2005

Soft-decision equalization techniques for frequency selective MIMO channels

Shoumin Liu

Michigan Technological University

Soft-Decision Equalization Techniques for Frequency Selective MIMO Channels

by

Shoumin Liu
B.S. Harbin Institute of Technology, 1993
M.S. Harbin Institute of Technology, 1996
M.S. Michigan Technological University, 2003

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical and Computer Engineering
in the College of Engineering
at Michigan Technological University
Houghton, Michigan

Spring Term
2005

Advisor: Dr. Zhi Tian

Abstract

Multi-input multi-output (MIMO) technology is an emerging solution for high data rate wireless communications. We develop soft-decision based equalization techniques for frequency selective MIMO channels in the quest for low-complexity equalizers with BER performance competitive to that of ML sequence detection.

We first propose soft decision equalization (SDE), and demonstrate that decision feedback equalization (DFE) based on soft-decisions, expressed via the posterior probabilities associated with feedback symbols, is able to outperform hard-decision DFE, with a low computational cost that is polynomial in the number of symbols to be recovered, and linear in the signal constellation size. Building upon the probabilistic data association (PDA) multiuser detector, we present two new MIMO equalization solutions to handle the distinctive channel memory. With their low complexity, simple implementations, and impressive near-optimum performance offered by iterative soft-decision processing, the proposed SDE methods are attractive candidates to deliver efficient reception solutions to practical high-capacity MIMO systems.

Motivated by the need for low-complexity receiver processing, we further present an alternative low-complexity soft-decision equalization approach for frequency selective MIMO communication systems. With the help of iterative processing, two detection and estimation schemes based on second-order statistics are harmoniously

put together to yield a two-part receiver structure: local multiuser detection (MUD) using soft-decision Probabilistic Data Association (PDA) detection, and dynamic noise-interference tracking using Kalman filtering. The proposed Kalman-PDA detector performs local MUD within a sub-block of the received data instead of over the entire data set, to reduce the computational load. At the same time, all the interference affecting the local sub-block, including both multiple access and inter-symbol interference, is properly modeled as the state vector of a linear system, and dynamically tracked by Kalman filtering. Two types of Kalman filters are designed, both of which are able to track an finite impulse response (FIR) MIMO channel of any memory length. The overall algorithms enjoy low complexity that is only polynomial in the number of information-bearing bits to be detected, regardless of the data block size.

Furthermore, we introduce two optional performance-enhancing techniques: cross-layer automatic repeat request (ARQ) for uncoded systems and code-aided method for coded systems. We take Kalman-PDA as an example, and show via simulations that both techniques can render error performance that is better than Kalman-PDA alone and competitive to sphere decoding.

At last, we consider the case that channel state information (CSI) is not perfectly known to the receiver, and present an iterative channel estimation algorithm. Simulations show that the performance of SDE with channel estimation approaches that of SDE with perfect CSI.

Acknowledgments

I would like to express my sincere gratitude to my advisor, Dr. Zhi Tian. She spends tremendous time and energy advising my research and studies. Dr. Tian is a renowned expert in the field of signal processing and wireless communications. Thanks to her profound learning and experience, she is always able to offer me interesting and challenging research topics. I have gained a great many skills by conducting research under her supervision.

I would also thank Dr. Jeffrey B. Burl, Dr. Seyed Zekavat, Dr. Seungjin Park, and Dr. Clark Givens for their interest in my work, their precious time reading my dissertation, and their willingness to be my examination committee members.

My thanks go to my fellow graduate students, Ming Ling, Yanfeng Gong, Lin Wu and Xianren Wu. I greatly appreciate their help during the past a few years. I enjoyed the discussions with them and benefited from their suggestions and comments on my work.

Finally, I would like to thank my parents and my wife for their love, care, and unending support. It is their encouragement that helps me keep overcoming challenges and succeed.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Research Goals

Wireless communications have brought great convenience and advances to all walks of life. Despite the progresses that have been made, researchers are working hard to further improve the data rate, capacity and quality of service that wireless systems can offer. Among many promising solutions, multiple-input-multiple-output (MIMO) systems emerge as an attractive approach. It has been shown that enormous increase in bandwidth efficiency is promised by the use of MIMO systems in wireless radio frequency links [1], [5], [6]. MIMO systems also enjoy high array gains and spatial diversity gains rendered by multiple receive antennas [8], [12]. Transmissions at high data rates typically encounter frequency selective channels, where channel-induced inter-symbol interference (ISI) can be mitigated using high-speed, low-cost equalization techniques such as decision feedback equalization (DFE) [2]. However, equalization in MIMO systems is remarkably challenging because of the need for signal detection in the presence of both multiple access interference (MAI) and inter-symbol interference (ISI) [11], [31], [32].

In ISI environments, the BCJR *a posteriori* (MAP) [28] and Viterbi equalizers perform optimum sequence detection to account for the channel memory, but incur prohibitive complexity that is exponential in the number of inputs and the channel memory. Their equalization complexity can be reduced using standard reduced-complexity methods for single-input channels (e.g., [29], [30]), but such methods do not apply to MIMO channels. For complexity considerations, typical equalizers consist of linear processing of the received signal, i.e., linear equalization (LE), and possibly past symbol estimation, e.g., decision feedback equalization (DFE). The optimum MIMO DFE settings in the minimum mean-square error (MMSE) sense have been derived in [30], [31], [32], [33]. In these schemes, tentative decisions on both past symbols and symbols from MAI sources are made by quantizing properly derived decision statistics.

However, the approaches in [30], [31], [32], [33] are hard decision based approaches. Such *hard-decision* based feedback approaches may suffer from catastrophic error propagation, which in most cases incurs non-trivial performance degradation relative to an optimal maximum likelihood (ML) detector, in terms of the bit error rate (BER) performance. Recent advances include turbo detection and equalization [34] – [36], in which low-complexity soft-input-soft-output (SISO) LE and DFE equalizers are devised based on the MMSE criterion [35].

In this dissertation, we focus on rate-oriented MIMO systems in which no coding is used at the transmitter. We assume that the receiver has perfect channel state information (CSI). Targeting low-complexity equalization of finite impulse response

(FIR) MIMO channels, we develop a set of soft-decision based approaches, in order to overcome the performance-degrading drawback of hard-decision feedback equalization. Our algorithms include soft-decision equalization (SDE), Kalman Probabilistic Data Association (Kalman-PDA) approach, and soft-decision oriented performance enhancement techniques. Firstly, this dissertation presents two SDE algorithms. The first SDE algorithm adopts a zero-padded transmission structure to convert the challenging sequence detection problem into a block-by-block least-square formulation. The second SDE algorithm takes advantage of the Toeplitz channel matrix structure embodied in an equalization problem. It processes the data samples through a series of overlapping sliding windows to reduce complexity, and at the same time performs implicit noise tracking to maintain near-optimum performance. Secondly, this dissertation presents a Kalman-PDA approach, which dissects the equalization problem for MIMO channels into two parts: local multi-user detection (MUD) using probabilistic data association (PDA), and noise-interference tracking using Kalman-filtering. Lastly, this dissertation discusses several performance enhancement strategies and joint channel estimation and equalization.

Compared with hard-decision driven approaches, the proposed soft-decision based algorithms have prominent advantages:

1. The overall complexity is reduced by invoking the Gaussian-forcing approximation. Different from the hard-decision multistage parallel interference cancellation (PIC), the proposed SDE approach generates tentative decisions on ISI and MAI symbols in the form of posterior probabilities instead of quantized bits,

and the decision updating is simplified by forcing the composite effect of noise and interference to be Gaussian, a strategy used in the PDA detector. The SDE detection performance improves with additional iterations, and stabilizes fast in 3-5 iterations for high signal-to-noise ratio (SNR), and 7-14 iterations for low SNR.

2. Near-optimal detection performance is achieved via soft-decision based signal processing. Compared with the sub-optimum hard-decision MMSE-DFE [31], the SDE method demonstrates close-to-optimal equalization performance, at a comparable low complexity. Compared with other quasi-ML methods such as sphere decoding [21], the SDE is not only competitive in both performance and complexity, but also applies to situations where sphere decoding does not work well, such as the fat channel case in which there are more transmit antennas than receive antennas.

3. As can be seen in Chapter 7, soft-decision approaches have the unique capability of sensing error-prone bits on line, which facilitates joint transmitter-receiver design to detect and correct unreliable transmissions.

The theme of this dissertation is to provide a set of soft-decision equalization algorithms that can fully exploit the bandwidth and capacity of MIMO systems. We also show the superiority of soft-decision based equalization algorithms compared with hard-decision based approaches. In the next section, we will lay out the organization of this dissertation.

1.2 Organization

The organization of the rest of this dissertation is as follows. Chapter 2 first briefly describes the background of MIMO wireless communication systems. The signal model is then established. Existing symbol detection and equalization algorithms based on this model are introduced. Pros and cons of these algorithms are discussed.

In Chapter 3 and 4, we present two soft-decision equalization (SDE) algorithms based on the signal model established in Chapter 2, SDE-1 and SDE-2, respectively. The first one is developed by way of PDA enhancement. The second one has a novel sliding windowing structure that is tailored for the Toeplitz zero-padding channel matrix, which is rendered by the signal model. Both of the two algorithms can achieve near-optimum performance when doing symbol detection and equalization in frequency selective MIMO channels.

In Chapter 5, we present simulation results and performance analysis of SDE algorithms. We first simulate SDE in full-rank MIMO channels. The performance of both SDE-1 and SDE-2 approaches that of the optimum ML detection. We then investigate the performance of SDE in various rank-deficient channels. Rank-deficient channels exist in many wireless systems, such as in the downlink of mobile transmission where there are typically more transmit than receive antennas. Many detection techniques either can not be applied to rank-deficient channels, or exhibit a large error floor. We observe that SDE works for rank-deficient channels, especially for BPSK-modulated systems.

SDE algorithms presented in Chapter 3 and 4 perform multiuser detection and equalization over all or multiple symbol periods. Motivated by the need for low-complexity receiver processing, we proposed in Chapter 6 a MIMO equalization approach that performs MUD within a sub-block of the symbol vector (local MUD), instead of doing it for the entire block (global MUD). The MIMO detection and equalization problem is partitioned into two parts: local MUD using PDA and noise-interference tracking using Kalman filtering. Simulations show that satisfactory performance is achieved compared with sphere decoding and MIMO decorrelator.

In Chapter 7, we present two types of performance enhancement techniques for soft-decision approaches. These two techniques are based on the observation that soft-decisions have the capability of indicating their own reliability. Taking advantage of this unique property, we devise two types of performance techniques, ARQ-aided approach for uncoded systems and coding-aided approach for coded systems. We will take Kalman-PDA as an example, and show that both enhancement techniques can noticeably enhance the detection performance of Kalman-PDA itself.

In Chapter 8, we study a realistic scenario when perfect channel knowledge is not available. We assume that the receiver can obtain a rough initial estimation of an unknown channel by using training sequences. For each information-bearing symbol block, the receiver performs symbol detection and channel equalization based on this coarse initial channel estimate. The symbol decisions are fed to recursive least square (RLS) based channel estimator to generate a refined channel estimate. This refined channel estimate, in turn, is used for a new round symbol detection. The iteration

continues until convergence is reached. Simulations show that there is only a narrow gap between the performance of SDE with perfect channel state information (CSI) and SDE with estimated CSI.

Chapter 9 concludes the dissertation and gives some directions for future work. A list of papers published during the course of this dissertation research is given in the appendix.

CHAPTER 2

Background and Signal Model

In this chapter, we first briefly introduce the background knowledge of multi-input multi-output systems. Then we establish the signal model of a generic frequency-selective MIMO channel. Our SDE and Kalman-PDA algorithms are both based on this model. In the third part of the chapter, we review some existing approaches to multiuser detection and channel equalization problems given this model, such as sphere decoding (SD), semi-definite relaxation programming (SDP), and probabilistic data association (PDA). Comparisons between these algorithms are provided.

2.1 Background of MIMO Systems

High data rate wireless communications is a rapid-growing business since it provide the backbone for wireless local area networks (WLAN) and home audio/video (AV) networks. Although IEEE 802.11g standard supports 54Mb/s data rate, people are looking for 1Gb/s data rate for future high-speed high-definition television (HDTV) data streams [3]. Practically speaking, a standard single-input single-output (SISO) wireless link can not be used to render such a high data rate. MIMO systems, on the

other hand, are an exciting solution for making the high data rate WLAN reality. The superior performance delivered by MIMO systems results from array gain, diversity, spatial multiplexing gain and interference reduction [3].

Array gain is achieved by using multiple antennas and the received signals add up coherently. Signal processing techniques are used at the transmitter or the receiver to increase the average SNR due to the coherent combining effect. Transmit or receive array gain requires knowledge of channel at the transmitter and the receiver respectively. The receiver can obtain channel information through processing training sequences, while the transmitter is supplied channel information in way of feedback. In general, channel knowledge at the receiver is relatively easier to obtain.

The signal power fades randomly in wireless communications. Diversity helps to mitigate the effect of fading by combining the arriving signals. MIMO systems provide spatial diversity which does not cost extra transmission time or bandwidth. Taking advantage of spatial diversity gains, the output signal demonstrates much less power variability compared with a SISO wireless link. Ideally, the wireless links in a MIMO channel are independent from each other, and the maximum order of spatial diversity is achieved. For a MIMO system with N_i inputs and N_o outputs, the maximum possible spatial diversity order is $N_i N_o$.

It has been shown the capacity of a MIMO system increases linearly in $\min(N_i, N_o)$ without invoking additional power or bandwidth cost [4], [5], [6], [7]. This gain in capacity referred to as spatial multiplexing gain. It is realized by transmitting independent data streams from different antennas. Under the rich scattering channel

condition, the receiver is able to separate the different data signals. Thus an increase in channel capacity is obtained.

Cochannel interference (CCI) occurs when the frequency is reused in wireless channels. In MIMO systems, the difference between the spatial signature of the desired signal and that of the interfering signals can be exploited to reduce interference. Interference reduction requires the desired user's channel information, while the interferers' channel knowledge may be unnecessary. Interference reduction can also be performed at the transmitter side. The transmitter can try to minimize the interference energy when sending signals to the desired user. Overall, interference reduction promised better frequency reuse and thus increase system capacity.

In practice, a key idea of MIMO systems is space-time signal processing in which the *time dimension* is complemented with the *spatial dimension* resulted from the use of multiple antennas. Such MIMO systems can be seen as an extension of the technique of smart antennas, which has been around for years [9]. For frequency-selective MIMO channels, Orthogonal Frequency Division Multiplexing (OFDM) can be used to deal with frequency selectivity. With the help of this MIMO-OFDM, a frequency-selective channel can be converted into a set of parallel frequency-flat fading channels. Each flat fading channel has a bandwidth that is a fraction of that of the original frequency-selective channel. MIMO-OFDM presents itself an attractive approach that avoids temporal equalization at the cost of a minor decrease in channel capacity [3], [10]. In the next section, we will establish the signal model of

a baseband MIMO system. We will develop the soft-decision based algorithms based on this model.

2.2 Signal Model

We consider the discrete-time block transmission equivalent model of a baseband communication system with N_i inputs and N_o outputs. A generic diagram of such a system is shown in Fig. 2.1. There are a total of N_iN_o links in this MIMO system, wherein the link between each input-output pair is modeled as a linear finite impulse response (FIR) dispersive channel with no greater than $L+1$ symbol-spaced taps in the channel response. The sampled channel response from the ith input to the jth output, including transmit and receive filters, is denoted by $\mathbf{h}^{(i,j)} := [h^{(i,j)}(0), \cdots, h^{(i,j)}(L)]^T$. This channel response vector subsumes all the channel information from the ith input to the jth output. Therefore, in the discussions hereafter, we develop soft-decision-based equalization algorithms based on the perfect or partial knowledge of the channel vectors. We adopt a block transmission structure with zero padding to eliminate inter-block interference. This structure helps to alleviate the performance degradation due to noise enhancement or error propagation [23]. Per the zero-padding structure, the information-bearing symbols at each link are parsed into N-long blocks, with the insertion of $P \geq L$ zeros at the end of each block. The input vector corresponding to the kth block of all inputs is denoted as $\mathbf{b}_k := [\mathbf{b}^T(kN), \cdots, \mathbf{b}^T(kN+N-1)]^T$. It consists of N sub-blocks, where the mth sub-block $\mathbf{b}(m) := [b^{(1)}(m), \cdots, b^{(N_i)}(m)]^T$

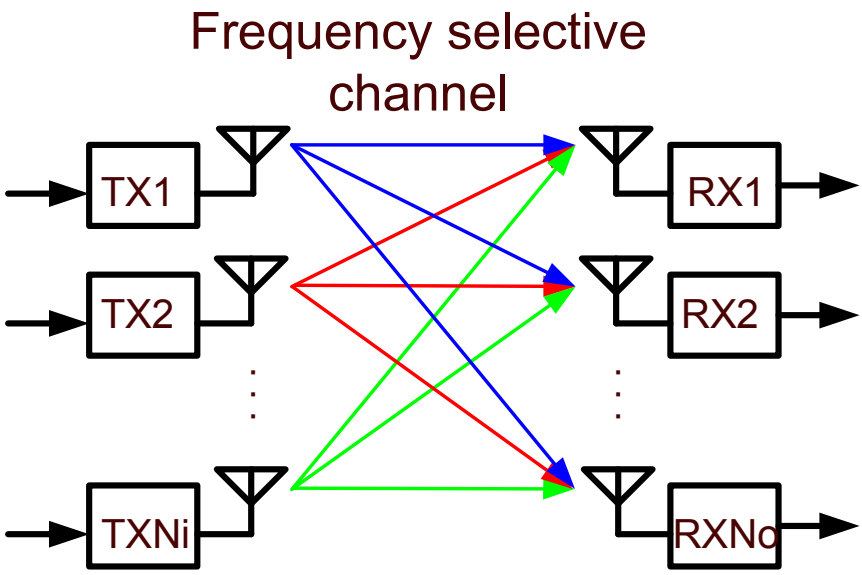

Figure 2.1: A generic MIMO system diagram

contains the information symbols from all N_i inputs at the mth sampling instant. Here, N_i is the number of data streams transmitted from the transmitter, and N is the number of symbols in each data stream. After padding PN_i trailing zeros, each N-long information-bearing symbol block $\mathbf{b}_k$ forms a transmit symbol block $\bar{\mathbf{b}}_k$ of block size $K := N + P$, where the first N entries convey messages, $\{\bar{b}^{(i)}(kK + n)\}_{n=0}^{N-1} = b^{(i)}(kN + n)$, followed by P trailing zeros $\{\bar{b}^{(i)}(kK + n)\}_{n=N}^{K-1} = 0$, for any block index k and input $i \in [1, N_i]$. Correspondingly, the received data vector at the kth block is a concatenation of K noise-contaminated sample vectors, $\mathbf{y}_k := [\mathbf{y}^T(kK), \cdots, \mathbf{y}^T(kK + K - 1)]^T$, where $\mathbf{y}(m) := [y^{(1)}(m), \cdots, y^{(N_o)}(m)]^T$ consists of the mth received signals at all N_o outputs. The redundancy per transmitted block is measured by the ratio P/K, while at the receiver the data rate is reduced by the same amount. We set $P = L$, and typically choose a large block size K (and N) to maintain the transmission rate.

In linear FIR channels, the received $y^{(j)}(kK+n)$ at the jth output is expressed by

$$y^{(j)}(kK+n) = \sum_{i=1}^{N_i} \sum_{l=0}^{L} h^{(i,j)}(l)\bar{b}^{(i)}(kK+n-l)$$
$$+v^{(j)}(kK+n), n = 0, \cdots, K-1; j = 1, \cdots, N_o, \tag{2.1}$$

where $v^{(j)}(kK+n)$ denotes the additive zero-mean Gaussian stationary noise received at the jth output. The noise terms at the kth transmission block are grouped into a $N_oK \times 1$ vector $\mathbf{v}_k := [\mathbf{v}^T(kK), \cdots, \mathbf{v}^T(kK+K-1)]^T$, where $\mathbf{v}(m) := [v^{(1)}(m), \cdots, v^{(N_o)}(m)]^T$, and the covariance of $\mathbf{v}_k$ is denoted by $\mathbf{R}_v := E\{\mathbf{v}_k\mathbf{v}_k^H\}$. With these definitions, the single-link input-output model (2.1), when assembled into the vector-matrix format, results in a MIMO channel model in the form of $\mathbf{y}_k = \mathbf{H}\mathbf{b}_k + \mathbf{v}_k$, where the MIMO channel matrix takes on a banded Toeplitz structure [23]:

$$\mathbf{H} := \begin{bmatrix} \mathbf{H}(0) & & \mathbf{0} \\ \vdots & \ddots & \\ \mathbf{H}(L) & & \mathbf{H}(0) \\ & \ddots & \vdots \\ \mathbf{0} & & \mathbf{H}(L) \end{bmatrix}, \tag{2.2}$$

$$\text{where } \mathbf{H}(l) := \begin{bmatrix} h^{(1,1)}(l) & \cdots & h^{(N_i,1)}(l) \\ \vdots & \ddots & \vdots \\ h^{(1,N_o)}(l) & \cdots & h^{(N_i,N_o)}(l) \end{bmatrix}, \tag{2.3}$$
$$l = 0, \cdots, L.$$

With this banded Toeplitz structure, the inter-block interference is eliminated at the cost of transmission redundancy. Sequence detection incurred by the channel memory can now be alternatively solved by block-by-block symbol detection, for which we can focus on symbol detection within one single block. From now on, we will drop the subscript k without raising confusion. This system model subsumes a broad range of MIMO transmission scenarios. The multiple input could result from the combination of three situations:

1. Multiple transmit antennas with a single user, e.g., the single-user space-time coding case.

2. Single transmit antenna with multiple users, e.g., as encountered in the single-channel multiuser detection problem.

3. Transmit-induced diversities including orthogonal frequency devision multiplexing (OFDM) and CDMA.

Thus, the number of inputs N_i is determined by both the number of transmit antennas and the number of multiple-access users. Meanwhile, multi-output is invoked when there are multiple receive antennas, and/or when fractional sampling is used. Thus, N_o is determined by the total number of distinct samplers operating within a symbol period.

With this MIMO model, MIMO channel equalization can be viewed as a general high-dimensional integer least-square (LS) problem in the form of $\mathbf{y} = \mathbf{Hb} + \mathbf{v}$, where $\mathbf{b}$ is an $NN_i \times 1$ input vector that takes on finite-alphabet vales. The channel matrix

is of size $\bar{K} \times \bar{N}$, where $\bar{K} := N_o(N + L)$ and $\bar{N} := N_i N$. The optimal ML solution to the MIMO system described above faces a major implementation challenge, as its complexity increases exponentially in N_i, N_o, N, and L. In this dissertation, we propose a set of soft-decision-based symbol detection and channel equalization schemes that achieve near-optimal BER performance at low polynomial complexity. In Chapter 3, 4 and 6, we develop soft-decision equalization and Kalman-PDA algorithms assuming that the receiver has perfect knowledge of channel information. In Chapter 8, we consider a more realistic scenario in which the receiver obtain a coarse, initial channel knowledge with the help of training sequences. Iterative channel estimation and symbol detection are combined together, and near-optimal performance is achieved. In the next section, we review the existing approaches to this MIMO equalization problem. Pros and cons of each approach are summarized.

2.3 Existing Approaches

Per the block transmission structure with zero-padding, which is established in the last section, we can express an L-tap FIR MIMO channel with N_i transmit antennas and N_o receive antennas by a $KN_o \times NN_i$ block matrix, $\mathbf{H}$, where $K = N + L$ is the number of output symbols after padding L trailing zeros in each input block of size N. The general input-output MIMO model is given by

$$\mathbf{y} = \mathbf{H}\mathbf{b} + \mathbf{v}, \tag{2.4}$$

where $\mathbf{y}$ and $\mathbf{v}$ are the output vector and the noise vector respectively, and $\mathbf{b}$ is an unknown $NN_i \times 1$ block input vector made of N sub-blocks, each consisting of the symbols transmitted by N_i transmit antennas at every sampling instant. The optimal maximum likelihood (ML) solution to $\mathbf{b}$ in (2.4) incurs high computational complexity that is exponential either in the block size N for block detection, or in the channel length L for sequence detection via the Viterbi algorithm. Specifically, when M-ary modulation is used for $\mathbf{b}$, the complexity of a brute-force ML detector is on the order of $\mathcal{O}(M^{NN_i})$ per block of NN_i symbols, while that of the Viterbi algorithm is $\mathcal{O}(NN_iM^{LN_i})$, which is still computationally ineffective when the channel memory length is long.

To avoid such practically infeasible complexity, a number of near-optimal algorithms have been proposed at polynomial-time complexity. In this section, we give a review of the existing solutions, including sphere decoding (SD) [21], semi-definite relaxation programming (SDP) [14], and probabilistic data association (PDA) [26]. The pros and cons of each algorithm are summarized. We also compare them with the proposed SDE, Kalman-PDA algorithms.

2.3.1 Sphere Decoding

The sphere decoding algorithm was first presented in [38] to find lattice vectors of short length. The complexity of sphere decoding was found to be polynomial in the dimension of the lattice [21]. Sphere decoding made its debut in communications

applications in [39]. For the years followed, it has been proposed as a near-optimum method for multiuser detection problems in code-division multiple access (CDMA) systems [41]. It has also been applied to space-time coded as well as uncoded multiple antenna systems [42].

Given the input-output equation,

$$\mathbf{y} = \mathbf{Hb} + \mathbf{v}, \tag{2.5}$$

sphere decoding requires minimization of the metric $||\mathbf{y} - \mathbf{Hb}||^2$ over all valid $\mathbf{b}$, i.e., all valid lattice points. The search for the optimum solution is restricted to the lattice points within the sphere of radius $\sqrt{C}$ that is centered at $\mathbf{y}$. Each time a valid point is found, the radius is shrunk so that the newly found point lies on the surface of the sphere. This process continues until the last point within the sphere is reached, and this last point is the optimum solution to (2.5), [43]. In other words, sphere decoding is an exhaustive search algorithm with early pruning of search subtrees that have been identified as impossible to contain an optimal solution by comparing with the present best solution [44]. Although this approach may seem to be straightforward, the implementation is sophisticated and ingenious. For a detailed description of the algorithm, readers are referred to [40].

There exist several improved sphere decoding algorithms based on the original one. One approach is to use near-orthogonal basis reduction as a preprocessing step, and another method is to essentially choose the most promising branch to check first [45].

One of the disadvantages of sphere decoding is that its complexity and performance are highly dependent on the choice of the initial radius of the sphere. This makes sense intuitively: if the initial radius is very large, the searching process requires huge computations; on the other hand, if the initial radius is small, the algorithm may fail to find a valid lattice point within the sphere. In [43], an enhancement technique was proposed to mitigate this problem. The search process was modified so that the computational load is insensitive to the initial search radius. Thus, the initial search radius can be sufficiently large to ensure the sphere encompasses the optimum solution (lattice point). However, sphere decoding still has another limitation when it is applied to "fat" channels, which translates to the case that there are more columns than rows in the channel matrix $\mathbf{H}$. This is because the original SD judiciously uses the lattice structure of the finite-alphabet input data to perform quasi-ML search at a low complexity. Unfortunately, such a lattice search is infeasible for $\bar{N} > \bar{K}$ [21], and the generalized SD (GSD) does not preserve optimality due to a reduced-dimension lattice projection [55]. The soft-decision equalization (SDE) algorithms proposed in this work, on the contrary, can be applied to fat channels. We will give detailed simulation results and discussions about the performance of SDE in fat channels in Chapter 5.

2.3.2 Semi-Definite Relaxation Programming

Although the maximum-likelihood (ML) detector is well known as the optimum solution for the problem slated in (2.5), ML detection (MLD) is considered as a non-deterministic polynomial-time hard (NP-hard) problem. The semi-definite relaxation algorithm was presented in [14] as a quasi-maximum-likelihood multiuser detection scheme. It is a sub-optimal, low-complexity substitute to the ML detector. Compared with multistage detection [15], the coordinate ascent algorithm [16], and the expectation-maximization (EM) approach [17], the semi-definite relaxation algorithm does not utilize the interference cancellation approach. It avoids the error-propagation that most interference cancellation algorithms suffer when the previous symbols are incorrectly detected.

Relaxation is an effective approximation technique for some complicated optimization problems. The basic idea of relaxation, namely, is to relax some constraints of the optimization problems. Thus, the relaxed problem is easier to solve than the original one [14]. However, for some finite alphabet detection problems, the solution for the relaxed problem may not directly applicable because it may not exist in the original alphabet. So some approximation algorithms are needed to convert the relaxation solution to an approximate solution for the original problem. For example, we can lift the finite alphabet constraint on (2.5) to reach a relaxed problem, where $\mathbf{b}$ can be any real vector. Suppose this is a antipodal synchronous CDMA case: $\mathbf{H}$ is the correlation matrix, and $\mathbf{v}$ is the zero mean noise vector with covariance $\sigma^2\mathbf{H}$. The minimum mean-square error (MMSE) solution for (2.5) is $[\mathbf{H} + \sigma^2\mathbf{I}]^{-1}\mathbf{y}$. Then

this solution needs to be converted to an approximate solution which lies in the finite alphabet set. There is a simple way to do this by quantizing each element in the vector.

There are many ways to relax the constraints. Semi-definite relaxation proposed in [14] is an efficient and accurate alternative to the NP-hard ML detector. The semi-definite relaxation algorithm stems from the solution for the Boolean quadratic programming (QP) problem [18]. It can be applied to the MLD problem with antipodal data transmissions. There are some advantages of employing semi-definite relaxation to solve the multiuser detection problem [14]:

1. The semi-definite relaxation algorithm is based on solving a convex optimization problem. Hence, this method does not suffer from local maxima.

2. The relaxed problem is a semi-definite programming problem, which is known to be efficiently solvable [19].

3. The semi-definite relaxation algorithm has a theoretical guarantee that the approximation accuracy is, at worst, moderate [18]. Moreover, the performance of this algorithm in practice is substantially better than that of the worst case.

The semi-definite relaxation method [14] to the problem in (2.5) achieves BER performance which is close to that of the ML detector. The complexity of semi-definite relaxation is on the order of $\mathcal{O}(\bar{N}^{3.5})$. Nevertheless, when compared with other detectors, the computational load of the semidefinite programming algorithm is relatively high in practice, even for small-sized problems [46]. In addition, the semi-definite

relaxation algorithm in [14] focuses on antipodal signaling, which is not often used in high-capacity wireless systems. Modulation schemes with large constellation sizes are preferred to achieve high spectral efficiency. For example, quadrature phase shift keying is used in the third-generation cellular communications and 16/64-quadrature amplitude modulation (QAM) in IEEE 802.11a. Higher level modulation is one of the primary reasons that obviate ML detection, but it has not been shown how semi-definite relaxation handles this task. On the other hand, the SDE algorithms proposed in this dissertation not only work for higher-level modulation schemes, but also deliver near-optimal performance with low complexity. The performance of SDE is verified via simulations in Chapter 5.

2.3.3 Probabilistic Data Association

The probabilistic data association (PDA) detector [26] provides a near-optimal solution for the MIMO equalization problem in (2.5) at a low overall complexity of $\mathcal{O}(\bar{N}^3)$. It employs a multistage detection structure, and replaces the intermediate finite alphabet symbol decisions by soft decisions, expressed via their associated posterior probabilities. The key assumption in PDA is that MAI is approximated to obey a Gaussian probability distribution, which is an idea originated from the PDA filter for target tracking [47]. This assumption and the soft-decision structure lead to significant computation reduction as well as superior performance. Despite its impressively low bit error rates, the PDA method in its original form [26] has limited

applications. The original PDA detector was derived for multi-user detection (MUD) within the framework of code-division multiple access CDMA in frequency flat fading channels, in which case the channel matrices $\mathbf{H}$ are confined to be square matrices made of users' cross-correlation coefficients. Moreover, zero-forcing preprocessing via square matrix inverse is performed in PDA, imposing an invertibility constraint on $\mathbf{H}$.

Symbol detection in practical MIMO systems encounters pronouncedly different system parameters $\mathbf{H}$ and $\mathbf{b}$, compared to those in MUD for single-channel CDMA. The channel matrix is typically non-square, and possibly rank-deficient. For example, downlink transmissions often face channels that are fat, that is, there are more columns than rows in $\mathbf{H}$. Such a channel matrix is no longer invertible, even though the system may still be identifiable for digital inputs. A typical case is a BLAST system [48] with a larger number of transmit than receive antennas. The original PDA is unable to work in these cases.

As far as the channel memory is concerned, the PDA method can only handle a short channel memory length of 2 induced by asynchronism [25], with the help of Kalman filtering. This result cannot be straightforwardly extended to channels with a longer memory length. On the contrary, SDE and Kalman-PDA proposed in this dissertation can perform channel equalization for a frequency-selective MIMO channel with an arbitrary length. In short, SDE has broader applications compared with the original PDA:

1. The SDE approach is tailored to handle near-optimum symbol sequence detection in the presence of channel memory, which is difficult to accommodate in PDA MUD. One of our main contributions is to adopt the block transmission structure via zero padding to enable block detection, and to apply sliding windows for ISI cancellation and noise tracking to attain near-MAP detection performance at a low complexity.

2. We propose an alternative implementation of the PDA principle that eliminates the zero-forcing pre-processing. As a result, the restriction on full-rank square channel matrices is lifted.

3. We extend the PDA algorithm to bandwidth-efficient higher-level modulation schemes. Through this work, the potential of the iterative soft-decision PDA philosophy can be fully enjoyed by practical wireless systems to achieve near-optimal, low-complexity detection and equalization.

2.4 Summary

In this chapter, we have established the signal model for a generic frequency-selective MIMO channel. This model embraces a broad range of situations, such as: multiple transmit antennas with a single user, single transmit antenna with multiple users, and transmit-induced diversities including OFDM and CDMA. The optimum solution to this problem incurs prohibitively high complexity, which prompts the research of some near-optimum solutions. We have introduced some of them in this chapter:

sphere decoding, semi-definite relaxation and probabilistic data association. We have provided brief reviews of these algorithms and compare them with the proposed SDE and Kalman-PDA algorithms.

CHAPTER 3

SDE by PDA Enhancement

Starting from this chapter, we will present the soft-decision equalization (SDE) algorithm, the Kalman-PDA equalization algorithm, and the soft-decision based performance enhancement techniques in detail. In this chapter, we first give a brief review of the original PDA method and then develop the SDE algorithm I, SDE-1, in way of PDA enhancement. We can see that the original PDA was developed for multiuser detection in synchronous CDMA, while SDE-1 can handle frequency-selective channels, rank-deficient channels and higher level modulations.

3.1 A Brief Review of PDA

The PDA algorithm for multiuser detection in communications originated from the probability data association filter used for target tracking [47]. PDA was applied to multiuser detection problems in CDMA systems in [26]. We express the matched-filter outputs at the receiver of a CDMA channel with N users as

$$\mathbf{y} = \mathbf{RAb} + \mathbf{n}, \tag{3.1}$$

where $\mathbf{R}$ is the symmetric normalized signature cross-correlation matrix, and $\mathbf{y} = [y_1, \cdots, y_N]^T$, $\mathbf{b} = [b_1, \cdots, b_N]^T$, $\mathbf{A} = diag\{A_1, \cdots, A_N\}$. b_i denotes the anti-podal bit transmitted by the ith users, and A_i denotes the corresponding amplitude. $\mathbf{n}$ is a zero-mean Gaussian random vector with the covariance matrix equal to $\sigma^2\mathbf{R}$. Premultiplying $\mathbf{A}^{-1}\mathbf{R}^{-1}$ on both sides of (3.1), we have

$$\tilde{\mathbf{y}} = \mathbf{b} + \tilde{\mathbf{n}}, \tag{3.2}$$

where $\tilde{\mathbf{y}} = \mathbf{A}^{-1}\mathbf{R}^{-1}\mathbf{y}$ and $\tilde{\mathbf{n}} = \mathbf{A}^{-1}\mathbf{R}^{-1}\mathbf{n}$. Let $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ denote the mean and covariance of $\tilde{\mathbf{n}}$. We have

$$\boldsymbol{\mu} = E\{\mathbf{A}^{-1}\mathbf{R}^{-1}\mathbf{n}\} = \mathbf{0} \tag{3.3}$$

$$\boldsymbol{\Sigma} = E\{\tilde{\mathbf{n}}\tilde{\mathbf{n}}^T\} = \sigma^2\mathbf{A}^{-1}\mathbf{R}^{-1}\mathbf{A}^{-1}. \tag{3.4}$$

Focusing on the detection of the ith element of $\mathbf{b}$, we rewrite (3.2) as

$$\tilde{\mathbf{y}} = b_i\mathbf{e}_i + \sum_{j\neq i} b_j\mathbf{e}_j + \tilde{\mathbf{n}}, \tag{3.5}$$

where $\mathbf{e}_i, i = 1, \cdots, N$ is the ith column of the $N \times N$ identity matrix. Let us denote the sum of noise and interference as $\mathbf{N}_i$, i.e.,

$$\mathbf{N}_i = \sum_{j\neq i} b_j\mathbf{e}_j + \tilde{\mathbf{n}}. \tag{3.6}$$

Then

$$b_i\mathbf{e}_i = \tilde{\mathbf{y}} - \mathbf{N}_i. \tag{3.7}$$

The key idea of PDA is that it assumes $\mathbf{N}_i$ to be Gaussian with matched mean and covariance. The conditional mean and covariance of $b_i\mathbf{e}_i$ are then given by

$$\mathbf{g}_i := E\{b_i\mathbf{e}_i|\tilde{\mathbf{y}}\} = \tilde{\mathbf{y}} - E\{\mathbf{N}_i|\tilde{\mathbf{y}}\} = \tilde{\mathbf{y}} - \sum_{j\neq i}(2P_{bj} - 1)\mathbf{e}_j \tag{3.8}$$

$$\mathbf{\Omega}_i \;:=\; Cov\{b_i\mathbf{e}_i|\tilde{\mathbf{y}}\} = Cov\{\mathbf{N}_i|\tilde{\mathbf{y}}\} = \sum_{j\neq i}[4P_{bj}(1-P_{bj})\mathbf{e}_j\mathbf{e}_j^T] + \mathbf{\Sigma}, \tag{3.9}$$

where $P_{bj} := Pr(b_j = 1|\tilde{\mathbf{y}})$, $j = 1, \cdots, N$.

The conditional probability density function (pdf) of $b_i\mathbf{e}_i$ is given by

$$Pr(b_i\mathbf{e}_i|\tilde{\mathbf{y}}) = \frac{\exp\{-\frac{1}{2}(b_i\mathbf{e}_i - \mathbf{g}_i)^T\mathbf{\Omega}_i^{-1}(b_i\mathbf{e}_i - \mathbf{g}_i)\}}{(2\pi)^{N/2}|\mathbf{\Omega}_i|^{1/2}}. \tag{3.10}$$

We can compute P_{bi} using [26]

$$\frac{P_{bi}}{1-P_{bi}} = \exp\{2\mathbf{g}_i^T\mathbf{\Omega}_i^{-1}\mathbf{e}_i\} \tag{3.11}$$

From the derivation above, we can see that the finite-alphabet ISI symbols are canceled out by their soft-decision alternative instead of tentative hard decisions. On the other hand, parallel interference cancellation (PIC) used hard-decisions to do cancellation which will generally suffer a 2dB performance loss compared with soft-decision based algorithms [27].

3.2 SDE by PDA Enhancement

We now develop soft-decision equalization (SDE) methods based on the PDA-type soft-decision multistage detection principle. We begin with enhancing the PDA detector to enable its applicability to a generic block transmission system with higher-level modulation. A key modification is to associate each symbol with its channel response vector when deriving its posterior probability density function (pdf) and updating its soft decisions. This strategy eliminates the pre-processing of channel matrix inversion, thus relaxes the constraint on channel invertibility. In addition, we present

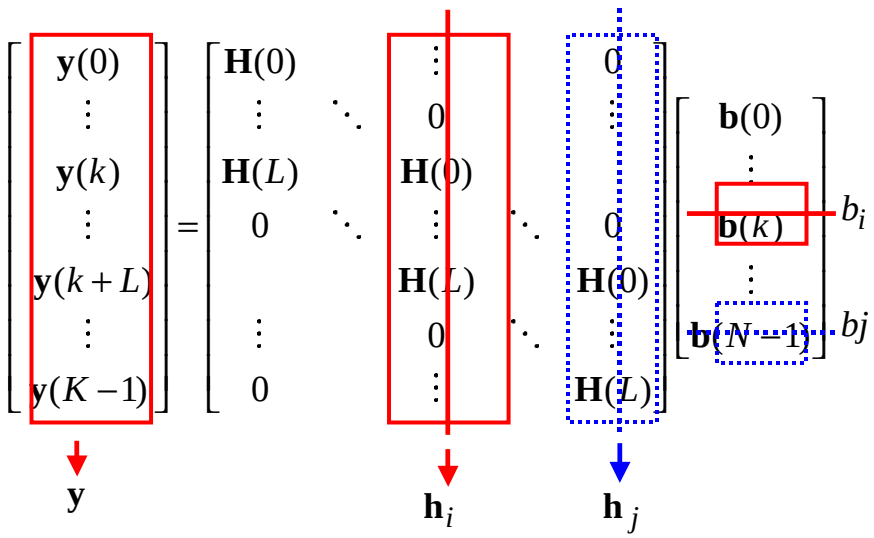

Figure 3.1: Channel model: SDE-1.

efficient implementations of the iterative soft-decision updating rule to further reduce the complexity.

In the general MIMO model $\mathbf{y} = \mathbf{H}\mathbf{b} + \mathbf{v}$ established in 2.2, we emphasize the ith element b_i of $\mathbf{b}$ by rewriting the received signal as

$$\mathbf{y} = \mathbf{h}_i b_i + \sum_{j=1, j\neq i}^{\bar{N}} \mathbf{h}_j b_j + \mathbf{v}, \quad i = 1, \cdots, \bar{N}, \tag{3.12}$$

where $\mathbf{h}_i$ and $\mathbf{h}_j \in \mathcal{R}^{\bar{K}\times 1}$ are the i-th and j-th column of $\mathbf{H}$ respectively, denoting the channel responses of b_i and b_j, $\forall i, j \in [1, \bar{N}]$, as shown in Figure 3.1. In Figure 3.1, each vertical line represents the signature vector of the corresponding input denoted by a horizontal line. The number of elements in each signature vector is $\bar{K}$. The transmitted bits $\{b_j\}_{j=1}^{\bar{N}}$ take values from a finite alphabet set $\{a_1, \cdots, a_M\}$ upon M-ary modulation, where the modulation format is typically chosen from PSK and QAM. Given $\mathbf{y}$, there are M posterior probability values associated with each digital input, which we denote as $\eta_{i,m} := Pr(b_i = a_m|\mathbf{y})$, for $m \in [1, M]$ and $i \in [1, \bar{N}]$.

Finite-alphabet symbol detection on b_i can be alternatively carried out by estimating $\eta_{i,m}$, giving rise to soft decisions. Unfortunately, direct evaluation of $\eta_{i,m}$ via the corresponding likelihood function $p(\mathbf{y}|b_i)$ still incurs exponential complexity in $\bar{N}$, considering that $p(\mathbf{y}|b_i)$ is a Gaussian mixture with $M^{\bar{N}-1}$ modes [50]. To avoid the combinatorial complexity, we adopt the PDA filtering idea and treat the transmitted symbols $\{b_i\}_{i=1}^{\bar{N}}$ as Gaussian random variables. Since the FIR MIMO channel is a linear system, the posterior pdf of b_i remains to be Gaussian, thus can be fully characterized by its mean and variance, conditioned on the received signal $\mathbf{y}$. Define $\mathbf{g}_i := E\{\mathbf{h}_i b_i|\mathbf{y}\}$ and $\mathbf{R}_i := Cov\{\mathbf{h}_i b_i|\mathbf{y}\}$ as the conditional mean and covariance of $\mathbf{h}_i b_i$, respectively. These definitions are different from [26], noting that each symbol b_i is now associated with its channel response vector $\mathbf{h}_i$, instead of a simple unit vector. Such definitions eliminate the need to perform the channel decorrelating pre-processing in [26], which is not applicable for non-square channel matrices and rank-deficient channels. When the transmitted symbols are independent and identically distributed (*i.i.d.*), it follows from (3.12) that

$$\mathbf{g}_i = \mathbf{y} - \sum_{j=1, j\neq i}^{\bar{N}} E\{b_j|\mathbf{y}\}\mathbf{h}_j, \tag{3.13}$$

$$\mathbf{R}_i = \sum_{j=1, j\neq i}^{\bar{N}} var\{b_j|\mathbf{y}\}\mathbf{h}_j\mathbf{h}_j^H + \mathbf{R}_v, \tag{3.14}$$

Here we assume the noise $\mathbf{v}$ is zero mean and with covariance $\mathbf{R}_v$. Thus,

$$E\{b_j|\mathbf{y}\} := \sum_{b_j\in\{a_1,\cdots,a_M\}} b_j Pr(b_j|\mathbf{y}) = \sum_{m=1}^{M} \eta_{j,m} a_m, \tag{3.15}$$

$$\begin{aligned} var\{b_j|\mathbf{y}\} &:= E\{|b_j|^2|\mathbf{y}\} - |E\{b_j|\mathbf{y}\}|^2 \\ &= \sum_{m=1}^{M} \eta_{j,m}|a_m|^2 - |E\{b_j|\mathbf{y}\}|^2, \end{aligned} \tag{3.16}$$

where $|\cdot|$ denotes the magnitude of a complex quantity.

In antipodal signaling, $b_j \in \{\pm 1\}$, which leads to $E\{b_j|\mathbf{y}\} = 1\cdot\eta_j + (-1)\cdot(1-\eta_j) = 2\eta_j - 1$, and $var\{b_j|\mathbf{y}\} = 4\eta_j(1-\eta_j)$, which corroborates [26].

When $\mathbf{h}_i b_i|\mathbf{y}$ is approximated as a Gaussian vector with the matched mean and covariance, its pdf can be described by $\mathbf{g}_i$ and $\mathbf{R}_i$ as follows:

$$p_{b_i|\mathbf{y}}(\mathbf{h}_i b_i|\mathbf{y}) = \frac{\exp\left\{-\frac{1}{2}(\mathbf{h}_i b_i - \mathbf{g}_i)^H \mathbf{R}_i^{-1}(\mathbf{h}_i b_i - \mathbf{g}_i)\right\}}{(2\pi)^{\bar{K}/2}|\mathbf{R}_i|^{1/2}}. \tag{3.17}$$

This pdf calibrates the posterior probabilities of all M possible constellation points of b_i, and the soft decisions can be derived from (3.17) by setting $\eta_{i,m} = c_i p_{b_i|\mathbf{y}}(\mathbf{h}_i a_m|\mathbf{y})$, where c_i is a normalization factor such that $\sum_{m=1}^{M}\eta_{i,m} = 1, \forall i$. Note that the conditional pdf in (3.17) is determined by $\mathbf{g}_i$ and $\mathbf{R}_i$, regardless of the constellation size M. Therefore the overall complexity of evaluating all M soft decisions $\{\eta_{i,m}\}_{m=1}^{M}$ is linear in M.

To obtain $\eta_{i,m}$ in an efficient manner, we introduce the ratios $\lambda_{i,m} := \eta_{i,m}/\eta_{i,1}$, for $m = 1, \cdots, M$, which can be deduced from (3.17) as

$$\begin{aligned}\lambda_{i,m} &= \exp\left\{-\frac{1}{2}(a_m\mathbf{h}_i - \mathbf{g}_i)^H\mathbf{R}_i^{-1}(a_m\mathbf{h}_i - \mathbf{g}_i)\right. \\ &+ \left.\frac{1}{2}(a_1\mathbf{h}_i - \mathbf{g}_i)^H\mathbf{R}_i^{-1}(a_1\mathbf{h}_i - \mathbf{g}_i)\right\}.\end{aligned} \tag{3.18}$$

Using the probability normalization condition $\sum_m \eta_{i,m} = 1$, we obtain

$$\eta_{i,m} = \begin{cases} \frac{1}{1+\sum_{m=2}^{M}\lambda_{i,m}}, & m = 1; \\ \lambda_{i,m}\eta_{i,1}, & m = 2, \cdots, M. \end{cases} \tag{3.19}$$

Subsequently, the maximum *a posteriori* (MAP) estimate of $\hat{b}_i$ is decided to be the a_m value that yields the largest $\eta_{i,m}$. The decision rule can be further simplified in the special binary modulation case [2], [26].

For algorithm implementation, we note that computing $\{\eta_{i,m}\}_{m=1}^{M}$ for one input b_i involves computing $\mathbf{g}_i$ and $\mathbf{R}_i$, both dependent on $\{\eta_{j,m}\}_{m=1}^{M}$ of all other inputs $\{b_j\}_{j=1,j\neq i}^{\bar{N}}$. This intertwined relationship among all unknown inputs prompts an iterative multi-stage procedure [26], where the $(\mathbf{g}_i, \mathbf{R}_i)$ pair is computed from (3.13) and (3.14) based on tentative soft decisions $\{\eta_{j,m}\}_{j\neq i}$ obtained at a previous stage. Each $\eta_{i,m}$ can be updated from (3.19) successively until all $\{\eta_{i,m}\}_{m=1}^{M}$ converge for all $i \in [1, \bar{N}]$, followed by decision making on this block of symbols $\mathbf{b}$ via the MAP rule.

In the above iterative soft-decision updating procedure, the most computationally-expensive operation is to compute the covariance matrix inverse $\mathbf{R}_i^{-1}$, which incurs complexity on the 3rd order of the number of outputs $\bar{K}$. Direct matrix inversion can be avoided using the matrix inverse lemma, which will lower the overall complexity by an order. This speed-up measure is discussed in [26], and it shares the same principle with the widely-used recursive least-square (RLS) adaptive filtering [49]. We now describe these results for our M-ary modulation case.

First, we form two auxiliary variables: $\mathbf{g} := \mathbf{y} - \sum_{j=1}^{\bar{N}} E\{b_j|\mathbf{y}\}\mathbf{h}_j$ is the conditional mean of the noise term $\mathbf{v} = \mathbf{y} - \mathbf{H}\mathbf{b}$, and $\mathbf{R} := \sum_{j=1}^{\bar{N}} var\{b_j|\mathbf{y}\}\mathbf{h}_j\mathbf{h}_j^H + \mathbf{R}_v$ is the conditional covariance matrix of $\mathbf{y}$. Apparently,

$$\mathbf{g}_i = \mathbf{g} + E\{b_i|\mathbf{y}\}\mathbf{h}_i, \tag{3.20}$$

$$\mathbf{R}_i = \mathbf{R} - var\{b_i|\mathbf{y}\}\mathbf{h}_i\mathbf{h}_i^H. \tag{3.21}$$

Table 3.1: Soft Decision Equalization Algorithm: SDE-1

Initialization	$\eta_{i,m} = \frac{1}{M}, \forall i, m;\ \mu_a = \frac{1}{M}\sum_{m=1}^{M} a_m; \sigma_a^2 = \frac{1}{M}\sum_{m=1}^{M} \lvert a_m - \mu_a\rvert^2;$ $\mathbf{g} = \mathbf{y} - \mu_a \sum_{i=1}^{\bar{N}} \mathbf{h}_i;\ \mathbf{R} = \sigma_a^2 \mathbf{H}\mathbf{H}^H + \mathbf{R}_v;$
Soft-decision iterative updating	*Do* *for* $i = 1 : \bar{N}$ – compute $\mathbf{g}_i$ and $\mathbf{R}_i^{-1}$ from (3.20) and (3.22); – compute $\{\eta_{i,m}\}_{m=1}^{M}$ from (3.18) – (3.19), and update $E\{b_i\lvert\mathbf{y}\}$ and $var\{b_i\lvert\mathbf{y}\}$ from (3.15) – (3.16); – update $\mathbf{g}$ and $\mathbf{R}^{-1}$ from (3.20) and (3.23); *end i* *Until* $\eta_{i,m}$ converges for all i and m
MAP Detection	$\hat{b}_i = \arg\max_{b_i \in \{a_1, a_2, \cdots, a_M\}} \eta_{i,m}(b_i)$

Applying the matrix inverse lemma on (3.21) yields [49]

$$\mathbf{R}_i^{-1} = \mathbf{R}^{-1} + \frac{var\{b_i|\mathbf{y}\}\mathbf{R}^{-1}\mathbf{h}_i\mathbf{h}_i^H\mathbf{R}^{-1}}{1 - var\{b_i|\mathbf{y}\}\mathbf{h}_i^H\mathbf{R}^{-1}\mathbf{h}_i}, \tag{3.22}$$

and, conversely

$$\mathbf{R}^{-1} = \mathbf{R}_i^{-1} - \frac{var\{b_i|\mathbf{y}\}\mathbf{R}_i^{-1}\mathbf{h}_i\mathbf{h}_i^H\mathbf{R}_i^{-1}}{1 + var\{b_i|\mathbf{y}\}\mathbf{h}_i^H\mathbf{R}_i^{-1}\mathbf{h}_i}. \tag{3.23}$$

By keeping the updated versions of $\mathbf{g}$ and $\mathbf{R}^{-1}$, $\{\mathbf{g}_i\}_{i=1}^{\bar{N}}$ and $\{\mathbf{R}_i^{-1}\}_{i=1}^{\bar{N}}$ can be obtained from (3.20) and (3.22) at a low complexity of $\mathcal{O}(\bar{K}^2)$ for each input b_i, and an overall complexity of $\mathcal{O}(\bar{N}\bar{K}^2)$ in one iteration. The overall soft-decision MIMO equalization algorithm, enabled by zero-padded block transmission, is summarized in Table 3.1.

3.3 Summary

In this chapter, we first review the original PDA algorithm briefly and then present the SDE-1 algorithm. SDE-1 is superior than PDA in that it is applicable to not only the synchronous CDMA scenario but also the frequency-selective MIMO channels and higher level modulations. We will present in next chapter a novel SDE algorithm that is tailored to the unique Toeplitz structure of the zero-padded frequency selective channel matrix.

CHAPTER 4

SDE by Sliding Windowing

The SDE-1 algorithm presented in the last chapter does not utilized the unique Toeplitz structure of the channel matrix in the equalization problem. When the pdfs in (3.17) are computed via matrix operations, the complexity of the Gaussian-forcing MAP detection is determined by the length of each input's channel response vector $\mathbf{h}_i$, which is $\bar{K} = (N + L)N_o$ in the SDE-1 algorithm. On the other hand, when the block size $K = N + L$ is chosen to be much larger than the channel length L to reduce the transmission redundancy, there are a large number of zeros in the channel matrix, which could be avoided to save computation. Next, we utilize the effective portion of each channel response vector, and construct a soft decision approach that is tailored to the equalization problem. The objective is to maintain the near-MAP detection accuracy of SDE-1, and at the same time reduce the computational complexity to be proportional to the channel memory L instead of the block size $K \gg L$.

4.1 Algorithm Development

Revisiting the signal model (3.12) illustrated in Figure 4.1, we divide the symbol vector $\mathbf{b}$ into N sub-blocks $\{\mathbf{b}(k)\}_{k=0}^{N-1}$, where each sub-block $\mathbf{b}(k) = [b^{(1)}(k) \cdots b^{(N_i)}(k)]^T$ contains the information symbols from all N_i inputs at the kth sampling instant. Due to the finite channel memory length, $\mathbf{b}(k)$ only affects $(L+1)N_o$ output elements, which we group into $\mathbf{y}_{L,k} := [\mathbf{y}^T(k), \cdots, \mathbf{y}^T(k+L)]^T$. This output block $\mathbf{y}_{L,k}$ contains the sufficient statistics of $\mathbf{b}(k)$, as well as contributions from residual ISI elements $\{\mathbf{b}(l)\}_{l \neq k}$. Compared with $\mathbf{y}$, the reduced-sized vector $\mathbf{y}_{L,k}$ contains all the observations relevant to $\mathbf{b}(k)$. It is thus possible to construct an optimum detector for $\mathbf{b}(k)$ from $\mathbf{y}_{L,k}$ in lieu of $\mathbf{y}$, provided that the contributions from other symbols to this output block are properly accounted for, possibly through ISI cancellation and noise tracking. Focusing on one input sub-block at a time, we will convert the signal model in (3.12) into a set of N sub-models, each describing one of the N sufficient statistics $\{\mathbf{y}_{L,k}\}_{k=0}^{N-1}$, such that each sub-model can be expressed by shorter channel response vectors compared with (3.12). Each vertical line in Figure 4.1 represents the signature vector of the corresponding input denoted by a horizontal line. Compared with Figure 3.1, the number of elements in this vector is $(L+1)N_o$ for SDE-2. In the k-th sub-model depicted by the shaded area in Figure 4.1, we denote the channel response matrix from the l-th input sub-block $\mathbf{b}(l)$ to $\mathbf{y}_{L,k}$ by $\mathbf{H}_L(l|k)$, $\forall l$, which refers to the sub-block matrix of $\mathbf{H}$ at the intersection of the k-th row sub-block and the l-th column sub-block. To be exact, it is the portion of $\mathbf{H}$ bordered by the (lN_i+1)-th to the $(l+1)N_i$-th columns, and the (kN_o+1)-th to the $(k+L+1)N_o$-th rows. There

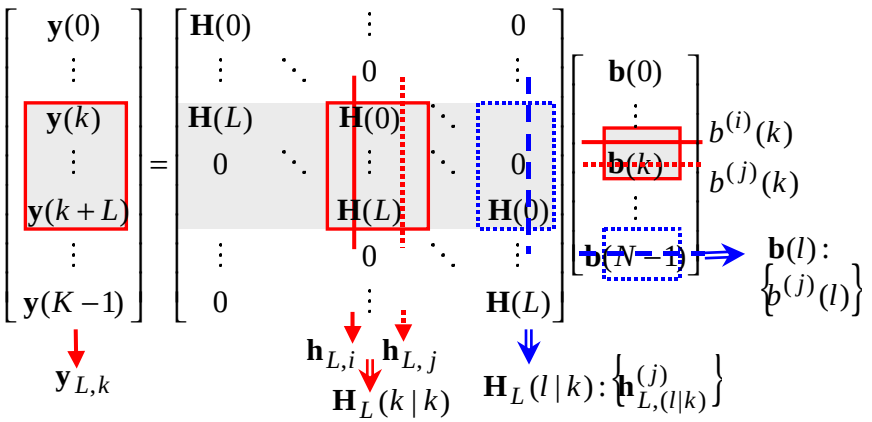

Figure 4.1: Channel model: SDE-2.

are N_i columns in $\mathbf{H}_L(l|k) = \left[\mathbf{h}^{(1)}_{L,(l|k)} \cdots \mathbf{h}^{(N_i)}_{L,(l|k)}\right]$, where $\mathbf{h}^{(i)}_{L,(l|k)}$ is the $(L+1)N_o \times 1$ channel response vector of $b^{(i)}(l)$ during the k-dependent observation window covering the k-th to the $(k+L)$-th symbol periods. It is further observed that, when $l = k$, $\mathbf{H}_L(k|k)$ represents the non-zero portion of each column block, and is independent of k. In fact, $\mathbf{H}_L(k|k) = \mathbf{H}_L := [\mathbf{H}^T(0) \cdots \mathbf{H}^T(L)]^T$, $\forall k$, which happens to be the $(L+1)$-tap FIR MIMO channel impulse response. With these definitions, $\mathbf{y}_{L,k}$ can be obtained from the general signal model (3.12) by sliding over $\mathbf{y}$ a k-dependent observation window of $L+1$ symbol periods, yielding

$$\mathbf{y}_{L,k} = \mathbf{H}_L \mathbf{b}(k) + \sum_{l=0, l\neq k}^{N-1} \mathbf{H}_L(l|k)\mathbf{b}(l) + \mathbf{v}_{L,k}, k = 0, \cdots, N-1 \qquad (4.1)$$

where $\mathbf{v}_{L,k} := [\mathbf{v}^T(k) \cdots \mathbf{v}^T(k+L)]^T$ is the zero-mean white Gaussian noise component that falls within the sliding window. Its covariance matrix $\mathbf{R}_{v,k} := E\{\mathbf{v}_{L,k}\mathbf{v}^H_{L,k}\}$ is readily available as a $(L+1) \times (L+1)$ sub-block of $\mathbf{R}_v$. Hence we have established N reduced-sized sub-models, resulted from overlapping sliding observation windows.

Next, we will show how to utilize the enhanced PDA MUD detector within each reduced-sized sub-model for symbol-by-symbol detection. Focusing on $\mathbf{b}(k)$ in the k-th sliding window described by (4.1), we further dissect the channel matrix $\mathbf{H}_L$, and represent the non-zero (effective) portion of the channel response vector for $b^{(i)}(k)$ as $\mathbf{h}_{L,i}$, which is the ith column of $\mathbf{H}_L$, and is independent of k. Define $\eta_{i,m}(k) := Pr(b^{(i)}(k) = a_m|\mathbf{y}_{L,k})$, $m = 1, \cdots, M$, as the posterior probabilities of the M-ary modulated symbol $b^{(i)}(k)$. The conditional mean and variance of $b^{(i)}(k)$ are then given by

$$\mu_b^{(i)}(k) := E\{b^{(i)}(k)|\mathbf{y}_{L,k}\} = \sum_{m=1}^{M} a_m \eta_{i,m}(k), \tag{4.2}$$

$$\sigma_b^{(i)}(k) := var\{b^{(i)}(k)|\mathbf{y}_{L,k}\} = \sum_{m=1}^{M} |a_m|^2 \eta_{i,m}(k) - |\mu_b^{(i)}(k)|^2, \tag{4.3}$$

respectively. To make MAP detection on $b^{(i)}(k)$, the task now is to evaluate its posterior probability distribution. We suppose the posterior pdf is Gaussian after the Gaussian forcing approximation, hence can be fully characterized by its mean and variance conditioned on $\mathbf{y}_{L,k}$.

Defining $\mathbf{v}_L(k) := \sum_{l \neq k} \sum_{j=1}^{N_i} \mathbf{h}_{L,(l|k)}^{(j)} b^{(j)}(l) + \mathbf{v}_{L,k}$, and emphasizing each individual element $b^{(i)}(k)$ in $\mathbf{b}(k)$, we rewrite (4.1) as

$$\mathbf{y}_{L,k} = \mathbf{h}_{L,i} b^{(i)}(k) + \sum_{j \neq i}^{N_i} \mathbf{h}_{L,j} b^{(j)}(k) + \mathbf{v}_L(k), \quad i = 1, \cdots, N_i. \tag{4.4}$$

Equation (4.4) follows the same structure as (3.12), hence the SDE-1 algorithm can be applied within this local time window for detecting $\mathbf{b}(k)$. To do so, we express the signal component from $b^{(i)}(k)$ by

$$\mathbf{h}_{L,i} b^{(i)}(k) = \mathbf{y}_{L,k} - \sum_{j \neq i}^{N_i} \mathbf{h}_{L,j} b^{(j)}(k) - \mathbf{v}_L(k). \tag{4.5}$$

The conditional mean and covariance of $\mathbf{h}_{L,i}b^{(i)}(k)$ are thus given by

$$\begin{aligned}\mathbf{g}_k^{(i)} &:= E\{\mathbf{h}_{L,i}b^{(i)}(k)|\mathbf{y}_{L,k}\} \\ &= \mathbf{y}_{L,k} - \sum_{j\neq i}^{N_i}\mathbf{h}_{L,j}\mu_b^{(j)}(k) - E\{\mathbf{v}_L(k)|\mathbf{y}_{L,k}\} \qquad (4.6)\end{aligned}$$

$$\begin{aligned}\mathbf{R}_k^{(i)} &:= Cov\{\mathbf{h}_{L,i}b^{(i)}(k)|\mathbf{y}_{L,k}\} \\ &= \sum_{j\neq i}^{N_i}\mathbf{h}_{L,j}\mathbf{h}_{L,j}^H\sigma_b^{(j)}(k) + Cov\{\mathbf{v}_L(k)|\mathbf{y}_{L,k}\} \qquad (4.7)\end{aligned}$$

To obtain the conditional mean and covariance of $\mathbf{v}_L(k)$, we adopt the following approximations for all $l \neq k$ and $j \in [1, N_i]$:

$$E\{b^{(j)}(l)|\mathbf{y}_{L,k}\} \approx E\{b^{(j)}(l)|\mathbf{y}_{L,l}\} = \mu_b^{(j)}(l), \qquad (4.8)$$

$$var\{b^{(j)}(l)|\mathbf{y}_{L,k}\} \approx var\{b^{(j)}(l)|\mathbf{y}_{L,l}\} = \sigma_b^{(j)}(l). \qquad (4.9)$$

As a result, we have

$$E\{\mathbf{v}_L(k)|\mathbf{y}_{L,k}\} = \sum_{l\neq k}\sum_{j=1}^{N_i}\mathbf{h}_{L,(l|k)}^{(j)}\mu_b^{(j)}(l), \qquad (4.10)$$

$$Cov\{\mathbf{v}_L(k)|\mathbf{y}_{L,k}\} = \sum_{l\neq k}\sum_{j=1}^{N_i}\mathbf{h}_{L,(l|k)}^{(j)}\mathbf{h}_{L,(l|k)}^{(j)H}\sigma_b^{(j)}(l) + \mathbf{R}_{v,k}. \qquad (4.11)$$

Now we are ready to use the enhanced PDA framework to compute the soft-detection on $b^{(i)}(k)$. As a result of the Gaussian forcing approximation, the posterior probabilities of $b^{(i)}(k)$ can be obtained as

$$Pr\left(\mathbf{h}_{L,i}b^{(i)}(k)|\mathbf{y}_{L,k}\right) = \frac{\exp\left\{-\frac{1}{2}(\mathbf{h}_{L,i}b^{(i)}(k) - \mathbf{g}_k^{(i)})^H\mathbf{R}_k^{(i)-1}(\mathbf{h}_{L,i}b^{(i)}(k) - \mathbf{g}_k^{(i)})\right\}}{(2\pi)^{N_o(L+1)/2}\left|\mathbf{R}_k^{(i)}\right|^{1/2}} \qquad (4.12)$$

where $b^{(i)}(k) \in \{a_1, \cdots, a_M\}$. Once the posterior probabilities $\eta_{i,m}(k) = Pr(\mathbf{h}_{L,i}a_m|\mathbf{y}_{L,k})$, $m = 1, \cdots, M$, are established from (4.12), the MAP detection on $\{b^{(i)}(k)\}_{i=1}^{N_i}$ can be made in the same manner as described in (3.18) and (3.19).

4.2 Iterative Implementations

As discussed in SDE-1, the evaluation of posterior probabilities $\{\eta_{i,m}(k)\}_{m=1}^{M}$ for each input $b^{(i)}(k)$ involves computing $\mathbf{g}_k^{(i)}$ and $\mathbf{R}_k^{(i)-1}$, which are dependent on not only the unknown $\{\eta_{j,m}(k)\}$ of MAI symbols $\{b^{(j)}(k)\}_{j\neq i}$ at the k-th sampling time, but also the unknown $\{\eta_{j,m}(l)\}$ of ISI symbols $\{b^{(j)}(l)\}_{l\neq k}$, $\forall j$. Algorithm implementation via iterative multi-stage processing is thus in order. Similar to the speed-up strategy described in (3.22) and (3.23), we introduce two auxiliary variables that are instrumental to computational saving in updating the $(\mathbf{g}_k^{(i)}, \mathbf{R}_k^{(i)-1})$ pair: one is the conditional mean $\mathbf{g}(k)$ of the noise term $\mathbf{v}_{L,k} = \mathbf{y}_{L,k} - \mathbf{H}_L\mathbf{b}(k) - \sum_{l\neq k}\mathbf{H}_L(l|k)\mathbf{b}(l)$, given by

$$\begin{aligned}\mathbf{g}(k) &:= \mathbf{y}_{L,k} - \sum_{j=1}^{N_i}\mathbf{h}_{L,j}E\{b^{(j)}(k)|\mathbf{y}_{L,k}\} \\ &- \sum_{l\neq k}\sum_{j=1}^{N_i}\mathbf{h}_{L,(l|k)}^{(j)}E\{b^{(j)}(l)|\mathbf{y}_{L,k}\}. \end{aligned} \tag{4.13}$$

The other is the conditional covariance matrix $\mathbf{R}(k)$ of $\mathbf{y}_{L,k}$ in the form of

$$\begin{aligned}\mathbf{R}(k) := &\sum_{j=1}^{N_i}\mathbf{h}_{L,j}\mathbf{h}_{L,j}^{H}var\{b^{(j)}(k)|\mathbf{y}_{L,k}\} + \\ &\sum_{l\neq k}\sum_{j=1}^{N_i}\mathbf{h}_{L,(l|k)}^{(j)}\mathbf{h}_{L,(l|k)}^{(j)H}var\{b^{(j)}(l)|\mathbf{y}_{L,k}\} + \mathbf{R}_{v,k}. \end{aligned} \tag{4.14}$$

Both $\mathbf{g}(k)$ and $\mathbf{R}(k)$ take into account of the channel responses to all the elements in $\mathbf{b}(k)$, and all the ISI components as well. For reasons that will be explained in Section 4.3, we term the updating of $\mathbf{g}(k)$ due to the ISI components as soft ISI cancellation, and that of $\mathbf{R}(k)$ as noise tracking.

Obvious contrast between (4.6)-(4.7) and (4.13)-(4.14) shows that

$$\mathbf{g}(k) = \mathbf{g}_k^{(i)} - \mu_b^{(i)}(k)\mathbf{h}_{L,i} \tag{4.15}$$

$$\mathbf{R}(k) = \mathbf{R}_k^{(i)} + \sigma_b^{(i)}(k)\mathbf{h}_{L,i}\mathbf{h}_{L,i}^H. \tag{4.16}$$

Therefore, for $i = 1, \cdots, N_i$ within each local window k, we have

$$\mathbf{R}_k^{(i)-1} = \mathbf{R}(k)^{-1} + \frac{\sigma_b^{(i)}(k)\mathbf{R}^{-1}(k)\mathbf{h}_{L,i}\mathbf{h}_{L,i}^H\mathbf{R}^{-1}(k)}{1 - \sigma_b^{(i)}(k)\mathbf{h}_{L,i}^H\mathbf{R}^{-1}(k)\mathbf{h}_{L,i}}, \tag{4.17}$$

$$\mathbf{R}^{-1}(k) = \mathbf{R}_k^{(i)-1} - \frac{\sigma_b^{(i)}(k)\mathbf{R}_k^{(i)-1}\mathbf{h}_{L,i}\mathbf{h}_{L,i}^H\mathbf{R}_k^{(i)-1}}{1 + \sigma_b^{(i)}(k)\mathbf{h}_{L,i}^H\mathbf{R}_k^{(i)-1}\mathbf{h}_{L,i}}. \tag{4.18}$$

During iterative processing, $\mathbf{g}_k^{(i)}$ and $\mathbf{R}_k^{(i)-1}$ are first updated via (4.15) and (4.17), using tentative soft information $\mu_b^{(i)}(k)$ and $\sigma_b^{(i)}(k)$ from the previous stage. Subsequently, the posterior pdf of $b^{(i)}(k)$ can be computed from (4.12), yielding updated soft information $\mu_{b,new}^{(i)}(k)$ and $\sigma_{b,new}^{(i)}(k)$ of the current stage. The auxiliary variables $\mathbf{g}(k)$ and $\mathbf{R}(k)$ are then updated via (4.15) and (4.18) using the new values. Bear in mind that $\mathbf{y}_{L,k}$ in each sliding window only yields the soft information of one input block $\mathbf{b}(k)$, but $\mathbf{b}(k)$ also affects L previous overlapping windows $\{\mathbf{y}_{L,l}\}_{l=k-L}^{k-1}$ and L future overlapping windows $\{\mathbf{y}_{L,l}\}_{l=k+1}^{k+L}$. Therefore, there are $\min(2L, N-1)$ additional pairs of auxiliary variables $\{\mathbf{g}(l), \mathbf{R}^{-1}(l)\}_{l\neq k}$ that need to be updated from the new estimates of $\mathbf{b}(k)$. The updating of these ISI pairs can be carried out instantaneously when any $\mu_{b,new}^{(i)}(k)$ and $\sigma_{b,new}^{(i)}(k)$ become available, or after these soft-information values are updated for all $b^{(i)}(k)$, $i \in [1, N_i]$ and $k \in [0, N-1]$, resulting in two implementation procedures of different computational loads.

In the first procedure, we update the related ISI auxiliary pairs whenever $\mu_{b,new}^{(i)}(k)$ and $\sigma_{b,new}^{(i)}(k)$ become available for any i and k. Following (4.13), an auxiliary ISI mean

can be updated by

$$\begin{aligned} \mathbf{g}_{new}(l) &= \mathbf{g}(l) + \mathbf{h}^{(i)}_{L,(k|l)}\mu^{(i)}_b(k) - \mathbf{h}^{(i)}_{L,(k|l)}\mu^{(i)}_{b,new}(k), \\ l &= k-L,\cdots,k-1,k+1,\cdots,k+L. \end{aligned} \tag{4.19}$$

To update $\mathbf{R}^{-1}_{new}(l)$, $l = k-L,\cdots,k+L$, it is observed from (4.14) that

$$\mathbf{R}_{new}(l) = \mathbf{R}_i(l|k) + \mathbf{h}^{(i)}_{L,(k|l)}\mathbf{h}^{(i)H}_{L,(k|l)}\sigma^{(i)}_{b,new}(k), \tag{4.20}$$

where $\mathbf{R}_i(l|k) := \mathbf{R}(l) - \mathbf{h}^{(i)}_{L,(k|l)}\mathbf{h}^{(i)H}_{L,(k|l)}\sigma^{(i)}_b(k)$ is independent of $b^{(i)}(k)$. Based on (4.20), we can apply the matrix inverse lemma twice to update $\mathbf{R}^{-1}_{new}(l)$ from $\mathbf{R}^{-1}(l)$ and $\sigma^{(i)}_{b,new}(k)$:

$$\mathbf{R}^{-1}_i(l|k) = \mathbf{R}^{-1}(l) + \frac{\sigma^{(i)}_b(k)\mathbf{R}^{-1}(l)\mathbf{h}^{(i)}_{L,(k|l)}\mathbf{h}^{(i)H}_{L,(k|l)}\mathbf{R}^{-1}(l)}{1-\sigma^{(i)}_b(k)\mathbf{h}^{(i)H}_{L,(k|l)}\mathbf{R}^{-1}(l)\mathbf{h}^{(i)}_{L,(k|l)}}. \tag{4.21}$$

$$\mathbf{R}^{-1}_{new}(l) = \mathbf{R}^{-1}_i(l|k) - \frac{\sigma^{(i)}_{b,new}(k)\mathbf{R}^{-1}_i(l|k)\mathbf{h}^{(i)}_{L,(k|l)}\mathbf{h}^{(i)H}_{L,(k|l)}\mathbf{R}^{-1}_i(l|k)}{1+\sigma^{(i)}_{b,new}(k)\mathbf{h}^{(i)H}_{L,(k|l)}\mathbf{R}^{-1}_i(l|k)\mathbf{h}^{(i)}_{L,(k|l)}}. \tag{4.22}$$

The intermediate matrix inverse $\mathbf{R}_i(l|k)$ plays a similar role to the updating of $\mathbf{R}(l)$ as $\mathbf{R}^{(i)}_k$ to $\mathbf{R}(k)$. The computational load of updating all the relevant auxiliary variables for each input is on the order of $\min(N, 2L+1)(L+1)^2N_o^2$, which represents an approximate L/K reduction compared with the SDE-1 algorithm. The overall algorithm is summarized in Table 4.1.

In the above procedure, N overlapping time windows are processed in serial, resulting in a total of $\bar{N} \times \min(N, 2L+1)$ times of updating the auxiliary pairs inside each stage. To reduce the number of covariance matrix inverse to be processed, we may take an alternative procedure to update the auxiliary matrices $\{\mathbf{R}^{-1}(k)\}^{N-1}_{k=0}$. In each stage, we first process the N local windows in parallel to update the soft decisions of all inputs $\{\mathbf{b}(k)\}^{N-1}_{k=0}$, using tentative decisions and auxiliary variables from

Table 4.1: Soft Decision Equalization Algorithm: SDE-2 (I)

Initialization	$\eta_{i,m}(k) = \frac{1}{M}$, $\mu_b^{(i)}(k) = \frac{1}{M}\sum_{m=1}^{M} a_m$, $\sigma_b^{(i)}(k) = \frac{1}{M}\sum_{m=1}^{M}\left\|a_m - \mu_b^{(i)}(k)\right\|^2$, $\forall i, m, k$; compute $\mathbf{g}(k)$, $\mathbf{R}(k)$ and $\mathbf{R}(k)^{-1}$ using (4.13) and (4.14).
Soft-decision iterative updating	*Do* *for* $k = 0 : N-1$, (for each input sub-block) *for* $i = 1 : N_i$, (for each symbol within a sub-block) Step a. Soft-decision MUD on $\mathbf{b}(k)$ – update $\mathbf{g}_k^{(i)}$ and $\mathbf{R}_k^{(i)-1}$ using (4.15) and (4.17); – compute $\eta_{i,m}(k) = Pr(\mathbf{h}_{L,i}a_m \mid \mathbf{y}_{L,k})$ using (4.12), and steps in (3.18)-(3.19); – update $\mu_b^{(i)}(k)$ and $\sigma_b^{(i)}(k)$ using (4.2) and (4.3); Step b. Soft ISI cancellation and noise tracking of auxiliary pairs – update the related $(2L+1)$ auxiliary pairs $\{\mathbf{g}(l), \mathbf{R}^{-1}(l)\}_{l=k-L}^{k+L}$ using (4.19), (4.21), (4.22) for $l \neq k$, and (4.15), (4.18) for $l = k$. *end i* *end k* *Until* $\eta_{i,m}(k)$ converges for all i, m, and k
Detection	$\hat{b}^{(i)}(k) = \arg\max_{b^{(i)}(k)\in\{a_1,a_2,\cdots,a_M\}} \eta_{i,m}(k)$

the previous stage. When all the new soft-information values are available, we re-calculate the N auxiliary pairs $\{\mathbf{g}(k), \mathbf{R}(k)\}_{k=0}^{N-1}$, possibly by using their definitions in (4.13) and (4.14). On average, there are only N pairs of auxiliary variables to be updated inside each stage, and the complexity order per symbol is the smaller of $\min(N, 2L+1)(L+1)^2 N_o^2$ (when the matrix inverse lemma is used), and $(L+1)^3 N_o^3/N_i$ (when direct matrix inversion is used). This procedure is summarized in Table 4.2. The implementations in both Tables 4.1 and 4.2 are expected to converge to the SDE-1 algorithm, which have been verified in our simulations. The second procedure offers a complexity advantage when the channel length is much less than the block size, i.e., $L \ll N$.

4.3 Comparisons with Existing Algorithms

The SDE-2 MIMO equalizer resembles a concatenation of a series of enhanced PDA/ SDE-1 detectors, each operating on a truncated sub-model to reduce the overall complexity. It is worth emphasizing that the pdf estimators in (3.17) for SDE-1 and in (4.12) for SDE-2 are equivalent, when both converge to the steady state. The only approximations involved are (4.8) and (4.9), which if at the steady state, does not incur performance loss. The multi-stage iterative processing nature of the SDE algorithms prompts their links with the parallel interference cancellation (PIC) [50], and the soft-input-soft-output (SISO) MMSE detector that was developed for turbo detection. In the next, we compare SDE with existing interference cancellation techniques.

Table 4.2: Soft Decision Equalization Algorithm: SDE-2 (II)

Initialization	same as in Table 4.1
Soft-decision iterative updating	*Do* Step a. Soft ISI cancellation and noise tracking of N auxiliary pairs – update all $\{\mathbf{g}(l), \mathbf{R}^{-1}(l)\}_{l=0}^{N-1}$ using (4.13) and (4.14). Step b. Soft-decision MUD over N overlapping truncated windows: *for* $k = 0 : N-1$ (for each input sub-block) *for* $i = 1 : N_i$ (for each symbol within a sub-block) – update $\mathbf{g}_k^{(i)}$ and $\mathbf{R}_k^{(i)-1}$ using (4.15) and (4.17); – compute $\eta_{i,m}(k)$ from (4.12), and steps in (3.18)-(3.19); – update $\mu_b^{(i)}(k)$ and $\sigma_b^{(i)}(k)$ using (4.2) and (4.3); – update $\mathbf{g}(k)$ and $\mathbf{R}^{-1}(k)$ using (4.15) and (4.18). *end* i *end* k *Until* $\eta_{i,m}(k)$ converges for all i, m, and k
Detection	$\hat{b}^{(i)}(k) = \arg\max_{b^{(i)}(k)\in\{a_1,a_2,\cdots,a_M\}} \eta_{i,m}(k)$

4.3.1 Comparison with Hard-Decision PIC

In each truncated output vector $\mathbf{y}_{L,k}$, $\{\mathbf{b}(l)\}_{l\neq k}$ can be viewed as the interference to the desired symbol block $\mathbf{b}(k)$, and the sub-model (4.4) can be written as $\mathbf{y}_{L,k} - \sum_{l\neq k}\mathbf{H}_L(l|k)\mathbf{b}(l) = \sum_{i=1}^{N_i}\mathbf{h}_{L,i}b^{(i)}(k) + \mathbf{v}_{L,k}$. This appears to be deduced from (4.1) using the PIC structure [50]. A close examination shows that the SDE-2 method treats the tentative decisions on $\mathbf{b}(l)$ differently from PIC. An equivalent model for SDE-2 is described below:

$$\tilde{\mathbf{y}}_{L,k} = \sum_{i=1}^{N_i}\mathbf{h}_{L,i}b^{(i)}(k) + \tilde{\mathbf{v}}_{L,k} \tag{4.23}$$

where the observation vector is modified to $\tilde{\mathbf{y}}_{L,k}$ in the form of

$$\tilde{\mathbf{y}}_{L,k} := \mathbf{y}_{L,k} - \sum_{l\neq k}\mathbf{H}_L(l|k)E\{\mathbf{b}(l)|\mathbf{y}_{L,k}\}, \tag{4.24}$$

and the noise term $\tilde{\mathbf{v}}_{L,k}$ is independent of $\mathbf{b}(k)$, and is assumed to be Gaussian with zero-mean, and covariance $\tilde{\mathbf{R}}_{v,k}$ being equal to $Cov\{\mathbf{v}_L(k)|\mathbf{y}_{L,k}\}$ in (4.11), i.e.,

$$\tilde{\mathbf{R}}_{v,k} := \mathbf{R}_{v,k} + \sum_{l\neq k}\sum_{j=1}^{N_i}\mathbf{h}^{(j)}_{L,(l|k)}\mathbf{h}^{(j)H}_{L,(l|k)}\sigma_b^{(j)}(l). \tag{4.25}$$

The efficacy of (4.23) can be established by the fact that both $\mathbf{g}_k^{(i)}$ and $\mathbf{R}_k^{(i)}$ in (4.6)-(4.7) that are required to fully characterize the posterior pdf of $b^{(i)}(k)$ can be equivalently obtained from (4.23) – (4.25). Compared with PIC, our SDE-2 method entails three major differences: (i) The finite-alphabet ISI symbols $\mathbf{b}(l)$ are canceled out by their soft-decision alternative $E\{\mathbf{b}(l)|\mathbf{y}_{L,k}\}$ instead of tentative hard decisions; (ii) In addition to the soft-decision interference cancellation, the conditional variances of the soft estimates are tracked and lumped into the variance of noise $\tilde{\mathbf{v}}_{v,k}$, as seen

in (4.25). In contrast, hard-decision PIC does not change the statistics of the noise term $\mathbf{v}_{L,k}$ during iterations. Due to this key noise tracking step, the formulation in (4.23) retains optimality subject to the Gaussian forcing approximation, while the conventional PIC is sub-optimum. (iii) The SDE-2 method performs interference cancellation along overlapping blocks, using a sliding window of size $(L+1)N_o$ outputs, such that all the observations related to each input are retained within the corresponding window. The conventional PIC receiver, on the other hand, operates on a non-overlapping sliding window of size 1.

4.3.2 Comparisons between SDE, PDA and SISO-MMSE Based Turbo Detection

The soft-decision iterative processing nature of our SDE algorithms prompts their links to turbo signal processing. In turbo detection [34], soft information in the form of log-likelihood ratios (LLR) is exchanged; interchangeably, the PDA, SDE-1, and SDE-2 methods iteratively feedback the means and variances of Gaussian distributed random symbols. We now compare the multi-stage Gaussian forcing principle used in PDA, SDE-1 and SDE-2, with the SISO-MMSE based turbo principle for uncoded systems. For different algorithms, we will explain the posterior probabilities derived for b_i in the general model $\mathbf{y} = \mathbf{h}_i b_i + \sum_{j \neq i} \mathbf{h}_j b_j + \mathbf{v}$. The comparison will be based on binary modulation.

In SDE-1 for binary signaling, the posterior probability distribution of b_i in (3.17) is reduced to

$$\frac{Pr(b_i = 1|\mathbf{y})}{Pr(b_i = -1|\mathbf{y})} = \exp\left\{2\mathbf{h}_i^H \mathbf{R}_i^{-1} \mathbf{g}_i\right\} \tag{4.26}$$

where $\mathbf{g}_i$ and $\mathbf{R}_i$ are given by (3.13) and (3.14), respectively.

The PDA algorithm focuses on MUD in synchronous CDMA multiple access, where $\mathbf{H}$ is a real-valued, square cross-correlation matrix, and the noise variance is $\mathbf{R}_v = \sigma_n^2 \mathbf{H}$. Let $\mathcal{N}(\mu, \sigma_n^2)$ represent a Gaussian random variable with mean μ and variance σ_n^2. Under the special system setup, the PDA MUD establishes a Gaussian model for b_i in the form of [Ref. [26]]

$$\mathbf{e}_i b_i = \bar{\mathbf{g}}_i + \mathcal{N}(0, \bar{\mathbf{R}}_i) \tag{4.27}$$

where $\bar{\mathbf{g}}_i = \mathbf{y} - \sum_{j \neq i} E\{b_j|\mathbf{y}\}\mathbf{e}_j$, $\bar{\mathbf{R}}_i = \sum_{j \neq i} \mathbf{e}_j \mathbf{e}_j^T var\{b_j|\mathbf{y}\} + \sigma_n^2 \mathbf{H}^{-1}$, and $\mathbf{e}_i$ is a column vector whose i-th element is 1 while all other components are 0. Pre-multiplying $\mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1}$ on both sides of (4.27) yields

$$\mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \mathbf{e}_i b_i = \mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \bar{\mathbf{g}}_i + \mathcal{N}\left(0, \mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \mathbf{e}_i\right). \tag{4.28}$$

The soft decision made by PDA is thus given by

$$\frac{Pr(b_i = 1|\mathbf{y})}{Pr(b_i = -1|\mathbf{y})} = \exp\left\{2\mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \bar{\mathbf{g}}_i\right\}. \tag{4.29}$$

Using the equality $\mathbf{h}_i = \mathbf{H}\mathbf{e}_i$, it can be established that $\mathbf{h}_i^H \mathbf{R}_i^{-1} \mathbf{g}_i = \mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \bar{\mathbf{g}}_i$. Therefore, the PDA result in (4.29) is the same as that of SDE-1 in (4.26) in this special case. As an enhancement to PDA, our SDE-1 algorithm does not restrict $\mathbf{R}_v$ to be proportional to $\mathbf{H}^{-1}$, and it applies even when $\mathbf{H}^{-1}$ does not exist.

The SISO-MMSE method generates a MMSE filtered decision statistic z_i for estimating b_i, where z_i boils down to [[34], (41) and (50)]

$$z_k = \mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \bar{\mathbf{g}}_i = \mathbf{e}_i^T \mathbf{R}_i^{-1} \mathbf{e}_i b_i + \mathcal{N}\left(0, \mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \mathbf{e}_i - \mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \mathbf{e}_i \mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \mathbf{e}_i\right). \tag{4.30}$$

The LLR of b_i is thus given by

$$\frac{p(\mathbf{y}|b_i = 1)}{p(\mathbf{y}|b_i = -1)} = \exp\left\{\frac{2\mathbf{e}_i^T \bar{\mathbf{R}}_i^{-1} \mathbf{g}_i}{1 - \mathbf{e}_i \bar{\mathbf{R}}_i^{-1} \mathbf{e}_i}\right\}. \tag{4.31}$$

The comparison between SISO-MMSE and PDA (a special case of SDE-1) is clearly illustrated by the similarities and differences between (4.28) – (4.29) and (4.30) – (4.31).

As to the SDE-2 method, it is specially tailored to the equalization problem with a Toeplitz channel structure, therefore is not directly comparable with the existing turbo detectors. Interestingly, the iterative processing in SDE-2 suggests the flow chart in Figure 4.2, which interprets the algorithm by a turbo structure in which two major function blocks: MUD and ISI cancellation, exchange information in an iterative manner. SDE-2 builds upon the Gaussian forcing idea, and incorporates sliding windowing to reduce the overall complexity without sacrificing the detection performance. The key to retain optimality after data truncation is to carry out the noise tracking step (4.25) prior to each reduced-dimension local SDE-1 detection on (4.23). Noise tracking via Kalman filtering has appeared in the context of PDA detection for asynchronous CDMA under frequency flat fading [25]. Such a PDA-Kalman tracker cannot be generalized to track the noise in a channel with a memory

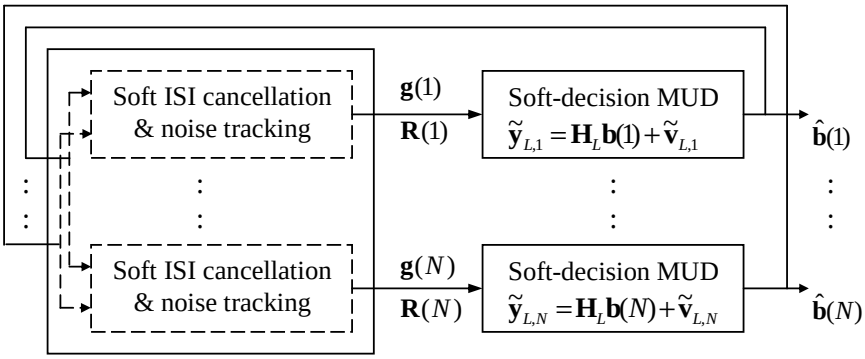

Figure 4.2: Turbo-like flow chart of the SDE-2 algorithm.

length $L > 2$, since the underlying dynamic model is no longer first-order, thus obviating Kalman filtering. In our development, we interpret noise tracking as a means to update the variance of the ISI estimates. This viewpoint allows us to generalize the noise tracking method easily to channels with long memory length, and there is no need to perform channel Cholesky decomposition and Kalman filtering as in [25].

4.4 Summary

In this chapter, we present a novel soft decision approach that is tailored to frequency selective MIMO channel equalization problem, taking advantage of the unique Toeplitz structure. Near-optimum BER performance is maintained, and the computational complexity is reduced to be proportional to the channel memory length L instead of the block size $K \gg L$. We also provide comparisons with parallel interference cancellation (PIC), and the soft-input-soft-output (SISO) MMSE detector that

was developed for turbo detection. We will provide simulation results and performance analysis in the next chapter.

CHAPTER 5

Simulations and Performance Analysis of SDE

In the last two chapters, we proposed two versions of soft-decision equalization algorithms. Compared with sphere decoding and the original PDA algorithm, SDE does not require computing the inverse of the channel matrix, the restriction of invertibility of the channel matrix is lifted. Therefore, SDE can be applied to MIMO systems that have less receive than transmit antennas [54], where many equalization algorithms fail. In this chapter, we investigate the characteristics of our two soft-decision equalization methods through computer simulations. In both the SDE-1 and SDE-2 algorithms, the soft decisions are derived to converge to the sequence MAP estimates through multi-stage iterations, therefore close-to-optimal symbol detection performance is anticipated. On the other hand, Gaussian forcing explains the low-complexity feature of these SDE methods. These claims will be verified here by comparisons with other competing methods, including the optimum ML detection by brute-force enumeration, quasi-ML by sphere decoding (SD) [21], [55], and the sub-optimal hard-decision MIMO FIR MMSE-DFE method [31]. Performance metrics of interest are the bit-error-rate (BER) performance in both full column-rank and rank-deficient channels,

and the computational complexity in terms of the number of operations versus the frame data size N.

In Section 5.1, we present the simulation results of the SDE-1 and SDE-2 algorithms in full rank channels. Then in Section 5.2, we try to probe the performance of SDE in rank-deficient channels. We set up various channel parameters and use SDE to do symbol detection. We plot out the performance in terms of bit-error-rate (BER) and summarize our observations. Finally, the complexity analysis of SDE-1 and SDE-2 is given in Section 5.3 to attest that SDE are low-complexity, near-optimum algorithms for MIMO symbol detection and equalization.

5.1 BER Performance in Full-Rank MIMO Channels

In the simulated MIMO system, each input-output radio link is generated independently from the broadband wireless High Performance European Radio LAN (HIPERLAN) model [56], [57]. The channels are complex-valued, and the noise is assumed to be complex white Gaussian. The time-varying FIR channels are generated according to the channel model A specified by ETSI for HiperLAN/2 [56], resulting in a maximum channel memory length of $L = 8$ symbols. Each channel tap varies according to Jakes' model with a maximum Doppler frequency of 52 Hz corresponding to a typical terminal speed and a carrier frequency of 5.2 GHz.

We study the BER performance versus the signal-to-noise ratio (SNR) of various detectors under different modulation schemes and numbers of antennas. The total

transmit power is held constant irrespective of the number of transmit antennas. For each given SNR, the simulation keeps running until the number of errors for the (near-optimum) sphere decoding algorithm reaches 100 or greater. With this number of errors, the simulated BER is within $\pm 20\%$ of the true BER.

We start with the case of more number of receive antennas than transmit antennas, i.e., $N_o > N_i$. Due to the block transmission structure with a transmit redundancy of L padded zeros, the Toeplitz channel matrix $\mathbf{H}$ in (2.2) has a full column-rank of $\bar{N} = NN_i$, and is guaranteed invertibility irrespective of channel nulls [23].

Figure 5.1 illustrates the performance comparison of SDE-1, SDE-2, SD, FIR MMSE-DFE, and ML (by brute-force search) for $N_i = 1$ and $N_o = 4$. The symbol block size N is chosen as 8. The results for both 16-QAM and 64-QAM are presented. The same MIMO setup is considered in Figure 5.2, except that the number of transmit antennas is increase to $N_i = 2$. In both figures, it can be seen that the BER curves of SDE-1, SDE-2 and SD are nearly identical for different high-bandwidth-efficiency modulation schemes. They all approach that of the optimum ML detection. This corroborates the near-optimum property of the reduced-complexity SDE-2 technique. FIR MMSE-DFE, however, experiences non-trivial performance degradation in all the above scenarios. The performance gap in the $N_i = 2$ case is more pronounced than that in the $N_i = 1$ case. As N_i increase, the information-theoretic capacity is expected to grow linearly in N_i, given $N_i \leq N_o$ [1]. MMSE-DFE cannot deliver the desired performance as capacity-driven MIMO systems exploit more transmit antennas. On

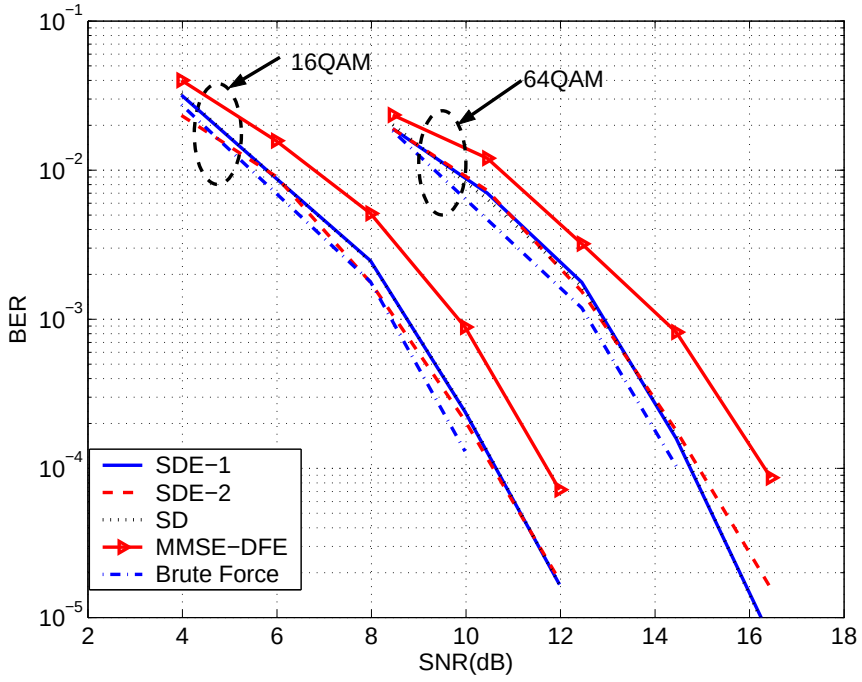

Figure 5.1: Performance comparison under 16/64QAM, $N_i = 1$, $N_o = 4$

the other hand, our SDE methods, with their near-optimum performance, are very promising candidates to bring the potency of MIMO systems to practice.

5.2 BER Performance in Rank-Deficient MIMO Channels

Rank-deficient MIMO channels exist in many wireless scenarios, such as in mobile downlink transmission where there are typically more transmit than receive antennas. A so-called fat channel matrix $\mathbf{H}$ arises when $KN_o < NN_i$, which means that $\mathbf{H}$ has more columns than rows, and its pseudo-inverse $\mathbf{H}^\dagger$ no longer exists. This poses a significant challenge for symbol detection, since a $NN_i \times 1$ input data vector is projected onto an output/observation space of a smaller dimension KN_o [55]. Such a

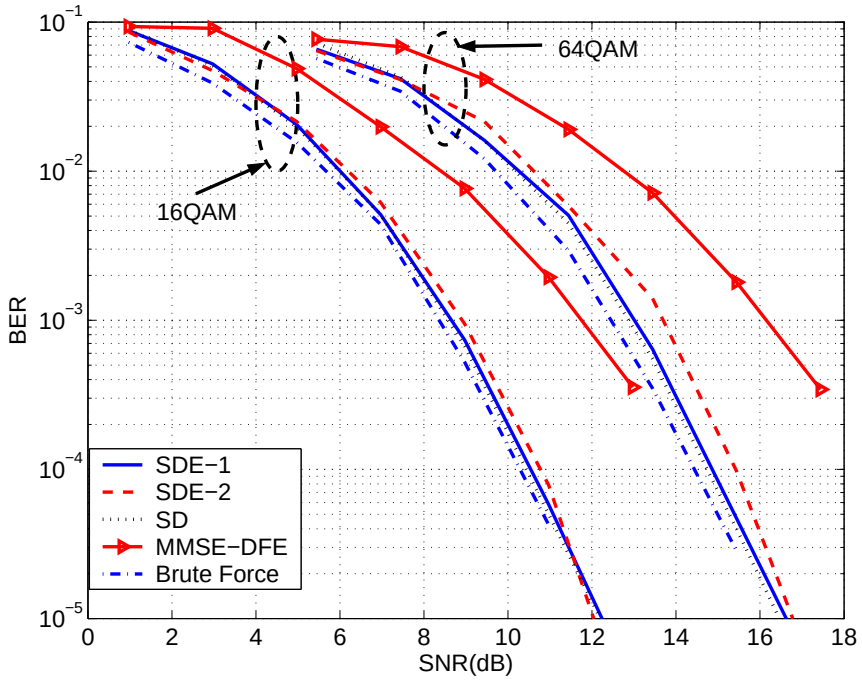

Figure 5.2: Performance comparison under 16/64QAM, $N_i = 2$, $N_o = 4$

channel is identifiable only when each distinct finite-alphabet input $\mathbf{b}$ can be mapped into a distinct and resolvable output $\mathbf{y}$ when free of noise. Even when the system identifiability condition is satisfied, a rank-deficient channel is difficult to process. First, many detection techniques that require channel invertibility do not apply. This includes the linear zero-forcing and MMSE detectors [50], and the original PDA MUD filter [26]. Second, even when a detector does not face implementation difficulty, its detection performance may exhibit an unacceptably large noise floor. Examples include FIR MMSE-DFE and SD. In MMSE-DFE, a fat channel matrix severely reduces the power efficiency of the feedforward filter, which in turn compromises ISI cancellation in the feedback filter design. The original SD judiciously uses the lattice structure of the finite-alphabet input data to perform quasi-ML search at a low complexity. Unfortunately, such a lattice search is infeasible for $NN_i > KN_o$ [21], and

the generalized SD (GSD) does not preserve optimality due to a reduced-dimension lattice projection [55].

To investigate the behavior of our SDE methods in the fat-channel case, we first take a look at a simple example in which we choose a MIMO setup with $N_o = 1$ receive antenna, $N_i = 3$ transmit antennas, and a block size of $N = 5$. The corresponding channel matrix $\mathbf{H}$ is thus 13×15 in dimension. In Figure 5.3, the performance of channel equalization by brute-force ML is plotted as a baseline, along that of SDE-1, SDE-2, GSD, and MMSE-DFE. The BER values for both MMSE-DFE and GSD stay above 10^{-1}, even for high SNR. The SDE methods also incur considerable performance degradation compared with the optimum ML, but do not seem to exhibit an error floor. Intuitively, soft-decision based methods with Gaussian forcing track the composite covariance of MAI components as noise. Even when there is a rank reduction, or some MAI components are too close in the signal space, the composite noise effect could still retain full rank under the ill-conditioned channel, thus leading to convergence in symbol detection.

Next, we further investigate the performance of SDE in various fat channels. In order to quantify the "shape" of the channel matrix, we define the dimension ratio as

$$\Lambda_F = \frac{N_i}{N_o}. \tag{5.1}$$

So for a system with a given number of receive antennas, the larger Λ_F is, the more transmit antennas it has. Because of the reasons listed in the beginning of this section, symbol detection and channel equalization algorithms face greater challenges as Λ_F increases, and thus the performance of SDE is expected to become worse. We

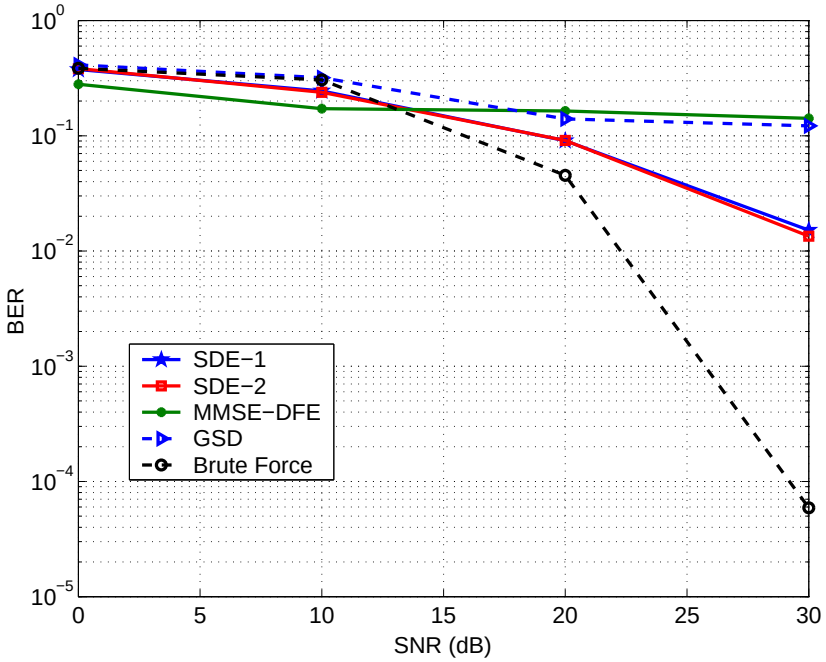

Figure 5.3: Performance comparison in fat channel case, $N_i = 3$, $N_o = 1$

would like to probe the trade-off between Λ_F and the BER performance. Intuitively, for a MIMO system with a fixed number of receive antennas, a higher Λ_F means more transmit antennas, which can accommodate more users, and thus offer a higher data rate. The penalty for having a larger number of transmit antennas is the degraded BER performance, because there is more MAI for each user. In the following simulations, we investigate the performance of SDE versus Λ_F in various scenarios.

5.2.1 BPSK-Modulated MIMO Systems

We first simulate a MIMO setup with $N_o = 4$ receive antennas. The number of transmit antennas ranges from 3 to 8. The modulation scheme used is binary phase

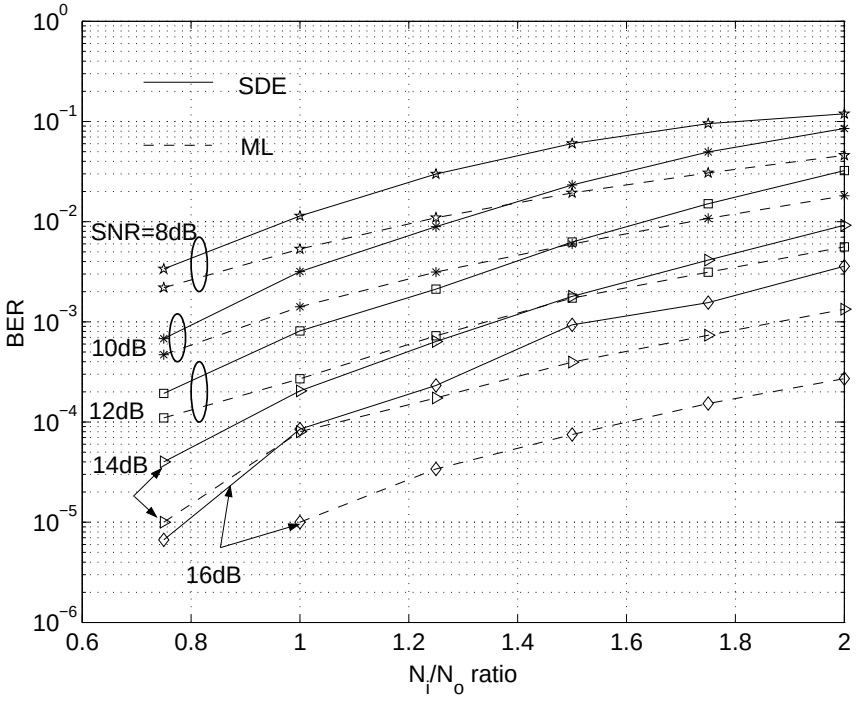

Figure 5.4: BPSK, N_o=4, N_i=3 – 8, flat fading

shift keying (BPSK), and the transmitted signals are assumed to go through flat-fading channels. We simulated the BER performance of SDE (SDE-1) along with the optimal ML detector by way of brute-force searching. The BERs at different SNRs are plotted versus Λ_F in Fig 5.4. At a given SNR, we keep the total transmit power same when increasing the number of transmit antennas. We can see that the BER performance does degrade as Λ_F increases, for a given SNR. But this can be mitigated by increasing the SNR. With SNR increased by 2 dB, the performance of SDE with $N_i + 1$ ($N_i = 3 - 7$) transmit antennas can be better than that of SDE with N_i transmit antennas. This can be cost-effective in some applications.

We also simulate a BPSK-modulated MIMO system with $N_o = 10$ receive antennas. We observed in Fig. 5.5 that the BER performance of SDE deteriorates again

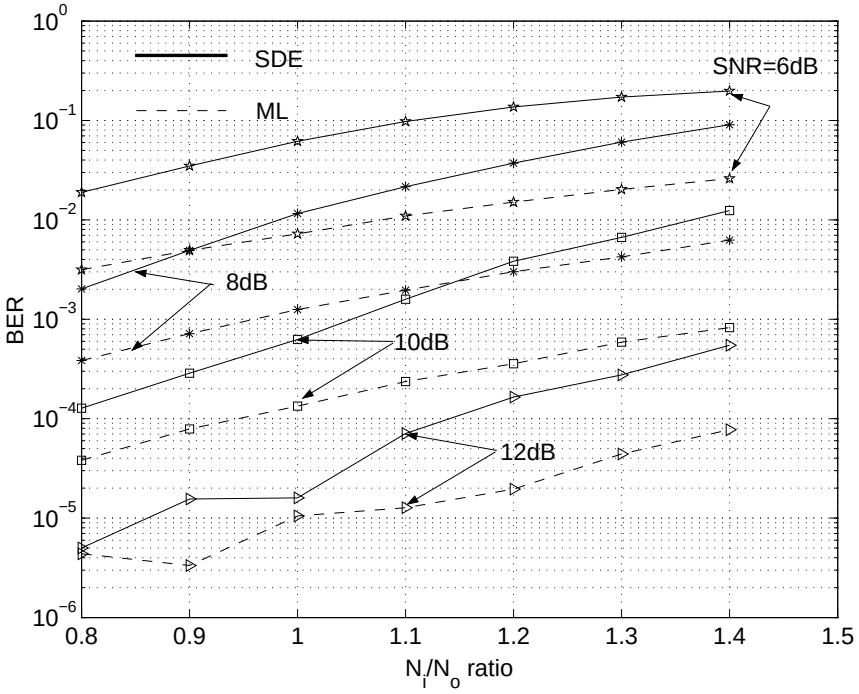

Figure 5.5: BPSK, N_o=10, N_i=8 – 14, flat fading

as Λ_F increases. Similar to the $N_o = 4$ case, the performance degradation can be alleviated or improved to even better by increasing SNR. In this setup, the BER is approximately equal to 1.3×10^{-4}, when $N_i = 8$, SNR=10dB. At the same SNR, when $N_i = 12$, the BER goes up to 3.8×10^{-3}. However, if we increase the SNR by 2dB, the BER is drastically lowered to 1.7×10^{-4}. Thus, with the cost of 2dB in SNR, the number of transmit antennas (users) can be increased from 8 to 12 while a competitive level of BER performance is retained.

In both aforementioned setups, the system can accommodate a few additional users at a reasonable cost of transmit power. It can be seen that ML also suffers from performance loss as Λ_F increases. We also notice that the performance gap between SDE and ML is small when $\Lambda_F < 1$, but grows larger as Λ_F increases.

The performance gap also becomes more pronounced for higher SNRs. However, we do not see any error floors in the plot, which implies we can always increase transmit power to mitigate the performance loss caused by a increasing number of antennas. This manifests that SDE has a prominent advantage in MIMO systems that have a fat channel matrix. Among existing detection algorithms, some (such as the linear zero-forcing, MMSE detection, the original PDA MUD filter, etc) simply do not apply, while others, e.g. FIR MMSE-DFE and sphere decoding, may exhibit an unacceptably large noise floor. Since fat channels exist widely in the downlink of mobile communications, this application advantage of SDE is significant. So SDE presents itself as a strong candidate for MIMO multiuser detection and equalization algorithms.

5.2.2 16QAM-Modulated MIMO Systems

We continue simulating the performance of SDE in fat channel MIMO systems. In this subsection, we focus on systems that adopt the 16-QAM (quadrature amplitude modulation) scheme since high-capacity wireless systems typically require large signal constellation size to improve spectral efficiency, and 16-QAM has been used in IEEE 802.11a. We simulate systems with 4, 6, 10 and 12 receive antennas. The results are plotted in Fig. 5.6, Fig. 5.7, Fig. 5.8 and Fig. 5.9. The BER performance of these setups share the same trend. We observe that:

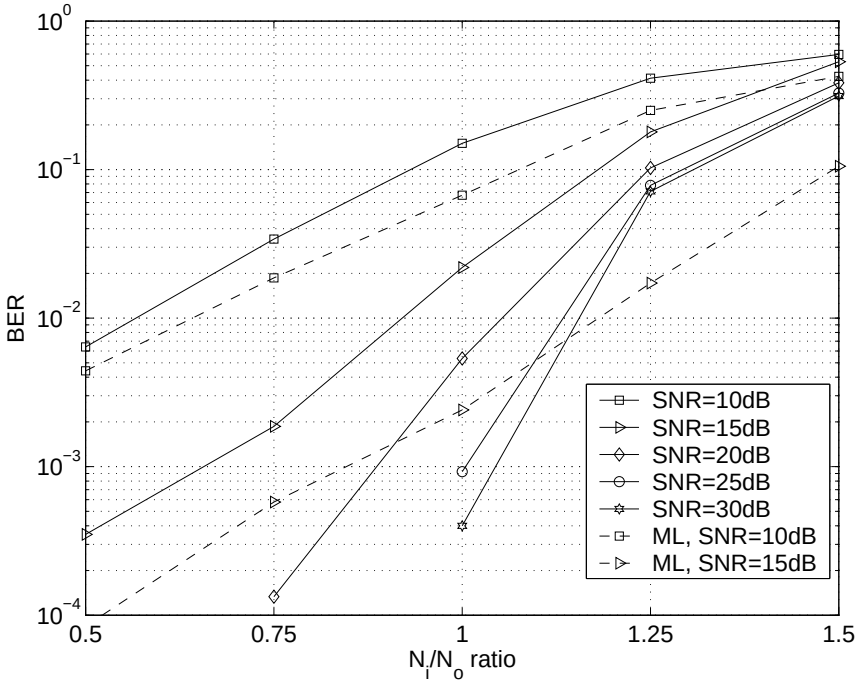

Figure 5.6: 16-QAM, N_o=4, N_i=2 – 6, flat fading

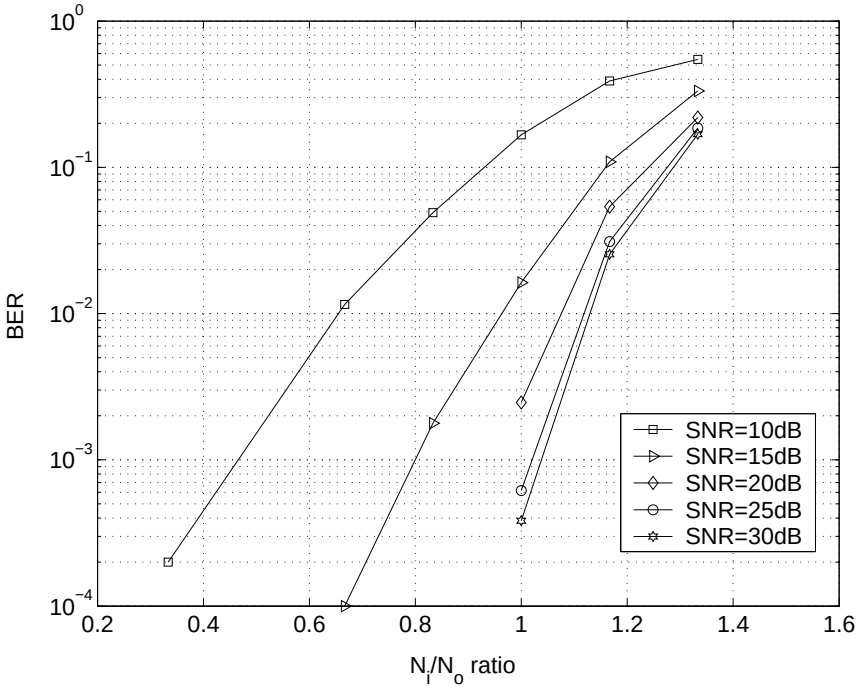

Figure 5.7: 16-QAM, N_o=6, N_i=2,4 – 8, flat fading

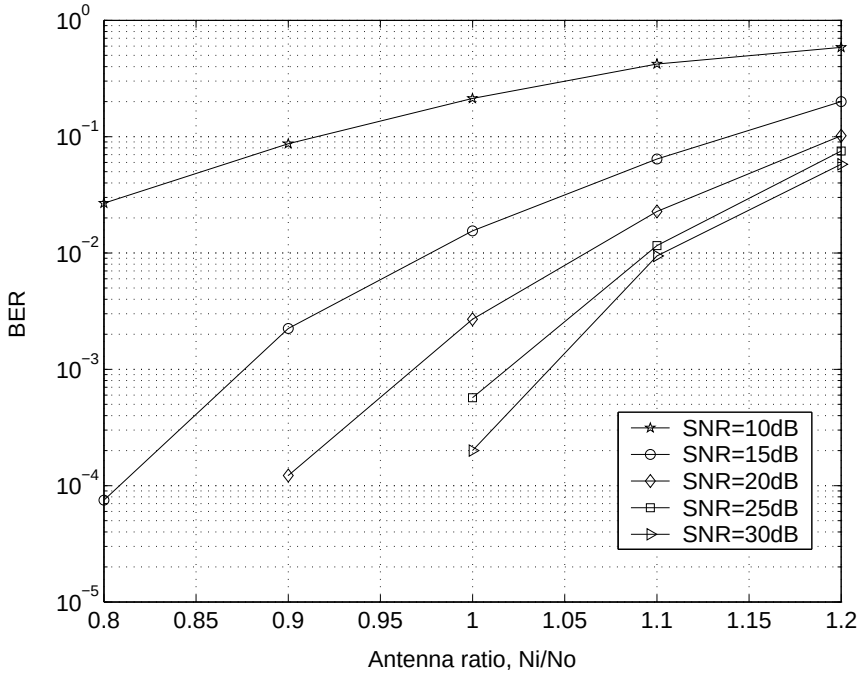

Figure 5.8: 16-QAM, N_o=10, N_i=8 – 12, flat fading

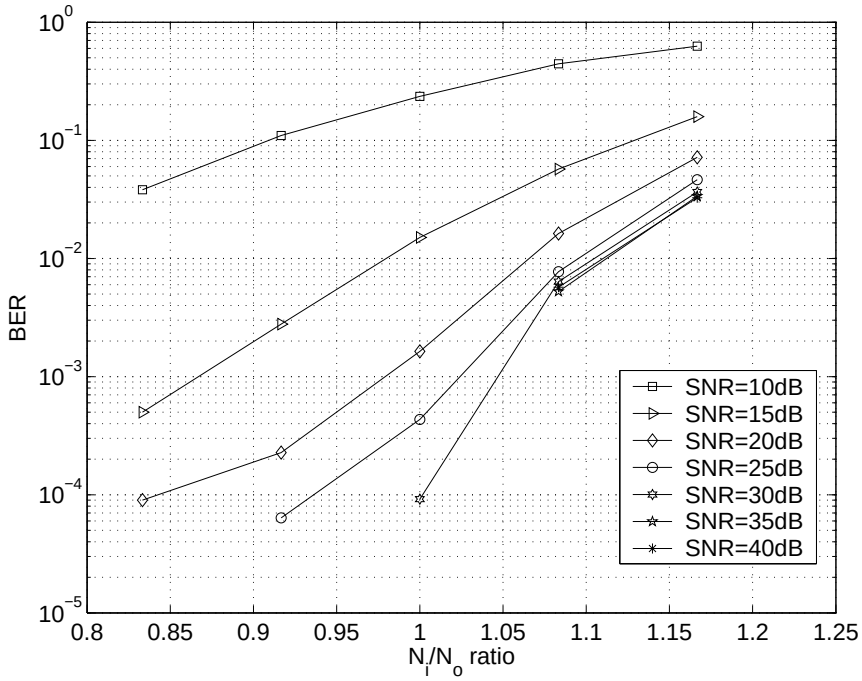

Figure 5.9: 16-QAM, N_o=12, N_i=10 – 14, flat fading

1. For $\Lambda_F \leq 1$ cases, the BER can always be lowered below a reasonable level (5×10^{-4}). This means SDE performs well when the channel matrix is tall or square. This corroborates what we claimed the previous sections.

2. However, when $\Lambda_F > 1$, the BER performance of SDE improves little although SNR increases up to 30 dB. This is different from BPSK cases. This implies that in 16-QAM systems, the application of SDE is restricted to tall or square channel to ensure a satisfactory performance. Although, SDE theoretically works for fat channels, regardless of the modulation scheme used, it suffers from error floors when applied to fat channels that use 16-QAM modulation.

3. We plot out the BER of the optimum ML detector for the $N_o = 4$ case. It can be seen that the performance of ML also deteriorates as Λ_F increases. But the performance gap between SDE and ML is much more pronounced than that in BPSK cases. For $N_i = 5$ and $N_i = 6$ cases, ML at SNR=15 dB outperforms SDE at SNR=30 dB. This shows that fat channels are extremely challenging for near-optimal detection algorithms.

In the simulations, we also observe that the posterior probabilities of each symbol in 16QAM cases are closer to each other than in BPSK cases. In fact, the closer the posterior probabilities locate to each other, the less reliable a decision is, so the more liable a detection is to be wrong. This observation unveils a unique advantage of soft-decision based detection algorithms: the detector/receiver can sense the reliability of the decisions on line. So it is possible for the receiver to enhance performance by

utilizing this property. We will discuss two categories of performance enhancement techniques for soft-decision equalization algorithms in Chapter 7.

5.3 Complexity Evaluation

As we explained in Chapter 2, the input size $\bar{N} = NN_i$ in a MIMO system may come from multiple access and/or multiple antennas, therefore is potentially very large for a high-capacity MIMO system. The computational load of the optimum ML detection is $\mathcal{O}(M^{\bar{N}})$, where M is the alphabet size of the input data. Such complexity is infeasible for a high-capacity (large $\bar{N}$), high-throughput (large M) system, which motivates the search for near-optimal, low-complexity symbol detection and channel equalization solutions.

Quasi-ML by sphere decoding entails polynomial complexity on the order of $\mathcal{O}((\bar{N})^2 + (1 + \frac{\bar{N}-1}{4dC})^{4dC})$, [21], where d^{-1} is a lower bound for the eigenvalues of the Gram matrix $\mathbf{G} := \mathbf{H}^T\mathbf{H}$, and C is the square of the initial searching radius. Choosing a large value for C improves the BER performance, but also incurs higher complexity. Typically, close-to-optimal performance can be achieved at a polynomial complexity index between 3–6.

The SDE algorithms are iterative routines that compute the posterior probabilities of each input symbol in a sequential fashion. The number of iterations required for convergence varies from one input to another. Based on our MIMO setups, we have observed from simulations that the posterior probabilities typically converge in 3–5

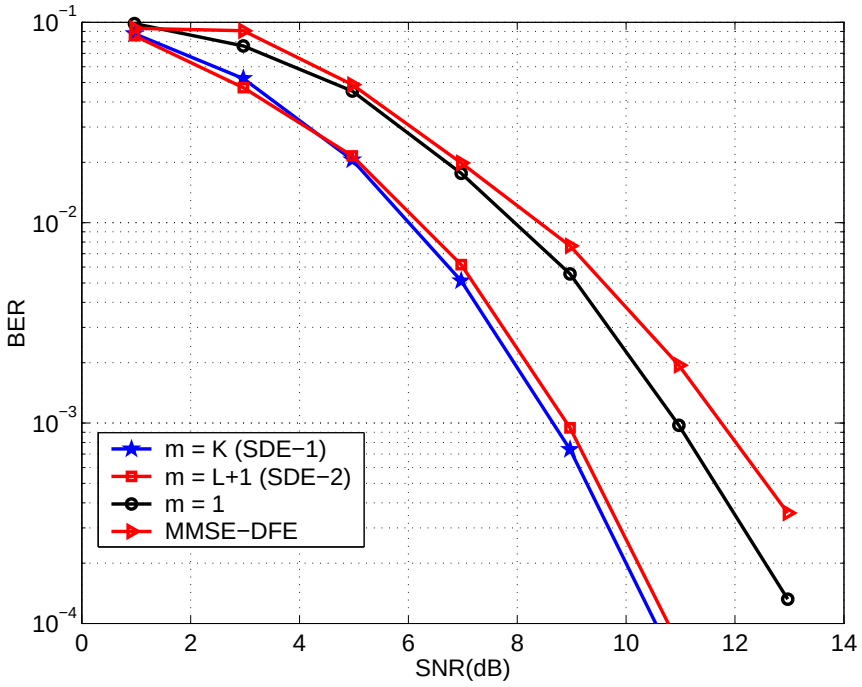

Figure 5.10: Performance vs. m

iterations for higher SNR (>10dB), and in 7–14 iterations for lower SNR (<10dB). In each iteration, the computational load is mainly composed of two parts: one is resulted from computing $\mathbf{R}_k^{(i)-1}$, and the other from evaluating the posterior probabilities using (3.17) or (4.12). If we define m as the sliding window length in terms of the number of sub-blocks, then the number of symbols within the window is defined by $\bar{m} := mN_o$. The $\bar{m}$ value is also the size of the covariance matrix $\mathbf{R}_k^{(i)}$. In the following discussions, we evaluate the performance-complexity tradeoff in choosing the window size $\bar{m}$.

5.3.1 Full-size Windowing $\bar{m} = KN_o$

This case corresponds to the SDE-1 algorithm. There is only one full-size window, therefore all the output sub-blocks are processed simultaneously. The dimension of the covariance matrix $\mathbf{R}_k^{(i)}$ and the auxiliary matrix $\mathbf{R}_k$ are both determined by $\bar{m}$. Because the matrix inverse lemma is used, the complexity of computing $\mathbf{R}_k^{(i)-1}$ is $\mathcal{O}(\bar{m}^2)$. The complexity involved in evaluating M posterior probabilities is $\mathcal{O}(M\bar{m}^2)$ per symbol. Hence the overall complexity per symbol is on the order of $\mathcal{O}(M\bar{m}^2)$. The complexity for detecting $\bar{N}$ symbols in one iteration is then given by $\mathcal{O}(M\bar{m}^2\bar{N})$. Noting $K = N + L$, the complexity of SDE-1 per symbol is on the 2nd order polynomial in the input size, and is only linear to the constellation size M.

5.3.2 Optimum Window Size $\bar{m} = (L+1)N_o$

This case corresponds to the SDE-2 algorithm. SDE-2 takes advantage of the sparse Toeplitz structure of the channel matrix to reduce the equalization complexity. The sliding window only contains the non-zero part of the channel response vector. A symbol can at most affect $L+1$ output blocks in a L-memory channel. These $L+1$ output blocks form the sufficient statistics of each symbol. As a result, SDE-2 can retain the near-optimum performance and at the same time save the computational cost. In SDE-2, computing $\mathbf{R}_k^{(i)}$ and updating the auxiliary matrix $\mathbf{R}_k$ costs $\mathcal{O}(N\bar{m}^2)$. Finding the pdfs of each symbol takes $\mathcal{O}(M\bar{m}^2)$ operations. The overall complexity per symbol

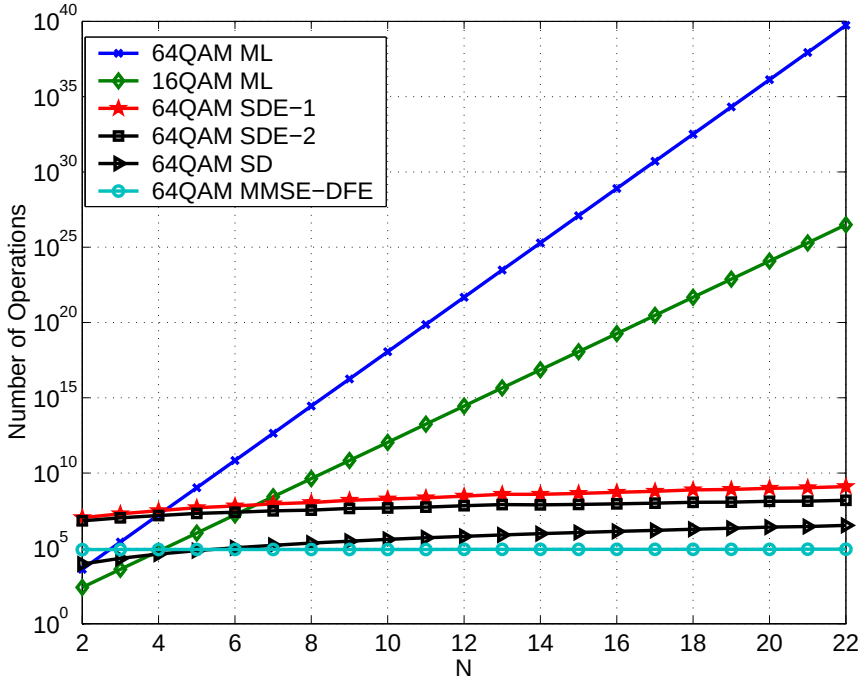

Figure 5.11: Complexity vs. N

is then $\mathcal{O}(\max(N, M)\bar{m}^2)$ and the complexity per iteration is $\mathcal{O}(\max(N, M)\bar{N}\bar{m}^2)$. Compared with SDE-1, the dimension of each conditional covariance matrix $\mathbf{R}_k^{(i)}$ is reduced from KN_o (corresponding to $\mathbf{R}_i$ in SDE-1) to $N_o(L+1)$, thus lowering the complexity order of N from 2 to 1 per input. More impressively, such a complexity reduction does not induce noticeable BER performance loss.

5.3.3 Sub-optimum Window Size $\bar{m} < (L+1)N_o$

The implementation procedures for SDE-2 can be used when the window size $\bar{m} < (L+1)N_o$. As $\bar{m}$ shrinks, the complexity $\mathcal{O}(\max(N, M)\bar{m}^2)$ decreases. However, Since the sliding window only covers a portion of the channel response for the intended

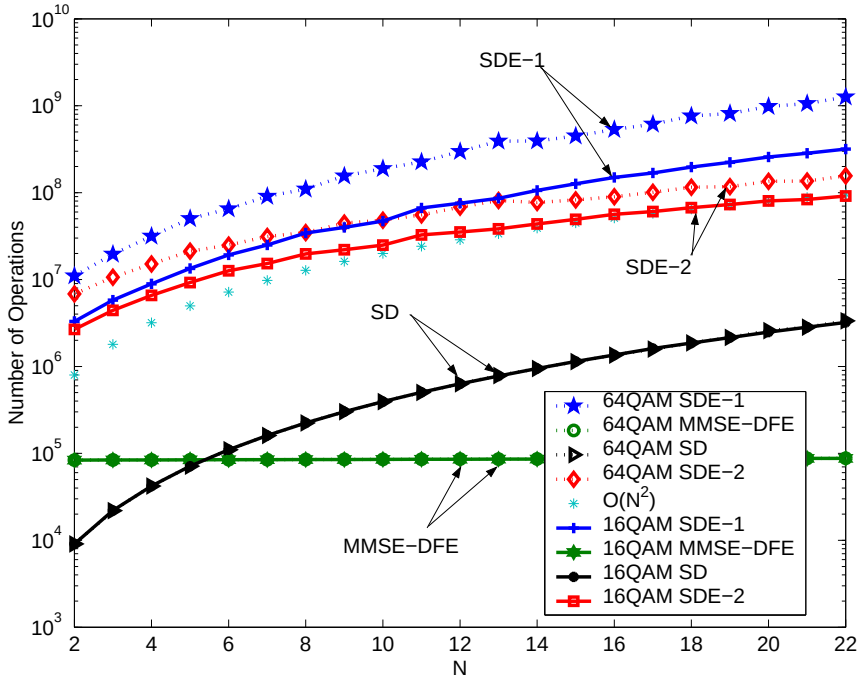

Figure 5.12: Complexity vs. N

input, the algorithm does not make full use of the sufficient statistics, leading to performance degradation. In Figure 5.10, we plot the BER curve at a sliding window size of $m = 1$, along with SDE-1 ($m = K$) and SDE-2 ($m = L + 1$). It is shown that the small window size yields inferior performance to the other two near-optimum schemes. In fact, $(L + 1)N_o$ is the optimal window length, because a longer length does not render better performance but the complexity increases, while a shorter length sacrifices the performance. When $m = 1$, the algorithm becomes a symbol-by-symbol equalization technique, therefore the performance cannot match up to that of sequence detection in ISI channels.

The complexity evaluation results are further verified by simulations in Figure 5.11 and 5.12. The system parameters are set to $N_i = 1$, $N_o = 4$, $L = 8$, and $M = 16$

and 64. The computational load in terms of the number of operations versus the data block size N is depicted for each detection method in Figure 5.11. Our SDE methods, along with SD and MMSE-DFE, avoid asymptotic computational explosion suffered by the brute-force ML algorithm at a large data size. More detailed comparison of these low-complexity algorithms are illustrated in Figure 5.12. In this simulation setting, the sphere decoder has the same 3rd-order complexity in N as the SDE-1 algorithm. This is not always the case, as the complexity of SD could be higher if the Gram matrix $\mathbf{G}$ has very small eigenvalues and the search radius C is chosen to be large. The overall complexity of the SDE-2 algorithm is between SD and SDE-1, but asymptotically its complexity order in N is only 2 instead of 3, as witnessed by its close match with the function N^2 at large N. Such a reduction in the complexity order will pay off for high capacity MIMO systems.

5.4 Summary

In this chapter, we have provided simulation results that demonstrate the performance of SDE algorithms in both full-rank and rank-deficient channels. In full-rank channels, SDE approaches the performance of the ML detector. It also works for rank-deficient channels where sphere decoding and PDA filter can not be applied. We further investigate the performance of SDE in various fat channel scenarios. We observe that SDE works better for BPSK modulation than 16-QAM when the channel is fat. By defining the dimension ratio of a channel and based on the simulation results, we

offer a qualitative guideline for the anticipated performance of SDE in a fat channel. Finally, we provide complexity evaluation to show the theoretical computational load that two SDE algorithms require.

CHAPTER 6

Kalman-PDA Algorithms

In Chapter 3 and 4, we presented SDE by PDA enhancement and SDE by sliding windowing. The performance of SDE algorithms has been attested by the simulation results in Chapter 5. For the zero-padding transmission model established in Chapter 2, SDE-1 performs multiuser detection (MUD) over all N symbol periods in this block, while SDE-2 performs MUD over $L + 1$ symbol periods. Motivated by the need for low-complexity receiver processing, we propose in this chapter a MIMO equalization approach that performs MUD within only one sub-block of $\mathbf{b}$ (local MUD), instead of doing it for multiple sub-blocks (global MUD). The number of symbols within each sub-block is only N_i, representing only one sampling period. This will lower the MUD complexity to only a small fraction of that of the global MUD. Meanwhile, we perform noise-interference tracking within each sub-block to mitigate all the MAI, ISI and noise that affect the current MUD sub-block. This will help retain close-to-optimal reception performance. Together, the MIMO detection and equalization problem is partitioned into two parts: local MUD detection and noise-interference tracking. It is possible to fulfill the local MUD subtask by various (existing) techniques on an individual basis. A similar mechanism using trun-

cated observation windows has been investigated in [13], but the overall performance largely depends on the estimation reliability of the ISI by a single-user correlator, which degrades as LN_i increases. In this paper, we choose probabilistic data association (PDA) detection for local MUD and Kalman filtering (KF) for noise-interference tracking, for the following major considerations: (i) PDA produces soft decisions, which can be useful in devising high-performance iterative receiver processing. (ii) As will be shown later, soft-decision based PDA has attractive capability of sensing error-prone bits on line, which facilitates joint transmitter-receiver design to detect and correct unreliable transmissions. (iii) KF is known as the optimal linear estimator of the state of a system given a set of known inputs and measurements. The tracking procedure is mature and easy to implement. (iv) Both PDA detection and KF estimation rely primarily on second-order statistics (in the forms of means and variances) of the underlying random system; thus, structurally they can be blended nicely thanks to the common theme of second-order statistics estimation. Although neither PDA nor KF are optimum for communication systems with finite-alphabet constraints on the input symbols, we combine them with the help of turbo-like iterative processing along with enhancement strategies to enable overall low-complexity, high-performance MIMO equalization.

Kalman filtering has been widely used in channel estimation and tracking [20], [24], [52], [53]. Under known channel knowledge, a Kalman noise tracker in combination with PDA detection is developed recently in [25] for MUD in decorrelated asynchronous multiuser CDMA systems operating in flat fading channels. This algo-

rithm not only yields detection performance that is close to the lower bound, but also possesses low polynomial-time complexity. However, as this algorithm stands in its current form, the channel memory length L is limited to 2 because the Kalman state equation is formed after factorizing special-structured correlation matrices [25]. Our Kalman-PDA proposed in this paper, in contrast, is tailored to equalization of frequency selective MIMO channels with $L > 2$. Treating a general L-tap FIR channel as an L-order Markovian process, we transform it into a first-order process so that a KF can be generalized to track an MIMO FIR channel with an arbitrary number of taps. The overall complexity of our Kalman-PDA for detecting NN_i symbols is on the order of $\mathcal{O}\{N(LN_i)^3\}$. We would like also to point out that the simplicity and effectiveness of PDA is based on the assumption that the sum of the interference and noise can be well approximated as a single-mode Gaussian random process, so-called Gaussian-forcing assumption. The error performance is dependent on how close this assumption models reality, and is subject to degradation induced by modeling mismatch. To improve the posterior bit probabilities computed in PDA under the Gaussian assumption, we also develop several performance-enhancement strategies: iterative processing, automatic repeat request (ARQ) and/or erasure decoding. The proposed Kalman-PDA approach achieves better performance than the MIMO decorrelator in terms of bit error rate (BER), as will be shown in our algorithm development and simulation results.

Focusing on the complexity-performance issue of MIMO channel equalization, we confines the treatment to systems with perfect channel knowledge of $\mathbf{H}$, for which

$N_o \geq N_i$ is assumed. Under imperfect channel settings, it is possible to combine our soft-decision based detectors with some channel estimators, such as recursive least square (RLS) channel estimators. The detection BER can be comparable with that under perfect channel knowledge, as long as a rough channel estimate is available initially. We will elaborate this in Chapter 8. Similar approach in a turbo-like fashion can be found in [44]. In developing the algorithms, the MIMO channel is assumed to be frequency selective and slowly varying. It is time invariant within each block of K symbol periods, but may change independently from block to block. We suppose that the receiver has perfect knowledge of the channel state information $\mathbf{H}$ and the noise variance $\mathbf{R}_v$. It is also assumed that $N_o \geq N_i$ and all the MIMO taps $\{\mathbf{H}_l\}_{l=0}^{L}$ are full-column rank matrices.

Next, we are going to develop MIMO detectors that combine local PDA MUD with Kalman filtering. The key step of Kalman filtering is to model the composite noise and ISI as the state vector of a first-order dynamic linear system, so that the ISI effect can be effectively tracked via state-vector updating. To this end, we offer two dynamic system models with two different sets of Kalman state and measurement equations. When introducing each of the two algorithms, we follow the steps below:

1. Construct the local PDA MUD structure;

2. Formulate the Kalman state and measurement equations that relate the local ISI to the state vector;

3. Lay out the Kalman iteration procedure for tracking the ISI.

6.1 The Kalman-PDA I Algorithm

The key idea in formulating a Kalman state vector is to set up a decorrelated received signal model and use Kalman filtering to track the mean and covariance of the interference. These statistics are furnished to PDA to facilitate local multiuser detection.

6.1.1 Local PDA MUD Structure

According to the zero-padded signal model described in Chapter 2, the kth received signal sub-block can be expressed by

$$\mathbf{y}(k) = \sum_{l=0}^{L} \mathbf{H}_l \mathbf{b}(k-l) + \mathbf{v}(k), \tag{6.1}$$

where $\mathbf{H}_0, \ldots, \mathbf{H}_L$ are the $N_o \times N_i$ FIR MIMO channel taps, $\mathbf{b}(m)$, $m = k, \ldots, k-L$, is the mth sub-block symbol vector with $\mathbf{b}(k-l) = \mathbf{0}$ for $l > k$, and $\mathbf{v}(k)$ is white Gaussian noise vector at the kth sampling period with zero mean and covariance $\sigma^2 \mathbf{I}_{N_o \times N_o}$.

Our goal is to construct a local MUD for detecting $\mathbf{b}(k)$ at the kth instant. To this end, we define a transformed output $\tilde{\mathbf{y}}(k)$ and a noise term $\tilde{\mathbf{v}}(k)$ that satisfy

$$\mathbf{y}(k) = \sum_{l=0}^{L} \mathbf{H}_l \tilde{\mathbf{y}}(k-l), \tag{6.2}$$

$$\mathbf{v}(k) = \sum_{l=0}^{L} \mathbf{H}_l \tilde{\mathbf{v}}(k-l), \tag{6.3}$$

respectively. Plugging these definitions into (6.1), it follows that

$$\tilde{\mathbf{y}}(k) = \mathbf{b}(k) + \tilde{\mathbf{v}}(k), \quad \forall k. \tag{6.4}$$

Eq. (6.4) is a standard decorrelated PDA model similar to (3.2); hence the PDA procedure in [26] can be applied directly, provided that $\tilde{\mathbf{y}}(k)$ and the statistics of $\tilde{\mathbf{v}}(k)$ can be obtained. The posterior probabilities are initialized as 0.5 (for BPSK modulation), and they are refined during iterations. Final decisions on $\mathbf{b}(k)$ are made after the probabilities converge. Note that the dimension of $\mathbf{b}(k)$ is $N_i \times 1$; thus the complexity of this local PDA MUD is only $\mathcal{O}(N_i^3)$. Meanwhile, $\tilde{\mathbf{v}}(k)$ contains all the ISI effect to this local block; thus there is little loss of optimality by this local MUD, as long as the statistics of $\tilde{\mathbf{v}}(k)$ can be tracked properly. On the other hand, since (6.4) involves zero-forcing in obtaining $\mathbf{b}(k)$, it is subject to noise-interference enhancement when the random channel tap $\mathbf{H}_0$ is not well conditioned[1].

Before we show the process of tracking $\tilde{\mathbf{v}}(k)$ in the next subsection, we now explain how to obtain $\tilde{\mathbf{y}}(k)$ from the observed data $\mathbf{y}(k)$. Let $\mathbf{H}_0^\dagger$ denote the Moore-Penrose pseudo-inverse of $\mathbf{H}_0$. Pre-multiplying (6.2) by $\mathbf{H}_0^\dagger$ and re-arranging the elements in (6.2) to emphasize $\tilde{\mathbf{y}}(k)$, we reach

$$\tilde{\mathbf{y}}(k) = \mathbf{H}_0^\dagger \mathbf{y}(k) - \sum_{l=1}^{L} \mathbf{H}_0^\dagger \mathbf{H}_l \tilde{\mathbf{y}}(k-l). \tag{6.5}$$

Based on (6.5), we can compute $\tilde{\mathbf{y}}(k)$ in an autoregressive (AR) manner, similar to that in [20]. When calculating $\{\tilde{\mathbf{y}}(k)\}_{k=0}^{N-1}$ from (6.5), we suppose $\tilde{\mathbf{y}}(-1), \ldots, \tilde{\mathbf{y}}(-L)$

[1]The undesired noise enhancement effect can be alleviated by an algorithm enhancement strategy we will present in Chapter 7, where an ill-conditioned channel may trigger an automatic repeat request (ARQ).

are all zeros, reflecting no data transmission prior to $t = 0$ and no inter-block interference.

It is worth pointing out that $\tilde{\mathbf{y}}(k)$ in (6.5) satisfies (6.2) exactly only when $N_o = N_i$. When $N_o > N_i$, (6.5) is the least square solution to $\tilde{\mathbf{y}}(k)$, given $\tilde{\mathbf{y}}(k-1), \ldots, \tilde{\mathbf{y}}(k-L)$. Similarly, the exact noise term $\tilde{\mathbf{v}}(k)$ cannot be found from (6.3) when $N_o > N_i$. Rather, we will use its least-square approximation in the form

$$\tilde{\mathbf{v}}(k) = \mathbf{H}_0^{\dagger}\mathbf{v}(k) - \sum_{l=1}^{L} \mathbf{H}_0^{\dagger}\mathbf{H}_l\tilde{\mathbf{v}}(k-l), \tag{6.6}$$

which also follows an AR structure similar to (6.5). It is the approximate $\tilde{\mathbf{y}}(k)$ and $\tilde{\mathbf{v}}(k)$ constructed from (6.5) and (6.6) that will be used in (6.4) during local PDA MUD. Despite of the approximations involved, the resulting Kalman-PDA I algorithm will be shown (via simulations) to have competitive error performance at slightly lower complexity compared to the Kalman-PDA II algorithm later developed in Section 6.2, which will not involve any approximation during the construction of its Kalman state equation.

6.1.2 Formulation of Kalman Equations

In order to carry out PDA MUD on (6.4), we need to know the conditional mean and variance of $\tilde{\mathbf{v}}(k)$ in (6.6) given the measurement $\tilde{\mathbf{y}}(k)$. This task can be fulfilled by a Kalman filter. We will lay out the formulation of Kalman state equation and measurement equation in this subsection.

To enable KF, it is necessary to convert (6.6) into a first-order linear dynamic system. To this end, we define an $N_iL \times 1$ state vector $\mathbf{x}(k)$ in the form of

$$\mathbf{x}(k) := \begin{bmatrix} \tilde{\mathbf{v}}(k) \\ \tilde{\mathbf{v}}(k-1) \\ \vdots \\ \tilde{\mathbf{v}}(k-L+1) \end{bmatrix}. \tag{6.7}$$

Defining $\tilde{\mathbf{H}}_l := -\mathbf{H}_0^{\dagger}\mathbf{H}_l$, (6.6) can be expressed as a state equation by

$$\mathbf{x}(k) = \mathbf{F}\mathbf{x}(k-1) + \mathbf{G}\mathbf{v}(k) \tag{6.8}$$

where

$$\mathbf{F} = \begin{bmatrix} \tilde{\mathbf{H}}_1 & \tilde{\mathbf{H}}_2 & \cdots & \tilde{\mathbf{H}}_{L-1} & \tilde{\mathbf{H}}_L \\ \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I} & \mathbf{0} \end{bmatrix} \qquad \mathbf{G} = \begin{bmatrix} \mathbf{H}_0^{\dagger} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \tag{6.9}$$

The measurement equation corresponding to (6.4) is thus given by

$$\tilde{\mathbf{y}}(k) = \mathbf{C}_1\mathbf{x}(k) + \mathbf{b}(k) \tag{6.10}$$

where $\mathbf{C}_1 = [\mathbf{I} \;\; \mathbf{0} \cdots \mathbf{0}]$ is an $N_i \times LN_i$ matrix, and $\mathbf{b}(k)$ appears as the measurement noise. In this set of KF equations, $\mathbf{F}$ and $\mathbf{G}$ can be easily obtained because the channel $\mathbf{H}$ is known; $\mathbf{v}(k)$ is the Gaussian noise vector with zero mean and covariance $\mathbf{R}_v$; and the measurement noise $\mathbf{b}(k)$ can be substituted by its tentative soft decisions obtained from the local MUD detector, when iterations between detection and tracking are in

place. With the above state and measurement equation formulation, we can utilize the standard Kalman update procedure to obtain the linearly optimum estimate of the state variable, whose first element is the desired $\tilde{\mathbf{v}}(k)$.

6.1.3 Kalman Update Procedure

With the state and measurement equations formulated in (6.8) and (6.10), we can utilize the standard Kalman update procedure to obtain the linearly optimum estimate of the state variable, whose first element is the desired $\tilde{\mathbf{v}}(k)$ required by local PDA detection in (6.4). Following the notational convention in KF, we let $\hat{\mathbf{x}}(k|k-1)$ denote the prediction of $\mathbf{x}(k)$ from past measurements $\tilde{\mathbf{y}}(k-1)$, $\tilde{\mathbf{y}}(k-2)$,$\cdots$, and let $\hat{\mathbf{x}}(k|k)$ denote the updated estimate of $\mathbf{x}(k)$ given measurements $\tilde{\mathbf{y}}(k)$, $\tilde{\mathbf{y}}(k-1)$, $\cdots$. Likewise, $\mathbf{P}(k|k-1)$ is the predicted covariance estimate of $\mathbf{x}(k)$ from $\tilde{\mathbf{y}}(k-1)$, $\tilde{\mathbf{y}}(k-2)$, $\cdots$, and $\mathbf{P}(k|k)$ is the updated covariance matrix of $\mathbf{x}(k)$ estimated from $\tilde{\mathbf{y}}(k)$, $\tilde{\mathbf{y}}(k-1)$, $\cdots$. The overall equalization and detection procedure is outlined in Table 6.1, while the equations involved are elaborated below.

6.1.3.1 Initialization

Since the information-bearing symbol transmission begins at $k = 0$, we can initialize the estimate and covariance of $\mathbf{x}(k)$ by

$$\hat{\mathbf{x}}(0|-1) \;=\; \mathbf{0}, \tag{6.11}$$

Table 6.1: Update Procedure for Kalman-PDA-I

Initialization	Set $\hat{\mathbf{x}}(0\|-1)$ and $\mathbf{P}(0\|-1)$; initialize the conditional probabilities as $P_i(k) = 0.5, \forall i, k$;
Kalman interference tracking	*for* $k = 0 : N-1$ – perform iterative local PDA MUD on (6.4) based on $\hat{\mathbf{x}}(k\|k-1)$ and $\mathbf{P}(k\|k-1)$, which yields the converged posterior probabilities $\{P_i(k)\}, \forall i$, and the noise statistics $\boldsymbol{\mu}(k)$ and $\boldsymbol{\Sigma}(k)$ in (6.13), (6.14); – decide $\hat{\mathbf{b}}(k)$ from $\{P_i(k)\}$, and compute the mean $\boldsymbol{\mu}_b(k)$ and covariance $\boldsymbol{\Sigma}_b(k)$ of $\mathbf{b}(k)$ via (6.15), (6.16); – compute the Kalman gain matrix $\mathbf{M}(k)$ via (6.17), and estimates $\hat{\mathbf{x}}(k\|k)$ and $\mathbf{P}(k\|k)$ via (6.18) and (6.19); – compute the predicted $\hat{\mathbf{x}}(k+1\|k)$ and $\mathbf{P}(k+1\|k)$ via (6.20) and (6.21); *end* k

$$\mathbf{P}(0|-1) \;=\; \sigma^2\mathbf{G}\mathbf{G}^T. \tag{6.12}$$

6.1.3.2 Iteration

For any $k \in [0, N-1]$, $\hat{\mathbf{x}}(k|k-1)$ and $\mathbf{P}(k|k-1)$ can be obtained either from initialization or the last round of iteration. Since the interested ISI term $\tilde{\mathbf{v}}(k)$ is the first sub-block of $\mathbf{x}(k)$, its mean and covariance statistics can be deduced from the first (diagonal) sub-blocks of $\hat{\mathbf{x}}(k|k-1)$ and $\mathbf{P}(k|k-1)$, respectively. This step is mathematically expressed by:

$$\boldsymbol{\mu}(k) \;=\; E\{\tilde{\mathbf{v}}(k)\} = \mathbf{C}_1\hat{\mathbf{x}}(k|k-1), \tag{6.13}$$

$$\boldsymbol{\Sigma}(k) \;=\; Cov\{\tilde{\mathbf{v}}(k)\} = \mathbf{C}_1\mathbf{P}(k|k-1)\mathbf{C}_1^T. \tag{6.14}$$

Based on the Gaussian forcing principle, the mean and covariance of $\tilde{\mathbf{v}}(k)$ are sufficient statistics to enable local PDA detection on (6.4). The output of the PDA detector includes the symbol decision $\hat{\mathbf{b}}(k)$ and the mean and covariance of $\mathbf{b}(k)$, $\boldsymbol{\mu}_b(k)$ and $\boldsymbol{\Sigma}_b(k)$, which are given by:

$$\boldsymbol{\mu}_b(k) \;=\; \sum_{i=1}^{N_i}(2P_{bi}(k)-1)\mathbf{e}_i, \tag{6.15}$$

$$\boldsymbol{\Sigma}_b(k) \;=\; \sum_{i=1}^{N_i}4P_{bi}(k)(1-P_{bi}(k))\mathbf{e}_i\mathbf{e}_i^T. \tag{6.16}$$

Let $\mathbf{M}(k)$ denote the Kalman gain matrix. Since $\mathbf{b}(k)$ acts as the measurement noise, its associated statistics $\boldsymbol{\mu}_b(k)$ and $\boldsymbol{\Sigma}_b(k)$ can be used in updating $\hat{\mathbf{x}}(k|k)$ and

$\mathbf{P}(k|k)$ from $\hat{\mathbf{x}}(k|k-1)$ and $\mathbf{P}(k|k-1)$. Thus, we can use the standard Kalman update procedures to compute the state estimate and covariance. The following KF equations describe the updating steps in Table 6.1, listed by their implementation order:

We first compute the Kalman gain matrix,

$$\mathbf{M}(k) = \mathbf{P}(k|k-1)\mathbf{C}_1^T[\mathbf{\Sigma}(k) + \mathbf{\Sigma}_b(k)]^{-1} \tag{6.17}$$

Then we update the state estimate and state covariance,

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{M}(k)\{\tilde{\mathbf{y}}(k) - \boldsymbol{\mu}(k) - \boldsymbol{\mu}_b(k)\} \tag{6.18}$$

$$\mathbf{P}(k|k) = [\mathbf{I} - \mathbf{M}(k)\mathbf{C}_1]\mathbf{P}(k|k-1) \tag{6.19}$$

Next, we compute the state prediction and state prediction covariance,

$$\hat{\mathbf{x}}(k+1|k) = \mathbf{F}\hat{\mathbf{x}}(k|k) \tag{6.20}$$

$$\mathbf{P}(k+1|k) = \mathbf{F}\mathbf{P}(k|k)\mathbf{F}^T + \sigma^2\mathbf{G}\mathbf{G}^T \tag{6.21}$$

Once (6.20) and (6.21) are obtained, the noise-interference statistics of $\tilde{\mathbf{v}}(k+1)$ can be predicted from (6.13) and (6.14), which enable local PDA MUD on $\mathbf{b}(k+1)$ in the next iteration step. The overall flow of the Kalman-PDA I algorithm is illustrated in Fig. 6.1.

6.2 The Kalman-PDA II Algorithm

In this subsection, we will briefly introduce an alternative way to construct the Kalman equations and thus the updating procedure. This approach intuitively groups

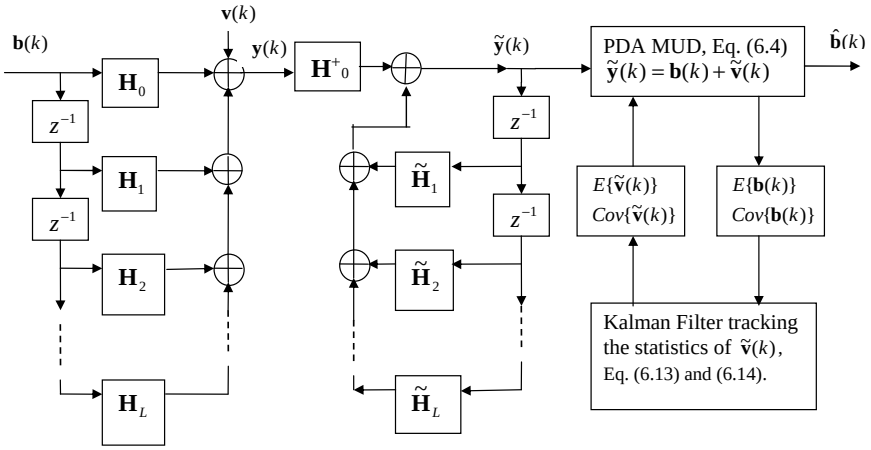

Figure 6.1: Diagram of Kalman-PDA I

all the ISI and noise components to form a new vector, $\bar{\mathbf{v}}(k)$, $k = 0, \cdots, N-1$, whose mean and covariance are then tracked by a Kalman filter. It does not require the autoregressive computing of the approximate $\tilde{\mathbf{y}}(k)$.

6.2.1 Local MUD Structure

At the k-th sampling instant, the L-tap FIR MIMO-ISI channel model can be written as:

$$\mathbf{y}(k) = \mathbf{H}_L \mathbf{b}(k-L) + \mathbf{H}_{L-1}\mathbf{b}(k-L+1) + \cdots + \mathbf{H}_0 \mathbf{b}(k) + \mathbf{v}(k). \tag{6.22}$$

Targeting detection of $\mathbf{b}(k)$ only from $\mathbf{y}(k)$, we naturally construct a composite noise term $\bar{\mathbf{v}}(k)$ in the form of

$$\bar{\mathbf{v}}(k) = \mathbf{H}_L \mathbf{b}(k-L) + \mathbf{H}_{L-1}\mathbf{b}(k-L+1) + \cdots + \mathbf{H}_1 \mathbf{b}(k-1) + \mathbf{v}(k) \tag{6.23}$$

Thus (6.22) is simplified to

$$\mathbf{y}(k) = \mathbf{H}_0\mathbf{b}(k) + \bar{\mathbf{v}}(k), \tag{6.24}$$

or equivalently,

$$\bar{\mathbf{y}}(k) := \mathbf{H}_0^\dagger\mathbf{y}(k) = \mathbf{b}(k) + \mathbf{H}_0^\dagger\bar{\mathbf{v}}(k) \tag{6.25}$$

upon which we can use the PDA detector to detect $\mathbf{b}(k)$ locally, after $\bar{\mathbf{v}}(k)$ is tracked. Note that we use HiperLAN-2 as the reference wireless application in our work [57]. The tap power of the power-delay profile decreases as the delay increases. This implies that $\mathbf{H}_0$ on the average has the largest norm, compared with other channel tap matrices. Hence we detect $\mathbf{b}(k)$ from $\mathbf{y}(k)$ in Kalman-PDA II. On the other hand, a general MIMO channel does not always have its first MIMO tap as the strongest one. In this case, $\mathbf{b}(k)$ might not be the best candidate to be extracted from $\mathbf{y}(k)$. For example, suppose $\mathbf{H}_D$ has the highest norm, then $\mathbf{b}(k-D)$ is the best candidate to be detected from $\mathbf{y}(k)$ first. In such a case, the Kalman state and measurement equations can be rearranged, and similar approaches can be used. More complicated user ordering strategies are also possible [26].

6.2.2 Formulation of Kalman Equations

To estimate the second-order statistics of $\bar{\mathbf{v}}(k)$, we construct an $N_o(L+1) \times 1$ state vector $\bar{\mathbf{x}}(k)$ in the form of

$$\bar{\mathbf{x}}(k) = \begin{bmatrix} \mathbf{H}_L \mathbf{b}(k-L) \\ \mathbf{H}_{L-1} \mathbf{b}(k-L+1) \\ \vdots \\ \mathbf{H}_1 \mathbf{b}(k-1) \\ \mathbf{v}(k) \end{bmatrix}. \tag{6.26}$$

Defining $\mathbf{C} := [\mathbf{I} \;\; \mathbf{I} \cdots \mathbf{I}]$, we relate $\bar{\mathbf{x}}(k)$ with the ISI term of interest, $\bar{\mathbf{v}}(k)$, by

$$\bar{\mathbf{v}}(k) = \mathbf{C}\bar{\mathbf{x}}(k), \tag{6.27}$$

By the definition of $\bar{\mathbf{x}}(k)$, a natural state equation emerges:

$$\bar{\mathbf{x}}(k+1) = \mathbf{\Phi}\bar{\mathbf{x}}(k) + \mathbf{\Gamma}_1 \mathbf{b}(k) + \mathbf{\Gamma}_2 \mathbf{v}(k+1) \tag{6.28}$$

where

$$\mathbf{\Phi} := \begin{bmatrix} \mathbf{0} & \mathbf{H}_L\mathbf{H}_{L-1}^{\dagger} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{H}_{L-1}\mathbf{H}_{L-2}^{\dagger} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{H}_2\mathbf{H}_1^{\dagger} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \end{bmatrix}_{(L+1)N_o \times (L+1)N_o},$$

$$\mathbf{\Gamma}_1 := \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{H}_1 \\ \mathbf{0} \end{bmatrix}_{(L+1)N_o \times N_i}, \quad \mathbf{\Gamma}_2 := \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{I} \end{bmatrix}_{(L+1)N_o \times N_o}$$

Substituting (6.27) into (6.25), we reach the measurement equation

$$\bar{\mathbf{y}}(k) = \mathbf{H}_0^\dagger \mathbf{C}\bar{\mathbf{x}}(k) + \mathbf{b}(k) = \mathbf{C}_2\bar{\mathbf{x}}(k) + \mathbf{b}(k), \tag{6.29}$$

where we define $\mathbf{C}_2 = \mathbf{H}_0^\dagger \mathbf{C}$, and $\mathbf{b}(k)$ is treated as measurement noise in this setup. Eq. (6.28) is a standard Kalman state equation with an input and plant noise, where the input is what is to be detected, $\mathbf{b}(k)$. The value of $\mathbf{b}(k)$, as well as its mean and covariance, are needed in the Kalman update. We use the *a priori* statistics to do the PDA detection, and then use the output of the PDA detector to update the Kalman estimates.

6.2.3 Kalman Equation Updating

The updating procedure is similar to that of Kalman-PDA-I. We summarize the procedure in Table 6.2 and give the equations used as follows.

Prior to the k-th local PDA MUD, the ISI term in (6.25) is tracked by the $(k|k-1)$ prediction:

$$\boldsymbol{\mu}(k) = E\{\mathbf{H}_0^\dagger \bar{\mathbf{v}}(k)\} = \mathbf{C}_2\hat{\bar{\mathbf{x}}}(k|k-1) \tag{6.30}$$

Table 6.2: Update Procedures for Kalman-PDA-II

Initialization	set $\hat{\hat{\mathbf{x}}}(0\|-1) = \mathbf{0}$, $\mathbf{P}(0\|-1) = \sigma^2\mathbf{\Gamma}_2\mathbf{\Gamma}_2^T$; initialize the conditional probabilities as $P_i(k) = 0.5, \forall i, k$;
Kalman interference tracking	*for* $k = 0 : N-1$ – Perform iterative local PDA MUD on (6.25), which yields the converged posterior probabilities $\{P_i(k)\}, \forall i$, and the noise statistics $\boldsymbol{\mu}$ and $\mathbf{\Sigma}$ in (6.30) and (6.31); – decide $\hat{\mathbf{b}}(k)$ from $\{P_i(k)\}$, and compute the covariance of $\mathbf{b}(k)$ via (6.32); – compute the Kalman gain matrix $\mathbf{M}_2(k)$ and then $\hat{\hat{\mathbf{x}}}(k\|k)$ and $\mathbf{P}(k\|k)$ via (6.33), (6.34) and (6.35); – compute the predicted $\hat{\hat{\mathbf{x}}}(k+1\|k)$ and $\mathbf{P}(k+1\|k)$ using (6.36), (6.37); *end* k

$$\boldsymbol{\Sigma}(k) \;=\; Cov\{\mathbf{H}_0^{\dagger}\bar{\mathbf{v}}(k)\} = \mathbf{C}_2\mathbf{P}(k|k-1)\mathbf{C}_2^T \tag{6.31}$$

The variance of $\mathbf{b}(k)$, $\boldsymbol{\Sigma}_2(k)$, is computed from the *a priori* or tentative probabilities $P_{bi}(k)$, $\forall i$:

$$\boldsymbol{\Sigma}_2(k) = \sum_{i=1}^{N_i} 4P_{bi}(k)(1-P_{bi}(k))\mathbf{e}_i\mathbf{e}_i^T \tag{6.32}$$

The Kalman Gain matrix is given by

$$\mathbf{M}_2(k) = \mathbf{P}(k|k-1)\mathbf{C}_2^T[\boldsymbol{\Sigma}(k)+\boldsymbol{\Sigma}_2(k)]^{-1} \tag{6.33}$$

From the $(k|k-1)$ prediction to the $(k|k)$ estimate, we have

$$\hat{\bar{\mathbf{x}}}(k|k) \;=\; \hat{\bar{\mathbf{x}}}(k|k-1) + \mathbf{M}_2(k)(\bar{\mathbf{y}}(k) - \boldsymbol{\mu}(k) - \hat{\mathbf{b}}(k)) \tag{6.34}$$

$$\mathbf{P}(k|k) \;=\; [\mathbf{I} - \mathbf{M}_2(k)\mathbf{C}_2]\mathbf{P}(k|k-1) \tag{6.35}$$

From the $(k|k)$ estimate to the $(k+1|k)$ prediction:

$$\hat{\bar{\mathbf{x}}}(k+1|k) \;=\; \boldsymbol{\Phi}\hat{\bar{\mathbf{x}}}(k|k) + \boldsymbol{\Gamma}_1\hat{\mathbf{b}}(k) \tag{6.36}$$

$$\mathbf{P}(k+1|k) \;=\; \boldsymbol{\Phi}\mathbf{P}(k|k)\boldsymbol{\Phi}^T + \sigma^2\boldsymbol{\Gamma}_2\boldsymbol{\Gamma}_2^T \tag{6.37}$$

The overall flow of the Kalman-PDA II algorithm is illustrated in Fig. 6.2.

6.3 Complexity Analysis

For the $KN_o \times NN_i$ channel matrix modeled in Chapter 2, the proposed Kalman-PDA detection algorithms enjoy polynomial-time low complexity in contrast to an exact ML solution with exponential-time complexity. Here we give a brief evaluation of the complexity order of Kalman-PDA.

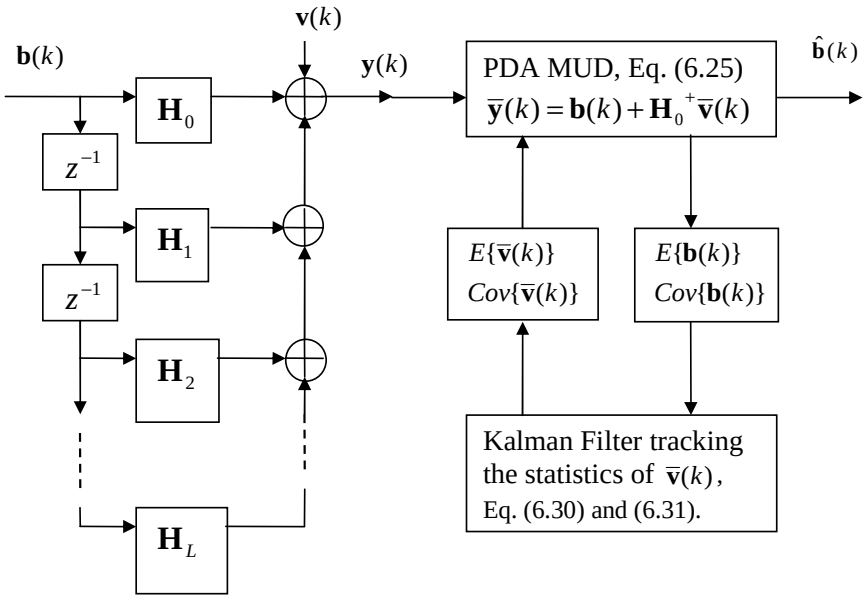

Figure 6.2: Diagram of Kalman-PDA II

When detecting each $N_i \times 1$ sub-block in the $NN_i \times 1$ block, the Kalman-PDA I algorithm goes through three major steps: computing $\tilde{\mathbf{y}}(k)$, local PDA MUD and Kalman updating. The complexity involved for computing $\tilde{\mathbf{y}}(k)$ and the local PDA are on the orders of $\mathcal{O}((LN_i)^2)$ and $\mathcal{O}(N_i^3)$, respectively. In Kalman updating, the highest order of computational burden results from computing $\mathbf{P}(k|k)$ and $\mathbf{P}(k+1|k)$, which is on the order of $\mathcal{O}((LN_i)^3)$. Since there are N sub-blocks, the overall complexity of Kalman-PDA I for detecting NN_i symbols is on the order of $\mathcal{O}\{N(LN_i)^3\}$. Similarly, we deduce that the complexity of the Kalman-PDA II is $\mathcal{O}\{N(L+1)^3N_o^3\}$.

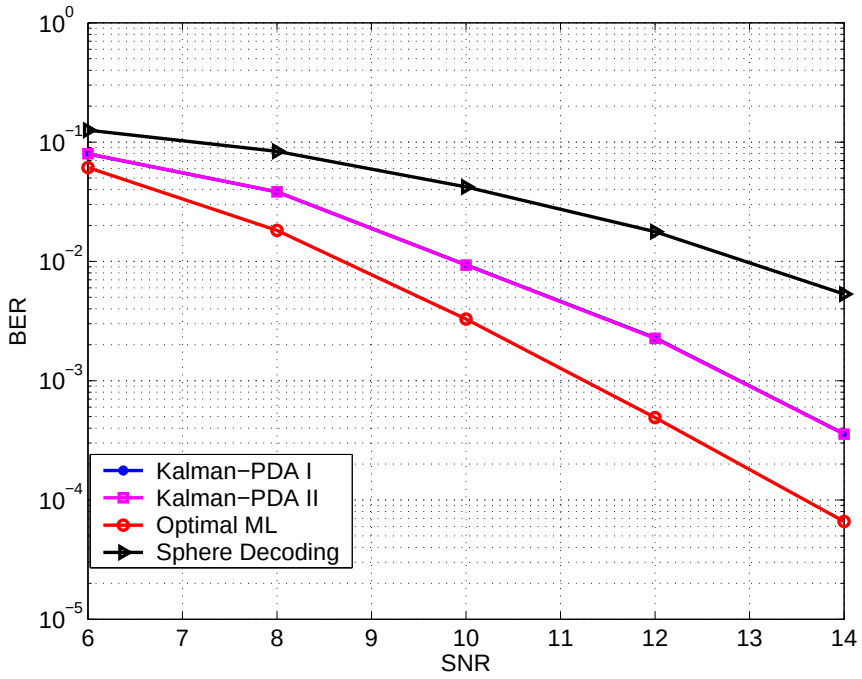

Figure 6.3: Performance Comparison in an overloaded CDMA system

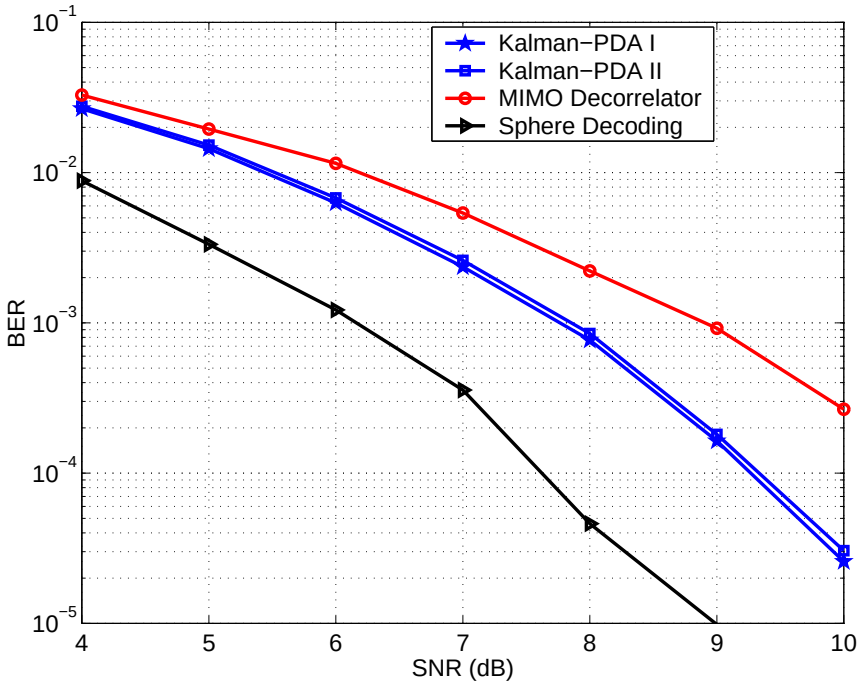

Figure 6.4: Performance Comparison in a 5-tap FIR MIMO system

6.4 Simulations

We simulate the Kalman-PDA I and II algorithms in order to test their performance. We use sphere decoding (SD) as a near-optimal performance benchmark. In all tests, the symbol block size is set to be[2] $N = 12$. The MIMO channels are assumed to be invariant within each symbol block, but are allowed to change independently from block to block. Monte Carlo simulations run 100,000 times, with a new channel realization generated in each simulation. The performance metric of interest is the bit-error-rate (BER) versus the bit energy to noise ratio E_b/N_o (SNR).

We first simulate both Kalman-PDA algorithms in an overloaded CDMA system. There are 10 users, each using a spreading code length of 6 and BPSK modulation. The channel model is the discrete-time asynchronous CDMA model described in [50], whereas the signature correlation matrices are generated according to [58]. The time delays of asynchronous users are random and uniformly distributed within a symbol duration. The spreading codes are used at the receiver to despread the received waveform and generate symbol-rate sampled data. We can see from Fig. 6.3 that Kalman-PDA outperforms SD in the overloaded case. In the next scenario, we assume a 5-tap FIR MIMO channel with one transmit antenna and four receive antennas. The channel tap coefficients of each link are generated independently from Rayleigh distributions with equal power, and the instantaneous tap gains are sorted

[2]We choose a relatively small block size N so that the simulation time is manageable for ML detection, which is depicted for reference.

in a descending order[3]. From Fig. 6.4, we can see that Kalman-PDA outperforms the MIMO zero-forcing decorrelator by 1 dB. We also notice that there is about 1.5 dB gap between Kalman-PDA and SD, which makes Kalman-PDA less attractive. The comparison between Kalman-PDA and SD in the above simulations confirms the assessment that the more accurate the Gaussian-forcing assumption is, the better Kalman-PDA performs. In an overloaded system, the sum of the interference and noise can be well approximated as Gaussian by the law of large numbers; as a result, both of the proposed Kalman-PDA algorithms render better performance than SD. On the other hand, in a FIR MIMO system where the Gaussian assumption is less accurate, the performance of Kalman-PDA may suffer, as seen in Fig. 6.4. Seeking to alleviate this performance gap, we are motivated to devise some techniques to improve the BER performance of Kalman-PDA. This is the topic of the next chapter.

6.5 Summary

For frequency selective MIMO channels, we have presented in this chapter low-complexity soft-decision equalization techniques based on a two-part structure: local MUD using soft-decision PDA detection, and dynamic noise-interference tracking using Kalman filtering. These two parts are combined naturally in our Kalman-PDA approach, thanks to the common thread of second-order statistics estimation. By constructing properly the state vector to incorporate the dynamic ISI information,

[3]Sorting the channel tap gains makes it convenient for simulations. However, this step is not necessary, nor is it realistic. In practice under known channel knowledge, the receiver may alternatively adjust the detection order by detecting $\mathbf{b}(k-D)$ from $\mathbf{y}(k)$, as suggested in Section 6.2.

two types of Kalman filters are designed, both of which are able to track the ISI in an FIR MIMO channel of any tap length. The computational complexity of Kalman-PDA is less than $(L/N)^3 \times 100\%$ that of sphere decoding, which is quite notable for a large block size N.

CHAPTER 7

Performance Enhancement Techniques

The basic idea of performance enhancement stems from the soft decisions generated by PDA MUD. According to the soft-decision principle of PDA in a BPSK system [2], [26], the detector computes the *a posteriori* probabilities of a given symbol, i.e., $P_i(k) = Pr(b_i(k) = 1|\tilde{\mathbf{y}}(k))$ in (6.4). Each symbol $b_i(k)$ is then decided to be 1 if $P_i(k)$ is greater than or equal to 0.5, or -1 otherwise. We observe in the simulations that most $P_i(k)$'s converge to the vicinity of either 1 or 0. Wrong decisions occur typically when $P_i(k)$ is relatively close to 0.5. This makes sense intuitively: if $P_i(k)$ is very close to 1 or 0, the detector tends to have strong "confidence" in the decision and the decision is reliable; on the other hand, if $P_i(k)$ is closer to 0.5 rather than 1 or 0, this means that the decision is less reliable and might be in error. This observation suggests that soft-decisions have the capability of indicating their own reliability. In contrast, decisions made in hard-decision techniques do not possess this property, because the reliability information is lost during quantization. Taking advantage of this unique property, we devise two performance enhancement techniques, automatic repeat request (ARQ) aided enhancement for uncoded transmissions and coding-aided enhancement for coded systems. We will use Kalman-PDA as an example to show

that both algorithms noticeably enhance the detection performance of Kalman-PDA itself. For brevity, the enhancement discussed hereinafter is based on Kalman-PDA I. However, all principles can be applied to Kalman-PDA II and other soft-decision based detection algorithms in a straightforward manner.

7.1 Kalman-PDA with ARQ

In Kalman-PDA with ARQ, the receiver computes a threshold, $\gamma_i(k)$, for each received bit $b_i(k)$ according to a given retransmission rate, R_{re}. If the absolute value of the *a posteriori* mean, $|E\{b_i(k)|\tilde{\mathbf{y}}(k)\}|$, is less than the corresponding threshold, this bit will be marked as an "error-suspect." It is termed an "error-suspect" because it is relatively more liable to be a detection error. We use an ARQ protocol according to which a data block should be retransmitted whenever it contains an error-suspect. We assume that there exists a reliable feedback channel from the receiver to the transmitter so that the repeat request can be sent to the transmitter via this channel. The retransmission ends when there is no error-suspect in the latest transmitted data block. We develop a systematic procedure to compute the threshold for a given retransmission rate. With this procedure, the retransmission rate can be preset to an arbitrary value to ensure a desired performance and data-rate tradeoff.

We define the retransmission rate as

$$R_{re} = \frac{N_{re}}{N_f}, \tag{7.1}$$

where N_f is the number of data blocks to be transmitted and N_{re} is the number of retransmissions incurred when transmitting N_f data blocks. Let us define α as the probability of current block containing at least one error-suspect. Because retransmission occurs as long as there is at least one error-suspect in the received block according to the ARQ protocol, we have

$$\alpha = \frac{R_{re}}{1 + R_{re}}. \tag{7.2}$$

We assume that all bits in a block are independent and equally liable to be an error-suspect. Letting β denote the probability of an error-suspect and N_b the data block length, the probability of an error-liable block is related to an error-liable bit as, $\alpha = 1 - (1 - \beta)^{N_b}$, i.e.,

$$\beta = 1 - (1 - \alpha)^{\frac{1}{N_b}}. \tag{7.3}$$

Using (7.2) and (7.3), the bit error-suspect probability β can be computed for any given retransmission rate, R_{re}. Next, we will show how to compute the threshold $\gamma_i(k)$ using β.

For the purpose of illustration and derivation simplicity, we consider a BPSK model with one transmit antenna. The extension to more than one antenna case can be derived by analogy. With this assumption, (6.4) is reduced to

$$\tilde{y}(k) = b(k) + \tilde{v}(k), \quad k = 0, 1, \cdots, N - 1. \tag{7.4}$$

All terms in (7.4) are scalars, and $\tilde{v}(k)$ is assumed to be Gaussian with mean zero and variance σ^2. The conventional ML decision rule for (7.4) is

$$\hat{b}(k) = \begin{cases} 1, & \tilde{y}(k) > 0; \\ -1, & \tilde{y}(k) < 0. \end{cases} \tag{7.5}$$

The probability of detection error is given by $Q(1/\sigma)$, where $Q(x) = 1/\sqrt{2\pi}\int_x^\infty e^{-t^2/2}dt$ is the complementary error function. After introducing error-suspects, the decision rule is now

$$\hat{b}(k) = \begin{cases} 1, & \tilde{y}(k) > y_c; \\ e, & -y_c < \tilde{y}(k) < y_c; \\ -1, & \tilde{y}(k) < -y_c; \end{cases} \tag{7.6}$$

where $y_c > 0$ is the threshold in terms of the amplitude of the received signal, and e stands for an error-suspect. If we take measures to remove the error-suspects and prevent them from becoming errors, the probability of detection error is reduced to $Q((1 + y_c)/\sigma)$. Because the Q-function is monotonically decreasing, we can see that introducing error-suspect does help to reduce the detection error probability.

Based on (7.6), β is given by the probability that $\{\hat{b}(k) = e\}$ occurs. With β known from (7.1)-(7.3), we can compute y_c by numerically solving

$$\beta = Q(\frac{1 - y_c}{\sigma}) - Q(\frac{1 + y_c}{\sigma}). \tag{7.7}$$

If we assume equi-probable transmissions, the corresponding soft-decision threshold γ_p generated by PDA is given by an one-to-one mapping from y_c:

$$\gamma_p \triangleq \frac{Pr(b(k) = 1|y_c)}{Pr(b(k) = -1|y_c)} = \frac{Pr(y_c|b(k) = 1)}{Pr(y_c|b(k) = -1)} = \exp(2\sigma^{-2}y_c). \tag{7.8}$$

In the simulations, we apply this ARQ-aided enhancement technique to Kalman-PDA. it is more convenient to set up a decision rule in terms of the absolute value of the soft-decision mean, $|\mu_b(k)|$. This is because $\mu_b(k)$ is furnished to the Kalman tracker by PDA so that it is easily obtained. The decision rule is

$$\hat{b}(k) = \begin{cases} 1, & \mu_b(k) > \gamma_b; \\ e, & -\gamma_b < \mu_b(k) < \gamma_b; \\ -1, & \mu_b(k) < -\gamma_b; \end{cases} \tag{7.9}$$

where $\gamma_b = \frac{\gamma_p - 1}{\gamma_p + 1}$. γ_b is guaranteed to be greater than zero because γ_p is greater than one in (7.8).

So far we have established a systematic way to compute the decision threshold γ_b (or γ_p) for a given retransmission rate. In Fig. 7.1, we simulate this ARQ-aided Kalman-PDA with the same system setup as in Fig. 6.4. The performance of Kalman-PDA is enhanced considerably by using ARQ. The larger the retransmission rate, the better the performance. Nevertheless, the benefit of ARQ becomes more prominent for higher SNRs: ARQ retransmissions result in faster BER fall-off slopes and thus larger diversity orders compared with sphere decoding. Kalman-PDA with 5% retransmission rate is not as good as sphere decoding until the SNR reaches 8dB. This is because retransmissions are still unreliable and contain error-suspects when the SNR is low.

In Fig. 7.1, we did not take into account the fact that retransmission costs extra signal energy and reduces the spectral efficiency. In order to present a fair comparison among Kalman-PDA, Kalman-PDA with ARQ and sphere decoding, we plot their

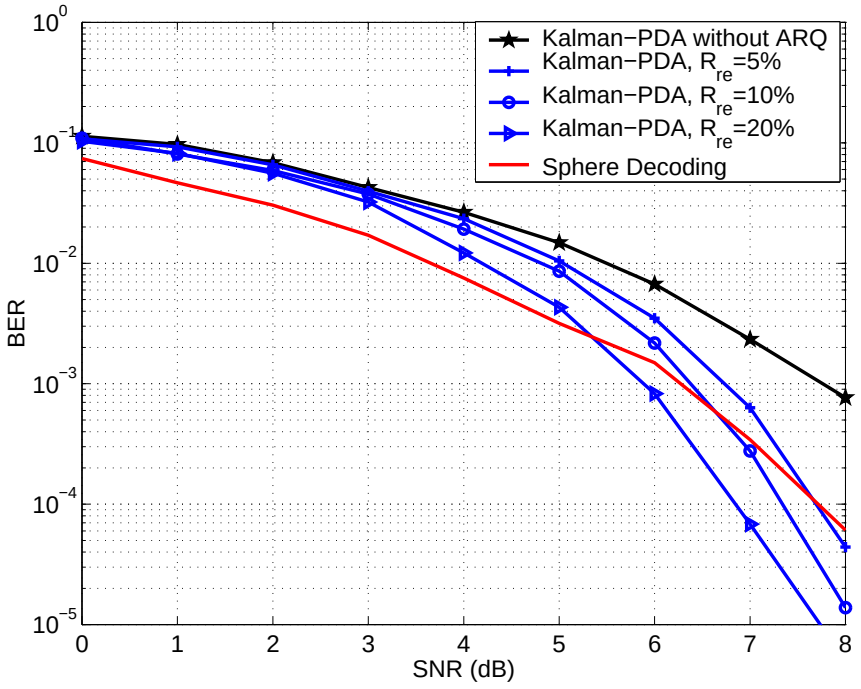

Figure 7.1: Kalman-PDA with ARQ

BER performances in Fig. 7.2, using the same effective data rate. The retransmission rate in Kalman-PDA with ARQ is set to 20% as an example. Since Kalman-PDA and SD do not perform retransmission, their curves in Fig. 7.2 are basically the same as that in Fig. 7.1. On the other hand, retransmissions in Kalman-PDA with ARQ reduce the effective power efficiency, which can be reflected by adjusting the corresponding SNRs. In Fig. 7.2, the simulated BER curve of Kalman-PDA with ARQ is shifted to the right by $10 \log_{10} 1.2 = 0.79$dB. We can observe that Kalman-PDA with ARQ still performs better than without ARQ, given the same effective data rate and energy cost. As SNR increases, Kalman-PDA with ARQ becomes competitive to or even better than SD.

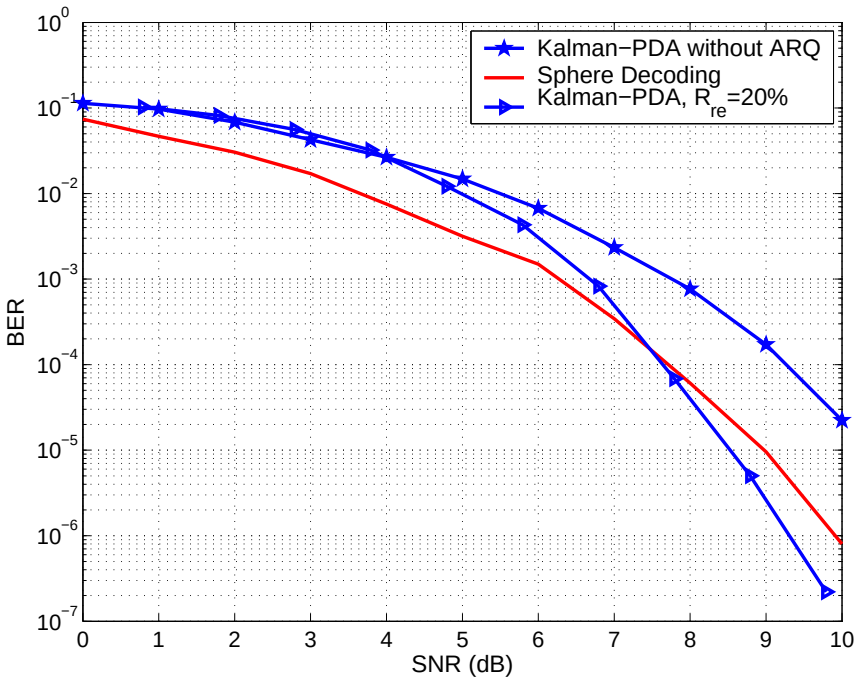

Figure 7.2: Performance comparison given same effective data rates

The average threshold is plotted versus the retransmission rate in Fig. 7.3. We can see that for a given SNR, the higher the retransmission rate R_{re}, the higher the threshold. This suggests that a higher threshold will mark more error-suspects and thus requires more retransmissions. On the other hand, for a certain R_{re}, the thresholds corresponding to higher SNRs are greater than that of lower SNRs. This is because the *a posteriori* probabilities converge closer to 1 or 0 in the high SNR region, in which case thresholds must be set more stringently in order to distinguish the error-suspects from the reliable detections. We also plot in Fig. 7.4 the average retransmission rate at different SNRs. We can see the simulated rates match well with the preset rates.

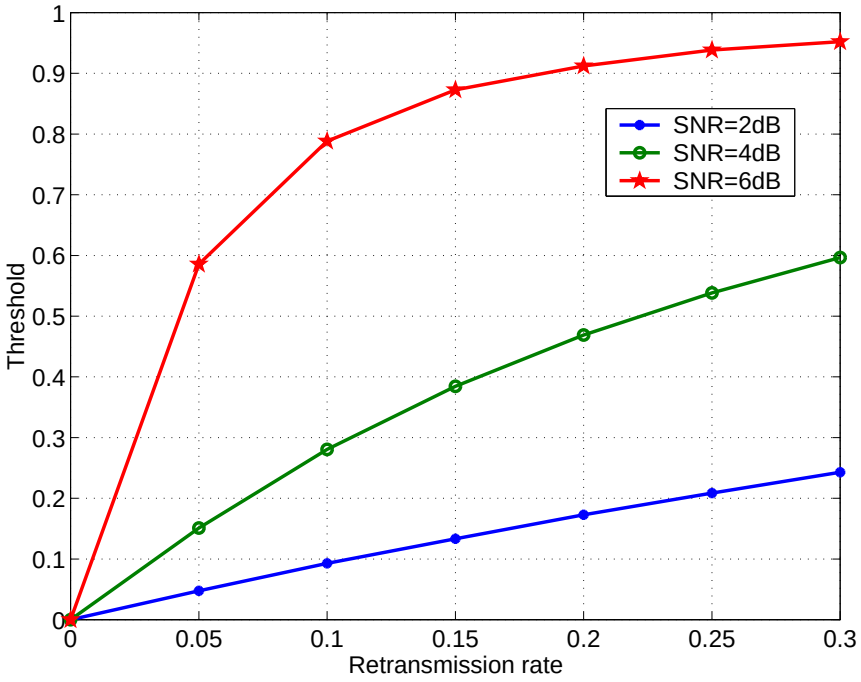

Figure 7.3: Threshold vs. retransmission rate

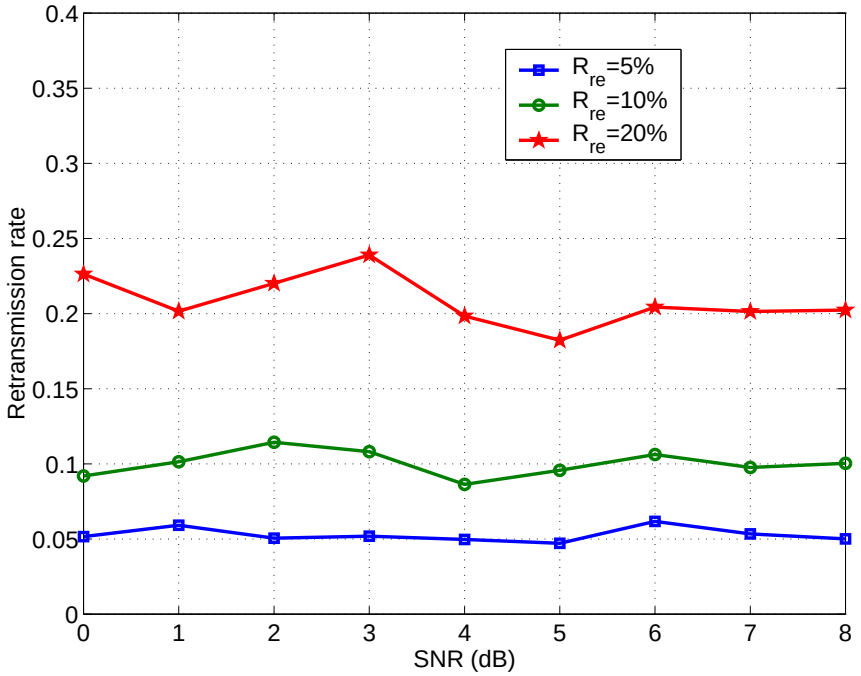

Figure 7.4: Average retransmission rate

7.2 Coding-aided Kalman-PDA

The ARQ-aided enhancement technique improves the BER performance by invoking a data-link layer protocol, which increases the system complexity. Alternatively, in coded systems, the BER performance can be enhanced solely within the physical layer via code puncturing. In this section, we derive a coding-aided enhancement technique for coded systems. It can boost the BER performance of Kalman-PDA to approach the performance of sphere decoding with coding.

The system structure of a coded system is shown in Fig. 7.5, where linear block codes are used to fit our block transmission scheme. Suppose a binary information sequence is denoted by $\mathbf{a} = [a_1, a_2, \cdots, a_{k_c}]$, $a_i = \{0, 1\}, i = 1, \cdots, k_c$. It is encoded using a rate $R_c = k_c/n_c$ ($k_c < n_c$) linear block code to generate a n_c-tuple codeword (or code vector) $\mathbf{c} = [c_1, c_2, \cdots, c_{n_c}]$, $c_i = \{0, 1\}, i = 1, \cdots, n_c$. There are 2^{k_c} codewords corresponding to 2^{k_c} k_c-long information sequences. For a specific linear block code, $\mathbf{c}$ can be computed for any given $\mathbf{a}$ via

$$\mathbf{c} = \mathbf{aG}, \tag{7.10}$$

where $\mathbf{G}$ is the $k_c \times n_c$ generator matrix. The codeword $\mathbf{c}$ is then modulated by antipodal signaling, and the modulated signal, $\mathbf{b}$, is transmitted through the FIR MIMO channel. At the receiver, Kalman-PDA outputs the detection of the modulated symbols $\hat{\mathbf{b}}$, which is in turn demodulated to the binary sequence $\hat{\mathbf{c}}$. Given $\hat{\mathbf{c}}$, the maximum-likelihood decoding (MLD) rule selects the optimal codeword $\mathbf{c}_a$ from a

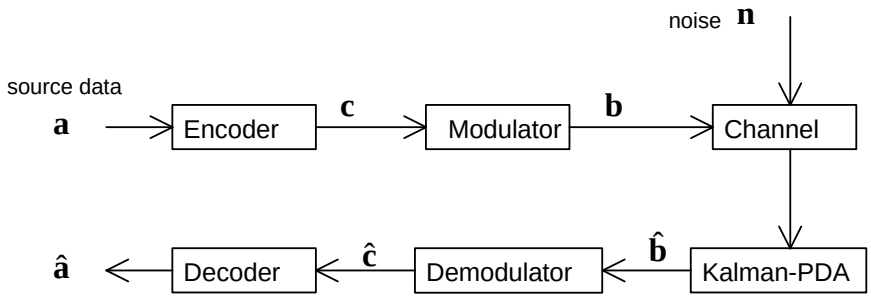

Figure 7.5: Block diagram of code-aided Kalman-PDA

codeword set $\mathbf{C}$ via

$$\mathbf{c}_a = \arg\min_{\forall \mathbf{c}\in\mathbf{C}} ||\mathbf{c} - \hat{\mathbf{c}}|| = \arg\min_{\forall \mathbf{c}\in\mathbf{C}} \sum_{i=1}^{n_c} |c_i - \hat{c}_i|, \tag{7.11}$$

where $||\cdot||$ denotes the Hamming distance. We can see that MLD always decodes $\hat{\mathbf{c}}$ to the codeword with the smallest Hamming distance. The final output of the decoder, $\hat{\mathbf{a}}$, is the data sequence that satisfies

$$\hat{\mathbf{a}}\mathbf{G} = \mathbf{c}_a. \tag{7.12}$$

MLD can decode correctly as long as the Hamming distance between $\mathbf{c}$ and the correct codeword is smaller than that of any other codeword. However, when some bits in $\hat{\mathbf{c}}$ are not reliable, the decoding performance will suffer. This can be mitigated by using erasure decoding. The erasure decoding approach marks the unreliable bits with "e", and these bits do not count when computing the Hamming distance. These erasures are the counterpart of the "error-suspects" in Kalman-PDA with ARQ. They are picked for having relatively lower absolute mean, $|E\{b_i|\tilde{\mathbf{y}}(k)\}|$. If we define $\mathbf{e}_M = [e_1, e_2, \cdots, e_{n_c}]$, $e_i \in \{0, 1\}$ as the erasure mask, the decoding rule is now

$$\mathbf{c}_a = \arg\min_{\forall \mathbf{c}\in\mathbf{C}} \sum_{i=1}^{n_c} e_i|c_i - \hat{c}_i|, \tag{7.13}$$

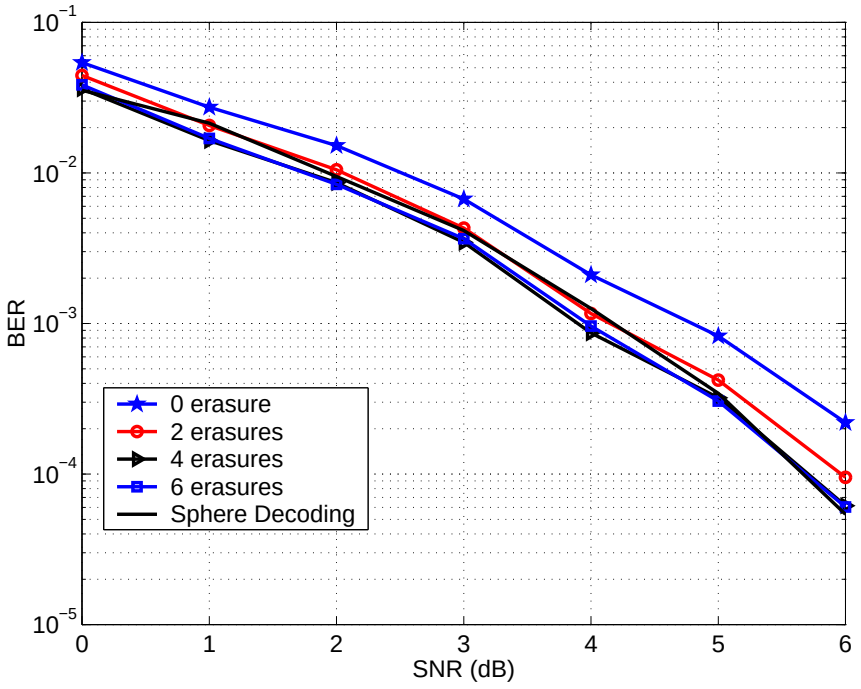

Figure 7.6: Coding-aided Kalman-PDA

where $e_i = 0$ indicates the ith bit is an erasure.

A code with a minimum distance d_{min} can correct up to $d_{min} - 1$ erasures [51]. If there are $d_{min} - 1$ erasures or less in $\hat{\mathbf{c}}$, and all other bits are reliable, $\hat{\mathbf{c}}$ can still be decoded to the right word using (7.13). It can be shown (see Appendix) that $\hat{\mathbf{c}}$ can be always decoded to the correct codeword provided that all the unreliable bits are covered by erasures. Under the same constraint, it follows that the decoding performance associated with more erasures can be no worse than that with fewer erasures. Therefore, codes with a larger d_{min} have an advantage over those with smaller d_{min} in terms of their erasure correction capability. Nevertheless, a code that can accommodate more erasures usually has a lower code rate, which means lower data transmission efficiency.

We simulate the performance of code-aided Kalman-PDA to investigate the benefit of erasure coding. In the simulation, we choose the BCH(15, 5) ($n_c = 15, k_c = 5$) code that has a minimum distance of 7; hence it can correct at most 6 erasures. The generator matrix is given in [51]. We simulate the cases of $N_e = 0, 2, 4$ and 6 erasures, where $N_e = 0$ means there is no erasure. The system setup is the same as Fig. 6.4 except that N is set to 15 to fit the codeword length. Among N soft-decisions corresponding to the N bits in a received codeword, we mark the N_e decisions that have the smallest $|E\{b_i|\tilde{\mathbf{y}}(k)\}|$ as erasures. The simulation result is plotted in Fig. 7.6. We use sphere decoding (SD) with the same encoder as the performance benchmark. The effective data rates of both algorithms are the same. SD is not enhanced with erasure decoding though, because it is impossible to pick error-prone erasure bits from the hard-decisions generated by SD. In contrast, erasure-marking is a handy by-product of the soft-decision process used in Kalman-PDA, without incurring much extra computation. Without erasure decoding, Kalman-PDA does not perform as well as sphere decoding. However, with the aid of erasure decoding, its performance improves noticeably as N_e increases. Kalman-PDA approaches SD when $N_e = 2$ and outperforms SD when N_e reaches 4 or more. We also note that $N_e = 6$ does not render significant improvement over $N_e = 4$, which means 6 erasures do not necessarily cover more unreliable bits than 4 erasures.

As a final remark on coded systems, we note that the decoding approach proposed in the paper in essence improves the performance of hard-decision decoding and keeps the complexity low. It makes use of $1-$bit reliability information at the decoder. On

the other hand, the standard decoding methods for block codes such as syndrome decoding and Massey algorithm do not utilize this reliability information. Nevertheless, the principle of the proposed decoder can be applied to any general block decoder.)

7.3 Summary

Two performance enhancement techniques have been presented in this chapter for soft-decision based symbol detection algorithms: ARQ aided enhancement and coding aided enhancement. We use Kalman-PDA as an example and construct Kalman-PDA with cross-layer ARQ for uncoded systems as well as coding-aided Kalman-PDA for coded systems. Simulations show that both techniques can render error performance that is better than Kalman-PDA alone and competitive to sphere decoding.

CHAPTER 8

SDE with Imperfect Channel Knowledge

In previous algorithm development and discussions, we have assumed that the receiver has perfect knowledge of the channel. Both SDE and Kalman-PDA are developed based on such an assumption, and satisfactory performance is achieved. Although this assumption has been largely adopted in the literature, it is only an ideal scenario. In a practical MIMO system, the receiver can not have perfect channel knowledge. What the receiver has is only an estimate of the actual channel. The estimate is typically obtained by transmitting a block of training symbols. Through processing the received training sequences, the receiver can reach estimated knowledge of the underlying channel, which is imperfect. There is enormous research interest in joint channel and data decoding recently. For coded transmission systems, detection decisions output from the decoder are used in lieu of additional tranining sequences to update the channel estimate. The decisions obtained from decoding can be either hard or soft [59] – [67].

A recursive least square (RLS) based iterative channel estimation combined with MIMO minimum mean square (MMSE) equalizer was proposed in [68]. This algorithm utilizes MMSE to perform channel equalization. In our research, we use SDE

to do symbol detection and RLS to do channel estimation. We follow the uncoded base band transmission structure established in Chapter 2. We focus on the scenario in which the transmitter transmits a training sequence, and the receiver uses the RLS method to estimate the channel, and obtains a "coarse" initial channel state information (CSI). After the training sequence, information bearing symbols are transmitted. In each sampling period, the receiver first uses the previously obtained channel knowledge to perform soft-decision symbol detection, then feeds the detection results to the RLS channel estimator to refine the channel knowledge. We chose RLS to performance channel estimation due to the following considerations:

1. For every new piece of data, we only need to update the channel estimate rather than to recompute the estimate over again, the latter of which is needed in the least square (LS) method. Thus, RLS saves considerable computation compared with LS.

2. RLS is a well-established algorithm that works fairly well. The SDE method can be blended with RLS channel estimation harmoniously. The overall performance loss due to the imperfect channel knowledge is small, which will be shown later in the simulations.

The rest part of this chapter is organized as follows. The MIMO system setup with training sequences is described in Section 8.1. The RLS channel estimation algorithm is developed in Section 8.2. Simulation results of SDE with RLS channel estimation is presented in Section 8.3.

8.1 MIMO System with Training Sequence

In this section we briefly describe the setup of the MIMO system with training sequences. The transmitter first transmit N_T training symbols followed by L padding zeros. Then the information-bearing transmission begins. We still adopt the block transmission structure used in Chapter 2. The information-bearing symbols are parsed into N-long blocks with the insertion of L zeros at the end of each block. We assume the channel remains invariant for N_B consecutive symbol blocks, but may vary from one N_B-block cluster to another.

As in previous chapters, the channel has $L+1$ FIR taps, and each link is modeled as Rayleigh fading. For each information-bearing symbol cluster, there is a preceding N_T long training symbol block. BPSK modulation scheme is used throughout this paper.

8.2 SDE+RLS Channel Estimation Algorithm Development

The basic structure of the algorithm contains two parts. First, use the training symbol sequence to obtain a "coarse" estimate of the channel. This is fulfilled by the RLS channel estimation method. Next, perform iterative SDE+RLS channel estimation for each information-bearing symbol block. The coarse channel estimate obtained from the training sequence is used to compute a tentative decision on the symbols transmitted. This tentative decision is then fed to the RLS channel estimator as

new data to update the channel estimate. The updated output of the RLS channel estimator is then fed to SDE again, and an updated symbol detection is computed. This iteration continues until a pre-defined cost metric can not be lowered any more. This iteration process takes place within each sampling period. The whole process is illustrated in Fig. 8.1.

Next, we derive the LS equations for MIMO channel estimation, and then migrate to RLS MIMO channel estimation. The FIR MIMO channel model in Chapter 2 can be depicted mathematically as follows:

$$\mathbf{y}(k) = \sum_{l=0}^{L} \mathbf{H}_l \mathbf{b}(k-l) + \mathbf{n}(k), \quad k = 0, 1, 2, \cdots \tag{8.1}$$

where $\mathbf{y}(k)$, $\mathbf{b}(k)$ and $\mathbf{n}(k)$ are the received signal vector, transmitted signal vector and white Gaussian noise vector, respectively. $\{\mathbf{H}_l\}_{l=0}^{L}$ are the FIR MIMO channel taps. In the channel estimation phase, $\{\mathbf{H}_l\}_{l=0}^{L}$ is unknown, and a training sequence is transmitted to facilitate estimation. so the objective of LS channel estimation is to compute the estimation of the channel taps, $\{\mathbf{H}_l\}_{l=0}^{L}$, given a set of know inputs $\mathbf{b}(-L), \cdots, \mathbf{b}(0), \cdots, \mathbf{b}(N-1)$ and the corresponding received signals, $\mathbf{y}(0), \cdots, \mathbf{y}(N-1)$.

If we define

$$\mathcal{Y} := [\mathbf{y}(0), \mathbf{y}(1), \cdots, \mathbf{y}(N-1)] \tag{8.2}$$

$$\mathcal{H} := [\mathbf{H}_0, \mathbf{H}_1, \cdots, \mathbf{H}_L], \tag{8.3}$$

$$\mathcal{B} := \begin{bmatrix} \mathbf{b}(0) & \mathbf{b}(1) & \cdots & \mathbf{b}(N-1) \\ \mathbf{b}(-1) & \mathbf{b}(0) & \cdots & \mathbf{b}(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{b}(-L) & \mathbf{b}(-L+1) & \cdots & \mathbf{b}(N-L-1) \end{bmatrix} = [\mathbf{B}(0), \mathbf{B}(1), \cdots, \mathbf{B}(N-1)], \tag{8.4}$$

$$\mathcal{N} := [\mathbf{n}(0), \mathbf{n}(1), \cdots, \mathbf{n}(N-1)] \tag{8.5}$$

Then the batched form of (8.1) can be expressed as:

$$\mathcal{Y} = \mathcal{H}\mathcal{B} + \mathcal{N}. \tag{8.6}$$

We denote the estimate of the channel taps $\{\hat{\mathbf{H}}\}_{l=0}^{L}$, and define $\hat{\mathcal{Y}} := \hat{\mathcal{H}}\mathcal{B}$. The difference between $\mathcal{Y}$ and $\hat{\mathcal{Y}}$ is denoted as:

$$\mathcal{E} := \mathcal{Y} - \hat{\mathcal{Y}}. \tag{8.7}$$

This can be seen as the estimation error. The LS method is to find the channel estimate that minimizes the sum of the square errors, i.e.,

$$\hat{\mathcal{H}} = \arg\min_{\mathcal{H}} Tr\{\mathcal{E}^T\mathcal{E}\}. \tag{8.8}$$

In order to obtain $\mathcal{H}$ that satisfies (8.8), we compute the derivative of $Tr\{\mathcal{E}^T\mathcal{E}\}$ with respect to $\hat{\mathcal{H}}$. We have

$$\begin{aligned} & \frac{\partial}{\partial\hat{\mathcal{H}}}\{Tr(\mathcal{E}^T\mathcal{E})\} \\ = \; & \frac{\partial}{\partial\hat{\mathcal{H}}}\{Tr[(\mathcal{Y}^T - \hat{\mathcal{Y}}^T)(\mathcal{Y} - \hat{\mathcal{Y}})]\} \\ = \; & \frac{\partial}{\partial\hat{\mathcal{H}}}\{Tr(\mathcal{Y}^T\mathcal{Y} - \hat{\mathcal{Y}}^T\mathcal{Y} - \mathcal{Y}^T\hat{\mathcal{Y}} + \hat{\mathcal{Y}}^T\hat{\mathcal{Y}})\}. \end{aligned} \tag{8.9}$$

In (8.9), $\mathcal{Y}^T\mathcal{Y}$ is independent of $\hat{\mathcal{H}}$, which implies that

$$\frac{\partial}{\partial\hat{\mathcal{H}}}\{Tr(\mathcal{Y}^T\mathcal{Y})\} = \mathbf{0}. \tag{8.10}$$

In the following discussions, we consider a real-valued transmission-detection scenario at the baseband. Correspondingly, we have

$$\begin{aligned} Tr(\hat{\mathcal{Y}}^T\mathcal{Y}) &= Tr(\mathcal{Y}^T\hat{\mathcal{Y}}) & (8.11)\\ Tr(\hat{\mathcal{Y}}^T\hat{\mathcal{Y}}) &= Tr(\hat{\mathcal{Y}}\hat{\mathcal{Y}}^T). & (8.12) \end{aligned}$$

Thus, (8.9) can be reduced to

$$\begin{aligned} & \frac{\partial}{\partial\hat{\mathcal{H}}}\{Tr(\mathcal{E}^T\mathcal{E})\} \\ = & \frac{\partial}{\partial\hat{\mathcal{H}}}\{-2Tr(\mathcal{Y}^T\hat{\mathcal{Y}}) + Tr(\hat{\mathcal{Y}}\hat{\mathcal{Y}}^T)\} \\ = & \frac{\partial}{\partial\hat{\mathcal{H}}}\{-2Tr(\mathcal{Y}^T\hat{\mathcal{H}}\mathcal{B}) + Tr(\hat{\mathcal{H}}\mathcal{B}\mathcal{B}^T\hat{\mathcal{H}}^T)\} \end{aligned} \tag{8.13}$$

Noting the following properties of the matrix trace [69],

$$\begin{aligned} \frac{\partial}{\partial\mathbf{X}}\{Tr(\mathbf{A}^T\mathbf{X}\mathbf{B}^T)\} &= \mathbf{A}\mathbf{B} & (8.14)\\ \frac{\partial}{\partial\mathbf{X}}\{Tr(\mathbf{X}\mathbf{A}\mathbf{X}^T)\} &= \mathbf{X}(\mathbf{A}+\mathbf{A}^T) & (8.15) \end{aligned}$$

we further simplify (8.13) to

$$\frac{\partial}{\partial\hat{\mathcal{H}}}\{Tr(\mathcal{E}^T\mathcal{E})\} = -2\mathcal{Y}\mathcal{B}^T + 2\hat{\mathcal{H}}\mathcal{B}\mathcal{B}^T. \tag{8.16}$$

At the minimum of $Tr\{\mathcal{E}^T\mathcal{E}\}$, we have

$$\frac{\partial}{\partial\hat{\mathcal{H}}}\{Tr(\mathcal{E}^T\mathcal{E})\} = \mathbf{0}. \tag{8.17}$$

Substituting (8.16) into above, we have

$$\hat{\mathcal{H}} = \mathcal{Y}\mathcal{B}^T(\mathcal{B}\mathcal{B}^T)^{-1} \tag{8.18}$$

Recalling $\mathcal{Y} = [\mathbf{y}(0), \mathbf{y}(1), \cdots, \mathbf{y}(N-1)]$, $\mathcal{B} = [\mathbf{B}(0), \mathbf{B}(1), \cdots, \mathbf{B}(N-1)]$, and defining

$$\boldsymbol{\theta} = \mathcal{Y}\mathcal{B}^T = \sum_{i=0}^{N-1} \mathbf{y}(i)\mathbf{B}(i)^T, \tag{8.19}$$

$$\boldsymbol{\Phi} = \mathcal{B}\mathcal{B}^T = \sum_{i=0}^{N-1} \mathbf{B}(i)\mathbf{B}(i)^T, \tag{8.20}$$

we have

$$\hat{\mathcal{H}} = \boldsymbol{\theta}\boldsymbol{\Phi}^{-1}. \tag{8.21}$$

Thus we have derived the algorithm to compute the channel estimate in the LS sense. The problem of the LS solution is that $\boldsymbol{\Phi}^{-1}$ has to be recomputed whenever a new data piece is available. This becomes cumbersome when the block length is large, since the computational load is on the third order of the matrix size. Recursive least square (RLS) algorithm has been widely used [70] to reduce the complexity in computing $\boldsymbol{\Phi}^{-1}$. The basic idea of RLS is to form a recursive update for the weights so that we do not have to recompute $\boldsymbol{\Phi}^{-1}$ for each new data piece. This will greatly reduce the computational load, and lower the complexity order by 1.

In RLS channel estimation, $\boldsymbol{\theta}$ and $\boldsymbol{\Phi}$ are defined as functions of n, where n is the timing index, i.e.,

$$\boldsymbol{\Phi}(n) = \sum_{i=0}^{n} \lambda^{n-i}\mathbf{B}(i)\mathbf{B}(i)^T, \tag{8.22}$$

$$\boldsymbol{\theta}(n) = \sum_{i=0}^{n} \lambda^{n-i}\mathbf{y}(i)\mathbf{B}(i)^T. \tag{8.23}$$

$\lambda \in (0, 1]$ is the forgetting factor, which is usually close to 1. Thus $\boldsymbol{\theta}(n)$ and $\boldsymbol{\Phi}(n)$ can be updated recursively,

$$\begin{aligned} \boldsymbol{\Phi}(n) &= \sum_{i=1}^{n} \lambda^{n-i} \mathbf{B}(i)\mathbf{B}(i)^T && (8.24) \\ &= \lambda \sum_{i=1}^{n-1} \lambda^{(n-1)-i} \mathbf{B}(i)\mathbf{B}(i)^T + \mathbf{B}(n)\mathbf{B}(n)^T && (8.25) \\ &= \lambda \boldsymbol{\Phi}(n-1) + \mathbf{B}(n)\mathbf{B}(n)^T. && (8.26) \end{aligned}$$

Likewise,

$$\boldsymbol{\theta}(n) = \lambda \boldsymbol{\theta}(n-1) + \mathbf{y}(i)\mathbf{B}(i)^T. \tag{8.27}$$

Using the matrix inversion lemma [70], we deduce from (8.26) that,

$$\boldsymbol{\Phi}(n)^{-1} = \lambda^{-1}\boldsymbol{\Phi}(n-1)^{-1} - \lambda^{-2} \frac{\boldsymbol{\Phi}(n-1)^{-1}\mathbf{B}(n)\mathbf{B}(n)^T\boldsymbol{\Phi}(n-1)^{-1}}{1 + \lambda^{-1}\mathbf{B}(n)^T\boldsymbol{\Phi}(n-1)^{-1}\mathbf{B}(n)}. \tag{8.28}$$

Define

$$\begin{aligned} \mathbf{P}(n) &:= \boldsymbol{\Phi}(n)^{-1} && (8.29) \\ \mathbf{k}(n) &:= \frac{\lambda^{-1}\mathbf{P}(n-1)\mathbf{B}(n)}{1 + \lambda^{-1}\mathbf{B}(n)^T\mathbf{P}(n-1)\mathbf{B}(n)} && (8.30) \end{aligned}$$

where $\mathbf{k}(n)$ is conventionally called the gain vector. Eq. (8.28) can be rewritten as

$$\mathbf{P}(n) = \lambda^{-1}\mathbf{P}(n-1) - \lambda^{-1}\mathbf{k}(n)\mathbf{B}(n)^T\mathbf{P}(n-1). \tag{8.31}$$

Given a new pair of data piece, the RLS channel estimation algorithm can be implemented via updating the gain vector. This can be summarized as follows:

1. Compute the new gain vector,

$$\mathbf{k}(n) = \frac{\lambda^{-1}\mathbf{P}(n-1)\mathbf{B}(n)}{1 + \lambda^{-1}\mathbf{B}(n)^T\mathbf{P}(n-1)\mathbf{B}(n)}, \tag{8.32}$$

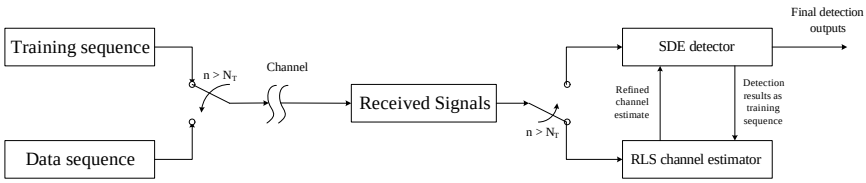

Figure 8.1: SDE + RLS joint symbol detection and channel estimation

2. Update the inverse correlation matrix,

$$\mathbf{P}(n) = \lambda^{-1}\mathbf{P}(n-1) - \lambda^{-1}\mathbf{k}(n)\mathbf{B}(n)^T\mathbf{P}(n-1). \tag{8.33}$$

3. Update

$$\boldsymbol{\theta}(n) = \lambda\boldsymbol{\theta}(n-1) + \mathbf{y}(n)\mathbf{B}(n)^T. \tag{8.34}$$

4. Compute

$$\hat{\mathcal{H}}(n) = \boldsymbol{\theta}(n)\mathbf{P}(n). \tag{8.35}$$

So far we have developed the RLS channel estimation algorithm. It is blended with SDE in the way shown in Fig. 8.1. The simulation results are presented in the next section.

8.3 Simulation Results: SDE+RLS Channel Estimation

We simulate the aforementioned SDE + RLS channel estimation algorithm in order to test its performance. We consider a 5-tap FIR MIMO channel with two transmit antennas and four receive antennas. The channel tap coefficients of each link are generated independently from Rayleigh distribution with equal power. We simulate 2,000 channel realizations for each SNR. For each channel realization, N_T-long

training symbols are transmitted to compute a "coarse" channel estimate. The RLS channel estimation algorithm presented in the previous section is used to compute this estimate. The N_T-long training symbols are followed by L trailing zeros to eliminate ISI. Then N_B information-bearing, zero-padded symbol blocks are transmitted. The channel estimate is first fed to the SDE detector. The decisions output from the SDE detector are utilized by RLS estimator to update the channel estimate until convergence is reached. Then the updated channel estimate is used to do SDE detection on next symbol block.

In Fig. 8.2, we plot the performance of SDE with perfect channel knowledge as a benchmark. The gap between SDE + RLS channel estimation and SDE + perfect channel is about 1dB. We also plot the SDE with the coarse channel estimate for the purpose of comparison. The coarse channel estimate is the estimate based only on the N_T training symbol vectors. It is not updated in the following SDE detection. We can see that the performance of SDE with coarse channel estimate is inferior to SDE + RLS channel estimation by more than 2dB. It shows that the channel estimate is improved over decision-feedback updating.

In the simulations we also adjust the values N_T and N_B, and observe the following:

1. It is possible to feed either hard decisions or soft decisions at the output of SDE to the RLS channel estimator. In the simulations, we try both cases. It makes little difference in the sense of BER performance whether to feed hard decisions or soft decisions to the RLS channel estimator. This is because reliable

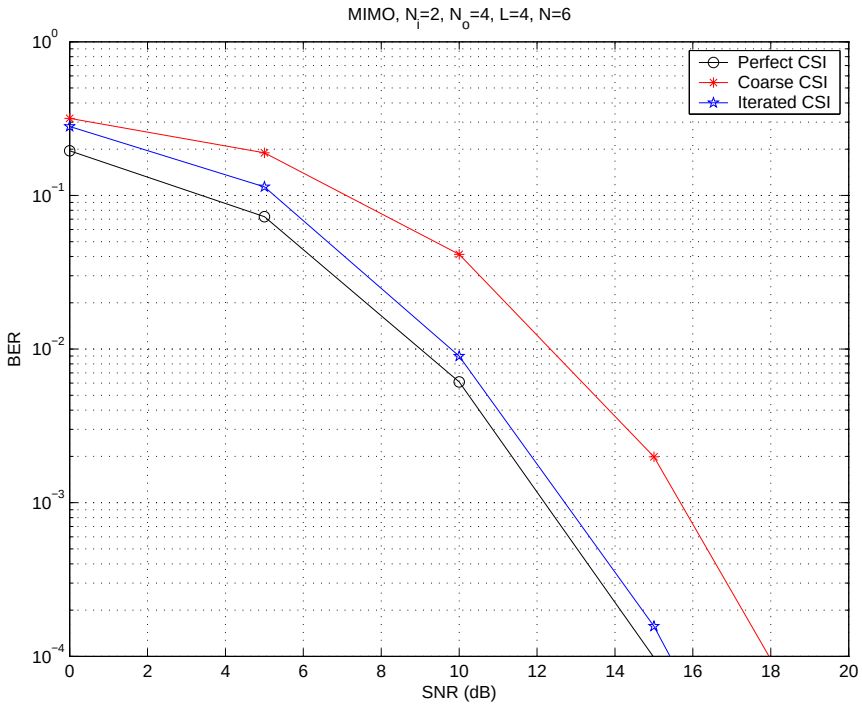

Figure 8.2: SDE + RLS joint symbol detection and channel estimation

soft decisions are very close to +1/-1. The difference is little between reliable decisions and the correct hard decisions.

2. The longer the training sequence, the better the coarse performance.

3. The longer the information-bearing data sequence, the better the iterated performance, especially for higher SNRs. For higher SNRs, the detections are fairly accurate, they are used to direct RLS, which acts as training symbols.

Fig. 8.2 shows the scenario of $N_T = 20$ and $N_B = 50$.

CHAPTER 9

Summary and Future Research

9.1 Summary

Our work focuses on exploiting the advantages of soft-decision based detection and equalization algorithms for frequency selective MIMO multipath channels. Relying on iterative posterior probability updating and PDA-type Gaussian forcing, the proposed SDE algorithms attain remarkable near-ML performance at low complexity that is polynomial (on the 3rd-order) in the input and output sizes, and linear in the modulation constellation size. Unlike existing MUD algorithms, our development for MIMO channel equalization relies on zero-padded block transmission to enable block detection for a sequence detection problem, and capitalizes on the distinct Toeplitz channel structure to simplify the equalization complexity. Near-optimum symbol detection in the presence of channel memory is attained by virtue of soft-decision MAP multiuser detection, multistage ISI cancellation, and implicit noise tracking. SDE algorithms also apply to rank-deficient channels, provided that the channels are identifiable for the signal constellation. In addition, we have also presented in this dissertation low-complexity soft-decision Kalman-PDA equalization techniques based

on a two-part structure: local MUD using soft-decision PDA detection, and dynamic noise-interference tracking using Kalman filtering. These two parts are combined naturally in our Kalman-PDA approach, thanks to the common thread of second-order statistics estimation. By constructing properly the state vector to incorporate the dynamic ISI information, two types of Kalman filters are designed, both of which are able to track the ISI in an FIR MIMO channel of any tap length. The computational complexity of Kalman-PDA is less than $(L/N)^3 \times 100\%$ that of sphere decoding, which is quite notable for a large block size N. We further offer two algorithms to enhance the BER performance of soft-decision based detection algorithms: cross-layer ARQ and the code-aided approach. We take Kalman-PDA as an example and show that both performance enhancement algorithms can render BER performance much better than the original Kalman-PDA. Finally, we present an iterative channel estimate algorithm for the case where perfect CSI is not available at the receiver. Utilizing training sequences, SDE with iterative channel estimation achieves BER performance which approaches that of SDE with perfect CSI.

9.2 Future Research

In this section, we briefly lay out possible future research directions to extend the work presented in this dissertation.

9.2.1 Convergence and Performance Analysis of SDE/PDA

Both SDE and Kalman-PDA use iterative updating of *a posteriori* probabilities to achieve near-optimum BER performance. The iteration plays a key role in delivering correct decisions. In the simulations, we observe that the SDE detection performance improves with additional iterations and stabilizes quickly in three to five iterations for high SNR, and seven to 14 iterations for low SNR. Similar observations are found for Kalman-PDA. However, it has never been proved the condition and rate of convergence. It is a very interesting topic to find out under what conditions the process of iterative posterior probabilities updating will converge. It is also important to analytically define the convergence rate given the SNR and channel state information. It is helpful to refer to the convergence analysis of expectation-maximization (EM) problem [71]. The EM algorithm is a well-known iterative algorithm for finding modes of a likelihood function and is characterized by simple implementation and stability [72]. If we can cast the structure of SDE/PDA to a model analgous to EM, we may then provide analysis of the SDE/PDA convergence problem. Other potential approaches include belief propagation methods used in analyzing (soft-decision) turbo codes.

Convergence analysis will also shed light on the steady state performance of the proposed algorithms. Steady state performance helps to understand the performance gap from an optimal (yet impractically complex) MIMO detector, under various operating conditions. These results will in turn help system designers strike a desired performance-complexity tradeoff.

9.2.2 Application in MIMO-OFDM

OFDM [10] is a promising approach that can convert a frequency selective channel into a set of narrow-band parallel subchannels. Each subchannel experiences flat fading, thus alleviating the need for channel equalization. OFDM has been used in digital audio and video broadcasting in Europe [73]. It is also an attractive scheme for future high data rate wireless communications. MIMO-OFDM can further improve system performance and capacity. In [10], MMSE-based successive interference cancellation (SIC) is proposed for signal detection at the receiver. Since we have seen from the simulations that SDE has better BER performance than the MMSE method, we can expect that using SDE for signal detection in MIMO-OFDM systems can render superior performance than what SIC can offer. The feasibility of applying SDE for MIMO-OFDM is confirmed by the generality of our signal model adopted in Section 2.2, which can subsume a MIMO-OFDM system. It is also of interest to compare in the future the detection performance and computational efficiency of an MIMO system with SDE equalization, and an OFDM system with SDE soft detection (but no need for equalization).

APPENDIX A

Erasure Decoding Proof

In Fig. 7.5, $\hat{\mathbf{c}}$ denotes the demodulated binary sequence. Given a code with a minimum distance d_{min}, it can correct up to $d_{min} - 1$ erasures. So if there are $d_{min} - 1$ or less erasures in $\hat{\mathbf{c}}$, and all other bits are reliable, $\hat{\mathbf{c}}$ can still be decoded to the right codeword. We would like to show here that a received data sequence can always be decoded to the correct codeword by using maximum-likelihood decoding (MLD) if we assume that all the unreliable bits are covered by erasures. Suppose $\mathbf{c}_1$ and $\mathbf{c}_2$ are two $1 \times n_c$ codewords, where $\mathbf{c}_1$ is the codeword transmitted while $\mathbf{c}_2$ is any other codeword in the set. Let us denote the received codeword as $\hat{\mathbf{c}}$, which contains both reliable and unreliable bits. The erasure mask is given by $\mathbf{e}_m$, in which $e_i = 0$ if the ith bit is unreliable (an erasure). Subject to the erasure correction ability of a given code, the number of erasures in $\mathbf{e}_m$, N_e, should be no more than $d_{min} - 1$. Denoting $\bigodot$ as the element-by-element array multiplication, we can see

$$\mathbf{e}_m \odot \mathbf{c}_1 = \mathbf{e}_m \odot \hat{\mathbf{c}}, \tag{A.1}$$

if all unreliable bits are marked by erasures. As a result the Hamming distance between $\mathbf{e}_m \bigodot \mathbf{c}_1$ and $\mathbf{e}_m \bigodot \hat{\mathbf{c}}$ is 0. Thus, $\hat{\mathbf{c}}$ is decoded correctly. On the other hand, $\mathbf{e}_m \bigodot \mathbf{c}_1$ differs from $\mathbf{e}_m \bigodot \mathbf{c}_2$ by at least 1 bit. This is because $\mathbf{c}_1$ differs from $\mathbf{c}_2$ by

d_{min} bits while $\mathbf{e}_m$ only contains at most $d_{min} - 1$ erasures. Thus, $\hat{\mathbf{c}}$ is decoded to $\mathbf{c}_1$ by MLD and this decoding is unique.

APPENDIX B

Publications

Journal Articles:

1. S. Liu, Z. Tian, "Near-Optimal Soft Decision Equalization for Frequency Selective MIMO Channels," *IEEE Transactions on Signal Processing*, vol. 52, no. 3, pp. 721-733, March 2004.

2. S. Liu, Z. Tian, "A Kalman-PDA Approach to Soft-Decision Equalization for Frequency Selective MIMO Channels," accepted to *IEEE Transactions on Signal Processing*.

Conference papers:

1. S. Liu, Z. Tian, "Performance Analysis of Adaptive Constrained Filtering," *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP'2002)*, Orlando, FL, vol. 4, pp. 4173, May 2002.

2. S. Liu, Z. Tian, "A Soft-Decision Approach for BLAST Systems with Less Receive than Transmit Antennae," *Proceedings of IEEE Global Communications Conference (Globecom'2003)*, San Francisco, CA, vol. 2, pp. 834-838, December 1-5, 2003.

3. S. Liu, Z. Tian, "A Cross-Layer Kalman-PDA Approach to Soft-Decision Equalization for FIR MIMO Channels," *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC'2004)*, Atlanta, GA, vol. 4, pp. 2307-2312, March 21-25, 2004.

4. S. Liu, Z. Tian, "A Kalman-PDA Approach to Soft-Decision Equalization for Frequency Selective MIMO Channels," *Proceedings of IEEE Signal Processing Workshop on Advances in Wireless Communications (SPAWC2004)*, Lisbon, July 11-14, 2004.

5. S. Liu, Z. Tian, "Sliding-window based Soft Decision Equalization for Frequency Selective MIMO Channels," *Proceedings of IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM'2004)*, Barcelona, July 2004.

LIST OF REFERENCES

[1] G. J. Foschini, M. J. Gans, "On the limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Comm.*, no. 6, pp. 315-335, 1998.

[2] Z. Tian, "Soft decision feedback equalization for pre-coded FIR-MIMO systems," *Proceedings of the 2nd IEEE Sensor Array and Multichannel Signal Processing Workshop* (SAM'2002), Rosslyn, VA, August 2002.

[3] A. J. Paulraj, D. A. Gore, R. U. Nabar, and H. Bolcskel, "An overview of MIMO communications – a key to gigabit Wireless," *Proceedings of the IEEE*, vol. 92, no. 2, pp.198-218, February, 2004.

[4] A. J. Paulraj and T. Kailath, "Increasing capacity in wireless broadcast systems using distributed transmission/directional reception," U.S. Patent 5 345 599, 1994.

[5] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs. Tech. J.*, vol. 1, no. 2, pp. 41-59, 1996.

[6] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *Eur. Trans. Telecommun.*, vol. 10, pp. 585-595, Nov. 1999.

[7] H. Bolcskei, D. Gesbert, and A. J. Paulraj, "On the capacity of OFDM-based spatial multiplexing systems," *IEEE Trans. Commun.*, vol. 50, pp. 225-234, Feb. 2002.

[8] P. Lombardo, D. Fedele, M.M. Rao, "MRC performance for binary signals in Nakagami fading with general branch correlations," *IEEE Trans. on Communications*, vol. 47, pp. 44-52, Jan. 1999.

[9] D. Gesbert, M. Shafi, D. Shiu, P. J. Smith, and A. Naguib, "From theory to practice: an overview of MIMO space-time coded wireless systems," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 3, pp. 281-302, April 2003.

[10] Y. Li, J. H. Winters, N. R. Sollenberger, "MIMO-OFDM for wireless communications: signal detection with enhanced channel estimation," *IEEE Trans. on Communications*, vol. 50, no. 9, pp. 1471-1477, Sep. 2002.

[11] A. Duel-Hallen, "Equalizers for multiple-input multiple-output channels and PAM systems with cyclostationary input sequences," *IEEE J. Selected Area in Communications*, vol. 10, pp. 630-639, Sept. 1992.

[12] J. Luo, J. Zeidler, S. McLaughlin, "Performance analysis of compact antenna arrays with MRC in correlated Nakagami fading," *IEEE Trans. Vehicular Technology*, vol. 50, pp. 267- 277, Jan. 2001.

[13] Z. Xie, R. T. Short, and C. K. Rushforth, "A family of sub-optimum detectors for coherent multiuser communications," *IEEE J. Select Areas on Communications*, vol. 8, no. 4, pp. 683-690, May 1990.

[14] W. K. Ma, T. N. Davidson, K. M. Wong, Z. Q. Luo, and P. C. Ching, "Quasi-maximum-likelihood multiuser detection using semi-definite relaxation with application to synchronous CDMA," *IEEE Trans. on Signal Processing*, vol. 50, no. 4, pp. 912-922, Apr. 2002.

[15] M. K. Varansi and B. Aazhang, "Near-optimum detection in synchronous code-division multiple-access systems," *IEEE Trans. Commun.*, vol. 39, pp. 725736, May 1991

[16] I. Sharfer and A. O. Hero, III, "A maximum likelihood digital receiver using coordinate ascent and the discretewavelet transform," *IEEE Trans. Signal Processing*, vol. 47, pp. 813825, Mar. 1999.

[17] L. B. Nelson and H. V. Poor, "Iterative multiuser receivers for CDMA channels: An EM-based approach," *IEEE Trans. Commun.*, vol. 44, pp. 17001710, Dec. 1996.

[18] Y. E. Nesterov, "Quality of Semidefinite Relaxation for Nonconvex Quadratic Optimization," Belgium, CORE Discussion Paper #9719, Mar. 1997.

[19] L.Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Rev.*, vol. 38, pp. 49-95, 1996.

[20] C. Komninakis, C. Fragouli, A. H. Sayed, R. D. Wesel, "Multi-input multi-output fading channel tracking and equalization using Kalman estimation," *IEEE Trans. on Signal Processing*, vol. 50, no. 5, pp. 1065-1076, May 2002.

[21] U. Finkce and M. Pohst, "Improved methods for calculating vectors of short length in lattice, including a complexity analysis," *Mathematics of Computation*, vol. 44, pp. 463–471, Apr. 1985.

[22] S. Liu and Z. Tian, "Near-Optimum Soft Decision Equalization for Frequency Selective MIMO Channels," *IEEE Trans. on Signal Processing*, to appear.

[23] Z. Wang, G. B. Giannakis, "Wireless multicarrier communications - where Fourier meets Shannon," *IEEE Signal Processing Magazine*, pp. 29-48, May 2000.

[24] R. A. Iltis, "Joint estimation of PN code delay and multipath using extended Kalman filter," *IEEE Trans. Commun.,* vol. 38, pp. 1677-1685, Oct. 1990.

[25] D. Pham, J. Luo, K. Pattipati, P. Willett, "A PDA-Kalman approach to multiuser detection in asynchronous CDMA," *IEEE Communications Letters,* vol. 6, no. 11, pp. 475-477, 2002.

[26] J. Luo, K. R. Pattipati, P. K. Willett, F. Hasegawa, "Near-optimal multiuser detection in synchronous CDMA using probabilistic data association," *IEEE Communications Letters,* vol. 5, no. 9, pp. 361-363, 2001.

[27] J. Proakis, *Digital Communications,* New York: McGraw-Hill, 1995.

[28] L. R. Bahl, J. Cocke, F. Jelinek, J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Information Theory*, vol. 44, pp. 744-765, Mar. 1998.

[29] M. V. Eyuboglu, S. U. Qureshi, "Reduced-state sequence estimation with set partitioning and decision feedback," *IEEE Trans. Communications*, vol. 36, pp. 12-20, Jan. 1988.

[30] A. Duel-Hallen, C. Heegard, "Delayed decision-feedback sequence estimation," *IEEE Trans. Communications,* vol. 37, pp. 428-436, May 1989.

[31] N. Al-Dhahir, A. H. Sayed, "The finite-length multi-input multi-output MMSE-DFE," *IEEE Trans. Signal Processing*, vol. 48, no. 10, pp. 2921-2936, 2000.

[32] A. Lozano, C. Papadias, "Layered space-time receivers for frequency-selective wireless channels," *IEEE Transactions on Communications*,vol. 50, no. 1, pp. 65-73, Jan. 2002.

[33] C. Komninakis, C. Fragouli, A. H. Sayed, R. D. Wesel, "Multi-input multi-output fading channel tracking and equalization using Kalman estimation," *IEEE Transactions on Signal Processing*, vol. 50, no. 5, pp. 1065-1076, May 2002.

[34] X. Wang, H.V. Poor, "Iterative (Turbo) soft interference cancellation and decoding for coded CDMA", *IEEE Trans. Communications*, vol. 47, pp. 1046-1061, July 1999.

[35] M. Tuchler, R. Koetter, A. C. Singer, "Turbo equalization: Principles and new results," *IEEE Transactions on Communications,* vol. 50, pp. 754-767, May 2002.

[36] G. Bauch, N. Al-Dhahir, "Reduced-complexity space-time turbo-equalization for frequency-selective MIMO channels," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 819-828, Oct 2002.

[37] W. K. Ma, T. N. Davidson, K. M. Wong, Z. Q. Luo, P. C. Ching, "Quasi-maximum-likelihood multiuser detection using semi-definite relaxation with application to synchronous CDMA," *IEEE Transactions on Signal Processing*, vol. 50, no. 4, pp. 912-922, April 2002.

[38] M. Pohst, "On the computation of lattice vectors of minimal length, successive minima and reduced basis with applications," *ACM SIGSAM Bull*, vol.15, pp.37-44, 1981.

[39] E. Viterbo and E. Biglieri, "A universal lattice decoder," in 14^{eme} *Colloq.GRETSI*, Juan-les-Pins, France, Sept. 1993, pp.611-614.

[40] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Inform. Theory*, vol. 45, no. 5, July 1999, pp. 1639-1642.

[41] L. Brunel and J. Boutros, "Euclidean space lattice decoding for joint detection in CDMA systems," in *ITW'99*, Kruger National Park, South Africa, June 1999.

[42] O. Damen, A. Chkeif, and J. C. Belfiore, "Lattice code deoder for space-time codes," *IEEE Commun. Letters*, vol. 4, pp. 161-163, May 2000.

[43] A. M. Chan and I. Lee, "A new reduced-complexity sphere decoder for multiple antenna systems," *Proc. Int. Conf. Commun. (ICC'02)*, New York City, April 2002.

[44] Y. Wu and S. Y. Kung, "Signal detection for MIMO-ISI channels: an iterative greedy improvement approach," *IEEE Trans. Signal Processing,* vol. 52, no. 3, pp. 703-720, March 2004.

[45] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inform. Theory,* vol. 48, pp. 2201-2214, Aug. 2002.

[46] J. Luo, "Improved multiuser detection in code-division multiple access communications," Ph.D. dissertation, University of Connecticut, 2002.

[47] Y. Bar-Shalom, X. R. Li, *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, 1993.

[48] G. D. Golden, G. J. Foschini, R. A. Valenzuela, P. W. Wolniansky, "Detection Algorithm and Initial Laboratory Results using the V-BLAST Space-Time Communication Architecture," *Electronics Letters*, Vol. 35, No. 1, pp. 14-15, 1999.

[49] S. Haykin, *Adaptive Filter Theory*, 3rd ed., Prentice Hall, Upper Saddle River, NJ, 1996.

[50] S. Verdu, *Multiuser Detection*, Cambridge University Press, 1998.

[51] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*, Prentice-Hall, Inc. Englewood Cliffs, New Jersey 07632.

[52] M. K. Tsatsanis, G. B. Giannakis, and G. Zhou, "Estimation and equalization of fading channels with random coefficients," *ICASSP–96,* vol. 2, pp. 1093-1096, May 7-10, 1996.

[53] S. Haykin, A. H. Sayed, J. R. Zeidler, P. Yee, and P. C. Wei, "Adaptive tracking of linear time-variant systems by extended RLS algorithms," *IEEE Trans. Signal Processing,* vol. 45, pp. 1118-1128, May 1997.

[54] S. Liu, Z. Tian, "A Soft-Decision Approach for BLAST Systems with Less Receive than Transmit Antennae," *Proceedings of IEEE Global Communications Conference (Globecom'2003)*, San Francisco, CA, December 1-5, 2003.

[55] M. O. Damen, K. Abed-Meraim, J.-C. Belfiore, "A generalized lattice decoder for asymmetrical space-time communication architecture", *Proceedings of ICASSP' 2000*, pp. 2581-2584, 2000.

[56] J. Medbo, H. Andersson, P. Schramm, H. Asplund and J. -E. Berg, "Channel models for HIPERLAN/2 in different indoor scenarios," *Tech. Rep. COST 259 TD(98)070*, Bradford, UK, Apr. 1998.

[57] R. D. J. van Nee, G. A. Awater, M. Morikura, H. Takanashi, M. A. Webster, and K. W. Halford, "New high-rate wireless LAN standards," *IEEE Commun. Mag.*, vol. 37, pp.82-88, Dec. 1999.

[58] M. Pursley, "Performance evaluation for phase-coded spread-spectrum multiple-access communication - Part I: System analysis," *IEEE Trans. Commun.,* vol. COM-25, pp.795-799, Aug. 1977.

[59] X. Deng, A. M. Haimovich, and J. Garcia-Frias "Decision directed iterative channel estimation for MIMO systems," *IEEE International Conference on Communications, 2003*, vol. 4, pp. 2326 - 2329, May 11-15, 2003.

[60] Z. B. Baranski, A. M. Haimovich, and J. Garcia-Frias, "EM-based iterative receiver for space-time coded modulation with noise covariance estimation," *IEEE Globecom*, Nov. 2002.

[61] E. Chiavaccini and G. M. Vitetta, "MAP symbol estimation on frequency-flat Rayleigh fading channels via a Bayesian EM algorithm," *IEEE Trans.Commun.*, vol. 49, no. 11, pp. 1869-1872, Nov. 2001.

[62] C. Cozzo and B. L. Hughes, "An iterative receiver for space-time communications," *2000 Conference on Information Sciences and Systems*, 2000.

[63] A. Grant, "Joint decoding and channel estimation for space-time codes," *IEEE Vehicular Technology Conference*, pp. 416-420, 2000.

[64] C. Komninakis and R. D. Wesel, "Joint iterative channel estimation and decoding in flat correlated Rayleigh fading," *IEEE J. Select. Areas Commun.*, vol. 19, no. 9, pp. 1706-1717, Sept. 2001.

[65] Y. Li, C. N. Georghiades, and G. Huang, "Iterative maximum-likelihood sequence estimation for space-time coded systems," *IEEE Trans. Commun.*, vol. 49, no. 6, pp. 948-951, June 2001.

[66] H.-J. Su and E. Geraniotis, "Low-complexity joint iterative channel estimation and decoding for pilot symbol-assisted modulation and multiple differential detection systems with correlated Rayleigh fading," *IEEE Trans. Commun.*, vol. 50, no. 2, pp. 249-261, Feb. 2002.

[67] M. C. Valenti and B. D. Woerner, "Iterative channel estimation and decoding of pilot symbol assisted turbo codes over flat-fading channels," *IEEE J. Select. Areas Commun.*, vol. 19, no. 9, pp. 1697-1705, Sept.2001.

[68] Y. Sun, M. Yee and M. Sandell, "Iterative channel estimation with MIMO MMSE-turbo equalisation", *IEEE 58th Vehicular Technology Conference*, vol. 2, pp. 1278 - 1282, Oct. 6-9, 2003.

[69] H. Lutkepohl, *Handbook of matrices*, Chichester ; New York : Wiley, c1996.

[70] S. Haykin, "Adaptive filter theory", 3rd. ed., Prentice Hall, Upper Saddle River, N.J., 1996.

[71] C. F. J.Wu, "On the convergence properties of the EM algorithm," *Ann. Statist.*, vol. 11, pp. 95-103, 1983.

[72] A. P. Dempster, N. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc. B*, vol. B39, pp. 1-38, 1977.

[73] H. Rohling, T. May, K. Bruninghaus, and R. Grunheid, "Broadband OFDM radio transmission for multimedia applications," *Proc. IEEE*, vol. 87, pp. 1778-1789, Oct. 1999.