國 立 交 通 大 學

電機與控制工程學系

碩 士 論 文

以子空間方法設計離線及適應性盲目等化器

Design of Offline and Adaptive Blind Equalizers Using Subspace Approach

研 究 生：林佳華
指導教授：鄭木火 博士

中華民國九十六年九月

以子空間方法設計離線及適應性盲目等化器

Design of Offline and Adaptive Blind Equalizers Using Subspace Approach

研 究 生 ： 林佳華　　Student : Chia-Hua Lin

指導教授 ： 鄭木火 博士　　Advisor : Dr. Mu-Huo Cheng

國 立 交 通 大 學

電機與控制工程學系

碩士論文

A Thesis

Submitted to Department of Electrical and Control Engineering

College of electrical and Computer Engineering

National Chiao Tung University

In Partial Fulfillment of the Requirement

For the Degree of

Master of Science

In

Electrics Engineering

September 2007

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 六 年 九 月

以子空間方法設計離線及適應性盲目等化器

研究生: 林佳華　　指導教授: 鄭木火 博士

國立交通大學電機與控制工學系

摘要

本論文的目的是針對單一輸入多重輸出 (single input multiple output, SIMO) 的情況下, 以子空間方法設計盲目等化器。由於在高速數位通訊傳輸下, 傳輸訊號經過多路徑無線通道, 會造成傳輸訊號有嚴重地符號間干擾 (intersymbol interference, ISI), 除此之外, 接收機對於無線通道與傳輸訊號一般而言是一無所知的, 所以傳統中的無線通訊系統是使用訓練序列 (training sequences) 讓接收機了解無線通道之特性, 不過使用訓練序列是相當浪費頻寬, 因此需要使用盲目等化器來改善頻寬的使用與降低符號間干擾。

本論文將提出兩種新型離線式盲目等化器之方法並改良其一方法使其爲適應性盲目等化器。第一種離線式方法, 吾人利用過度取樣 (Oversampling) 生成特殊之 Toeplitz 架構的通道矩陣, 且將此通道矩陣之特性與接收訊號的二階統計 (second order statistics, SOS) 的子空間運用, 在與 MRE(mutually referenced filters) 方法搭配即是吾人提出的第一種盲目等化器方法。第二種離線式方法, 利用 MRE 擁有地特性並與最小平方法 (Least Square Method) 和二階統計的子空間相互結合, 得到了疊代式的最小平方法亦即是吾人提出的第二種盲目等化器方法。最後, 應用了子空間追蹤 (subspace tracking) 與二次疊代 (Bi-iteration) 的奇異值分解 (singular value decomposition, SVD) 將所提出的第一種離線式方法改進爲可以不斷更新等化器參數的適應性盲目等化器。

在論文最後, 使用電腦的數值模擬與其他文獻提出之方法比較。所使用的比較標準分別爲符號間干擾 (ISI) 與訊號雜波比 (signal-to-interference-noise ratio, SINR), 經由數值模擬的驗證, 判斷方法的優越性。

關鍵詞: 盲目等化器, 適應性盲目等化器, 符號間干擾, 正交投影子空間追蹤

Design of Offline and Adaptive Blind Equalizers Using Subspace Approach

Student: Chia-Hua Lin   Advisor: Dr. Mu-Huo Cheng

Institute of Electrical and Control Engineering

National Chiao-Tung University

Abstract

In wireless communication systems, equalization is often required in order to suppress the intersymbol interference (ISI) caused by multipath channels. Conventional approaches use training sequences for equalizer design which wastes the bandwidth. The blind equalizer can perform equalization in no need of the training sequences and thus achieves more efficient channel bandwidth usage. In this thesis, we present new methods based on the subspace approach for computing fractionally spaced blind equalizers in single input multiple output (SIMO) systems.

We first present a new offline method using the properties of the channel matrix structure and the idea of mutually referenced filters. This method is later used to develop an adaptive blind equalizer by employing the OPAST algorithm and the bi-iteration singular value decomposition. We also use the idea of mutually reference filters to develop a new blind equalizer design using the iterative least squares method. Finally, simulations are performed to demonstrate the better performance of the proposed algorithms compared to existing approaches.

Index Terms - Blind equalization, Adaptive equalization, intersymbol interference, OPAST

致謝

這篇論文可以順利完成，首先要特別感謝我的指導教授鄭木火博士，在這短短兩年碩士研究過程中，在研究上循循善誘，一次一次不斷地引導我解決困難，並且讓我的學習分析問題並有解決問題的能力，所以無論在待人接物與處事治學上的嚴謹態度，都讓我獲益良多。因此本論文付梓之際，對於耐心傳道授業的老師致上最誠摯的感謝。

再來要感謝在電控所 Synchronization Laboratory 裡所提供的完備資源。也承蒙學長浩緯、信良、俊維與啟峰的提攜與照顧，在研究上與生活上都可以順利解決問題。而在實驗室一起打拼的夥伴衍禎、嘉明、宏揚與志全還有室友嘉川、立卓與存智在課業上的砥礪以及生活上的互相鼓勵與協助，讓我在忙碌的研究生生活中一就可以保持愉悅的心情。也要感謝學弟立中、英哲與浩淵，在你們的幫助下，使我的論文得以順利完成。最後亦要感謝士芳與隆炯讓我的碩士生活多采多姿。

最後，要感謝我的家人，這個溫暖的家讓我可以無後顧之憂地涉取專業知識，努力與我的學業之上，順順利利地完成我的學業，並做好面對下一個人生階段挑戰的面對。

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

In wireless communication systems, equalization process is often necessary in order to suppress the intersymbol interference (ISI) caused by channels with multi-path phenomena. The conventional equalization design is usually performed by using the information known in a priori contained in a training sequence. Since the communication channels are usually time varying, the training sequence must be sent periodically. Hence, much bandwidth is wasted by sending the training sequence. The equalizer designed in no need of the training sequence, referred to as the blind equalizer, draws much research interest in recent years in order to obtain a more efficient channel bandwidth [1], [2].

Early blind equalizer is designed using a symbol-rate sampled mono-channel transmission model in terms of second order or high order statistics of the received signals [1]-[4]. The corresponding adaptive methods are classically designed via the stochastic gradient descent scheme for minimizing one designed cost function. Lately, the use of either the fractionally spaced samples in one sensor or multiple sensors reception arrays is employed for the design of blind equalizer mainly for two reasons. One is that some high order statistics methods such as the constant modulus algorithm (CMA) can be shown to be globally convergent in this scheme and the other is that the blind equalizer is shown to be achievable in this scheme merely based on the second order statistics of the received signals [5].

Based on the multiple FIR channel model and second order statistics of the channel outputs, many methods for blind equalizer design have been proposed. Some methods perform the blind channel identification first, then the equalizer is designed from the obtained channel. Some methods, such as the linear prediction-based approach [6], require partial channel information; these methods are robust to channel order overestimation but the performance will be degraded by the error in channel estimation.

1.2 Type of Blind Equalization Design

Recently, the subspace approach is an important technique for blind equalizer design because the closed form solution can be derived via this approach. This approach, in noise-free condition, yields perfect channel estimation using only a finite number of data. The subspace algorithms decompose the received data into two mutually orthogonal subspaces, namely the signal subspace and the noise subspaces. The orthogonality between the signal and nose subspace enables direct channel identification or even direct estimation of input symbols [1]-[4].

There are also some other methods that use neither channel estimation nor the structure of channel matrix, like mutually referenced equalizers (MREs) [5]. In these cases, these methods requires nonlinear optimization, which may affect their convergence speed and lager number of computations.

For adaptive blind equalization algorithms, like linear prediction methods, MREs, and CMA. Direct blind equalization methods are also based on RLS and LMS solutions. These algorithm use the algebraic structure of the received data sequence together with statistical source properties.

1.3 Organization of the Thesis

The remainder of this thesis is divided into five chapters including conclusions. Chapter 2 explains that problem formulation including over-sampling a single sensor, multiple sensor, and blind equalization design. Chapter 3 realizes the difference between current method and our methods for off-line situation. Chapter 4 use some algorithm let off-line situation of blind equalization become adaptive blind equalization. The final chapter is the conclusions.

Chapter 2

Problem Formulation and Subspace Blind Equalizer Design

In this chapter, we discuss the subspace approach for the design of blind equalizers. We first discuss the formulation of received data either by oversampling a single sensor or by sampling multiple sensors. Then the zero-forcing equalizer is obtained directly from the formulation by the pseudoinverse technique. The subspace decomposition is shortly reviewed and its application to the design of blind equalizer is then elaborated.

2.0.1 Received Data

Denote s_n as the transmitted digital symbol sequence at time nT where the T represents the symbol duration. These digital symbols are modulated, filtered, and transmitted through the wireless communication channel; then the signal is received via antennas, after filtering and demodulation, yielding the baseband continuous signal given by

$$x(t) = \sum_{m=-\infty}^{\infty} s_m h(t - mT) + n(t) \tag{2.1}$$

where $n(t)$ is the assumed additive noise and $h(t)$ denotes the equivalent channel impulse response including the transmitter filter, receiver filter, channel response, and modulation/demodulation. In this thesis, we assume that the channel impulse response $h(t)$ has a finite duration; moreover we also assume that several measurements can be performed during one transmission symbol period T. The latter is realized either by sampling the received signal on several sensors at the rate T, or by oversampling the received signal on a single sensor, or by combining both methods above. The digitized received signal and its formulation via oversampling or multi-sensor is discussed below.

2.0.2 Oversampling on a Single Sensor

In this approach, the received signal is sampled at the rate of Δ which is a fraction of the symbol interval T; that is, normally we have an integer L called the oversampling factor such that $L = T/\Delta$. Hence, in one symbol interval T, we obtain L digitized received data $x_i(n) = x(t_0 + i\Delta + nT)$ for $0 \leq i \leq L-1$. Using (2.1), we obtain

$$\begin{aligned} x_i(n) &= x(t_0 + i\Delta + nT) \\ &= \sum_{m=-\infty}^{\infty} s_m h(t_0 + i\Delta + nT - mT) + n(t_0 + i\Delta + nT) \\ &= \sum_{m=-\infty}^{\infty} s_m h_i(n-m) + n_i(n) \end{aligned} \tag{2.2}$$

where $h_i(n) = h(t_0+i\Delta+nT)$, $n_i(n) = n(t_0+i\Delta+nT)$, and the index $i = 0, \cdots, L-1$. Then we can treat the received data as obtained from an equivalent multichannel setup shown in Fig. 2.1 where $h_i(n)$ denotes the T-sampled impulse of the i-th channel, $x_i(n)$ the baud-rate signal measured at the output of the i-th channel, $n_i(n)$ the corresponding noise sequence, and L the number of channels.

Collecting all L channel measured outputs, channel responses, and measured noise together as vectors, we have

$$\mathbf{x}(n) = \begin{bmatrix} x_0(n) \\ \vdots \\ x_{L-1}(n) \end{bmatrix}, \ \mathbf{h}(n) = \begin{bmatrix} h_0(n) \\ \vdots \\ h_{L-1}(n) \end{bmatrix} \mathbf{n}(n) = \begin{bmatrix} n_0(n) \\ \vdots \\ n_{L-1}(n) \end{bmatrix}$$

The vector form of (2.2) can be obtained as below

$$\mathbf{x}(n) = \sum_{m=-\infty}^{\infty} s_m \mathbf{h}(n-m) + \mathbf{n}(n). \tag{2.3}$$

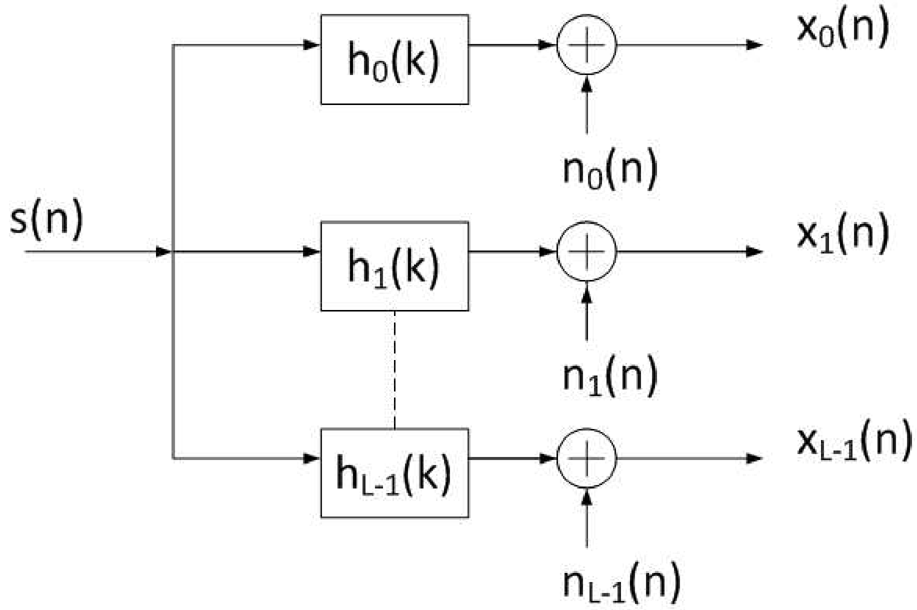

Fig. 2.1: Baud-rate digital multiple sensor model

Since we assume that $h(t)$ is causal and of finite duration, then all channel response $h_i(n)$ is finite. Assume that the maximum length of $h_i(n)$ is of order M, then the equation (2.3) can be represented in a matrix form given by

$$\begin{aligned}\mathbf{x}(n) &= \begin{bmatrix} x_0(n) \\ \vdots \\ x_{L-1}(n) \end{bmatrix} \\ &= \begin{bmatrix} h_0(0) & h_0(1) & \cdots & h_0(M) \\ h_1(0) & h_1(1) & \cdots & h_1(M) \\ \vdots & \vdots & \ddots & \vdots \\ h_{L-1}(0) & h_{L-1}(1) & \cdots & h_{L-1}(M) \end{bmatrix} \begin{bmatrix} s_n \\ s_{n-1} \\ \vdots \\ s_{n-M} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{h}(0) & \mathbf{h}(1) & \cdots & \mathbf{h}(M) \end{bmatrix} \begin{bmatrix} s_n \\ s_{n-1} \\ \vdots \\ s_{n-M} \end{bmatrix}\end{aligned} \tag{2.4}$$

Collect all received data of NT interval in a vector $\mathbf{x}_N(n)$ as shown below

$$\mathbf{x}_N(n) = \begin{bmatrix} \mathbf{x}(n) \\ \mathbf{x}(n-1) \\ \vdots \\ \mathbf{x}(n-N+2) \\ \mathbf{x}(n-N+1) \end{bmatrix}$$

Using (2.4), we can then express the received data above in a form below

$$\mathbf{x}_N(n) = H\mathbf{s}(n) + \mathbf{n}_N(n) \tag{2.5}$$

where H is called the channel matrix given by

$$H = \begin{bmatrix} \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(M) & \mathbf{0} & \dots & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(M) & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(M) & \mathbf{0} \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(M) \end{bmatrix}$$

and

$$\mathbf{s}(n) = \begin{bmatrix} s_n \\ s_{n-1} \\ \vdots \\ s_{n-N-M+2} \\ s_{n-N-M+1} \end{bmatrix} \mathbf{n}_N(n) = \begin{bmatrix} \mathbf{n}(n) \\ \mathbf{n}(n-1) \\ \vdots \\ \mathbf{n}(n-N) \\ \mathbf{n}(n-N+1) \end{bmatrix}$$

The equation (2.5) describes the relation between the received data and the transmitted input sequence; the dimension of each element is listed in Table 2.1. Note that this formulation will be used throughout this thesis. Similar expressions can be obtained for the case of multiple sensor, as discussed below.

2.0.3 Multiple Sensors

For the case with multiple sensors, in each sensor we receive a signal that may have gone through different propagation channels. Hence, the complex envelop of the signal

Table 2.1: Matrix Size

matrix	size	type
H	$NL \times (N+M)$	block Toeplitz matrix
$\mathbf{s}(n)$	$(N+M) \times 1$	vector
$\mathbf{n}_N(n)$	$NL \times 1$	vector
$\mathbf{x}_N(n)$	$NL \times 1$	vector

received on the i-th sensor can be expressed as below

$$x_i(n) = \sum_{m=-\infty}^{\infty} s_m h_i(n-m) + n_i(n) \tag{2.6}$$

Note that the above equation (2.6) is in the same form as (2.2); hence the received data obtained by oversampling on a single sensor can be expressed in the identical form as the received data from multiple sensors when the oversampling factor L is equal to the number of sensors. Similar relations can be obtained straightforwardly by combining both the oversampling and multiple sensors; that is, given K sensors and the oversampling factor P, the received data can be expressed in the same form as (2.5) when $L = KP$.

2.1 Zero-Forcing Equalizer

The channel matrix H in (2.5) is of size $LN \times (M+N)$. Let $LN \geq M+N$ and assume H is full rank, the zero-forcing equalizer can be obtained from the pseudoinverse of the channel matrix H which is $(H^H H)^{-1} H^H$ where the superscript H denotes the hermitian operation. Multiplying (2.5) on the left by $(H^H H)^{-1} H^H$, we obtain

$$\hat{\mathbf{s}}(n) = (H^H H)^{-1} H^H \mathbf{x}_N(n) = \mathbf{s}(n) + (H^H H)^{-1} H^H \mathbf{n}_N(n) \tag{2.7}$$

Clearly, $\hat{\mathbf{s}}(n) = \mathbf{s}(n)$ for noise-free condition. Note that the zero-forcing equalizer cannot be obtained directly in practice because the channel matrix is unknown. The reason we list the result here is that most blind equalizer design via subspace approach can be seen as a way to obtain the pseudoinverse of the channel matrix. The pseudoinverse of the channel matrix is called the equalization matrix denoted as below

$$Z = (H^H H)^{-1} H^H = \begin{bmatrix} \mathbf{z}_0^H \\ \vdots \\ \mathbf{z}_{M+N-1}^H \end{bmatrix} \tag{2.8}$$

Note that any row of the equalization matrix Z can be realized as an FIR equalizer as shown in Fig. 2.2, where $g_i(n)$ for $i = 0, 1, \ldots, L-1$ is the component of the d-th row of Z where d can be either one chosen from 0 to $N + M - 1$. That is,

$$\mathbf{z}_d = \begin{bmatrix} g_0(0) & \ldots & g_{L-1}(0) & \ldots & g_0(N-1) & \ldots & g_{L-1}(N-1) \end{bmatrix}^H \tag{2.9}$$

The output of this equalizer, denoted as $\hat{s}_d(n)$, is obtained as below

$$\hat{s}_d(n) = \mathbf{z}_d^H \mathbf{x}_n(n) \tag{2.10}$$

which will approximate s_{n-d}, the d delayed input data.

2.2 Subspace Blind Equalizer

In this section, we present shortly how the subspace approach is used to design a blind equalizer. The subspace approach needs first to obtain the information of the autocorrelation matrix R_0 of the received signal vector $\mathbf{x}_N(n)$, given by

$$R_0 = E(\mathbf{x}_N(n)\mathbf{x}_N^H(n)) \tag{2.11}$$

where $E(\cdot)$ denotes the expectation operator. Normally, the additive received noise is assumed to be independent of the transmitted source signal, the autocorrelation matrix R_0 using (2.5) becomes

$$R_0 = HR_sH^H + R_n \tag{2.12}$$

where $R_s = E(\mathbf{s}(n)\mathbf{s}^H(n))$ and $R_n = E(\mathbf{n}_N(n)\mathbf{n}_N^H(n))$ respectively denote the autocorrelation matrices of the transmitted discrete signal vector $\mathbf{s}(n)$ and the received noise

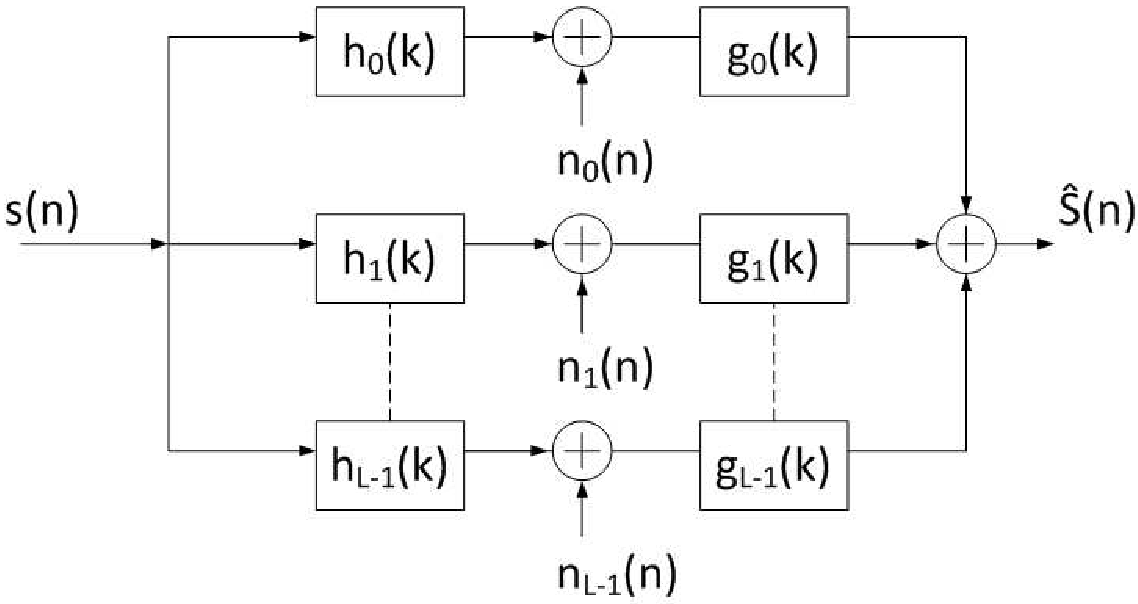

Fig. 2.2: Linear equalization for fractionally spaced channels

vector $\mathbf{n}_N(n)$. Note that the dimension of R_s is $(M+N) \times (M+N)$, and is normally assumed to be full-rank. The dimension of the matrix R_n is $LN \times LN$. Assume as normally done in the literature that the transmitted signal and the noise are both white; that is, $R_s = \sigma_s^2 I$ and $R_n = \sigma_n^2 I$ where I is an identity matrix with the corresponding size. Substituting these assumption into (2.12) yields

$$R_0 = \sigma_s^2 H H^H + \sigma_n^2 I \tag{2.13}$$

Note that R_0 equals a sum of two matrices; the first term on the right side of the above equation constitutes the signal subspace while the second term expresses the noise subspace. Note that the signal subspace has the dimension of at most $N+M$ if the channel matrix H is full rank. The subspace approach derives from R_0 to obtain the signal subspace by matrix decomposition, as discussed in the following section.

2.2.1 Subspace Decomposition

Assume the channel matrix H is full rank, then by simple eigenvalue decomposition for the correlation matrix R_0 with its eigenvalues in descending order, we obtain

$$R_0 = \begin{bmatrix} S & N \end{bmatrix} \begin{bmatrix} \Lambda_1 & \\ & \Lambda_2 \end{bmatrix} \begin{bmatrix} S & N \end{bmatrix}^H = S\Lambda_1 S^H + \sigma_n^2 N N^H \tag{2.14}$$

where Λ_1, Λ_2 are diagonal matrices, S is of dimension $LN \times (N+M)$ which constitutes the signal subspace, and N of dimension $LN \times (LN - (N+M))$ represents the noise subspace. The diagonal elements of Λ_1 are all larger than σ_n^2 while each diagonal element of Λ_2 is equal to σ_n^2. These two subspaces S, N are obviously orthogonal. The dimension of each matrix in the above equation is listed in Table 2.2.

2.2.2 Blind Equalizer Matrix

When the noise is assumed free, we then obtain

$$R_0 = \sigma_s^2 H H^H = S\Lambda_1 S^H \tag{2.15}$$

Given R_0 and its subspace decomposition, for noise-free condition, we may estimate the channel matrix as below

$$\hat{H} = cS\Lambda_1^{1/2} Q^H \tag{2.16}$$

where c is an unknown scale constant and Q is also an unknown orthonormal matrix. Therefore, we can obtain via the subspace decomposition the estimate of the channel matrix up to an unknown orthonormal matrix and an unknown scale constant. Note that the design of single-input single-output blind equalizers inevitably results in an ambiguity of a scale constant; hence we assume the constant c equal to 1 in the sequel, that is, $\hat{H} = S\Lambda_1^{1/2} Q^H$. The critical step in the design of blind equalizers using the subspace approach then lies in how to determine the unknown matrix Q; several approaches will be discussed in this thesis.

The equalization matrix in (2.8) can be obtained using (2.16) and assuming $c = 1$,

Table 2.2: Subspace Matrix Size

matrix	size
$\Lambda_1 = \Lambda + \sigma_n^2 I$	$(M+N) \times (M+N)$
$\Lambda_2 = \sigma_n^2 I$	$(LN-M-N) \times (LN-M-N)$
S	$LN \times (M+N)$
N	$LN \times (LN-M-N)$

yielding

$$\begin{aligned} Z &= (\hat{H}^H \hat{H})^{-1} \hat{H}^H \\ &= Q \Lambda_1^{-1/2} S^H \end{aligned} \tag{2.17}$$

The equalized output after the operation of equalization matrix can be shown below to be white:

$$Z R_0 Z^H = I \tag{2.18}$$

The above result means that the equalized output for any orthonormal matrix Q will be white; the whitened output, however, is not enough to assure the equalizer performance. Hence, extra criteria will be required to determine an adequate design of the matrix Q; that is the main topic in the next chapter.

Chapter 3

Subspace Off-Line Equalizer Design

In this chapter, we will exploit subspace decomposition that is showed in previous chapter to process received signals, and obtain equalizer matrix. Therefore, in first section [1], we introduce method of other paper that also use subspace approach and transmitted signals is white. In second and third section, we show our methods that are also use signal subspace but simulations demonstrate the good performance of our methods.

3.1 Estimate Q with Crosscorrelation Matrix

In this section, a direct blind equalization method proposed in [1] is discussed first, then two new methods are proposed.

3.1.1 Formulation Q Matrix

In noiseless case, the output of an equalization matrix Z, denoted as $\hat{\mathbf{s}}_M(n)$, is given below

$$\hat{\mathbf{s}}_M(n) \triangleq \begin{bmatrix} \hat{s}_0(n) \\ \vdots \\ \hat{s}_{M+N-1}(n) \end{bmatrix} = Z\mathbf{x}_N(n) \tag{3.1}$$

which ideally satisfies the whitening condition below

$$E\{\hat{\mathbf{s}}_M(n)\hat{\mathbf{s}}_M^H(n-1)\} = E\left\{\begin{bmatrix} \hat{s}_0(n) \\ \ddots \\ \hat{s}_{M+N-1}(n) \end{bmatrix}\begin{bmatrix} \hat{s}_0^*(n-1) & \dots & \hat{s}_{M+N-1}^*(n-1) \end{bmatrix}\right\} = J \tag{3.2}$$

where J is

$$J = \begin{bmatrix} 0_{1\times(M+N-1)} & 0 \\ I_{(M+N-1)\times(M+N-1)} & 0_{(M+N-1)\times 1} \end{bmatrix} \tag{3.3}$$

because the transmitted signal is assumed white. The equalizer matrix Z is obtained from this condition derived as follows.

¿From (3.1) and (3.2), we know

$$J = E\{\hat{\mathbf{s}}_M(n)\hat{\mathbf{s}}_M^H(n-1)\} = ZE\{\mathbf{x}_N(n)\mathbf{x}_N^H(n-1)\}Z^H \tag{3.4}$$

Define the crosscorrelation matrix

$$R_1 \triangleq E\{\mathbf{x}_N(n)\mathbf{x}_N^H(n-1)\} \tag{3.5}$$

The equation (3.4) becomes

$$ZR_1Z^H = J \tag{3.6}$$

We have discussed in Chapter 2 that the equalizer matrix (2.17) has the form $Z = Q\Lambda_1^{-1/2}S^H$; substituting it into the above equation yields the following equation

$$J = Q\Lambda_1^{-1/2}S^H R_1 S\Lambda_1^{-1/2}Q^H \triangleq QR_2Q^H. \tag{3.7}$$

where $R_2 = \Lambda_1^{-1/2}S^H R_1 S\Lambda_1^{-1/2}$. Note that R_2 can be evaluated from the subspace decomposition of R_0 and the direct evaluation of R_1. Hence, the above equation enables us to obtain Q.

3.1.2 Q Matrix Estimation

It is clear that the rank of R_2 is $M+N-1$ as we have

$$QR_2R_2^HQ^H = JJ^H = \begin{bmatrix} 0 & \\ & I \end{bmatrix} \tag{3.8}$$

Since $Q^H = [\mathbf{q}_0, \ldots, \mathbf{q}_{M+N-1}]$, $\mathbf{q}_0$ therefore is the eigenvector of $R_2 R_2^H$ corresponding to the eigenvalue of zero. The eigenvector can be obtained by performing the singular value decomposition (SVD) of R_2 as below

$$R_2 = U \Sigma V^H \tag{3.9}$$

with descending singular values in Σ, then $\mathbf{q}_0$ can be selected as last column of U. After we have $\mathbf{q}_0$, the other columns of Q can also be obtained recursively as below,

$$\mathbf{q}_i = R_2 \mathbf{q}_{i-1} \quad i = 1, \ldots, M + N - 1. \tag{3.10}$$

which can be shown directly from (3.7). One way to choose the best delayed equalizer in the equalizer matrix Z is to minimize the following constant modulus index

$$D(d) = E\{(\left|\mathbf{z}_d^H \mathbf{x}_N(n)\right|^2 - 1)^2\} \tag{3.11}$$

The equalizer having the smallest D value will be considered as the best delayed equalizer.

In summary, the channel output whitening algorithm for direct equalization is listed in following:

1. Define the equalizer length N, which should be used reasonably large to satisfy the condition $LN \geq (M + N)$.

2. Compute the correlation matrix R_0 and R_1 in (2.11) and (3.5).

3. Estimate the signal subspace of rank $M + N$ by computing the subspace decomposition of R_0.

4. Compute R_2 (3.7) and its SVD.

5. Estimate Q according to (3.10), and compute the equalization matrix Z (2.17).

6. Choose the best delayed equalizer in Z according to (3.11)

3.2 Estimate Q with Channel Matrix structure Method

In this section we exploit the idea of mutually referenced filters (MREs) and the special property Toeplitz structure of the channel matrix to design the blind equalizer. Instead of estimating all equalizers in a equalization matrix, the method proposed Here estimates one equalizer with the delay $d = \text{int}((M+N)/2)$ which, as commonly done in equalizer design, results in good performance.

We observe that the channel matrix H is a Toeplitz structure of $LN \times (M+N)$; excluding the first row of H yields H_r as below

$$H_r = \begin{bmatrix} \mathbf{0} & \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(M) & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(M) & \mathbf{0} \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(M) \end{bmatrix} \tag{3.12}$$

Since the channel matrix H is assumed full rank of $M+N$, the rank of H_r, of course, is $M+N-1$ because of the existing first zero column in H_r. ¿From (2.16) $H = S\Lambda_1^{-1/2}Q^H$ and define $F = S\Lambda_1^{-1/2}$; it follows that,

$$H = FQ^H \tag{3.13}$$

Also define F_r obtained from F by truncating its first row; we obtain the following equation:

$$H_r = F_r Q^H \tag{3.14}$$

Again using $Q^H = [\mathbf{q}_0, \dots, \mathbf{q}_{M+N-1}]$ in (3.14), we obtain

$$F_r \mathbf{q}_0 = \mathbf{0} \tag{3.15}$$

Therefore, the vector $\mathbf{q}_0$ can be obtained as the eigenvector of $F_r^H F_r$ corresponding to the zero eigenvalue; equivalently, $\mathbf{q}_0$ can be obtained from the SVD of F_r. The solution of $\mathbf{q}_0$ can also be obtained from the following optimization problem

$$\min_{\mathbf{q}_0} \ \mathbf{q}_0^H F_r^H F_r \mathbf{q}_0 \qquad s.t. \ \|\mathbf{q}_0\| = 1 \tag{3.16}$$

Therefore, if we have the equalizer matrix Z that can recover original transmitted signal perfectly, then the conditions in the delayed equalizer $\mathbf{z}_d^H$ are simply rewritten as

$$\left[\begin{array}{c} I_{M+N-1}, \mathbf{0} \end{array}\right] Z\mathbf{x}_N(n) = \left[\begin{array}{c} \mathbf{0}, I_{M+N-1} \end{array}\right] Z\mathbf{x}_N(n+1), \tag{3.17}$$

Using (2.17) in (3.17) yields

$$\left[\begin{array}{c} I_{M+N-1}, \mathbf{0} \end{array}\right] Q\Lambda_1^{-1/2} S^H \mathbf{x}_N(n) = \left[\begin{array}{c} \mathbf{0}, I_{M+N-1} \end{array}\right] Q\Lambda_1^{-1/2} S^H \mathbf{x}_N(n+1) \tag{3.18}$$

Since S^H and $\Lambda_1^{-1/2}$ are already obtained from the subspace decomposition, denote $\mathbf{y}_N(n) = \Lambda_1^{-1/2} S^H \mathbf{x}_N(n)$; the equation (3.18) becomes

$$\left[\begin{array}{c} I_{M+N-1}, \mathbf{0} \end{array}\right] Q\mathbf{y}_N(n) = \left[\begin{array}{c} \mathbf{0}, I_{M+N-1} \end{array}\right] Q\mathbf{y}_N(n+1) \tag{3.19}$$

which reads equivalently

$$\begin{bmatrix} \mathbf{q}_0^H \\ \mathbf{q}_1^H \\ \vdots \\ \mathbf{q}_{M+N-2}^H \end{bmatrix} \mathbf{y}_N(n) = \begin{bmatrix} \mathbf{q}_1^H \\ \mathbf{q}_2^H \\ \vdots \\ \mathbf{q}_{M+N-1}^H \end{bmatrix} \mathbf{y}_N(n+1) \tag{3.20}$$

Thus we obtain

$$\mathbf{q}_i^H \mathbf{y}_N(n) = \mathbf{q}_{i+1}^H \mathbf{y}_N(n+1) \quad \text{for } i = 0, \ldots, M+N-2 \tag{3.21}$$

Given the delay d, by the above recursion we have

$$\mathbf{q}_i^H \mathbf{y}_N(n) = \mathbf{q}_{i+d_g}^H \mathbf{y}_N(n+d_g) \quad \text{for } i = 0, \ldots, M+N-d_g-1 \tag{3.22}$$

Thus

$$\mathbf{q}_0^H \mathbf{y}_N(n) = \mathbf{q}_{d_g}^H \mathbf{y}_N(n+d_g) \tag{3.23}$$

Hence, given $\mathbf{q}_0^H$, $\mathbf{y}_N(n)$, and $\mathbf{y}_N(n+d_g)$ we can obtain $\mathbf{q}_{d_g}^H$ by least-square solution, yielding

$$\mathbf{q}_0^H E\{\mathbf{y}_N(n)\mathbf{y}_N^H(n+d_g)\} = \mathbf{q}_{d_g}^H E\{\mathbf{y}_N(n+d_g)\mathbf{y}_N^H(n+d_g)\} \tag{3.24}$$

$$\mathbf{q}_0^H R_{d_g} = \mathbf{q}_{d_g}^H \tag{3.25}$$

where the autocorrelation matrix of $\mathbf{y}_{N+d_g}$ is an identity matrix.

In summary, this algorithm is listed in the following:

1. Define the equalizer length N such that $LN \geq (M + N)$.

2. Compute the correlation matrix R_0 in (2.11).

3. Estimate the signal subspace of rank $M + N$ from the subspace decomposition of R_0.

4. Compute $\mathbf{q}_0$ using SVD from F_r.

5. Compute R_{d_g} and use (3.25) to find $\mathbf{q}_{d_g}^H$.

6. Compute $\mathbf{z}_{d_g}^H = \mathbf{q}_{d_g}^H \Lambda_1^{-1/2} S^H$.

3.3 Estimate Q with Iterative Least Square Method

In above section, we observe that to estimate the equalizer matrix Z usually needs to compute the auto-correlation and cross-correlation matrices of received signals. This section presents one new blind equalization algorithm; this algorithm uses the idea similar to that of mutually referenced filters (MREs) to derive equalizer matrix but it only uses the autocorrelation matrix of received signals.

As discussed above, given $\mathbf{q}_0$, we can obtain $\mathbf{q}_1$ and $\mathbf{q}_i$ for $i > 1$ recursively. Denote A_N as below

$$A_N = \begin{bmatrix} \mathbf{y}_N(n)^H \\ \mathbf{y}_N(n+1)^H \\ \vdots \\ \mathbf{y}_N(n+N+M-2)^H \\ \mathbf{y}_N(n+N+M-1)^H \\ \vdots \end{bmatrix} \tag{3.26}$$

where the row number of A_N is greater than the rank of channel matrix H $(rank(H) = M + N)$. Let B_N obtained by excluding the first low of A_N and adding the last received signal $\mathbf{y}_{new}^{H}$ in the last row as below

$$B_N = \begin{bmatrix} \mathbf{y}_N(n+1)^H \\ \vdots \\ \mathbf{y}_N(n+N+M-2)^H \\ \mathbf{y}_N(n+N+M-1)^H \\ \vdots \\ \mathbf{y}_N(new)^H \end{bmatrix}, \tag{3.27}$$

The MRE (3.21) enables us to write the following equation

$$A_N \mathbf{q}_i = B_N \mathbf{q}_{i+1} \quad \text{for } 0, \ldots, M+N-2 \tag{3.28}$$

Since the row numbers of A_N and B_N are greater than the rank of channel matrix H, we can assume that both A_N and B_N are full rank, thus both A_N and B_N have pseudoinverse matrices,

$$A_N^{\dagger} = \left(A_N^H A_N\right)^{-1} A_N^H \tag{3.29}$$

$$B_N^{\dagger} = \left(B_N^H B_N\right)^{-1} B_N^H \tag{3.30}$$

Multiplying (3.28) by $\mathbf{B}_N^{\dagger}$ on the left yields

$$B_N^{\dagger} A_N \mathbf{q}_i = \mathbf{q}_{i+1} \quad \text{for } 0, \ldots, M+N-2 \tag{3.31}$$

Hence, the equation (3.31) enables us to obtain from an initial $\mathbf{q}_0$ to compute $\mathbf{q}_i$ $i \leq 1$ as shown in the flow path below

$$\mathbf{q}_0 \longrightarrow \mathbf{q}_1 \longrightarrow \ldots \longrightarrow \mathbf{q}_{M+N-2} \longrightarrow \mathbf{q}_{M+N-1}$$

Similarly, the same operation (3.21) can be reversed; multiplying (3.21) by $A_N^{\dagger}$ we obtain

$$\mathbf{q}_i = A_N^{\dagger} B_N \mathbf{q}_{i+1} \quad \text{for } M+N-2, \ldots, 0 \tag{3.32}$$

Given $\mathbf{q}_{M+N-1}$ we can obtain $\mathbf{q}_i$ via (3.32) as shown in the flow path below

$$\mathbf{q}_0^{'} \longleftarrow \mathbf{q}_1^{'} \longleftarrow \ldots \longleftarrow \mathbf{q}_{M+N-2}^{'} \longleftarrow \mathbf{q}_{M+N-1}^{'}$$

Combining (3.31) and (3.32) together, a new flow loop is formed as shown below:

$$
\begin{array}{ccccccccc}
\mathbf{q}_0 & \longrightarrow & \mathbf{q}_1 & \longrightarrow & \ldots & \longrightarrow & \mathbf{q}_{M+N-2} & \longrightarrow & \mathbf{q}_{M+N-1} \\
\Uparrow & & & & & & & & \Downarrow \\
\mathbf{q}_0^{'} & \longleftarrow & \mathbf{q}_1^{'} & \longleftarrow & \ldots & \longleftarrow & \mathbf{q}_{M+N-2}^{'} & \longleftarrow & \mathbf{q}_{M+N-1}^{'}
\end{array}
$$

Given an initial $\mathbf{q}_0$, the loop will continue until the loop recursion converges. One criterion is to set $\left|\mathbf{q}^{0-\mathbf{q}_0^{'}}\right| \leq \epsilon$ where ϵ is a user defined threshold. From our experience, the loop converges with less than 10 iterations for $\epsilon \leq 10^{-3}$.

To summarize, this algorithm is listed in the following:

1. Define the equalizer length N, which can be used reasonably large to satisfy the condition $LN \geq (M + N)$.

2. Compute the correlation matrix R_0 in (2.11).

3. Estimate the signal subspace of rank $M + N$ from the subspace decomposition of R_0.

4. Set the initial g_0 and a threshold ϵ.

5. Use the iteration loop to estimate Q and compute the equalizer matrix Z.

Chapter 4

Adaptive Subspace Blind Equalizer

The blind equalizers discussed above are designed by batch data or called off-line data. This chapter presents an adaptive blind equalizer by using the OPAST algorithm [7], [8], generally Gram-Schmidt orthogonalization [10], and bi-iteration algorithm [9]. We discuss OPAST first and then the adaptive equalizer is developed.

4.1 OPAST Algorithm

Define $\mathbf{r}(k)$ is a sequence of $n \times 1$ random vectors, and the autocorrelation matrix of $\mathbf{r}(k)$ is $C = E\{\mathbf{r}(k)\mathbf{r}(k)^H\}$. Consider the problem of estimating the principal subspace spanned by the sequence of dimension $r < n$. So consider the following scalar function:

$$J(W) = E\{\left\|\mathbf{r} - WW^H\mathbf{r}\right\|^2\} \tag{4.1}$$

where a subspace matrix argument $W \in \mathbb{C}^{n\times r}$. It has been shown in PAST algorithm that

1. W is a stationary point of $J(W)$ iff $W = S\widetilde{Q}$, where S is an $n\times r$ matrix containing any r distinct eigenvectors of C, and $\widetilde{Q}$ is any $r \times r$ unitary matrix.

2. All stationary points of $J(W)$ are saddle points, expect when S contains that r dominant eigenvectors of C. In this case, $J(W)$ obtain the global minimum.

In tracking applications, we can replace the autocorrelation matrix C with its recursive version $C(t) = \alpha C(t-1) + \mathbf{r}(t)\mathbf{r}(t)^H$ at the tth iteration where α is a forgetting factor chosen between (0,1]. In PAST algorithm, a fast implementation is proposed based on the projection approximation that if the subspace matrix $W(t)$ is slowly varying with t, then $C(t)W(t) \approx C(t)W(t-1)$. Using this projection approximation, the matrix product $C(t)W(t-1)$ and the matrix inverse $\left(W^H(t-1)C(t)W(t-1)\right)^{-1}$ can be computed in $O(np)$. Define $Z(n) \triangleq \left(W^H(t-1)C(t)W(t-1)\right)^{-1}$, then PAST algorithm can be written in Table 4.1.

Table 4.1: PAST Algorithm

$$
\begin{aligned}
\mathbf{q}(t) &= \tfrac{1}{\alpha} Z(t-1)\mathbf{y}(t) \\
\mathbf{y}(t) &= W^H(t-1)\mathbf{r}(t) \\
\gamma(t) &= \tfrac{1}{(1+\mathbf{y}^H(t)\mathbf{q}(t))} \\
\mathbf{p}(t) &= \gamma(t)\left(\mathbf{r}(t) - W(t-1)\mathbf{y}(t)\right) \\
Z(t) &= \tfrac{1}{\alpha} Z(t-1) - \gamma(t)\mathbf{q}(t)\mathbf{q}(t)^H \\
W(t) &= W(t-1) + \mathbf{p}(t)\mathbf{q}^H(t)
\end{aligned}
$$

The OPAST algorithm consists of the PAST algorithm and an orthonormalization step of the subspace matrix at each iteration

$$W(t) \triangleq W(t)\left(W^H(t)W(t)\right)^{-1/2} \tag{4.2}$$

where $\left(W^H(t)W(t)\right)^{-1/2}$ defines an inverse square root of $\left(W^H(t)W(t)\right)$. we will use the iterating equation of $W(t)$ to keep $W(t-1)$ be an orthonormal matrix, we get

$$W^H(t)W(t) = I + \|\mathbf{p}(t)\|^2 \mathbf{q}(t)\mathbf{q}^H(t) = I + \mathbf{x}\mathbf{x}^H \tag{4.3}$$

where we know the fact that $W^H(t-1)\mathbf{p}(t) = \mathbf{0}$ from PAST algorithm, and define $\mathbf{x} \triangleq \|\mathbf{p}(t)\| \, \mathbf{q}(t)$. Therefore

$$\left(W^H(t)W(t)\right)^{-1/2} = I + \frac{1}{\|\mathbf{x}\|^2}\left(\frac{1}{\sqrt{1+\|\mathbf{x}\|^2}} - 1\right)\mathbf{x}\mathbf{x}^H \tag{4.4}$$

$$= I + \tau(t)\mathbf{q}(t)\mathbf{q}(t)^H \tag{4.5}$$

where

$$\tau(t) \triangleq \frac{1}{\|\mathbf{q}(t)\|^2}\left(\frac{1}{\sqrt{1+\|\mathbf{p}(t)\|^2\,\|\mathbf{q}(t)\|^2}} - 1\right).$$

Exploiting (4.3), (4.5) and the iterating of $W(t)$, we can have

$$W(t) = \left(W(t-1) + \mathbf{p}(t)^H\mathbf{q}(t)\right)\left(I + \tau(t)\mathbf{q}(t)\mathbf{q}(t)^H\right) \tag{4.6}$$

$$= W(t-1) + \mathbf{p}'(t)\mathbf{q}^H(t) \tag{4.7}$$

where

$$\mathbf{p}'(t) \triangleq \tau(t)W(t-1)\mathbf{q}(t) + \left(1 + \tau(t)\,\|\mathbf{q}(t)\|^2\right)\mathbf{p}(t).$$

The OPAST algorithm can be rewritten as the PAST (Table 4.1) combined with that shown in Table 4.2.

4.2 Modified Algorithm for r-Dominant Generalized Eigenvectors

The PAST algorithm obtains $W = S\widetilde{Q}$ where the columns of S consists of r principal eigenvectors of C and $\widetilde{Q}$ is any $r \times r$ unitary matrix. Consider the following constraint:

$$W^H C W = M \tag{4.8}$$

where $M = \text{diag}(m_1, m_2, \ldots, m_r)$ and $m_1, m_2, \ldots, m_r > 0$. Therefore, we can obtain that

$$\widetilde{Q}^H\Sigma = M\widetilde{Q}^H \tag{4.9}$$

Table 4.2: OPAST Algorithm

$W(t)$	$= W(t-1) + \mathbf{p}^{'}(t)\mathbf{q}^{H}(t)$
$\tau(t)$	$= \frac{1}{\\|\mathbf{q}(t)\\|^2}\left(\frac{1}{\sqrt{1+\\|\mathbf{p}(t)\\|^2\\|\mathbf{q}(t)\\|^2}} - 1\right)$
$\mathbf{p}^{'}(t)$	$= \tau(t)W(t-1)\mathbf{q}(t) + \left(1+\tau(t)\\|\mathbf{q}(t)\\|^2\right)\mathbf{p}(t)$

where the diagonal matrix Σ is a generalized eigenvalue matrix of C.

Since the r principal generalized eigenvalues are different, so it is simple to show that

$$M = \Sigma$$

$$\widetilde{Q} = I$$

$$W = S$$

This means that if we use the cost function (4.1) and the constraint (4.8), then we can obtain the r principal generalized eigenvectors. In fact, the constraint (4.8) implies that W should be C orthogonal. This can be obtained by orthogonalizing the matrix W via iterating algorithm for dominant eigen-subspace. The algorithm for orthogonalization is given as

$$\mathbf{m}_1 = \mathbf{w}_1 \tag{4.10}$$

$$\mathbf{m}_j = \mathbf{w}_j - \sum_{i=1}^{j-1} \frac{\mathbf{m}_i^H C \mathbf{w}_j}{\mathbf{m}_i^H C \mathbf{m}_i} \mathbf{m}_i \tag{4.11}$$

where $\mathbf{w}_i$ is the ith column of W. When $C = I$, the C orthogonalization is the same as the Gram-Schmidt orthogonalization (GSO) method. ¿From the GSO method, we also can obtain eigenvalues of C, because in (4.11) we must compute $\mathbf{m}_i^H C \mathbf{m}_i$ and $\mathbf{m}_i^H C \mathbf{m}_i$ equal to $\mathbf{m}_i$ eigenvalue of C that is useful for blind equalization algorithm.

Thus, the above idea results in the following modified algorithm Table 4.3 for r principal generalized eigenvectors.

Table 4.3: Modified Algorithm

$$
\begin{aligned}
\mathbf{q}(t) &= \tfrac{1}{\alpha} Z(t-1)\mathbf{y}(t) \\
\mathbf{y}(t) &= W^H(t-1)\mathbf{r}(t) \\
\gamma(t) &= \tfrac{1}{(1+\mathbf{y}^H(t)\mathbf{q}(t))} \\
\mathbf{p}(t) &= \gamma(t)\left(\mathbf{r}(t) - W(t-1)\mathbf{y}(t)\right) \\
Z(t) &= \tfrac{1}{\alpha} Z(t-1) - \gamma(t)\mathbf{q}(t)\mathbf{q}(t)^H \\
\tau(t) &= \tfrac{1}{\|\mathbf{q}(t)\|^2}\left(\tfrac{1}{\sqrt{1+\|\mathbf{p}(t)\|^2\|\mathbf{q}(t)\|^2}} - 1\right) \\
\mathbf{p}'(t) &= \tau(t)W(t-1)\mathbf{q}(t) + \left(1+\tau(t)\|\mathbf{q}(t)\|^2\right)\mathbf{p}(t) \\
W(t) &= W(t-1) + \mathbf{p}'(t)\mathbf{q}^H(t) \\
C(t) &= \alpha C(t-1) + \mathbf{r}(t)\mathbf{r}(t)^H \\
(W(t), \Sigma(t)) &= C(t) \text{ orthogonalize } (W(t))
\end{aligned}
$$

4.3 Adapive equalization algorithm Using Channel matrix Method

The adaptive equalizer starts to use OPAST algorithm to find the signal subspace $W = S\widetilde{Q}$. Since S and Λ_1 are required in the adaptive equalizer, we use the modified algorithm to achieve this goal. Define $F = S\Lambda_1^{1/2}$ and F_r contain the last $L(N-1)$ rows of the F. Then the classical bi-iteration algorithm is used to compute the dominant singular values and vectors of F_r to obtain $\mathbf{q}_0$; the algorithm is shown in the following Table 4.4.

In the bi-iteration algorithm, $A(t)$ and $B(t)$ denote auxiliary matrix of size $(M+N) \times L(N-1)$ and $L(N-1) \times (M+N)$. The QR factorizations in (Table 4.4) produce the corresponding dominant left and right singular vectors in the SVD of F_r. The triangular

Table 4.4: Bi-Iteration Algorithm

$Q_A(t=0)$	$= I_{M+N}$	for $t = 1, 2, 3, \ldots$ for each time step do
$B(t)$	$= F_r(t)Q_A(t-1)$	
$B(t)$	$= Q_B(t)R_B(t)$: $L(N-1) \times (M+N)$ QR-factorization	
$A(t)$	$= F_r(t)^H Q_B(t)$	
$A(t)$	$= Q_A(t)R_A(t)$: $(M+N) \times L(N-1)$ QR-factorization	

matrix will converge toward the diagonal matrix of the dominant singular values.

$$Q_A(t) \longrightarrow V(t)$$

$$Q_B(t) \longrightarrow U(t)$$

$$R_A(t), R_B(t) \longrightarrow \Sigma_{SVD}(t)$$

and the structure of the exact SVD of the F_r

$$F_r(t) = U(t)\Sigma_{SVD}(t)V^H(t) \tag{4.12}$$

where

U(t) is matrix of left singular vectors;

V(t) is matrix of right singular vectors;

$\Sigma_{SVD}(t)$ is diagonal matrix of singular values.

Then $\mathbf{q}_0$ can be selected as the last column vector of right singular vector $V(t)$.

¿From (3.25), we need to obtain R_{d_g}. Define $R_{d_g} = E\{\mathbf{y}_N \mathbf{y}_{N+d_g}^H\}$; it can be rewritten as

$$R_{d_g} = \Lambda_1^{-1/2} S^H E\{\mathbf{x}_N(n)\mathbf{x}_N(n+d_g)^H\} S \Lambda_1^{-1/2} \tag{4.13}$$

$$= \Lambda_1^{-1} F^H E\{\mathbf{x}_N(n)\mathbf{x}_N(n+d_g)^H\} F \Lambda_1^{-1} \tag{4.14}$$

Define $R_d = E\{\mathbf{x}_N(n)\mathbf{x}_N(n+d_g)^H\}$; the computation of R_d can be obtain below

$$R_d(n) = \alpha R_d(n-1) + \mathbf{x}_N(n)\mathbf{x}_N(n+d_e)^H \tag{4.15}$$

where α is forgetting factor chosen between $(0,1]$. Therefore, R_{d_g} also can be rewritten as

$$R_{d_g}(n) = \Lambda_1^{-1} F^H R_d(n) F \Lambda_1^{-1} \tag{4.16}$$

Combining all iterating algorithms together we obtain the adaptive equalizer which is shown in Table 4.5.

Table 4.5: Subspace Blind Equalization Algorithm

$$\mathbf{q}(t) = \frac{1}{\alpha} Z(t-1)\mathbf{y}(t)$$

$$\mathbf{y}(t) = S^H(t-1)\mathbf{x}_N(t)$$

$$\gamma(t) = \frac{1}{(1+\mathbf{y}^H(t)\mathbf{q}(t))}$$

$$\mathbf{p}(t) = \gamma(t)\left(\mathbf{x}_N(t) - S(t-1)\mathbf{y}(t)\right)$$

$$Z(t) = \frac{1}{\alpha} Z(t-1) - \gamma(t)\mathbf{q}(t)\mathbf{q}(t)^H$$

$$\tau(t) = \frac{1}{\|\mathbf{q}(t)\|^2}\left(\frac{1}{\sqrt{1+\|\mathbf{p}(t)\|^2\|\mathbf{q}(t)\|^2}} - 1\right)$$

$$\mathbf{p}^{'}(t) = \tau(t)S(t-1)\mathbf{q}(t) + \left(1+\tau(t)\,\|\mathbf{q}(t)\|^2\right)\mathbf{p}(t)$$

$$S(t) = S(t-1) + \mathbf{p}^{'}(t)\mathbf{q}^H(t)$$

$$R_0(t) = \alpha R_0(t-1) + \mathbf{x}_N(t)\mathbf{x}_N(t)^H$$

$$(S(t), \Lambda_1(t)) = R_0(t) \text{ orthogonalize } (S(t))$$

$$F(t) = S(t)\Lambda_1^{1/2}(t)$$

$$F_r(t) = \text{the last } L(N-1) \text{ rows of } F(t)$$

$$B(t) = F_r(t)Q_A(t-1)$$

$$B(t) = Q_B(t)R_B(t)$$

$$A(t) = F_r(t)^H Q_B(t)$$

$$A(t) = Q_A(t)R_A(t)$$

$$\mathbf{q}_0(t) = \text{the last column of } Q_A(t)$$

$$R_d(t) = \alpha R_d(n-1) + \mathbf{x}_N(n)\mathbf{x}_N(n+d_e)^H$$

$$R_{d_g}(t) = \Lambda_1^{-1}(t)F(t)^H R_d(t)F(t)\Lambda_1^{-1}(t)$$

$$\mathbf{q}_{d_g}^H(t) = \mathbf{q}_0^H(t)R_{d_g}(t)$$

Chapter 5

Simulations

In this chapter, we use simulations to examine the performance of our new algorithms described in this thesis. We compare the performance of the proposed methods with some typical existing algorithms below:

- the output whitening method in [1] (LF) for direct equalization;
- the MRE algorithm in [5] (MRE) for estimating equalizers with all passible delays;
- the subspace algorithm [3] (TXK) for channel identification and equalization;
- the linear prediction-based algorithm in [6] (PS) for adaptive equalization.

5.1 Criteria of Performance Measure

As a performance measure, we estimate the residual intersymbol interference (ISI) over 100 Monte Carlo runs as done in most other measurements. Let the "overall" channel impulse response be

$$c(n) = \sum_{i=0}^{L-1} \sum_{j=0}^{N-1} g_i(j) h_i(n-j). \tag{5.1}$$

The residual ISI is defined as

$$\text{ISI} = \frac{\sum_n |c(n)|^2 - \max_n |c(n)|^2}{\max_n |c(n)|^2} \tag{5.2}$$

We also estimate the signal-to interference-and-noise ratio (SINR) of the equalizer output, which may be more suitable for studying the noise performance

$$\text{SINR} = \frac{\sigma_s^2 \max_n |c(n)|^2}{\sigma_s^2 \left(\sum_n |c(n)|^2 - \max_n |c(n)|^2\right) + \sigma_n^2 \sum_i \sum_j |g_i(j)|^2} \tag{5.3}$$

where σ_s^2 is the variance of the transmitted symbols.

For all simulations, the signal-to-noise ratio (SNR) is for the input to the equalizer

$$\text{SNR} = \frac{E\left\{|x(n) - n(n)|^2\right\}}{E\left\{|n(n)|^2\right\}}. \tag{5.4}$$

For each experiment, we use an i.i.d. input sequence drawn from a 16-QAM constellation. The noise is drawn from a white Gaussian distribution at a varying SNR. The channel is drawn from Table 5.1 [2].

Table 5.1: Channel Coefficients [2]

$\mathbf{h}(0)$	$\mathbf{h}(1)$	$\mathbf{h}(2)$	$\mathbf{h}(3)$	$\mathbf{h}(4)$
$-0.049 + 0.359i$	$0.482 - 0.569i$	$0.556 + 0.587i$	1	$-0.171 + 0.061i$
$0.443 - 0.0364i$	1	$0.921 - 0.194i$	$0.189 - 0.208i$	$-0.087 - 0.054i$
$-0.221 - 0.322i$	$-0.199 + 0.918i$	1	$-0.284 - 0.524i$	$0.136 - 0.190i$
$0.417 + 0.030i$	1	$0.873 + 0.145i$	$0.285 + 0.309i$	$-0.049 + 0.161i$

The number of subchannels is $L = 4$, the channel order is $M = 4$, and chooses $N = 4$ to let the delay be $d = 4$. In addition to above channel model [2], we also use the channel impulse response in [3] that is an approximation of two-ray multipath environment and the channel is obtained from delayed raised consine pulses. A single pulse is described by $c_t(t, \beta)$ where β is a roll-off factor. The channel impulse response is:

$$h(t) = (0.2c(t, 0.11) + 0.4c(t - 0.25, 0.11))\, W_{6T}(t) \tag{5.5}$$

where $W_{6T}(t)$ is a square window of duration 6 symbol interval and is shown in Fig. 5.1

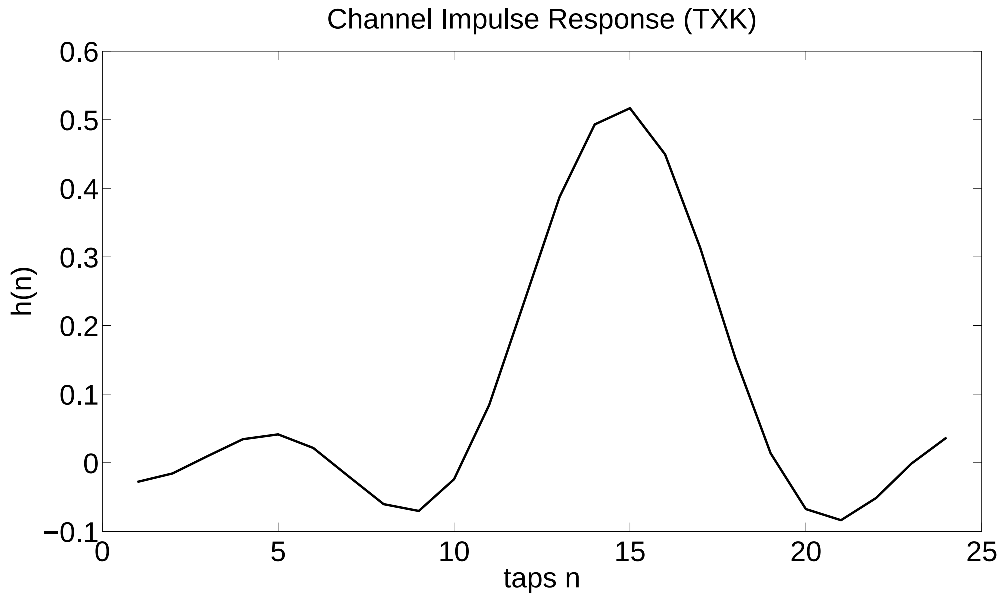

Fig. 5.1: Oversampling ratio L=4 and channel length M=6

5.2 Simulation Results

In this section, we use several examples to demonstrate that our method is effective to design an equalizer. We also use ISI and SINR as performance indexes to compare the difference of all algorithms.

5.2.1 Simulations of Off-Line Designed Blind Equalizer

First, the channel model in Table 5.1 is used to generate 500 symbols for equalizer design under SNR=25dB; the equalizer outputs result in the constellation shown in Fig. 5.2 and 5.3 which clearly shows that the resolution using the iterative least square method and the channel structure method is be better than that of the Fan's method. In Fig. 5.4 and 5.5, we compare the performance indexes (ISI and SINR) and observe that the channel structure method has better performance; the iterative least square method is better as the SNR exceeding 25dB. In Fig. 5.6 and 5.7, we use different symbols numbers and SNRs to show that our methods also have better performance. Next, we use the channel of TXK to produce the received data, and compute ISI and SINR for comparison. In Fig. 5.8 and 5.9, we observe that the change of N can influence the performance; when N=7 the iterative least square method improves its performance and in Fig. 5.10 and 5.11 using different N to display channel structure method can resist different channel and iterative least square method in $N = 7$ still have better performance.

5.2.2 Simulations of On-Line Blind Equalization

In this subsection, we use the adaptive channel structure method to illustrate its performance. In Fig. 5.12 we show equalizer estimation this algorithm for 25dB SNR and with 500 symbols and in Fig. 5.13 and 5.14 show that our algorithm have better performance than the linear prediction algorithm. For the channel of TXK, the performance of our algorithm for SNR below 25dB still remains an acceptable ISI and SINR.

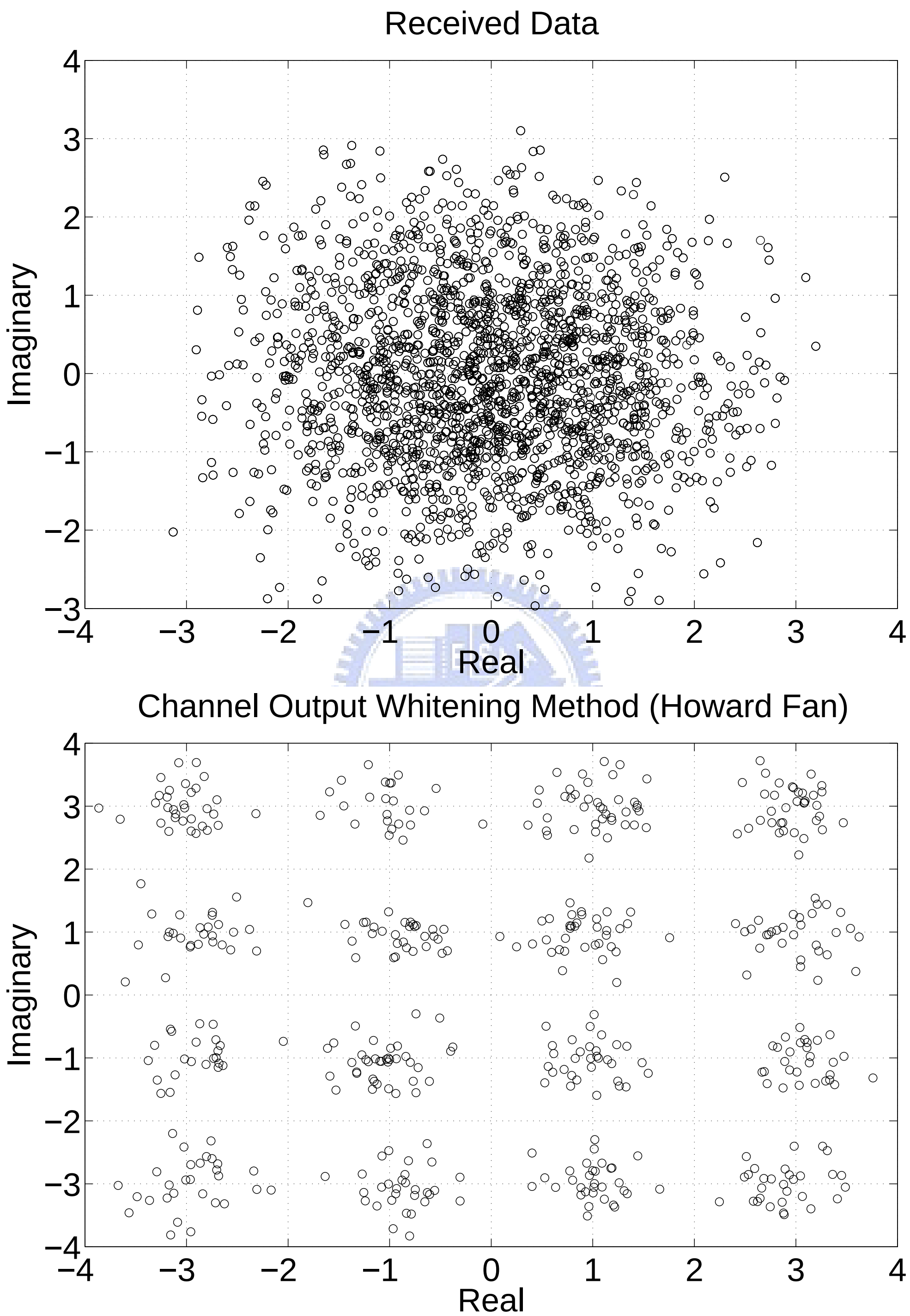

Fig. 5.2: Signal constellations for 500 symbols at SNR=25dB. (a) Without equalization. (b) After equalization for Fan's method.

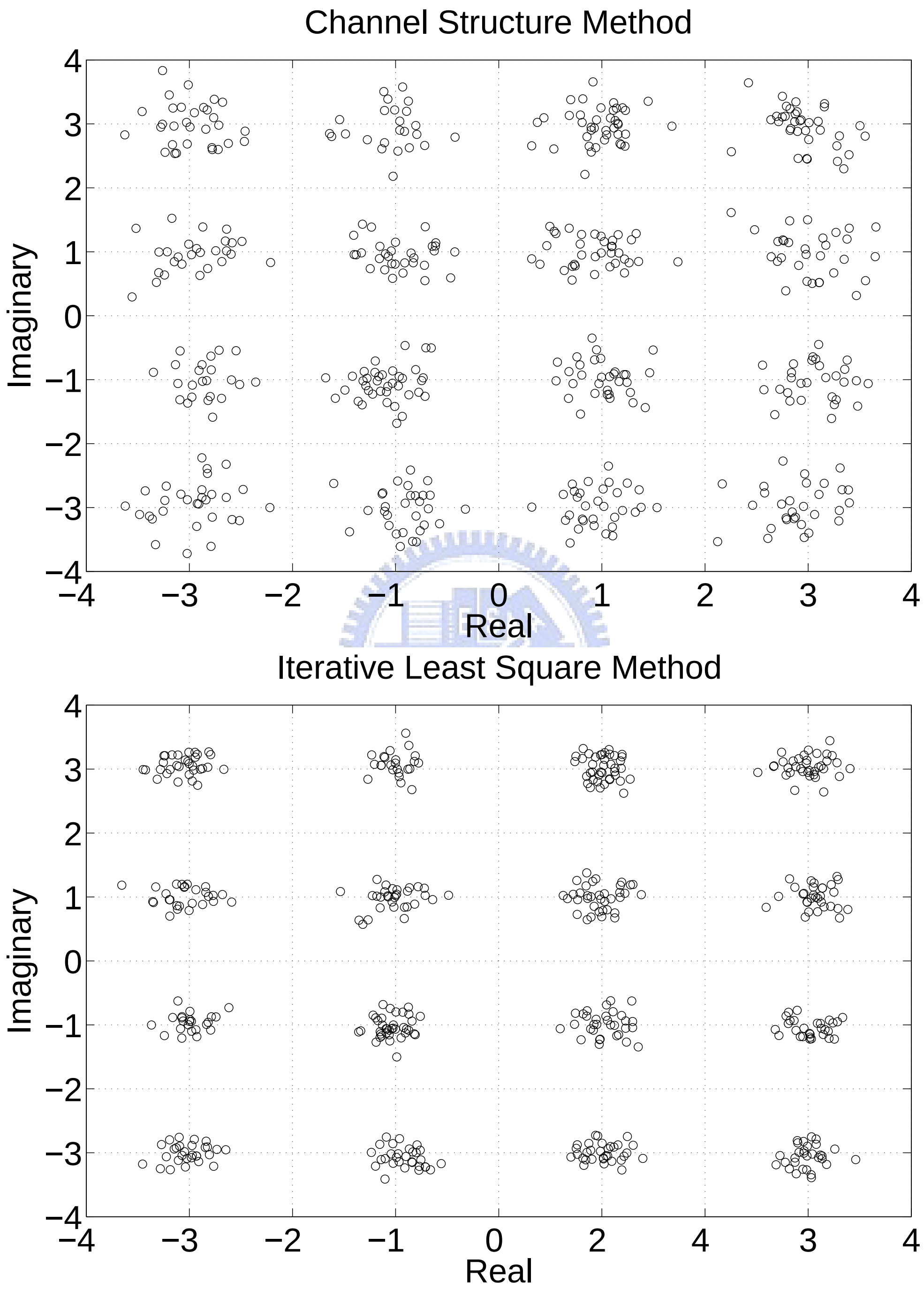

Fig. 5.3: After equalization for Channel Structure and Iterative Least Square method.

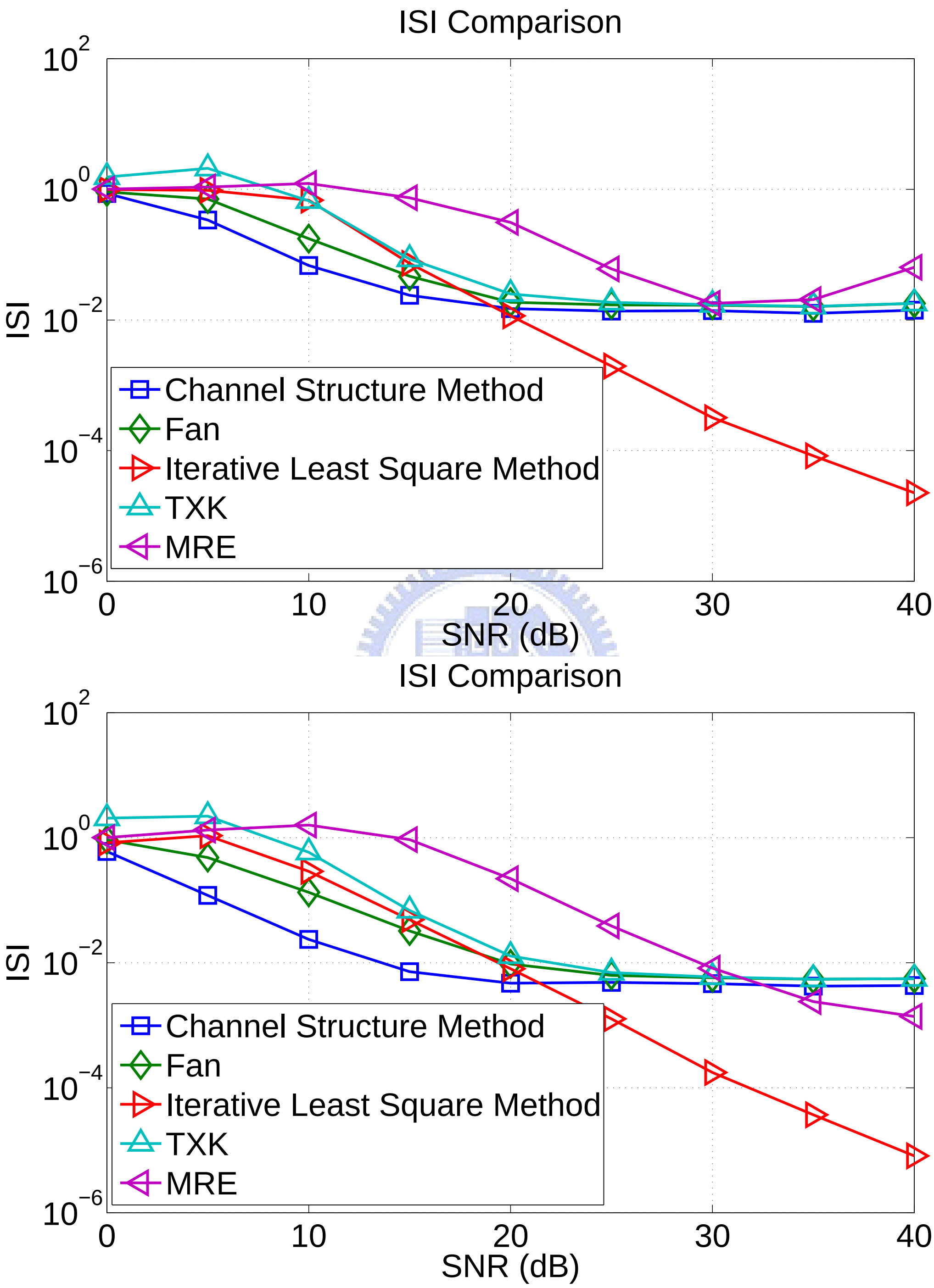

Fig. 5.4: Performance comparison versus different SNR for ISI. ((a) 250 and (b) 750 symbols)

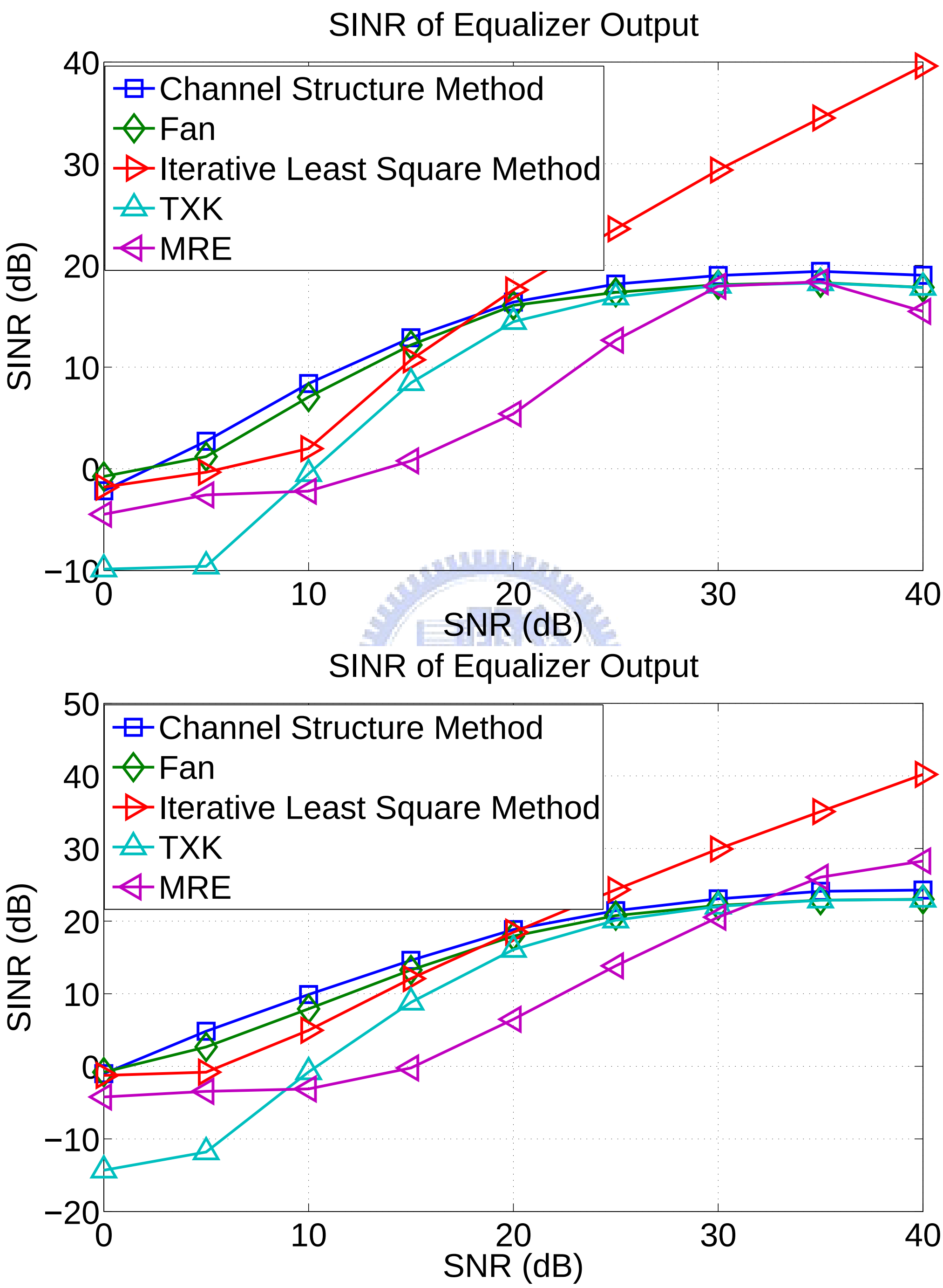

Fig. 5.5: Performance comparison versus different SNR for SINR. ((a) 250 and (b) 750 symbols)

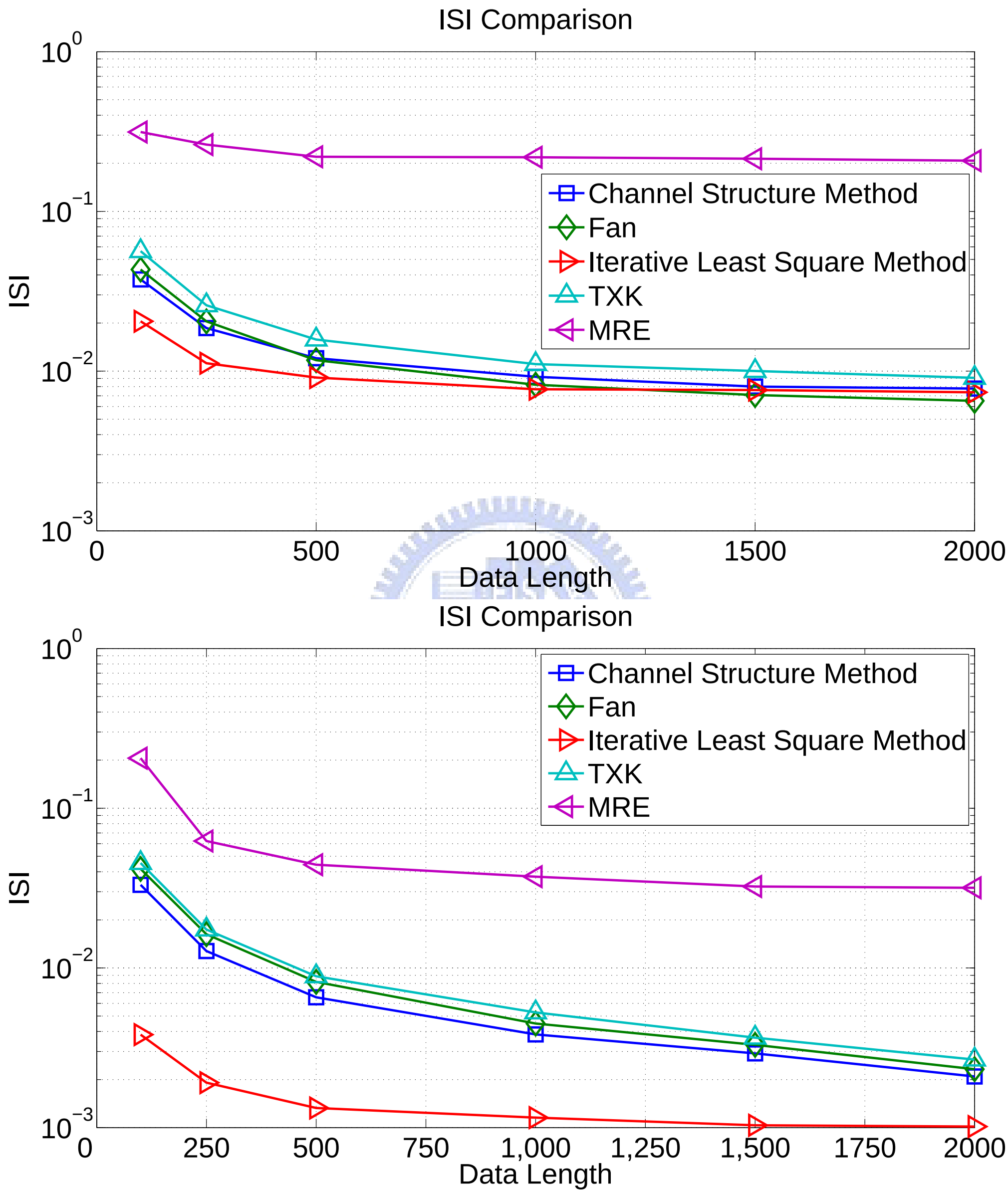

Fig. 5.6: Performance comparison versus different symbol number for ISI. ((a) 20dB and (b) 25dB)

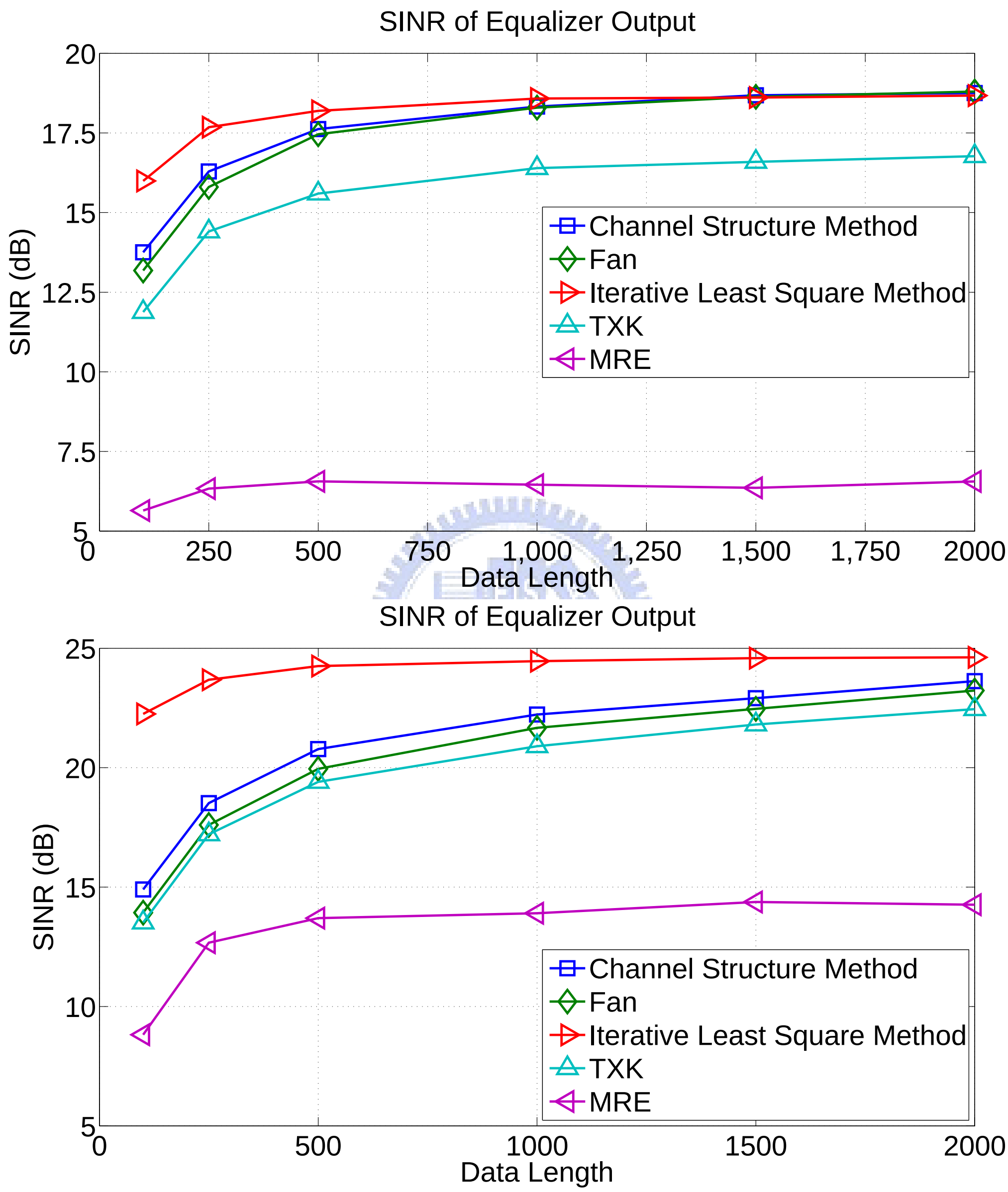

Fig. 5.7: Performance comparison versus different symbol number for SINR. ((a) 20dB and (b) 25dB)

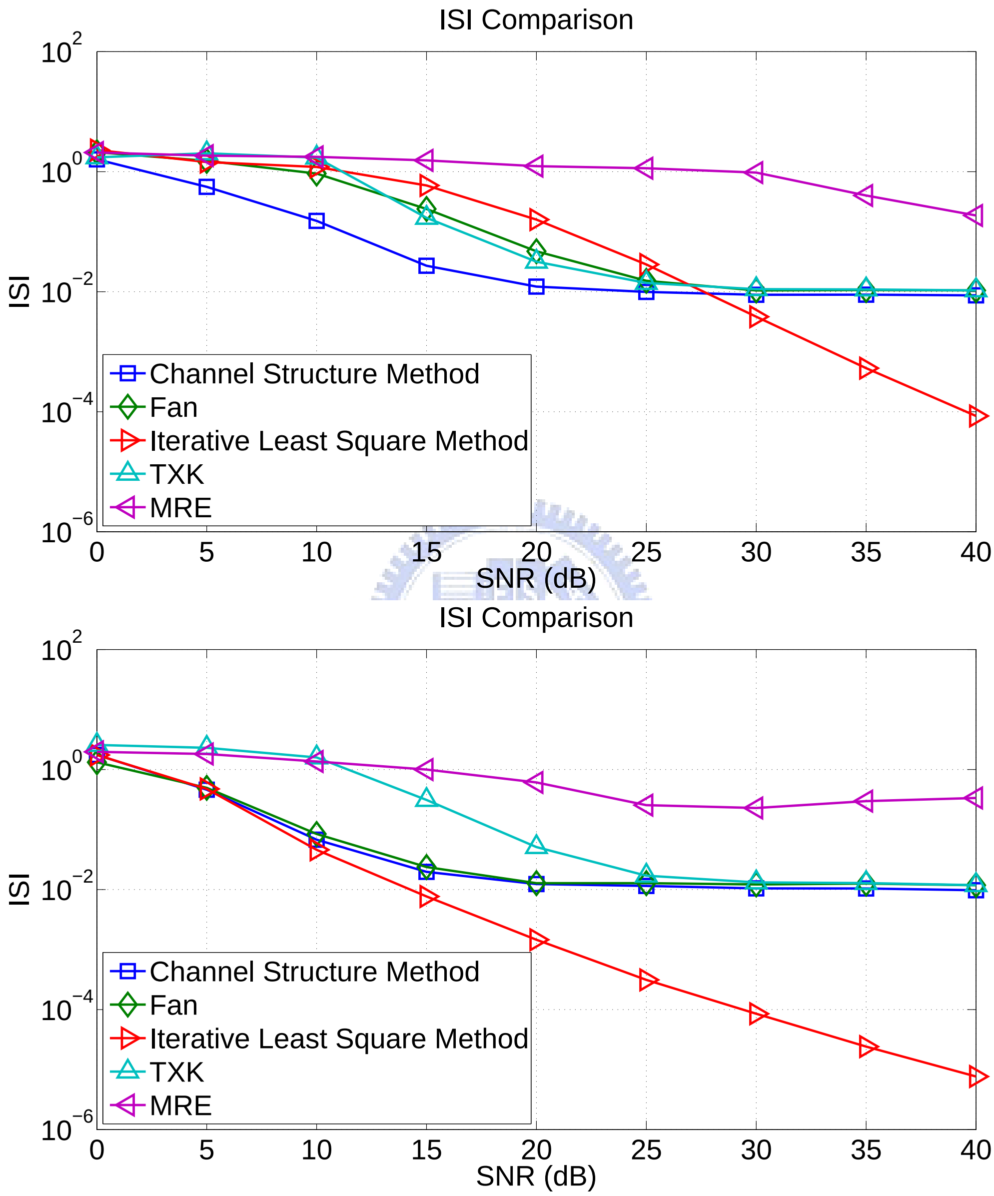

Fig. 5.8: Using Channel of TXK to performance comparison versus different SNR for ISI with 500 symbols. ((a) N=5 and (b) N=7)

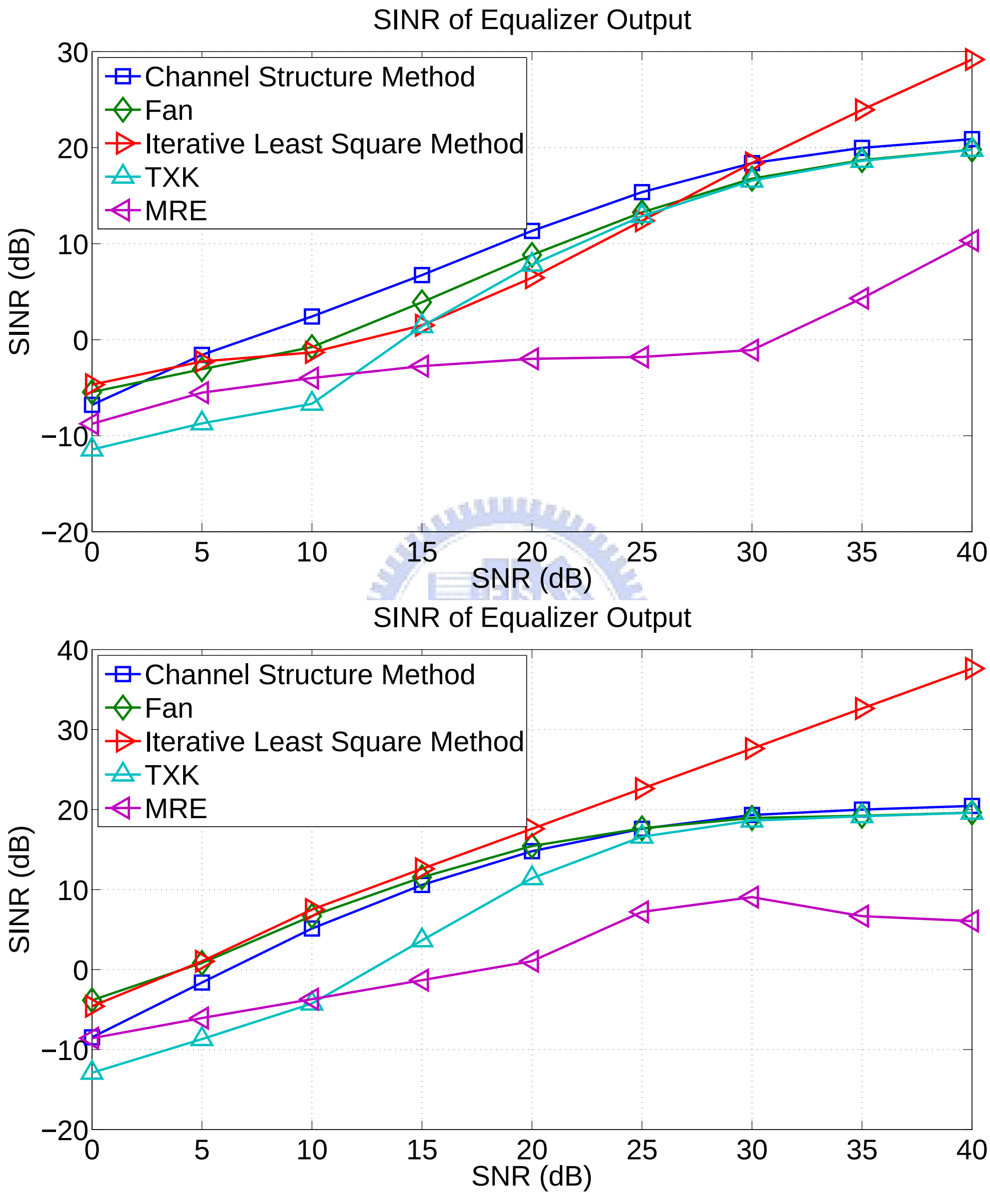

Fig. 5.9: Using Channel of TXK to performance comparison versus different SNR for SINR with 500 symbols. ((a) N=5 and (b) N=7)

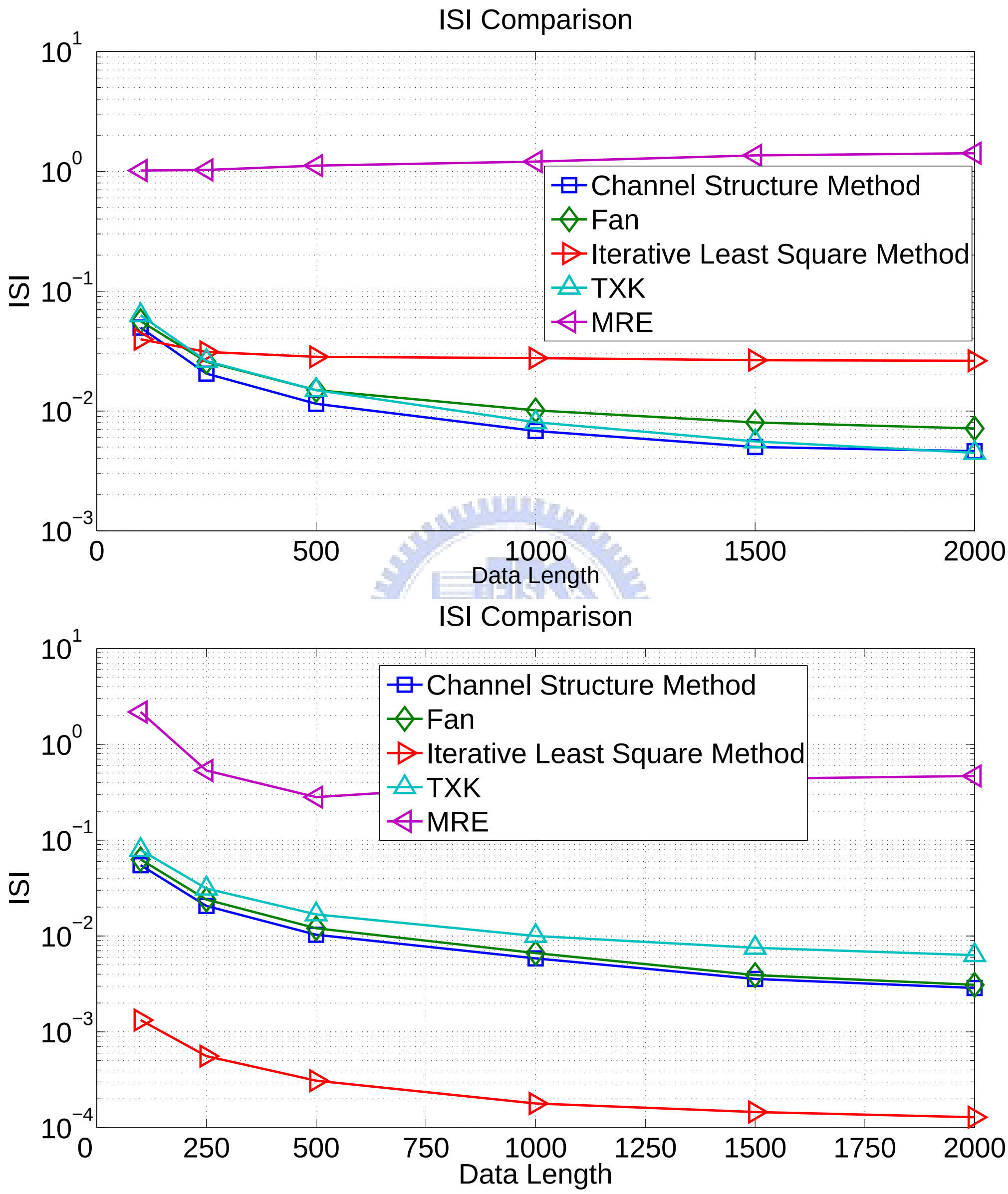

Fig. 5.10: Using Channel of TXK to performance comparison versus different symbol number for ISI with SNR=25dB. ((a) $N = 5$ and (b) $N = 7$)

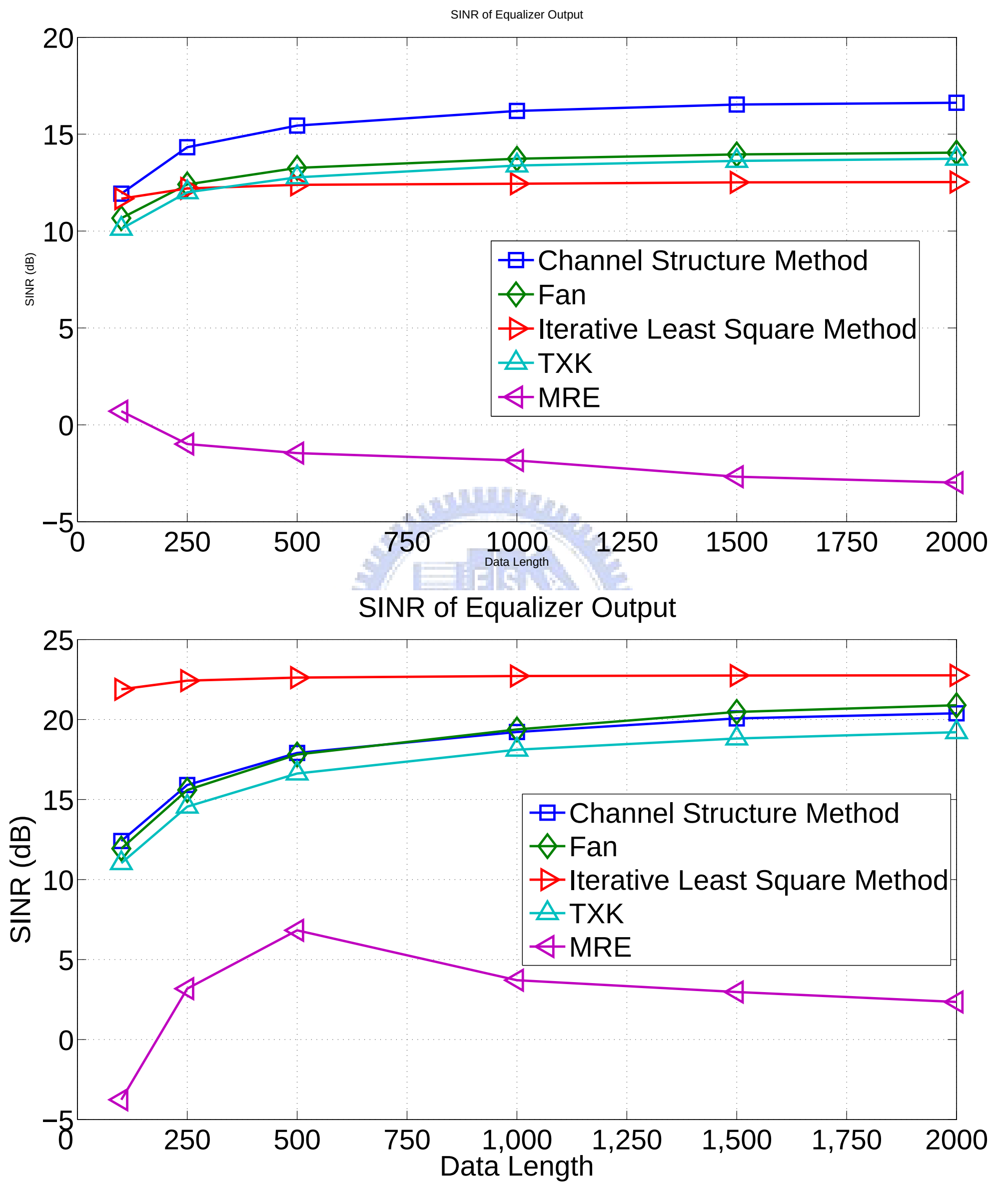

Fig. 5.11: Using Channel of TXK to performance comparison versus different symbol number for SINR with SNR=25dB. ((a) $N = 5$ and (b) $N = 7$)

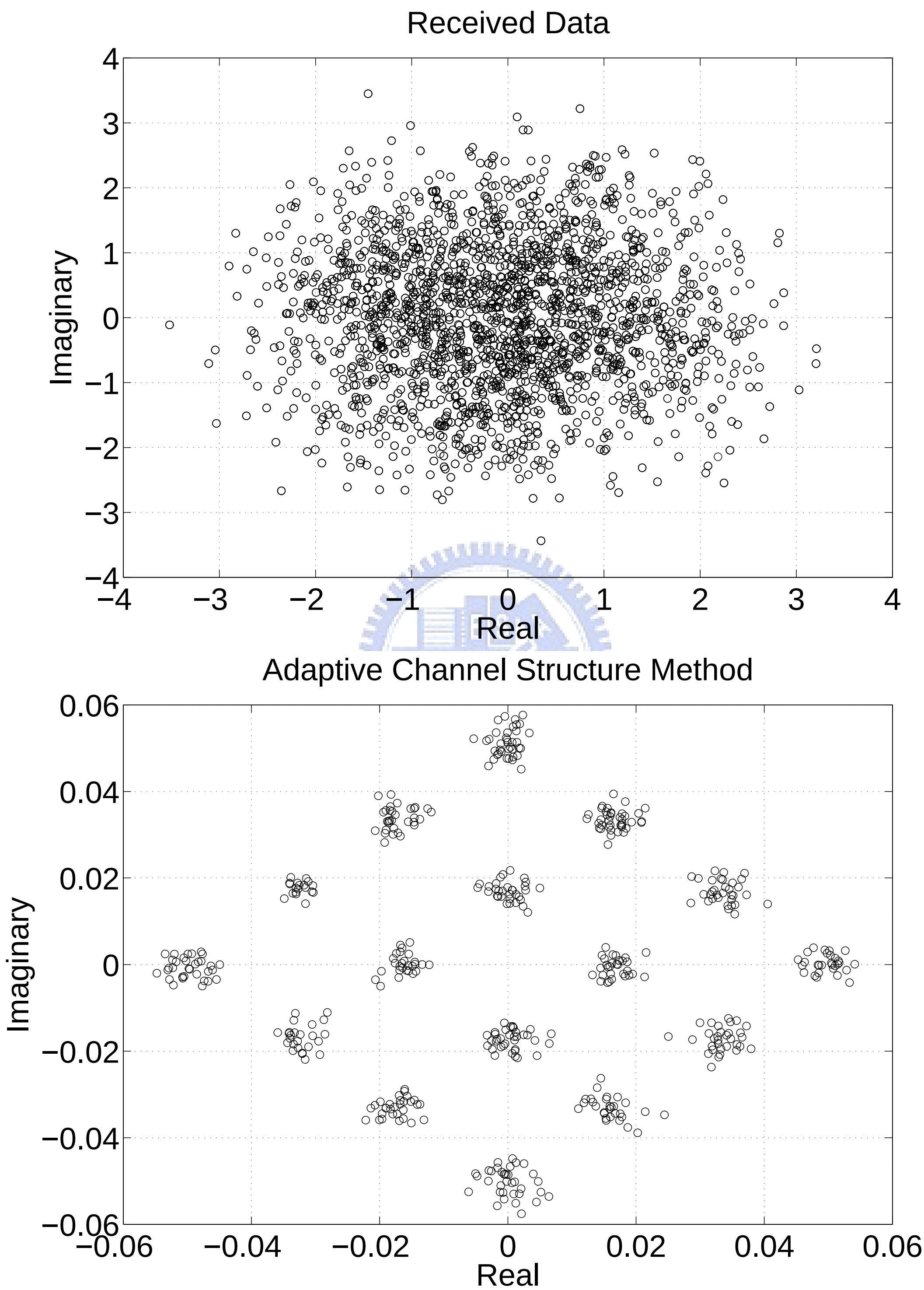

Fig. 5.12: Performance comparison versus different SNR for ISI and SINR with 20dB.

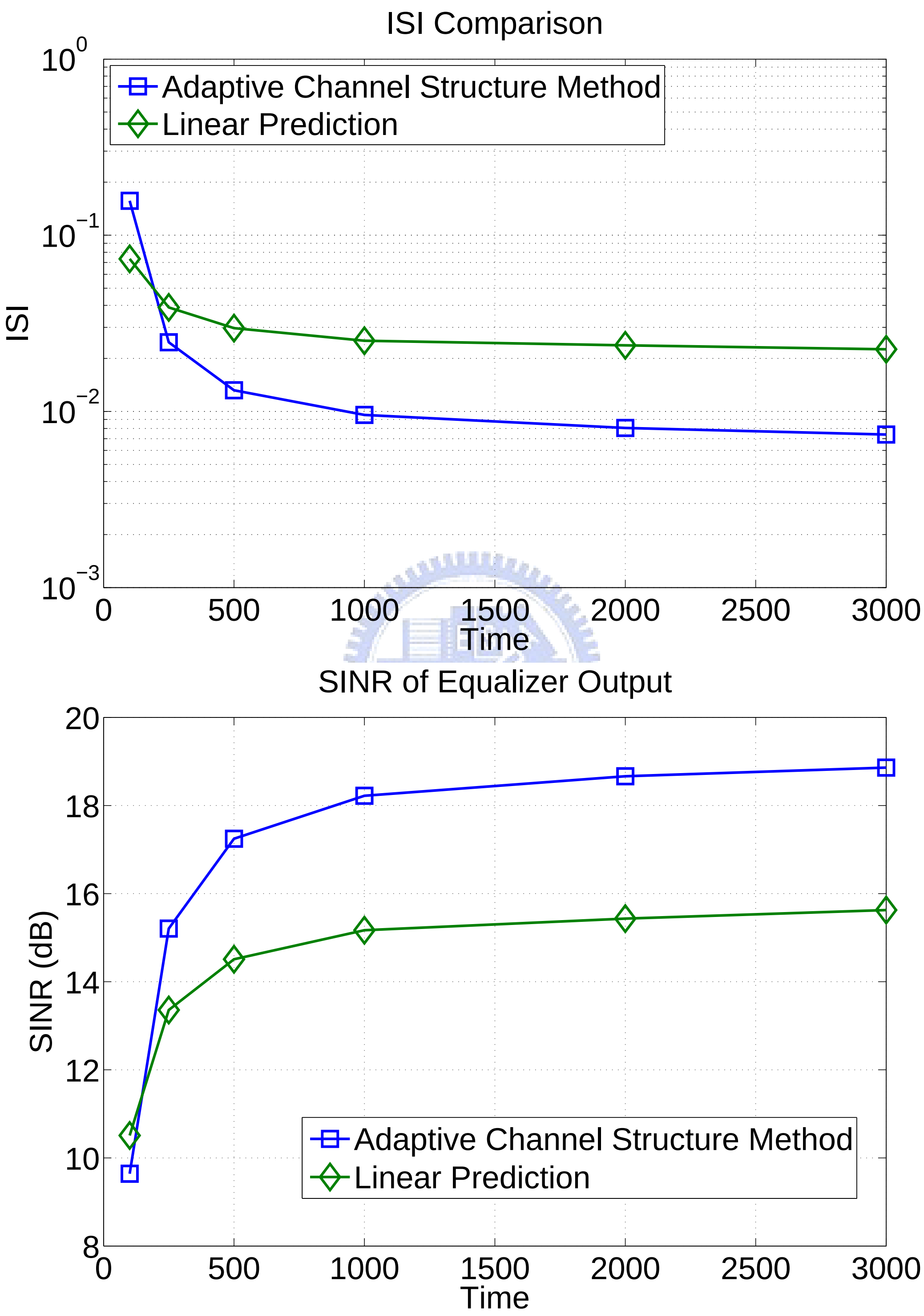

Fig. 5.13: Performance comparison versus different SNR for ISI and SINR with 20dB.

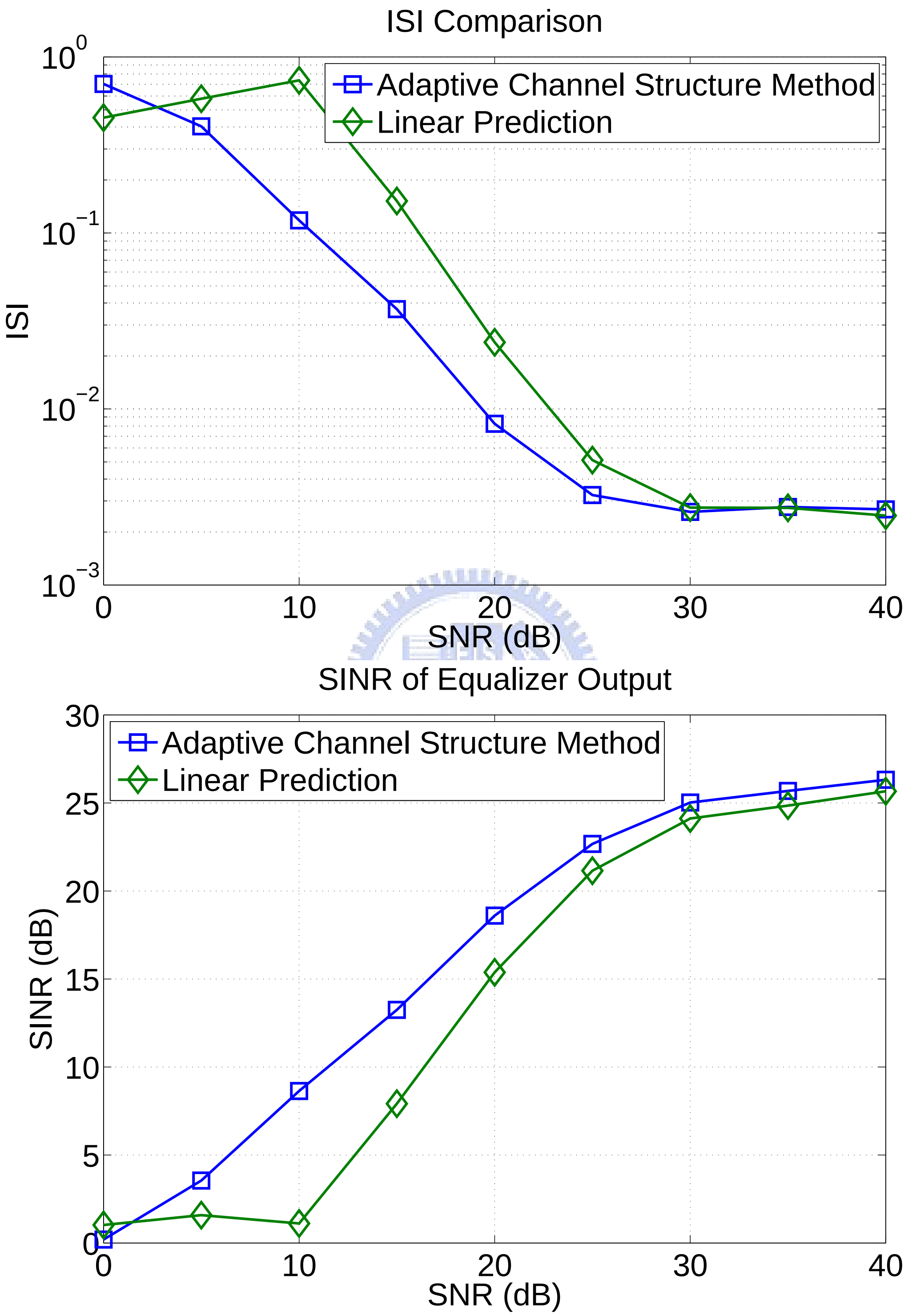

Fig. 5.14: Performance comparison versus different SNR for ISI and SINR with N=4 and 1500 symbol.

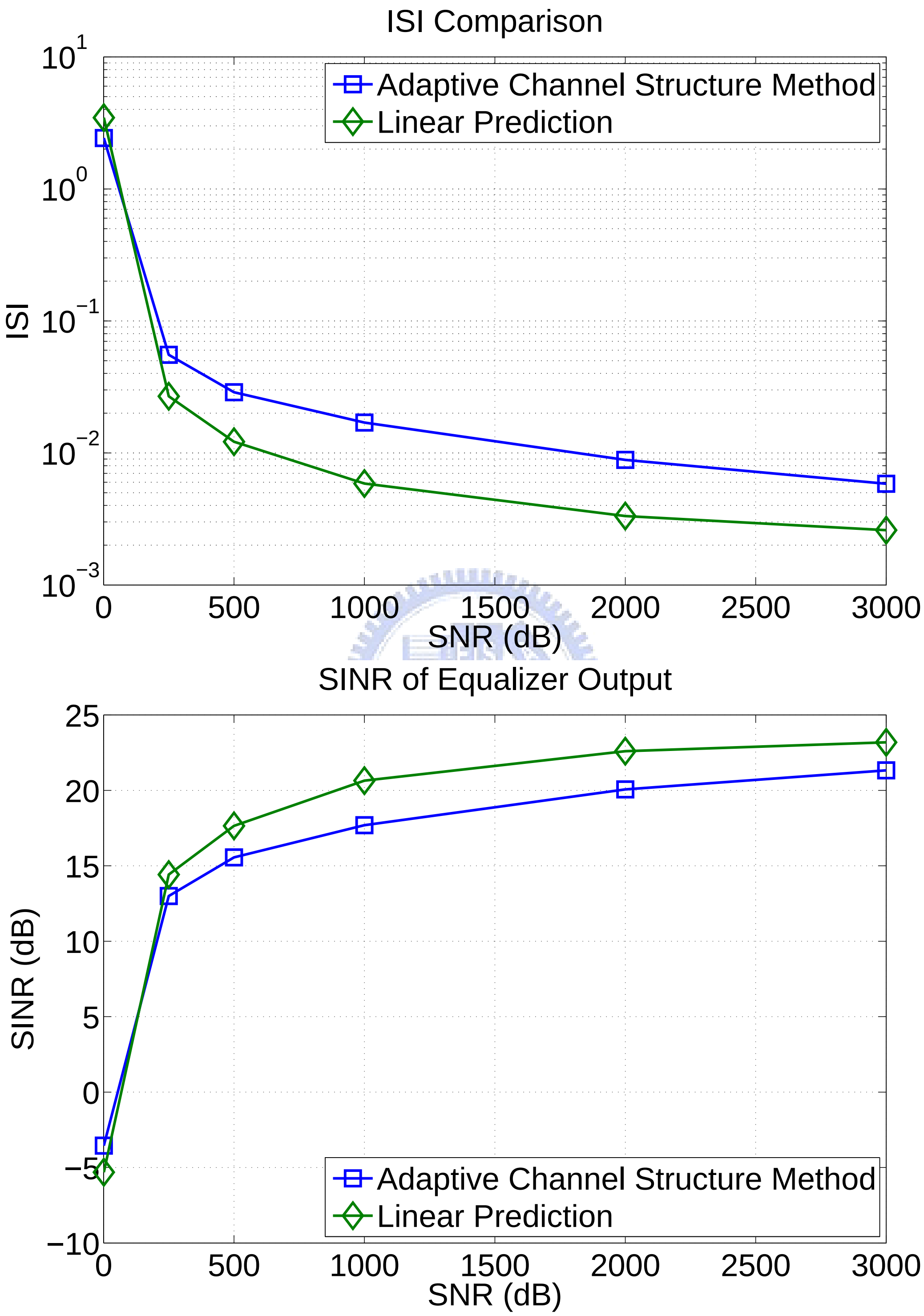

Fig. 5.15: Using channel of TXK to performance comparison versus different symbol number for ISI and SINR with N=7 and 30dB.

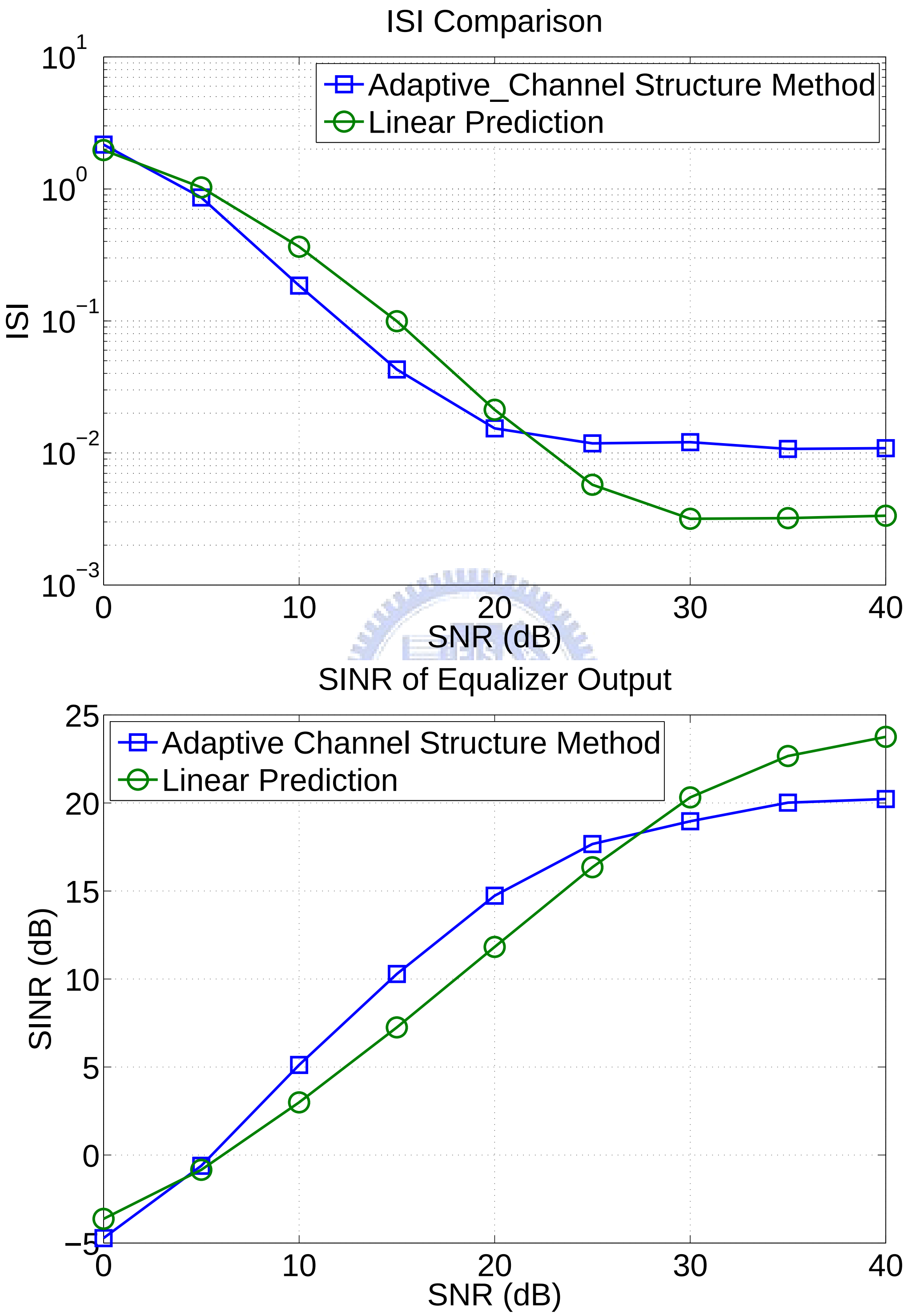

Fig. 5.16: Using channel of TXK to performance comparison versus different SNR for SINR with N=7 and 1500 symbols.

Chapter 6

Conclusion

This thesis presents two design algorithms for subspace blind equalization and an adaptive subspace blind equalizer. We use the property of Toeplitz structure of channel matrix and the concept of mutual-reference filters to develop a new blind equalizer and its adaptive algorithm. We also use the iterative least squares approach to design the blind equalizer. Simulations are also performed to demonstrate that our algorithms yield better performance with respect to the measures of ISI and SINR.

Bibliography

[1] X. Li and H. Fan, "Direct Blind Equalization with Best Delay by Channel Output Whitening," *IEEE Trans. Signal Processing*, Vol. 49, pp.1556-1563, July 2001.

[2] E. Moulines, P. Duhamel, J.F. Cardoso, and S. Mayrargue, "Subspace Methods for the Blind Identification of Multichannel FIR Filters," *IEEE Trans. Signal Processing*, Vol. 32, pp.516-525, Feb. 1995.

[3] L. Tong, G. Xu, and T. Kailath "Blind Identification and Equalization Based on Second-Order Statistic: A Time Domain Approach," *IEEE Trans. Information Theory*, Vol. 40, pp.340-349, March 1994.

[4] A.A. Farid, Z.Q. Luo, and Z.Ding "Blind Channel Equalization Based on Second Order Statistics," *IEEE Int. Conf. Acoust, Speech, and Signal Process*, vol. 3, pp. 557-560, March 2005.

[5] D. Gesbert, P. Duhamel, and S. Mayrargue "On-Line Blind Multichannel Equalization Based on Mutually Referenced Filters," *IEEE Trans. Signal Processing*, Vol. 45, pp. 2307-2317 Sept. 1997.

[6] X. Li and H. Fan, "Linear Prediction Methods for Blind Fractionally Spaced Equalization," *IEEE Trans. Signal Precessing*, vol. 48, pp. 1667-1675 , June 2000.

[7] B. Yang, "Projection Approximation Subspace Tracking," *IEEE Trans. Signal Precessing*, vol. 44, pp. 95-107, Jan. 1995.

[8] K.A. Meraim, A. Chkeif, and Y. Hua, "Fast Orthonormal PAST Algorithm," *IEEE Trans. Signal Precessing*, vol. 7, pp. 60-62, March 2000.

[9] P. Strobach, "Bi-Iteration Recursive Instrumental Variable Subspace Tracking and Adaptive Filtering," *IEEE Trans. Signal Precessing*, vol. 46, pp. 2708-2725, Oct. 1998.

[10] J. Yang, H. Xi, F. Yang, and Y. Zhao, "RLS-Based Adaptive Algorithms for Generalized Eigen-Decomposition," *IEEE Trans. Signal Precessing*, vol. 57, pp. 1177-1188 , April 2006.

[11] J. Fang, A.R. Leyman, and Y.H. Chew, "Blind Equalization of SIMO FIR Channels Driven by Colored Signals with Unknown Statistics," *IEEE Trans. Signal Precessing*, vol. 54, pp. 1998-2008 , June 2006.

[12] G.B. Giannakis and S.D. Halford, "Blind Fractionally Spaced Equalization of Noisy FIR Channels: Direct and Adaptive Solutions," *IEEE Trans. Signal Precessing*, vol. 45, pp. 2277-2292 , Sept. 1997.

[13] P. Balaban and J. Salz, "Optimum Diversity Combining and Equalization in Digital Data Transmission with Applications to Cellular Mobile Radio. I. Theoretical Considerations," *IEEE Trans. Comm.*, vol. 40, pp. 885-894 , May 1992.

[14] G.H. Golub and C.F. Van Loan, *Matrix Computations*, Baltimore, MD: Johns Hopkins Univ. Press, 1983.

[15] Z.ding and Y. Li, *Blind Equalization and Identification*, Marcel Dekker, Inc., 2001.