ input auto-correlation matrix is given by

$$\mathbf{R}_{x^{(l)}x^{(l)}} = E[\mathbf{x}^{(l)}_{k+N_f-1:k-N_c}\, \mathbf{x}^{(l)H}_{k+N_f-1:k-N_c}] \quad (1 \le l \le n_i) \tag{6.2}$$

where

$$x^{(l)}_{k+N_f-1:k-N_c} = \begin{bmatrix} x^{(l)}_{k+N_f-1} \\ x^{(l)}_{k+N_f-2} \\ \vdots \\ x^{(l)}_{k-N_c} \end{bmatrix}$$

The l^{th} channel's input-output cross-correlation matrix is given by

$$\mathbf{R}_{x^{(l)}y} = E[\mathbf{x}^{(l)}_{k+N_f-1:k-N_c}\, \mathbf{y}^H_{k+N_f-1:k}] = R_{xx}H^H \tag{6.3}$$

Denote $\mathbf{1}_\Delta \equiv [\underbrace{0\,0\,\cdots\,0}_{\Delta} 1\ \underbrace{0\,0\,\cdots\,0}_{N_f+N_c-\Delta-1}]^H$. Then $x^{(l)}_{k-\Delta}$ can be rewritten as $x^{(l)}_{k-\Delta} = \mathbf{1}^H_\Delta x^{(l)}_{k:k-N_f-v+1}$. Furthermore, denote $\mathbf{r}^{(l)}_\Delta = \mathbf{R}_{x^{(l)}y}\mathbf{1}_\Delta$. Assuming that the source signals $\{x^{(l)}\}$ are stationary, we have $E[\, {x^{(l)}_{k-\Delta}}^2\,] = \delta^2_{x^{(l)}}$, where $\delta^2_{x^{(l)}}$ is the $l^{(th)}$source signal power.

Using the above notations, Equation 6.7 can be rewritten as

$$\mathrm{MSE}_l = \delta^2_{x^{(l)}} - \mathbf{w}^H \cdot \mathbf{r}^{(l)}_\Delta - {\mathbf{r}^{(l)}_\Delta}^H \mathbf{w} + \mathbf{w}^H R_{yy}\mathbf{w}, \tag{6.4}$$

The $\mathrm{MSE}_{\mathrm{total}}$ is used to measure the system's overall performance, and is defined as the sum the MSE of all input-output pairs, i.e., $\mathrm{MSE}_{\mathrm{total}} = \sum_{m=1}^{n_i} \mathrm{MSE}_m$. The MSE discussed for the MIMO case In Section 3.1.3 is exactly the MSE_{total}.

■ 6.2 Problem Formulation and Visualization

In Chapter 4, the sparse equalizer design problem is formulated under a single MSE constraint. For the MIMO case we discussed in Section 3.1.3, it corresponds to minimizing the equalizer coefficient l_0-norm subject to a pre-defined $\mathrm{MSE}_{\mathrm{total}}$ tolerance. However, this formulation is not applicable when specific MSE constraint is placed on several $\{MSE_l\}$s.

In this section, we formulate the sparse equalizer design problem under two MSE constraints. Specifically, the $\mathrm{MSE}_{\mathrm{total}}$ is required to be less than a pre-defined value ξ, and the $\mathrm{MSE}_{\mathrm{a}}$ is required to be less than another pre-defined value ξ_a. The $\mathrm{MSE}_{\mathrm{a}}$ denotes the MSE constraint asked by a subset of users, and is equal to the sum of the $\mathrm{MSE}_{\mathrm{l}}$ for these users.

Therefore, this problem can be formulated as

$$\begin{aligned} &\min_{w} \|\mathbf{w}\|_{\mathbf{0}} \\ s.t.\ \ &\mathrm{MSE}_a \leq \xi_a \\ &\mathrm{MSE}_{total} \leq \xi_{total} \end{aligned}$$

or equivalently

$$\begin{aligned} &\min_{w} \|\mathbf{w}\|_{\mathbf{0}} \\ s.t.\ \ &\mathrm{MSE}_a \leq \xi_a \\ &\mathrm{MSE}_b \leq \xi_b \end{aligned} \tag{6.5}$$

where $\mathrm{MSE_b}$ is the sum of the $\mathrm{MSE_l}$ for the users not in subset a, and $\xi_b = \xi - \xi_a$. From Equation 6.7, we obtain

$$\begin{aligned} \mathrm{MSE}_a &= \delta^2_{x^{(a)}} - \mathbf{w}^H \cdot \mathbf{r}^{(a)}_{\Delta} - \mathbf{r}^{(a)H}_{\Delta}\mathbf{w} + \mathbf{w}^H R_{yy}\mathbf{w} \\ &= \mathrm{MMSE_a} + (\mathbf{w} - \mathbf{w}_{Ma})^H R_{yy}(\mathbf{w} - \mathbf{w}_{Ma}), \end{aligned} \tag{6.6}$$

where $\mathbf{w}_{Ma} = \mathbf{R}^{-1}_{yy}\mathbf{r}^{(a)}_{\Delta}$, and the superscript (a) denotes the grouping of the users in a.

$$\begin{aligned} \mathrm{MSE}_b &= \delta^2_{x^{(b)}} - \mathbf{w}^H \cdot \mathbf{r}^{(a)}_{\Delta} - \mathbf{r}^{(a)H}_{\Delta}\mathbf{w} + \mathbf{w}^H R_{yy}\mathbf{w} \\ &= \mathrm{MMSE_b} + (\mathbf{w} - \mathbf{w}_{Ma})^H R_{yy}(\mathbf{w} - \mathbf{w}_{Ma}), \end{aligned} \tag{6.7}$$

where $\mathbf{w}_{Ma} = \mathbf{R}^{-1}_{yy}\mathbf{r}^{(a)}_{\Delta}$, and the superscript (a) denotes the grouping of the users in a.

The problem is visualized in Figure 6-1. The two quadratic constraints correspond to the two ellipses centered at $\mathbf{w}_{Ma}$ and $\mathbf{w}_{Mb}$ respectively. The ellipse sizes are determined by the excess error allowance ξ_{ea} and ξ_{eb} respectively. The excess error allowance is the difference between the MMSE and the MSE constraint. The problem is find the most sparse vector $\mathbf{w}^*$ in the intersection of the two ellipses.

6.3 Sub-optimal Solution

Finding the most sparse vector $\mathbf{w}^*$ in the intersection of the two ellipses shown in Figure 6-1 is NP-hard. There is no efficient algorithm on the l_0-norm minimization problem under

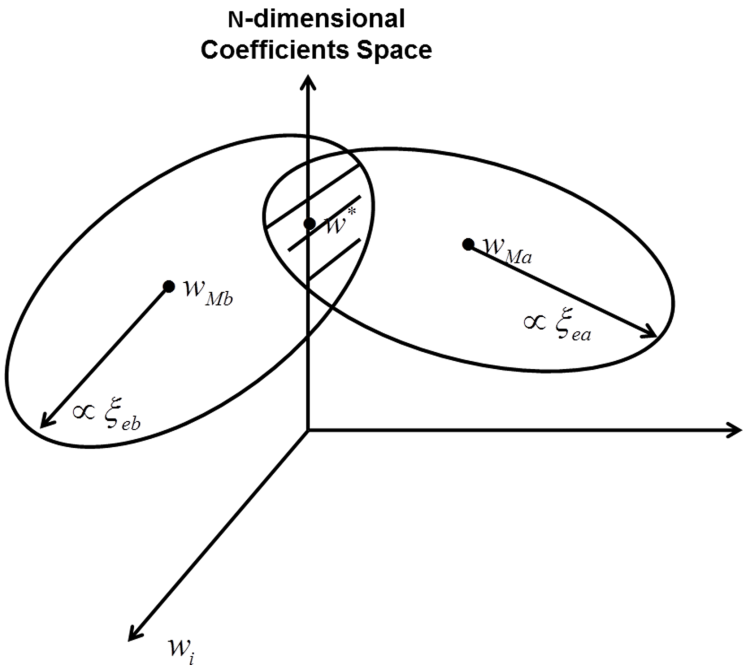

Figure 6-1: Visualization of the l_0-norm minimization under two quadratic constraints

two quadratic constraints. In this section, we propose a low-complexity method that does not search for the most sparse solution but can find a sparse solution in most cases.

The idea is to decouple the two constraint to two independent constraints so that we can solve the two sub-problems independently. There are two parts in this method.

Part I

The equalizer filter coefficients are separated into two sets, w_a and w_b, where $w = w_a + w_b$. w_a is used to control MSE_a and the other set is used to control MSE_b. The goal of this part is to find a valid separation that both the MSE_a constraint and MSE_b constraint are satisfied. The process is visualized in Figure 6-2. Separating w into w_a and w_b is equivalent to projecting the two ellipses onto two orthogonal lower-dimensional spaces. In Figure 6-2, P and Q denote the dimensionality in w_a and w_b respectively, where $N = P + Q$.

In the following paragraph, we propose two methods to find the valid separation. One is a numerical search algorithm and the other one is an iterative approach. Denote U_{ab} as the binary indicator vector to represent the separation, where the value 1 means the corre-

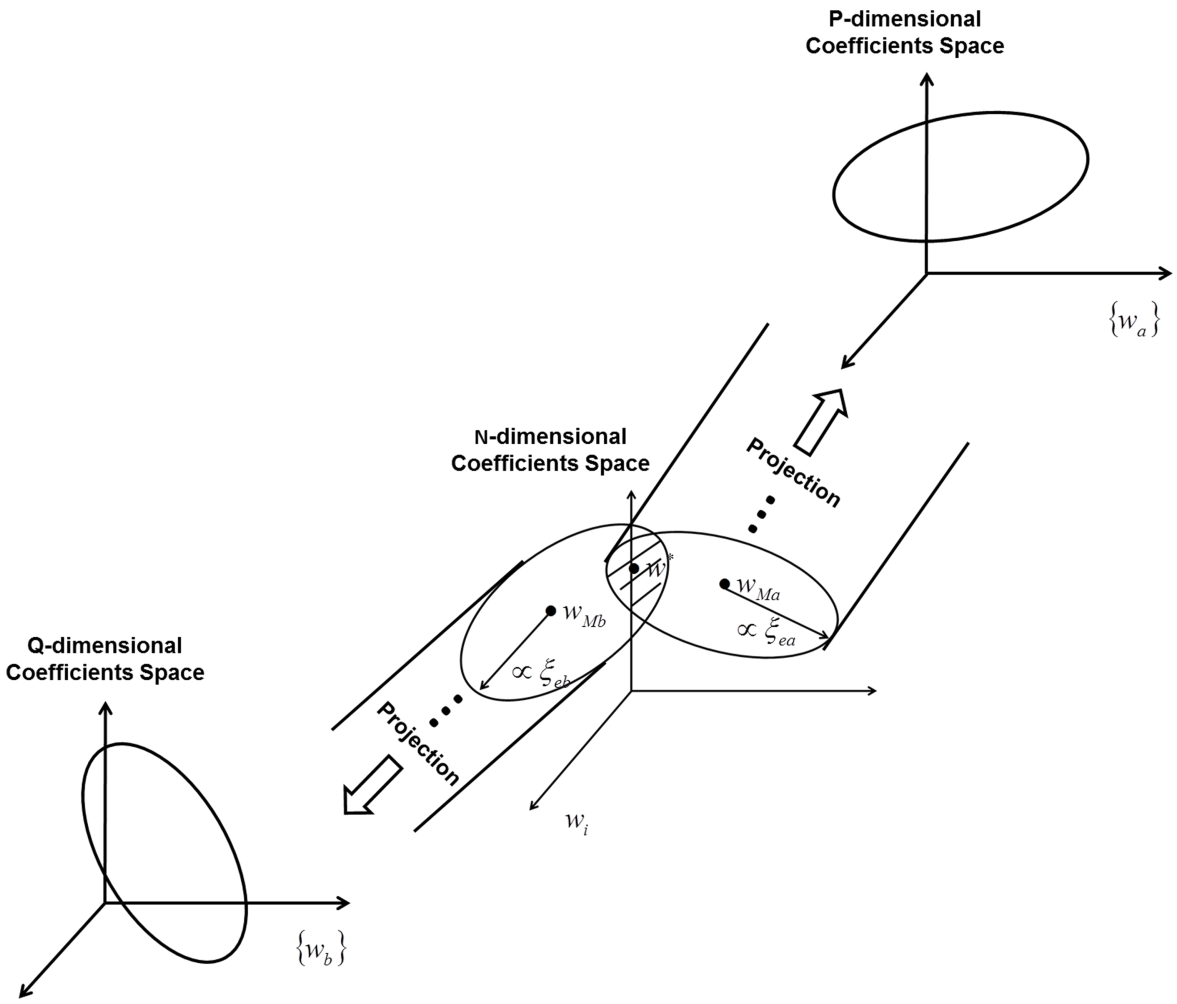

Figure 6-2: Visualization of the constraints separation in the sub-optimal method

sponding coefficient is separated to w_a and the value 0 means the corresponding coefficient is separated to w_b.

Figure 6-3 shows an example of the numerical search method for equalizer length $N = 4$. Each generation in the tree represents a coefficient, and the left branch denotes that the coefficient is separated to $\{w_a\}$ and vice versa. If the equalizer length is N, there are 2^N leaves, and each leave represents a separation. When visiting a specific node, the algorithm checks if the MSE_a constraint and the MSE_b constraint can both be satisfied. If the constraint can be satisfied, we put indicator 1 in the leave and 0 otherwise. To show that this indicator can help reduce the number of enumerations, we refer to the example in Figure 6-3. The leave in the middle uses $[w_1\ 0\ w_3\ 0]$ to equalize the channel for the

user set a, and uses $[0\ w_2\ 0\ w_4]$ to equalize the channel for the use set b. Indicators 01 represent that the MSE_a constraint cannot be satisfied, thus the first leave node can be eliminated from the searching candidates because it can never achieve a better result than the current leave. When the equalizer length N is larger, the indicators can help reduce the computational cost.

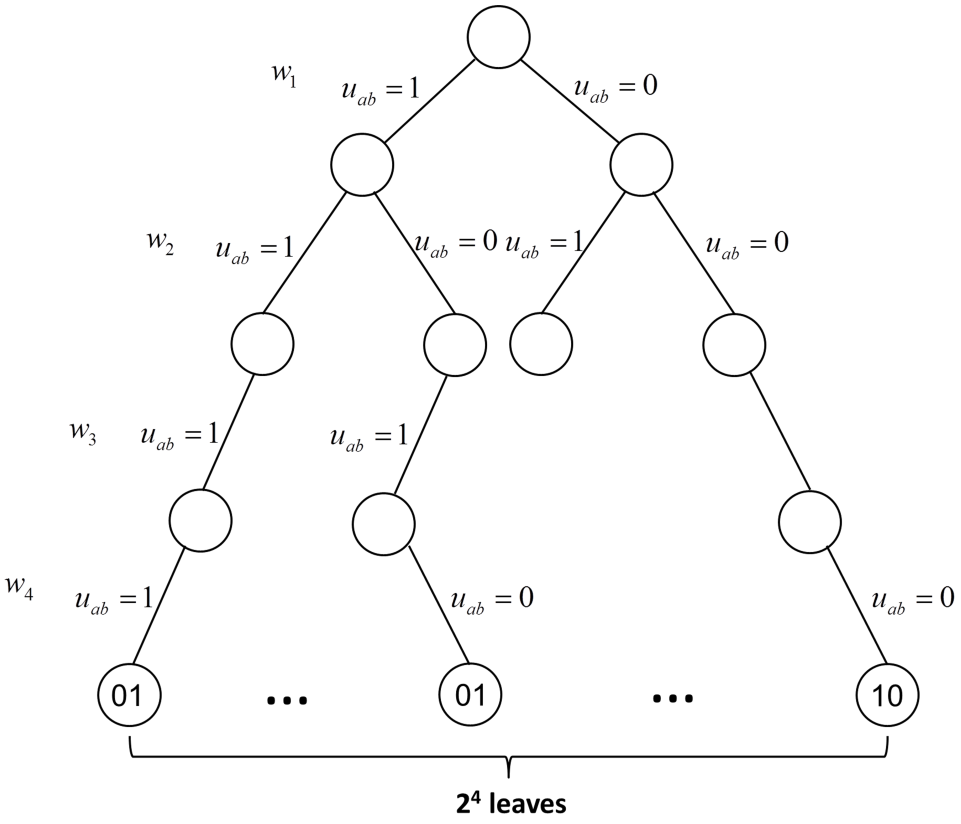

Figure 6-3: The tree of feasible separation searching

Figure 6-4 gives the flow graph of the the iterative algorithm to find a valid separation U_{ab} in problem 6.9. The algorithm is initialized by w_{MMSE}, the coefficients that minimizes MSE_{total}. The left rectangular takes the input coefficient values and finds the best separation that minimizes the sum of MSE_a and MSE_b based on the given value. Then the right rectangular uses the separation, and optimizes the coefficient values in w_a and w_b individually. If the new coefficients w_a and w_b can satisfy the MSE_a and MSE_b constraint, then the valid separation is given by the current U_{ab}. At each step, the MSE_a and MSE_b are non-increasing, so this algorithm is able to find the local optimal separation.

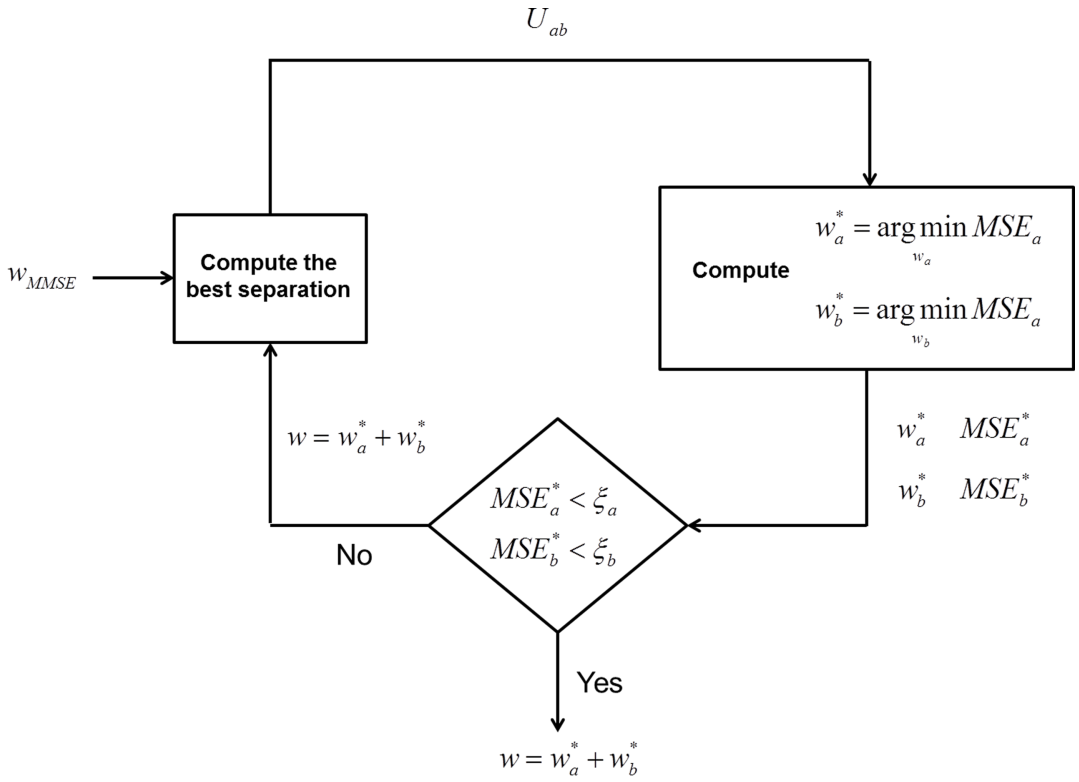

Figure 6-4: Iterative approach for finding the valid separation in problem 6.9

Part II

Using this separation, the original problem is decoupled into two independent problems, and we solve the two problems individually. Specifically, the two sub-problems are

$$\begin{aligned} &\min_{w_a} \|\mathbf{w_a}\|_\mathbf{0} \\ s.t. \quad &(\mathbf{w_a} - \mathbf{w}_{Ma})^H R(\mathbf{w_a} - \mathbf{w}_{Ma}) \leq \xi_{ea}, \end{aligned}$$

and

$$\begin{aligned} &\min_{w_b} \|\mathbf{w_b}\|_\mathbf{0} \\ s.t. \quad &(\mathbf{w_b} - \mathbf{w}_{Mb})^H R(\mathbf{w_b} - \mathbf{w}_{Mb}) \leq \xi_{eb}. \end{aligned} \tag{6.8}$$

Note that both sub-problems are reduced to the l_0-norm minimization problem under a single quadratic constraint, and can be solved using the methods discussed in Chapter 4.

The two sub-problems in 6.8 can be rewritten in a more compacted form

$$\begin{aligned} & \min_{w} \|\mathbf{w}\|_{\mathbf{0}} \\ s.t. \quad & (\mathbf{w_a} - \mathbf{w}_{Ma})^H R(\mathbf{w_a} - \mathbf{w}_{Ma}) \leq \xi_{ea}, \\ & (\mathbf{w_b} - \mathbf{w}_{Mb})^H R(\mathbf{w_b} - \mathbf{w}_{Mb}) \leq \xi_{eb} \\ & w_a = U_{ab}. * w \\ & w_b = (I - U_{ab}). * w \end{aligned} \tag{6.9}$$

where U_{ab} is the binary indicator vector which denotes the separation.

This sub-optimal approach fails when a valid separation does not exist. This means that in Figure 6-2, the projection of the ellipses onto any two orthogonal sub-spaces is empty. In this case, even when the two ellipses have intersections, i.e., the original problem is feasible, the sub-optimal solution is unable to provide any optimal or non-optimal result.

Chapter 7
Conclusion and Future Work

In this thesis, we have considered the design of discrete-time equalizers according to a measure of the number of nonzero coefficients compared to a more conventional measure based on the total number of filter coefficients. The measure of the number of nonzero taps can be more closely aligned with the actual implementation cost . Using this measure, we obtain equalizers with less number of nonzero elements, which we refer to as sparse equalizers. Specifically, this thesis focuses on the sparse equalizer design for a class of linear, time-invariant multi-path channels.

Chapter 2 of this thesis focused on the intrinsic sparsity of the multi-path channel equalizers. We showed in Figure 2-1 that the ZF equalizer exhibits a sparse pattern for sparse multi-path channels. We showed in Figure 2-2 that the FIR MMSE equalizers often have a significant percentage of small coefficients. This becomes the basis of the sparse equalizer design for multi-path channels with a small performance loss.

Chapter 3 of this thesis evaluated the equalization MSE under various channel and equalizer specifications. We demonstrated that the LE MSE can be described by a quadratic function for the four channel types, i.e., the SISO channel, the SIMO channel, the MISO channel and the MIMO channel. Moreover, we showed that the DFE MSE can also be described by a quadratic function. This extended the sparse filter design for the decision feedback scenarios. Additionally, the dependency of the MMSE on the channel and equalizer specifications were investigated. Figure 3-7 indicated that the MMSE decreases with the equalizer length N. Figure 3-8 verified the empirical optimal decision delay, which is $\Delta = \frac{N_c+N_f}{2}$. Figure 3-9 showed that the MMSE decreases with the channel SNR. Table 1 proved the optimal decision delay Theorems of the DFE.

Chapter 4 of this thesis formulated the sparse equalizer design problem under a single MSE constraint. We defined it as an l_0-norm minimization problem under a quadratic constraint. For the scope of this thesis, three algorithms were presented to solve the sparse

equalizer design problem. We discussed the advantage and the disadvantage of each method and then verified them in Chapter 5.

Then in Chapter 5 we simulated the problem defined in Chapter 4. We plotted the number of nonzero coefficients in the designed equalizer versus different channel specifications and equalizer specifications. These simulation results can be explained by referring to Chapter 2 and 3. The most significant result is that the number of non-zero coefficients decrease rapidly as the excess error ratio goes from 0 to 0.02. This indicates that a sparse equalizer can be achieved with a very small MSE increase. This is related the discussion in Chapter 2. We also found that the successive thinning algorithm presented in Chapter 4 produces the optimal solution in over 95% cases. Therefore, it would be a good choice to use the successive thinning method instead of the optimal branch-and-bound approach especially when the computational cost is a major concern.

Chapter 6 of this thesis developed the sparse equalizer design problem under two MSE constraints required by two groups of users. We formulated it as an l_0-norm minimization problem under two quadratic constraints. There is no effective algorithm that is applicable to this problem. We proposed a low-complexity algorithm that decouples the problem into two subproblems. This method does not find the optimal solution but provides a possibility to solve the problem.

■ 7.1 Future Work

First, more research can be conducted on the channel. In this thesis, we focused on the sparse filter design on sparse multi-path channels. It would be helpful to investigate the sparse equalizer design for non-sparse channels and compare them with the results in this thesis. Obtaining the general characterstics of channels that are suitable for the sparse equalizer design is very important for the applications.

Second, from an optimization point of view, the development of algorithms to solve the l_0-norm minimization problem under one quadratic constraint is a potentially rich area for future study. It may be possible to devise more efficient optimal algorithms by combining the linear and diagonal relaxations in such a way that the new relaxation is stronger than

either alone. It is also interesting to explore the sub-optimal heuristic algorithm by defining new subset selection criteria.

Third, it is clear from Chapter 6 that an optimal algorithm for sparse filter design under two quadratic constraint has not been developed. Moreover, the non-optimal solution proposed in 6.3 requires simulations on the real data to evaluate its performance. It is also intriguing to develop computationally efficient algorithms on finding the valid separation U_{ab}.

Last, the sparsity concept can be applied to other problems either in the scope of the filter design or other applications beyond signal processing. For example, the sparsity measure can be extended to IIR filters. It can also be applied to sensor arrays since the array elements can be expensive to manufacture or operate. Other applications include portfolio optimization [22]. Adopting the sparsity measure can help improve the savings in computation, power consumption, hardware, or communication resulting from the elimination of operations involving zero-valued coefficients.

Bibliography

[1] S. Raghavan, J. Wolf, L. Milstein, and L. Barbosa, "Nonuniformly Spaced Tapped-Delay-Line Equalizers," *IEEE Transactions on Communications,* vol. 41, pp. 1290-1295, 1993.

[2] D. Wei, "Design of Discrete-Time Filters for Efficient Implementation," *Doctoral Thesis*, pp. 40-46, 2011.

[3] "Guidelines for The Evaluation of Radio Transmission Technologies for IMT-2000," *Recommendation ITU-R M.1225,* 1997.

[4] Carlos A. Belfiore, John H. Park, "Decision Feedback Equalization," *Proceedings of the IEEE,* vol. 67, No.8, pp. 1143-1156, 1979.

[5] S. Ariyavisitakul, N. Sollenberger, and L. Greenstein, "Tap-Selectable Decision Feedback Equalization," *IEEE Transactions on Communications,* vol. 45, pp. 1497-1500, 1997.

[6] Robert Gallager, *Principles of Digital Communications*, book, Fall 2006.

[7] R. J. Hartnett, G. F. Boudreaux-Bartels, "On the use of cyclotomic polynomial prefilters for efficient FIR filter design," *IEEE Trans. Signal Process.*, vol. 41, pp. 1766-1779, May 1993.

[8] T. Kailath, Linear Systems. Englewood Cliffs, NJ: Prentice-Hall, 1980, pp. 390.

[9] A. Fertner. "Improvement of bit-error-rate in decision feedback equalizer by preventing decision-error propagation". *Signal Processing, IEEE Transactions on*, 46(7):18721877, 1998.

[10] M. Reuter, J.C. Allen, J.R. Zeidler, and R.C. North. "Mitigating error propagation effects in a decision feedback equalizer". *Communications, IEEE Transactions on*, 49(11):20282041, 2001.

[11] J.E. Smee and N.C. Beaulieu. Error-rate evaluation of linear equalization and decision feedback equalization with error propagation. *Communications, IEEE Transactions on*, 46(5):656665, 1998

[12] D.L. Duttweiler, J. Mazo, D.G. Messerschmitt , "An Upper Bound on the Error Probability in Decision-Feedback Equalization", *Information Theory, IEEE Transactions on*, vol.20, no.4, pp. 490- 497, Jul 1974

[13] Stanford EE379a lecture notes, Chapter 3.

[14] Y. Gong, C.F.N. Cowan, "Optimum Decision Delay of the Finite-Length DFE," *IEEE Signal Processing Letters.*, vol. 11, pp. 858-861, November 2004.

[15] N. Bourbaki, "Chapters 15". Topological vector spaces. Springer. ISBN 3-540-13627-4, 1987.

[16] Donoho, D. L., "Compressed Sensing", *IEEE Transactions on Information Theory*, V. 52(4), 12891306, 2006

[17] D. Bertsimas and R. Weismantel, Optimization Over Integers. Belmont, MA: Dynamic Ideas, 2005.

[18] J. Diestel, Sequences and series in Banach spaces, Springer-Verlag, ISBN 0-387-90859-5, 1984.

[19] H. Sui, E. Masry, and B. D. Rao, "Chip-level DS-CDMA downlink interference suppression with optimized finger placement, IEEE Trans. Signal Process., vol. 54, no. 10, pp. 39083921, Oct. 2006.

[20] R.G. Baraniuk, "Compressive sensing". *IEEE Signal Processing Magazine*, 24(4):118, 2007

[21] T. Baran, D. Wei, and A.V. Oppenheim. "Linear Programming Algorithms for Sparse Filter Design. *IEEE Transactions on Signal Processing*, 58.3 (2010): 16051617. Web. 30 Mar. 2012.

[22] D. Bertsimas and R. Shioda, "Algorithm for cardinality-constrained quadratic optimization, *Comput. Optim. Appl.,* vol. 43, no. 1, pp. 122, 2009.

[23] R. J. Dakin, "A tree-search algorithm for mixed integer linear programming problems, *Comp. J.*, vol. 8, no. 3, pp. 250-255, 1965.

[24] D. Bertsimas and J. Tsitsiklis, "Introduction to Linear Optimization" (1st ed.). Athena Scientific.1997.